Learning to Fly: Design and Construction of an Autonomous Airplane

Stephen B. Stancliff
stanclif@eel.ufl.edu

Jennifer L. Laine
jenny@mil.ufl.edu

Michael C. Nechyba
nechyba@mil.ufl.edu

Machine Intelligence Laboratory
Electrical and Computer Engineering
University of Florida
Gainesville, FL 32611-6200

Abstract

Humans are, and for the foreseeable future will remain our best and only example of true intelligence. Therefore, a lot of work has been done in recent years to abstract computational models of human control strategy that are capable of accurately emulating dynamic human control behavior. Land-based autonomous vehicles, both in simulation and on real roads, have, for example, made successful use of this modeling formalism. Very little work in the past has, however, attempted similar such skill transfer from humans to robots for aerial vehicles, such as helicopters or airplanes. Although control can be quite different from land-based vehicles, we contend that human pilots can potentially serve as excellent guides in the development of intelligent autonomous aerial vehicles. As a first step in modeling human control strategies in airplanes, we are developing and building a robotic airplane as an experimental platform for studying human-to-robot skill transfer in aerial vehicles. This paper describes the design of this airplane, the status of the project and future planned experiments.

1. Introduction

Over the past two decades, rapid advances in computer performance have not been matched with similar advances in the development of intelligent robots and systems. Although humans are quite adept at mastering complex and dynamic skills, we are far less impressive in formalizing our behavior into algorithmic, machine-codable strategies. Therefore, it has been difficult to duplicate the types of intelligent skills and actions, we witness every day as humans, in robots and other machines. This not only limits the capabilities of individual robots, but also the extent to which humans and robots can safely interact and cooperate with one another. Nevertheless, human actions are currently our only examples of truly "intelligent" behavior. As such, there exists a profound need to abstract human skill into computational models, capable of realistic emulation of dynamic human behavior.

Thus, a number of different researchers have endeavored in recent years to abstract models of human skill directly from observed human input-output data (see [1,2] for overviews of the literature). Autonomous control and navigation of land-based vehicles is one area of robotics research that has benefited especially from learning through observation of humans. Pomerleau [3, 4], for example, has implemented real-time road-following with data collected from a human driver. A static feedforward neural network with a single hidden layer, ALVINN, learned to map coarsely digitized camera images of the road ahead to a desired steering direction, whose reliability was given through an input-reconstruction reliability estimator. The system has been demonstrated successfully at speeds up to 70 mi/h. Pentland and Liu [5] have applied hidden Markov models towards inferring a particular driver's high-level intentions, such as turning and stopping. Finally, [1, 2, 6] address the autonomous control of a dynamic car — including steering and acceleration — through observation and modeling of extreme human driving inside a driving simulator.

Perhaps surprisingly, little similar such work has been done in creating intelligent, autonomous aerial vehicles from observing human pilots. Although the control and task challenges in autonomous flying may be quite different from those in autonomous driving, the basic paradigm of learning from humans is nevertheless equally applicable. Therefore, we propose to extend some of the methods of learning for land-based vehicles to aerial vehicles. As a first step in this process, we are developing and building a robotic airplane as an experimental platform. Once built, such an airplane could find application in a number of important tasks or missions.

Many of the activities that currently involve remotely piloted vehicles (RPVs) would benefit in some way from automation. It is foreseeable that in many applications, such as surveying, reconnaissance, and target acquisition, entire missions could be fully automated. For other applications, where more sophisticated flying may be required, the human pilot and robotic plane could share and coordinate control of the overall system. Moreover, adding intelligence to RPVs can reduce or eliminate the need for skilled human pilots in situations where substantial risk to the human pilot would otherwise be unavoidable, such as in peace-keeping missions over Kosovo, for example.

We have divided our overall project into three stages, where each stage is characterized principally by the level of autonomy required:

1. The body and electronics of the airplane will be completed. The airplane should be able to fly straight and level, maintain a given heading, and make turns to adjust to new headings.
2. The airplane should be able to autonomously take off and land.
3. The airplane should be able to navigate to a set of given map coordinates.

