Learning to Fly like a Bird

Russ Tedrake, Zack Jackowski, Rick Cory, John William Roberts, Warren Hoburg
Massachusetts Institute of Technology
Computer Science and Artificial Intelligence Lab
32 Vassar St, MIT 32-380
Cambridge, MA, 02139 USA
{russt,zackbass,recory,jwr,whoburg}@mit.edu

ABSTRACT

Birds routinely execute aerial maneuvers that are far beyond the capabilities of our best aircraft control systems. The complexity and variability of the aerodynamics during these maneuvers are formidable, with dominant flow structures (e.g., vortices) that are difficult to predict robustly from first-principles (Navier-Stokes) models. Here we argue that machine learning will play an important role in the control design process for agile flight by building data-driven approximate models of the aerodynamics and by synthesizing high-performance nonlinear feedback policies based on these approximate models and trial-and-error experience. This article highlights some of the more remarkable characteristics of nature's flyers, and describes the challenges involved in replicating this performance in our machines. We conclude by describing our two-meter wingspan autonomous robotic bird and some initial results using machine learning to design control systems for bird-scale, supermaneuverable flight.

Categories and Subject Descriptors

I.2.9 [**Robotics**]: Autonomous vehicles; I.2.9 [**Robotics**]: Propelling Mechanisms

General Terms

Algorithms, Experimentation

Keywords

Machine learning, Reinforcement learning, Control theory, Flying robots, Flapping-Wings, Fluid Dynamics

1. INTRODUCTION

Watch carefully the next time you see a neighborhood bird fly past the window and land on the branch of a tree. That little bird is casually, but dramatically, outperforming some of the best control systems ever designed by humans.

During a "perching" maneuver, birds rotate their wings and bodies so that they are almost perpendicular to the direction of travel and oncoming airflow. This maneuver increases the aerodynamic drag on the bird both by increasing the surface area exposed to the flow and by creating a low-pressure pocket of air behind the wing. Viscous and pressure forces combine for the desired rapid deceleration, but the maneuver has important consequences: the wings become "stalled", meaning they experience a dramatic loss of lift and potentially control authority, and the aerodynamics become unsteady (time-varying) and nonlinear, making the aerodynamic forces difficult to model and predict accurately. The task is further complicated by uncertain wind dynamics (which affect both the bird and the perch) and the partial observability of the airflow. Yet birds perch with apparent ease.

By comparison, helicopters and vertical take-off and landing (VTOL) airplanes require considerably more time and energy to land on a target. Airplanes today can't yet land on a perch with the dynamic performance of a bird. It turns out that we might not be too far from providing this capability, but success will require control systems which reason about the complicated unsteady post-stall aerodynamics. Flared perching is just one example in which birds exploit complicated dynamic interactions with the airflow to outperform our best engineered vehicles in metrics of speed, efficiency, and/or maneuverability.

This article is intended as a call to arms for computer scientists. Our fascination with birds is as old as time, and thanks to recent advances in UAV technology we may soon replicate the performance of birds with our machines. Computer science is suddenly well-positioned to make a substantial impact on this fascinating problem.

2. AMAZING FLYING MACHINES

Modern airplanes are extremely effective for steady, level flight in still air. Propellers produce thrust very efficiently, and today's cambered airfoils are highly optimized for speed and/or efficiency. But examining performance in more interesting flight regimes reveals why birds are still the true masters of the sky.

Birds are extremely efficient flying machines; some are capable of migrating thousands of kilometers with incredibly small fuel supplies. The wandering albatross can fly for hours, or even days, without flapping its wings by exploiting the shear layer formed by the wind over the ocean surface in a technique called dynamic soaring. Remarkably, the metabolic cost of flying for these birds is indistinguishable from the baseline metabolic cost[5], suggesting that they can travel incredible distances (upwind or downwind) powered almost completely by gradients in the wind. Other birds achieve efficiency through similarly rich interactions with the air - including formation flying, thermal soaring, and ridge soaring[32]. Small birds and large insects, such as butterflies and locusts, use 'gust soaring' to migrate hundreds or even thousands of kilometers carried primarily by the wind[32].

Birds are also incredibly maneuverable. The roll rate of a barn swallow is in excess of 5000 deg/sec[22].Bats can be flying at full-speed in one direction, then be flying at full-speed in the opposite direction, using a turning maneuver that is accomplished in just over 2 wing-beats and in a distance less than half the wingspan[27]. Although quantitative flow visualization data from maneuvering flight is scarce, a dominant theory is that the ability of these animals to produce sudden, large forces for maneuverability can be attributed to unsteady aerodynamics, e.g., the animal creates a large suction vortex to rapidly change direction[28]. These astonishing capabilities are called upon routinely in maneuvers like flared perching, prey-catching, and high speed flying through forests and caves. Even at high speeds and high turn rates, these animals are capable of incredible agility - bats sometimes capture prey on their wings, Peregrine falcons can pull 25 G's out of a 240 mph dive to catch a sparrow in mid-flight[29], and even the small birds on the MIT campus can be seen diving through a chain-link fence to grab a bite of food.