At the outset of the project, we set nine months and the relatively small sum of $1,000 as the time and budgetary constraints for stage one. In this paper, we describe what design choices these constraints forced, the hardware and software configurations of the resulting airplane, and the future goals for the project.

2. Platform design

2.1 Initial configuration

When we decided to build an aerial vehicle for human-to-robot skill transfer experiments, our first design decision required a choice between a fixed-wing aircraft or a helicopter design. (Bird-like, flapping-wing designs were eliminated at the outset as a possibility, due to their inherent complexity and our relatively poor understanding of the mechanics of such flight.) Each type of platform has advantages and disadvantages. On the one hand, a fixed-wing aircraft will tend to have a greater payload range, be more stable, and operate with fewer and simpler control mechanisms. Consequently, a fixed-wing aircraft will require a less expensive radio system and less skilled humans from whom to learn. A helicopter, on the other hand, can hover (useful for birds-eye photography or surveillance), and land and take off more easily in smaller areas. We ultimately decided that either design could adequately serve as an experimental platform. Therefore, cost and simplicity became overriding factors in leading us to choose a fixed-wing over a helicopter design.

One of the most critical decisions in the airplane design revolves around the type of power plant to be used — either an electric motor or an internal-combustion engine. As late as a few years back, this decision would not have required much thought, since only very recently have electric motors approached performance specs comparable to those of combustion engines. Nevertheless, each technology offers distinct advantages. Electric motors exhibit lower vibration, lower long-term maintenance costs, reduced electromagnetic interference (EMI) and noise, and do not require caustic fluids to operate. Internal combustion engines, on the other hand, offer better-tested technology, greater payload ranges and lower initial cost. Although this decision is not necessarily clear cut, we chose an electric power plant to avoid the design effort and weight that would have been required to shield the on-board electronics from the vibrations, EMI and fluids of an internal combustion engine.

Next, we needed to decide between brushed and brushless motors. Brushless motors available for our application are generally much more efficient than brushed motors, produce less EMI, and do not require the periodic rebuilding that brushed motors do, potentially lowering the expected long-term costs of the project. Once again, however, the full story is not so one-sided. Brushed motor are cheaper and allow the use of simpler motor-speed controllers, perhaps offsetting the potential cost savings of brushless motors. Ultimately though, our concern about electronic interference between the motors and the intelligent control hardware led us to select brushless motors for the airplane.

Third, we needed to select appropriate batteries for our project, without which the whole project would be academic. We considered three battery technologies: nickel-cadmium (NiCd), nickel-metal-hydride (NiMH), and lithium-ion (Li^+). While the newer technologies (i.e. NiMH and Li^+) offer much greater energy densities than NiCd batteries, they cannot generate the high currents (10-15C[1]) required for our motor. Therefore, NiCd batteries were the only option for powering the motor. For the electronics, however, the more advanced NiMH or Li^+ batteries are a better option, since the weight of the battery packs can be reduced significantly with the newer technology; consequently, we are currently pursuing low-cost (i.e. donated) NiMH or Li+ batteries for this purpose.

With the initial configuration in place, we now had to decide on an appropriate airframe. We concluded that the airframe should support a payload of at least 3.5 lb, a capacity which allows for future expansion (e.g. cameras, increased computing), and for large weight tolerances in the motor, electronics and batteries. We also concluded that the airplane must be able to stay aloft for at least 5 minutes at a time in order to allow for sufficient data collection and control experimentation. Finally, we wanted a stable, forgiving airframe. Given these parameters, we consulted with experienced electric model pilots to determine possible airframes that would meet both these design constraint and our cost requirements. Based on their advice, we settled on the *Sig Kadet Senior*, a radio-controlled (RC) airplane kit. Although this model is somewhat challenging to build with many parts needing to be cut and shaped, rather than being ready to assemble, it is well suited for our project, because of its relatively large wing area ($1150 in^2$), roomy fuselage and inherent stability.

---

1. C is the one-hour current capacity of the cell.

2.2 Drive train design

In designing the drive train for the airframe chosen above, we aim (1) to provide enough power to maintain flight, (2) to provide enough energy to fly for a given period of time, and (3) to minimize energy losses. Several more-or-less scientific methods exist for sizing motors and batteries to meet such criteria. In our calculations, we relied on a combination of the method outlined by Orme [7] and the instructions included with the software *MotoCalc* [8], a design tool that simplifies the repetitive calculations involved.

As the first step in the design process, we estimated the number of sub-C size NiCd cells necessary for flight. A useful rule-of-thumb [7] suggests that a large, slow, trainer-type model aircraft requires approximately one cell per 50in^2 of wing area, with a resultant wing loading of about $20\text{oz}/\text{ft}^2$. Thus, for 1150in^2 of wing area, we require about 24 cells (28.8V).

Given this approximate cell count, we then evaluated several motors based on each motor's voltage rating, where we explicitly preferred motors that had previously been flown by others in the *Kadet Senior* airframe. A useful tool for this evaluation is MotoCalc, which contains a database of common motors, motor controllers, batteries and airframes, and can compute power usage, efficiencies and flight envelopes for specific combinations of motors, controllers and airframes. In the preliminary design, we used 24 sub-C NiCd batteries, since we had not yet settled on specific batteries. That battery configuration is very similar to our current design of five 6V camcorder batteries (equivalent in voltage to approximately 25 sub-C cells). Although our specific airframe is not explicitly included in the MotoCalc database, we estimated the unknown aerodynamic properties of the *Kadet Senior* airframe from similar airframes that were contained in the database.

In using MotoCalc, we were first interested in determining the most efficient current for a given motor and number of cells. MotoCalc does this by computing the losses in the motor and wiring at various currents. For our eventual motor choice (Aveox 106/4 YSE), the peak motor efficiency with 25 cells occurs at approximately 18A at an efficiency of approximately 89%, as shown in Figure 1 below. Note, also, that a relatively high efficiency (greater than 88%) is maintained for this motor for currents between 14A and 25A), a typical characteristic of brushless motors. This wide range of high-efficiency currents allows us significant leeway in selecting the propeller, gearbox and cells.

At the desired current of 18A, MotoCalc predicts that the motor will turn at approximately 33,000 rpm. In order to reduce this rate to the normal prop-speed range of 8,000-12,000 rpm, we chose a gear ratio of 3.69:1, and then repeated our calculations with this gear box and a range of prop sizes. As can be observed from Figure 2, the two prop sizes[1] that come closest to the desired current of 18A, while still maintaining flight, are 11×7 and 12×6. Detailed performance predictions for these two prop sizes are given in Table 1 below. Note from the table that the 11×7 prop provides greater power for a somewhat shorter period of flight time.

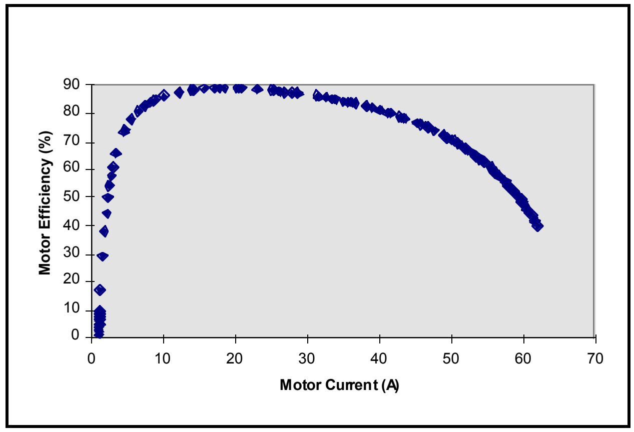

Fig. 1: Motor efficiency.