Although many impressive statistics about avian flight have been recorded, our understanding is partially limited by experimental accessibility - its is quite difficult to carefully measure birds (and the surrounding airflow) during their most impressive maneuvers without disturbing them. The dynamics of a swimming fish are closely related, and can be more convenient to study. Dolphins have been known to swim gracefully through the waves alongside ships moving at 20 knots[28]. Smaller fish, such as the bluegill sunfish, are known to possess an escape response in which they propel themselves to full speed from rest in less than a body length; flow visualizations indeed confirm that this is accomplished by creating a large suction vortex along the side of the body[30] - similar to how bats change direction in less than a body length. There are even observations of a dead fish swimming upstream by pulling energy out of the wake of a cylinder; this passive propulsion is presumably part of the technique used by rainbow trout to swim upstream at mating season[6].

3. MANIPULATING THE AIR

These examples illustrate that the key attribute which separates birds from our machines is not their flapping wings, but rather their more delicate interaction with the surrounding airflow. These interactions occur at many different time-scales - from relatively persistent aerodynamic structures like the shear layer on the ocean or updrafts along a ridge, to transient thermal and wind gusts, all of the way down to the unsteady aerodynamics that occur with a single flap of the wings.

One of the primary reasons why many of these topics have not received sufficient attention from the engineering community to date is that birds are operating in a different dynamic regime than most of our man-made vehicles. Because they are considerably smaller, and correspondingly slower than an airplane, small-scale atmospheric events which are of minor consequence to a fighter jet or an airliner can

Figure 1: Smoke visualization of the perching plane.

dominate the dynamics of a bird-scale flying machine; this presents both a challenge and an opportunity. Advances in miniaturization of power, actuator, sensing, and computational resources, along with the advances in technologies for unmanned vehicles, have only recently made these small vehicles practical.

One only has to watch a video of an eagle soaring at the edge of a cliff[1] to appreciate the fundamental problem of flying like a bird. These birds are able to remain incredibly stationary with respect to the world (lift occurs because they are moving with respect to the air), presumably keeping their eyes still to watch for something the size of a mouse moving hundreds of feet below. The station keeping occurs not by flapping, but by making constant minor adjustments with the fingers and tail, a ritual that might best be described as "playing" with the incredibly complicated airflow coming off the cliff. The fundamental problem in flying like a bird is not generating lift and thrust with flapping wings, nor optimizing an airfoil for lift-to-drag ratio, but rather using small adjustments on the aerodynamic surfaces to achieve a refined manipulation of the complicated, unsteady airflow. As roboticists, the authors cannot help but think of this as the ultimate robotic manipulation problem.

4. A MACHINE LEARNING PROBLEM

In fluid dynamics, the dimensionless Reynolds number (Re) characterizes the ratio of inertial forces to viscous forces in a fluid flow. Roughly speaking, at very low Re, viscosity dominates and the fluid- (or air-) flow is laminar and smooth. At very high Re, inertia dominates and flows are turbulent. Bird flight is an *intermediate Reynolds number* problem, typified by Reynolds numbers between 10^3 and 10^5, where the mean aerodynamic chord of the wing is used as the reference length. At these intermediate Reynolds numbers, the flow is complicated but structured, dominated by features such as vortices, and just beginning the transition to turbulence. Figure 1 is a smoke visualization of the airflow behind the wing of a bird-scale airplane at 10^4 which illustrates this clear but complicated structure.

Although the governing equations for intermediate Reynolds number flows are known (the Navier-Stokes equations), solving these partial differential equations accurately can require prohibitive spatial resolution. As such, computational fluid dynamic (CFD) codes which simulate a single wing-beat at Re 10^3 can take hours or days to compute, while Re 10^5 flows are generally considered to be computationally intractable. Furthermore, the details of a flow of this complexity will be very sensitive to modeling assumptions about boundary conditions and the exact properties of the wing. Computational investigations of flapping flight have been, and will certainly continue to be, an important tool for understanding avian flight, but these techniques may not be the most direct tool for designing a control system for a robotic bird.

4.1 Learning Approximate Models

Machine learning potentially has an important role to play in building tractable approximate models of the aerodynamics in this regime. Perhaps analagous to speech recognition and machine vision, we suspect that data-driven models will be more successful than models based directly on the physics involved; speech and vision rarely make extensive use of the detailed mechanics of the vocal tract/ear/eye nor sound/light propagation. In fact, there is substantial evidence of a robust low-dimensional structure embedded within the complicated fluid dynamics of bird flight. Experimental fluid dynamicists are often able to write down suitable first-principle models which describe the dominant dynamics of their experiments[4, 8], and the considerable literature on model-order reduction for CFD has demonstrated the potential for dimensionality reduction, even in the governing equations[14, 24]. However, the derived models must be rich and robust, as small atmospheric disturbances that would go unnoticed to a large vehicle can dominate the dynamics of these small vehicles.