It is important to realize that the flight durations in Table 1 are upper limits and cannot be achieved in reality. They are based on level flight at a partial-throttle setting (in our case, approximately 89%), and therefore do not account for the energy required for take off and climb to altitude. Since the lower bound on the flight duration, given by a 100% throttle setting, is approximately 4 minutes for each prop, it is reasonable to assume that in practice we will be able to reach flight times of 5 to 10 minutes.

Over time, we have frequently revisited our performance estimates, as detailed component data has become available. Throughout, the design predictions have remained rather stable as the design has solidified.

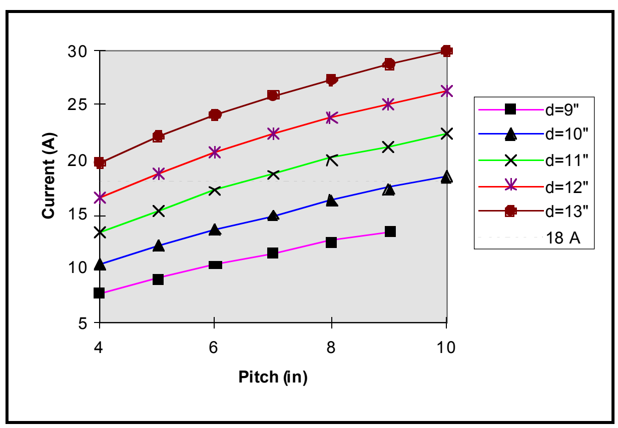

Fig. 2: Prop selection.

1. diameter(in) × pitch(in)

Table 1: Prop performance

Prop size	*Motor current (A)*	*Motor efficiency (%)*	*Prop speed (rpm)*	*Stall speed (mph)*	*Max. level speed (mph)*	*Rate of climb (fpm)*	*Duration (min)*
11×7	18.6	89.0	8,853	21	43	359	14:26
12×6	20.6	88.9	8,489	21	41	342	16:41

3. Intelligent system design

3.1 On-board sensors

Since we are designing this aerial platform for experiments in transferring human control strategies, it is important that our platform have adequate sensing and computing power. On an aerial platform, a careful balance must be struck between the amount of sensing on-board and the weight of the sensors. Too few sensors will not allow the plane to execute even the simplest maneuvers; too many sensors, on the other hand, may unduly burden the plane with excess weight. After some consideration, we anticipate that the following suite of sensors will be sufficient, yet not too heavy, for stage one maneuvers: (1) a two-axes tilt sensor, (2) an altimeter, (3) a compass, and (4) an air speed sensor. In other words, we expect that these sensors will be sufficient to maintain level flight at a given heading and to make turns to adjust to new headings (definition of stage one).

We have designed and are in the process of constructing an altimeter sensor around the Motorola ASB210 pressure sensor module. This sensor provides an analog voltage output proportional to the differential pressure between two ports. To convert this sensor to an altimeter, we will seal one port of the sensor before take-off, so that the differential pressure measurement will then be proportional to altitude. Based on the sensor specs, we anticipate approximate altitude resolution of about 7ft.

For heading detection, we intend to use the Honeywell HMR3000 Electronic Compass Module. This module provides an RS-232 output for heading, pitch, and roll, where the heading is electronically compensated for pitch and roll angles up to $45°$. Although we have also built a separate tilt sensor around the Analog Devices ADXL2020 dual-axes $\pm 2g$ accelerometer, we will most likely use the compass module's tilt information as our primary tilt sensor.

Finally, finding a cost-effective air speed sensor has proven to be our greatest challenge. We hope to employ another Motorola ASB210 differential pressure sensor with a pitot-static tube, but have been unable to find an appropriate tube. Another possibility is to use a turbine-type wind-speed sensor removed from a handheld unit.

3.2 On-board computing

We are currently planning to have two computers on-board, one to handle low-level I/O and coordination of sensor information, and a higher-level computer with sufficient processing to implement intelligent control behaviors.