A primary challenge in this modeling exercise is that the flows which dominate the wings are difficult to sense, and are partially observable at best. Flow visualizations and quantitative flow measurements typically require specialized experimental setups - these are appropriate for wind-tunnel testing but typically not for outdoor flight. Local flow sensors that can be placed on the wing (such as mechano-receptors at the wing-tip insertion) can be strongly effected by boundary layer shear, and are presumably limited in their ability to predict oncoming wind gusts early enough for the bird to respond. There is relatively little known about how detailed and predictive of a flow model birds use to accomplish their aerodynamic prowess.

4.2 Algorithms for Feedback Control Design

Even if the model was known exactly and the airflow was fully-observable, designing a feedback control system for systems of this complexity remains an open and challenging problem due to the severe dynamic constraints. The control system is underactuated, relying on thrust and local control surfaces for orientation to control six degrees of freedom. Furthermore, aerodynamic control forces take time to develop; for example, the response in lift force to leading-edge flow control actuators occurs only after a noticeable delay [34]. Separation on the wings, as experienced during an aerodynamic stall, can also result in intermittent losses of control authority. Furthermore, the dynamics of the fluid are described as a continuum model - even relatively low-order lumped-parametert approximations of these complicated dynamics will likely reside in a very high-dimensional state space compared to the capabilities of our current methods.

There is considerable work on model-based algorithms which attempt to systematically design nonlinear feedback for complicated control systems; often these are based on sample-based motion planning or dynamic programming and optimal control[26]. Many of these are based on discretizing the state-action space, and nearly all struggle with designing controllers in high-dimensional state spaces. Abbeel recently demonstrated aggressive maneuvering with a model helicopter based on learning trajectories from human experts and tracking them with receding-horizon control[2]; however, our primary interest here is in accomplishing maneuvers that are beyond the capabilities of our best human pilots. Our current model-based control design efforts are focused on algorithms which efficiently stitch together locally-valid linear control laws into a nonlinear policy by explicitly reasoning (with semi-definite programming) about the basins of attraction of the local controllers[25]. Many other approaches could succeed, and more work is certainly needed.

4.3 When Modeling Fails

To date, the flow control community has yet to make serious progress in low-dimensional models for transient regimes, and problems with aero-elasticity. Due to limited observability, strong nonlinearities, and simply the variability of the flow conditions, there will inevitably be non-neglible differences between the actual dynamics and our approximate models. The standard approach to this problem in systems theory is robust control synthesis[35], which attempts to bound the modeling errors then design a feedback control which is guaranteed to stabilize all possible models in this defined class. The well-known problems with robust control synthesis are that it tends to require considerable control authority, and tends to result in very conservative performance.

A potentially unique and essential contribution from machine learning to this domain will be in the use of "model-free" control methods from reinforcement learning[23], including the policy gradient methods (e.g,[17]). These methods stochastically optimize an expected value cost function using sampling in policy parameter space and trial-and-error experience to estimate the local *policy gradient* - the expected change in cost based on a change in the parameters. Because they operate directly on experimental data (naively ignoring any previously design approximate models), these methods can potentially be used to overcome approximations in the model - though typically requiring many trials.

An essential property of the policy gradient algorithms, which makes them compelling in this domain, is that the learning performance depends only on the number of control parameters being optimized, and is *locally invariant* to the dimensionality and complexity of the dynamics. As the number of parameters increases, it requires more samples to obtain a low-variance estimate of the gradient. Careful analysis of the signal-to-noise ratio (SNR) of the basic policy gradient algorithms[20] reveals that the SNR degrades as a simple function:

$$\mathrm{SNR} = \frac{3}{N-1}, \tag{1}$$

where N is the number of policy parameters. Note that the dynamics and the cost function appear nowhere in this expression; for a controller with N parameters, it requires the same number of samples to estimate the policy gradient whether that controller is attached to a simple pendulum or a Navier-Stokes simulation. Thus, if a clever policy representation can be found that requires only a small number of parameters, policy gradient learning for bird flight may in fact be relatively very efficient.

Control design based on approximate models or on direct policy search are both viable approaches to avian flight. The correct answer will almost certainly involve both. In the remaining sections, we will present a few initial results using these machine learning ideas to generate experimental control solutions.