Our primary computer will be a 386-class processor on a PC/104 board. It is small in size and weight (e.g. $3.6" \times 3.8"$ and less than 4oz.), consumes very little power, expands to hold a large amount of RAM (for storing human control data), and offers multiple serial ports to connect to sensors, the secondary processor and a host terminal. The PC/104 format allows for all the features found on a typical desktop computer and provides a standard bus for easy expansion, while the 386 processor is sufficiently powerful to implement many modern machine learning algorithms, but consumes relatively little power. In addition, we will add a hard drive to the basic setup in order to eventually implement computer vision algorithms, which often require significant data storage.

Our secondary computer will be a Motorola 68HC11 on a single-chip board that has been previously designed in the Machine Intelligence Laboratory (MIL) and that has been used in many of our robots as a key control component. Because of our prior experience with this board, existing HC11 code should be readily adaptable to our current application. Non-serial sensory data, as well as the radio inputs and servo outputs, will run through this board which we will link serially to the main processor through an RS-232 interface.

In order to safeguard the airplane against software errors, we have built a multiplexing system which will use one of the channels on the radio to switch between computer and human control. This circuit is designed fail-safe to allow a human supervisor pilot to assume control of the plane at any time, even in the event of a on-board computer failure. The control-switching circuit consists of a 2 MHz oscillator, a counter, an AND gate, and a multiplexer. The counter calculates the width of the control-channel by comparing it with the oscillator. A pulse of more than 1.5ms corresponds to pilot control, and a pulse of less than 1.5 ms corresponds to computer control. The resulting digital output is then used as the select signal for the multiplexer.

4. Software design

In previous work [1, 2, 6], we have successfully modeled human control strategies (HCS) in the driving domain. We have, for example, successfully learned extreme human driving behaviors in the driving simulator shown in Figure 3 below. We expect that much of that work will transfer readily to the flying domain.

Broadly speaking we can classify all control as either *continuous* or *discontinuous*. In the driving domain, for example, steering varies continuously with sensor inputs, while acceleration varies discontinuously with sensor inputs, because of the necessary switching between the gas and brake pedals. We have previously found that continuous human control can be modeled through *cascade neural networks*, which are powerful nonlinear function approximators that offer several advantages over more traditional neural network architectures: (1) the network architecture is not fixed prior to learning, but rather adapts as a function of learning [9]; (2) hidden units in the neural network can assume variable activation functions [10]; and (3) the weights in the neural network are trained through the fast-converging node-decoupled extended Kalman filter [10]. The flexibility of these cascade networks is ideal for HCS modeling, since few *a priori* assumptions are made about the underlying structure of the human controller.

Discontinuous human control can only be poorly approximated through a continuous learning formalism [2]. Therefore, we have developed a statistical, Markovian discontinuous learning architecture [6], that has successfully abstracted the switching control between the gas and brake pedals in human driving. Both the learning paradigms for continuous as well as discontinuous control may eventually

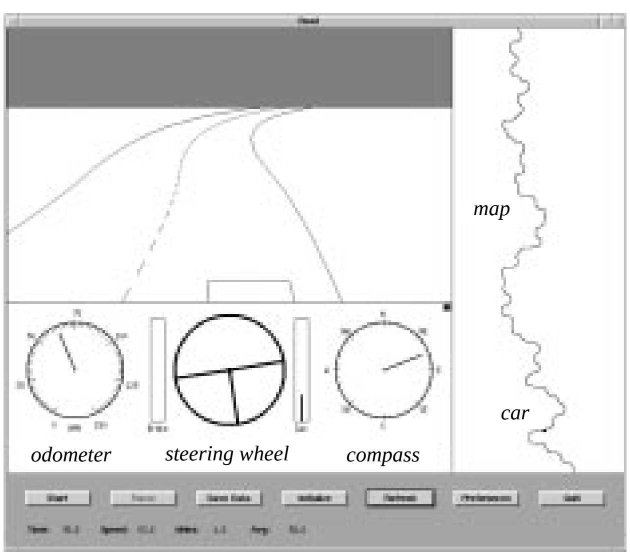

Fig. 3: Dynamic driving simulator used previously in modeling human control strategies. The user has independent controls of steering and acceleration.