5. A PERCHING AIRPLANE

Can an airplane land on a perch like a bird? Most modern control systems impose a hard limit on the *angle of attack* (AoA) of the aircraft, ensuring that the wings stay at a low angle relative to the oncoming airflow in order to avoid stall. These vehicles achieve only a fraction of the desired drag (scaled) of a perching bird and consequently require long runways for landing[12]. A number of "supermaneuverable" fighter jets have demonstrated post-stall angle-of-attack maneuvers in airshows (such as the famous "Pugachev's Cobra" maneuver), using high thrust-to-weight ratios for lift, which are energetically expensive to maintain, along with thrust vectoring or passive stability in the aircraft design to return from stall[11]. The flight control systems used in these maneuvers operate without high fidelity models of the airflow[3], so probably cannot execute with sufficient position precision to, for instance, land on a (scaled) perch. To the authors' knowledge, the most dynamic landings achieved by a piloted fixed-wing vehicle to date are the 24 degree AoA runway landings achieved by the X-31 research vehicle[33]. In practice, a 747 lands with about 15 degrees AoA; an F-18 lands on a carrier at a very moderate 8.1 degree AoA[18], because it must be ready to take-off again if the tail-hook misses the arrestor cable. Birds, on the other hand, routinely surpass a 90 degree AoA during flared perching.

We designed an indoor experiment for flared perching in a motion capture studio with a closed-aerodynamic environment. We designed a small foam glider (77g, 260mm wingspan, 98mm mean chord) with passive roll and yaw stability and a single actuator for the elevator. We hypothesized that by designing an appropriate nonlinear feedback control law, the small glider would be capable of autonomously landing on a perch by executing the flared perching maneuver. The glider was launched at 6 m/s towards a perch 3.5 m away, and constrained to land in less than one second. Under these tight dynamic constraints, the (un-propelled) glider must exploit viscous and pressure drag forces for a successful perch. In order to test our hypothesis, we first collected post-stall flight data by launching our glider from a custom launching device approximately 200 times in a motion capture arena, executing open loop trajectories that would cover a wide set of angle-of-attack and elevator angle combinations, capturing data that was representative of our perching dynamics. Through careful analysis of the kinematic flight data, we were able to recover surprisingly clean aerodynamic coefficients at very high angles of attack [7] (see Figure 2).

Given this acquired model, we formulate the optimal control problem:

$$\min_{\alpha,t_f} J(\mathbf{x}_0) = \mathbf{x}^T(t_f)\mathbf{Q}_f\mathbf{x}(t_f), \text{ subject to}$$

$$\mathbf{x}(0) = \mathbf{x}_0, \quad \dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), u(t)), \quad u(t) = \pi_\alpha(\mathbf{x}(t), t)),$$
$$\mathbf{x}_{mintol} \leq \mathbf{x}(t_f) \leq \mathbf{x}_{maxtol}, \quad |u| \leq u_{lim}$$

where $\mathbf{x}$ is the seven-dimensional state of the glider in a coordinate system relative to the perch, $\mathbf{Q}_f$ is a positive definite final-cost matrix, $\mathbf{f}$ is the acquired dynamic model, π_α is

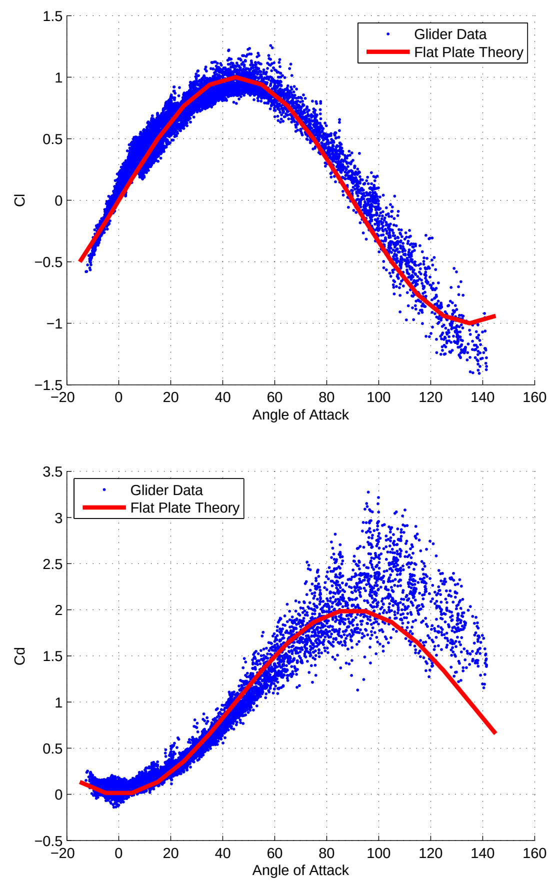

Figure 2: Lift and drag coefficients of the perching plane.

the feedback policy parameterized by vector α, u_{lim} is the actuator contraint, and $\mathbf{x}_{mintol}$, $\mathbf{x}_{maxtol}$ are hard final-value constraints which require that the plane lands within the tolerance of the latching mechanism on the perch. We have approximated solutions to this optimal control problem with a number of methods: initially with a dynamic programming algorithm[7], and more recently with open-loop trajectory optimization and local trajectory stabilization[13]. These approaches differ in their robustness to initial conditions and disturbances[19], as well as their computation time; dynamic programming on the large mesh took hours to run, while local trajectory optimization and stabilization run at speeds compatible with real-time planning. Transferring the acquired policy onto the real plane then allowed the glider to autonomously land on a perch from initial conditions within $\pm$ 1 m/s of the initial training data. Figure 3 shows a successful perching trajectory.

Although our simplified glider design has fixed wings, it still has the capability of inducing and exploiting complicated airflow structures during such a dynamic maneuver. During perching, our foam glider easily exceeds a 90 degree AoA, causing the airflow to separate over the wings and produce considerable pressure drag. The smoke visualization in Figure 1 is taken from a perching trajectory, and clearly reveals the dominant vortex dynamics behind the wing during the maneuver. The success of this experiment was primarily due to our ability to simplify the dynamics of the problem down into a planar dynamics with a seven-dimensional state space - just at the limits of our capabilities for dynamic programming-type algorithms - and our ability to identify a sufficiently accurate model of the aircraft to perform model-based feedback design.

We are currently working to demonstrate perching in an outdoor environment. One of the primary challenges in the perching problem for a fixed-wing aircraft is that as the airspeed goes to zero near the perch, so does the aerodynamic control authority[19]. When the vehicle is near the perch, even a small gust of wind can knock it off course. Birds nearly always flap their wings during landing - this contributes quite effectively to the deceleration of the bird. But we hypothesize that flapping during the final moments of landing also serves to maintain airspeed on the wings, and therefore maintain control authority. Although there are a number of actuation schemes possible for investigating super-maneuverable flight (thrust vectoring is the most popular choice on fighter jets), we believe that flapping wings will enable a more intricate interaction with the surrounding airflow.

6. LEARNING FLAPPING FLIGHT

Unlike the perching glider, the complex aerodynamics of flapping wings are not as easily captured by a low-order statistical model (yet). In lieu of a descriptive dynamical model, we investigated the feasibility of using only policy gradient methods to optimize flapping flight. Initial experiments were carried out with Jun Zhang at the Applied Math Lab at NYU on an experimental laboratory system he developed to study the fluid dynamics of a flapping wing[31]. In this experiment, a rigid flat plate is driven by a linear motor vertically in a fluid, while it is free to spin about its center (see Figure 4). This spinning motion replicates the features of a simple wing flying forward, but allows experiments to be carried on indefinitely without resetting the system. This

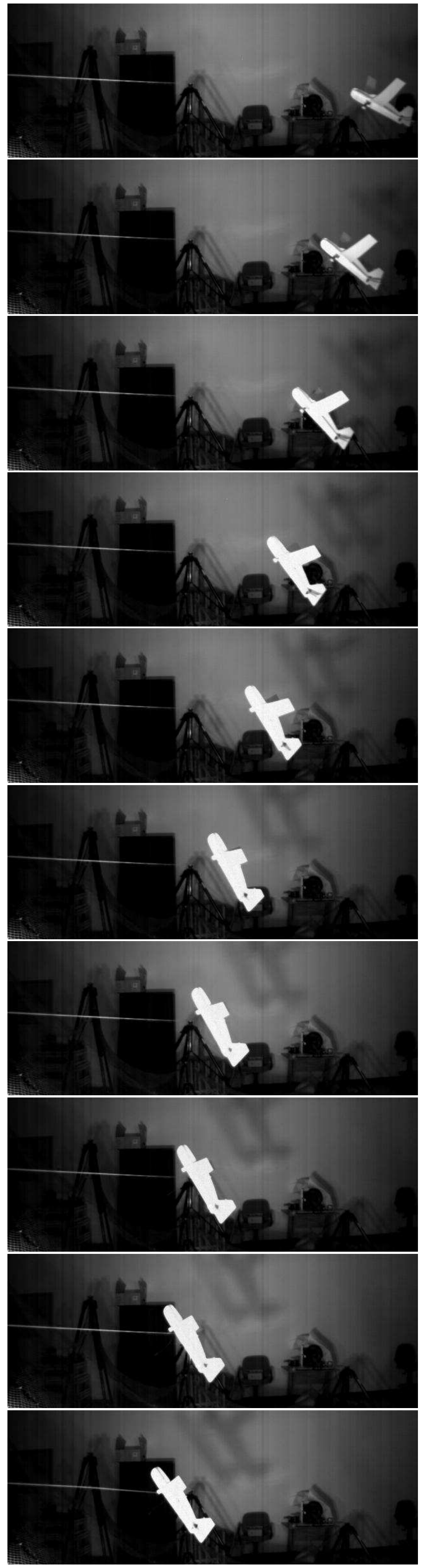

Figure 3: Stills from a successful perching trajectory.

system acts as a simple model for forward flapping flight, possessing a Reynolds number of approximately 16,000 during these experiments, placing it in the same regime as a large dragonfly flying forward.