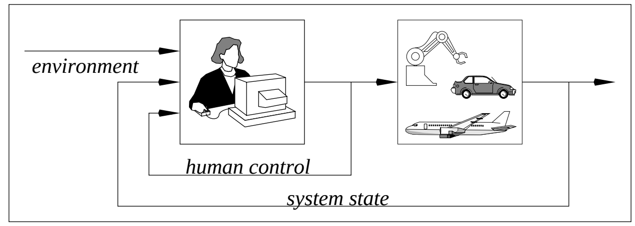

Fig. 4: In modeling human control strategy, we hope to replace the human control with a HCS model.

be required for fully developing an intelligent, autonomous airplane.

In general, we plan to proceed with the human modeling experiments as follows. The plane will first be flown completely under human control, while the computer collects and records sensor and control data. HCS models similar to those developed previously in the driving domain will then be trained off-line with the recorded human control data. The final HCS model(s) will then be downloaded on the on-board computers and the system as a whole will be tested in flight. Since the computationally intensive learning that is part of the human-to-robot skill transfer will take place off-line, the 386 computer is more than sufficient to execute any possible computed human model.

5. Conclusion and future work

This paper describes the first stage of development of an autonomous airplane. At the time of this writing, we have completed much of the hardware design, with the airspeed indicator being the only significant remaining design issue. Final assembly of the platform awaits arrival of all of the parts, since some modification of the airframe may be necessary to provide access to all of the components while maintaining proper balance. The multiplexing safety system is near completion, and the development of the HC11 software has begun. We expect to complete stage one by the end of May.

Acknowledgments

We would like to acknowledge the support of the following people and companies, without whom this project would not have been possible: (1) the members of the Machine Intelligence Laboratory; (2) Motorola, a major sponsor of the MIL; (3) Honeywell, which provided the HMR3000 compass module; (4) Eveready, which provides rechargeable batteries to the MIL; and (5) Analog Devices, which provided the ADXL202 accelerometer chips.

References

[1] M. C. Nechyba and Y. Xu, "Human Control Strategy: Abstraction, Verification and Replication," *IEEE Control Systems Magazine*, vol. 17., no. 5, pp. 48-61, 1997.

[2] M. C. Nechyba, *Learning and Validation of Human Control Strategies*, Ph.D. thesis, Carnegie Mellon University, 1998.

[3] D. A. Pomerleau, “Neural Network Perception for Mobile Robot Guidance,” Ph.D. Thesis, School of Computer Science, Carnegie Mellon University, 1992.

[4] D. A. Pomerleau, “Reliability Estimation for Neural Network Based Autonomous Driving,” *Robotics and Autonomous Systems*, vol. 12, no. 3-4, pp. 113-9, 1994.

[5] A. Pentland and A. Liu, “Toward Augmented Control Systems,” *Proc. Intelligent Vehicles*, vol. 1, pp. 350-5, 1995.

[6] M. C. Nechyba and Y. Xu, “On Discontinuous Human Control Strategies,” *Proc. IEEE Int. Conference on Robotics and Automation*, vol. 3, pp. 2237-43, 1998.

[7] M. Orme, personal communication, October 26, 1998.

[8] *MotoCalc*, http://www.motocalc.com, February 28, 1999.

[9] S. E. Fahlman, L. D. Baker and J. A. Boyan, “The Cascade 2 Learning Architecture,” Technical Report, CMU-CS-TR-96-184, Carnegie Mellon University, 1996.

[10] M. C. Nechyba and Y. Xu, “Cascade Neural Networks with Node-Decoupled Extended Kalman Filtering,” *Proc. IEEE Int. Symp. on Computational Intelligence in Robotics and Automation*, vol. 1, pp. 214-9, 1997.