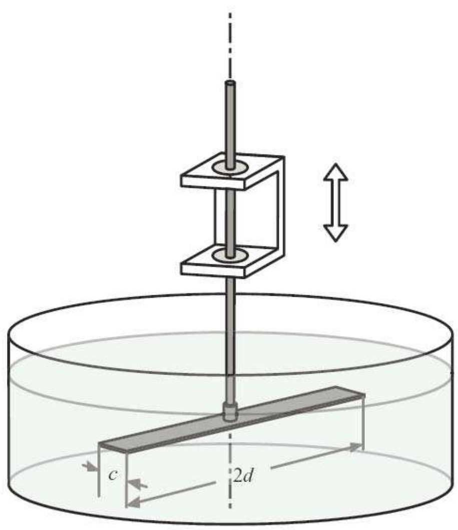

Figure 4: A schematic of the rigid flat plate flapping experiment (courtesy of Jun Zhang, NYU).

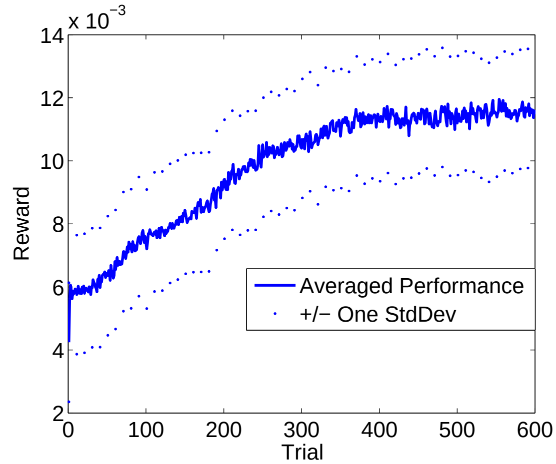

Figure 5: The average of five learning curves using online learning (an update every second, after each full flapping cycle), with markers for +/- one standard deviation. The high variance is the result of large inter-trial variance in the cost, rather than large differences between different learning curves.

The control task in this problem was to control the heave of the foil vertically in order to maximize the horizontal propulsive efficiency, as measured by the dimensionless cost of transport, defined over one period T as follows:

$$c_{mt} = \frac{\int_T |F_z(t)\dot{z}(t)|dt}{mg\int_T \dot{x}(t)dt}, \tag{2}$$

where F_z is the vertical force, m is the mass of the body, x is the angular position of the plate, z is its vertical position and g is gravitational acceleration. The vertical motion of the plate was parameterized as a symmetric periodic cubic spline with fixed amplitude and frequency with five independent parameters. With every flap of the flat plate, the policy gradient algorithm makes small random changes to these control parameters and correlates these changes with the resulting change in the reward. By optimizing the random sampling distribution to maximize the signal-to-noise ratio of the algorithm, we found we were able to learn the optimal flapping motion in as few as four hundred flapping cycles, or approximately seven minutes (see Figure 5 and [20]). In contrast, cutting edge CFD simulations of this system take approximately 36 hours to simulate 30 seconds of flight. Unlike the simulation model, the policy gradient algorithm easily accommodates changes in the experimental setup, including a flexible trailing edge and passive pitching. This versatility is due to the essential property of the policy gradient algorithms - learning performance (as measured by the number of trials required to accurately estimate the gradient) degrades with the number of policy parameters, but is insensitive to the complexity of the dynamics in the task.

Learning trials in this periodic flapping system were experimentally cheap (the system didn't fall out of the sky every time we executed a bad policy), and the task was to optimize a single nominal trajectory. These features made the system quite compatible with online policy gradient learning. In contrast, nearly every trial in the perching experiment ended in a crash landing, requiring human intervention to restart the system, and the goal was to design a controller that could respond to a range of initial conditions - in this case model-based control was more appropriate. We anticipate that most systems will make use of both methods: using an initial model-based controller design to achieve reasonable performance, then policy-gradient optimization to overcome limitations imposed by the approximations in the model.

7. A TWO-METER WINGSPAN AUTONOMOUS ORNITHOPTER

In order to continue our investigations of flapping-winged flight, we have designed a large, two-meter wingspan, robotic bird - commonly referred to as an ornithopter (Figure 6). This machine was designed as a platform for control experiments, with enough payload to carry plentiful sensing and computational resources, and with sufficient robustness to endure the controller development process. We affectionately call this robot the "Phoenix", because of its large red wings and its propensity to crash dramatically, then rise from the ashes.

There have been a number of successful demonstrations of small-scale radio-controlled ornithopters; Many of these were not developed in research labs, but rather by hobbyists and enthusiasts of radio-controlled aircraft. Perhaps the most impressive are Sean Kinkade's 'Park Hawk' and 'Slow Hawk' designs[15]. There have even been demonstrations of steady-level autonomous ornithopter flights using standard off the shelf autopilots (e.g. [16]), and investigations with simplified models of stabilizing controllers[21, 10], again for steady-level flight. There has even been a short demonstration of flight by a human-scale ornithopter[9].

The design of the Phoenix ornithopter was based on the wing design of Kinkade's Slow Hawk - a nylon membrane stretched across a carbon-fiber frame. In order to accommodate the very large forces present during the downstroke of our vehicle, which has a wingspan almost twice as large as the Slow Hawk, the drive train was completely redesigned. In the current model, we use a large brushless motor and a titanium welded geared four-bar linkage transmission. The bird currently carries a small processor running Linux, a solid-state inertial measurement unit, altimeters, and all of the necessary communication hardware. The mechanical design also features engineered failure points, including a breakaway fiberglass beak and easily replaceable wing spars - these have proven essential elements for our initial control experiments. The powertrain consumes approximately 300 watts during flight, sourced from a large lithium polymer battery pack. The bird flaps its wings at approximately 2.4Hz. The control system directly commands the voltage to the powertrain motor and the orientation of the tail via two position-controlled servo motors.

To date, we have only achieved simple steady, level flight with this machine, using a hand-tuned linear controller acting on state information low-pass filtered below the flapping frequency. Figure 6 shows frames from a successful outdoor autonomous steady-level flight at approximately 4 m/s. The machine is now ready to serve as a test-bed for more extensive control experiments; our goals include making this bird land on the branch of a tree and turn 180 degrees at full speed in less than a meter.

Figure 6: The MIT Phoenix Ornithopter. Photos by Jason Dorfman.

8. CONCLUSIONS

Birds fly using a delicate controlled interaction with the surrounding airflow that is, so far, unparalleled by our machines. Engineering this prowess into our machines will enable more efficient and more agile unmanned aerial vehicles, will address fundamental issues in nonlinear control, and will contribute to our scientific understanding of avian flight. Due to the dynamical complexity of the problem, we expect that designing high-performance control systems with approximate dynamical models will be an essential theme. Our initial results suggest that techniques from machine learning for building approximate dynamic models, for model-based feedback design, and for model-free policy improvement, are well-matched to the dynamic complexity of the task. Indeed, it may actually be easier to design a control system for agile flight on a robotic bird than it is to describe the aerodynamics; birds probably don't solve Navier-Stokes.

9. ACKNOWLEDGMENTS

The authors would like to thank Jun Zhang at NYU for his generous collaboration with the heaving foil experiment. The authors gratefully acknowledge support from MIT, NEC, MIT Lincoln Labs, and the Microsoft New Faculty Fellowship.

10. REFERENCES

[1] Spy on the wild. Discovery Channel in cooperation with BBC, `http://animal.discovery.com/videos/spy-on-the-wild-tiny-tv-camera.html`, May 2006.

[2] P. Abbeel, A. Coates, M. Quigley, and A. Y. Ng. An application of reinforcement learning to aerobatic helicopter flight. In *Proceedings of the Neural Information Processing Systems (NIPS '07)*, volume 19, December 2006.

[3] C. W. Alcorn, M. A. Croom, and M. S. Francis. The X-31 experience: Aerodynamic impediments to post-stall agility. In *Proceedings of the 33rd Aerospace Sciences Meeting and Exhibit*. AIAA 95-0362, January 9-12 1995.

[4] A. Anderson, U. Pesavento, and Z. J. Wang. Unsteady aerodynamics of fluttering and tumbling plates. *J. Fluid Mech.*, 541:65–90, 2005.

[5] J. Arnould, D. Briggs, J. Croxall, P. Prince, and A. Wood. The foraging behaviour and energetics of wandering albatrosses brooding chicks. *Antarctic Science*, 8(3):229–236, 1996.

[6] D. Beal, F. Hover, M. Triantafyllou, J. Liao, and G. V. Lauder. Passive propulsion in vortex wakes. *Journal of Fluid Mechanics*, 549:385âĂŞ402, 2006.

[7] R. Cory and R. Tedrake. Experiments in fixed-wing UAV perching. In *Proceedings of the AIAA Guidance, Navigation, and Control Conference*. AIAA, 2008.

[8] J. DeLaurier. An aerodynamic model for flapping-wing flight. *The Aeronoutical Journal of the Royal Aeronautical Society*, April 1993.

[9] J. DeLaurier. The development and testing of a full-scale piloted ornithopter. *Canadian Aeronautics and Space Journal*, June 1999.

[10] J. M. Dietl and E. Garcia. Ornithopter flight stabilization. volume 6525. SPIE, 2007.

[11] W. B. Herbst. Supermaneuverability. In *AFOSR Proc. of the Workshop on Unsteady Separated Flow*, pages 1–9, August 1983.

[12] W. Hoburg, J. W. Roberts, J. Moore, and R. Tedrake. The perching number: A dimensionless analysis of post-stall maneuvering in birds and planes. *Working Draft*, 2009.

[13] W. Hoburg and R. Tedrake. System identification of post stall aerodynamics for UAV perching. In *Proceedings of the AIAA Infotech@Aerospace Conference*, Seattle, WA, April 2009. AIAA.

[14] P. Holmes, J. Lumley, and G. Berkooz. *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*. Cambridge University Press, 1998.

[15] S. Kinkade. Hobbytechnik. http://www.flappingflight.com., 2009.

[16] R. Krashanitsa, D. Silin, and S. Shkarayev. Flight dynamics of flapping-wing air vehicle. In *Proceedings of the AIAA Atmospheric Flight Mechanics Conference*, pages 1–12. AIAA, August 2008.

[17] J. Peters, S. Vijayakumar, and S. Schaal. Reinforcement learning for humanoid robotics. In *Proceedings of the Third IEEE-RAS International Conference on Humanoid Robots*, Sept. 29-30 2003.

[18] A. Prickett and C. Parkes. Flight testing of the f/a-18e/f automatic carrier landing system. *Aerospace Conference, 2001, IEEE Proceedings.*, 5:2593–2612 vol.5, 2001.

[19] J. W. Roberts, R. Cory, and R. Tedrake. On the controllability of fixed-wing perching. In *To Appear in the Proceedings of the American Controls Conference (ACC)*, 2009.

[20] J. W. Roberts and R. Tedrake. Signal-to-noise ratio analysis of policy gradient algorithms. In *Advances of Neural Information Processing Systems (NIPS) 21*, page 8, 2009.

[21] L. Schenato, W. Wu, and S. Sastry. Attitude control for a micromechanical flying insect via sensor output feedback. In *IEEE International Conference on Robotics and Automation*. IEEE, May 2003.

[22] W. Shyy, Y. Lian, J. Teng, D. Viieru, and H. Liu. *Aerodynamics of Low Reynolds Number Flyers*. Cambridge Aerospace Series. Cambridge University Press, 2008.

[23] R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, 1998.

[24] G. Tadmor, M. D. Centuori, B. R. Noack, M. Luchtenburg, O. Lehmann, and M. Morzynski. Low Order Galerkin Models for the Actuated Flow Around 2-D Airfoils. In *45th AIAA Aerospace Sciences Meeting and Exhibit*, 8 - 11 January, Reno, Nevada 2007. Paper AIAA 2007-1313.

[25] R. Tedrake. LQR-Trees: Feedback motion planning on sparse randomized trees. In *Accepted for the Proceedings of Robotics: Science and Systems (RSS)*, 2009.

[26] R. Tedrake. *Underactuated Robotics: Learning, Planning, and Control for Efficient and Agile Machines: Course Notes for MIT 6.832*. Working draft edition, 2009.

[27] X. Tian, J. Iriarte-Diaz, K. Middleton, R. Galvao, E. Israeli, A. Roemer, A. Sullivan, A. Song, S. Swartz, and K. Breuer. Direct measurements of the kinematics and dynamics of bat flight. *Bioinspiration & Biomimetics*, 1:S10–S18, 2006.

[28] M. S. Triantafyllou and G. S. Triantafyllou. An efficient swimming machine. *Scientific American*, 272(3):64, March 1995.

[29] V. A. Tucker. Gliding flight: Speed and acceleration of ideal falcons during diving and pull out. *Journal of Experimental Biology*, 201:403–414, Nov 1998.

[30] E. D. Tytell and G. V. Lauder. Hydrodynamics of the escape response in bluegill sunfish, lepomis macrochirus. *The Journal of Experimental Biology*, 211:3359–3369, 2008.

[31] N. Vandenberghe, S. Childress, and J. Zhang. On unidirectional flight of a free flapping wing. *Physics of Fluids*, 18, 2006.

[32] J. J. Videler. *Avian Flight*. Oxford University Press, October 2006.

[33] H. Visser. Optimization of high angle-of-attack approach to landing trajectories. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 219(6):497–506, 2005.

[34] D. Williams, J. Collins, V. Quach, W. Kerstens, S. Buntain, T. Colonius, G. Tadmor, and C. Rowley. Low reynolds number wing response to an oscillating freestream with and without feed forward control. slides from talk at AIAA ASM Orlando 2009, 2009.

[35] K. Zhou and J. C. Doyle. *Essentials of Robust Control*. Prentice Hall, 1997.