

# Correlating Kinematics & Control Inputs on Small-Scale Helicopters

Johnhenri R. Richardson<sup>1</sup> and Anouck R. Girard<sup>2</sup>  
*University of Michigan, Ann Arbor, MI, 48109*

We present a time-domain technique for correlating the control inputs given to hovering, small-scale helicopters with their corresponding accelerations. The resulting correlations give a simple plant model that can be used for proportional-integral-derivative (PID) controller gain tuning, helicopter trimming, and simulations to test low-level controllers. We have collected our data in the Autonomous Controls Environment (ACE), an indoor cooperative control test bed for small, unmanned rotorcraft and ground vehicles. The data collection utilizes pre-existing low-level controllers and the data logs automatically generated by those controllers, so no special system identification arrangements need to be made. We also present two applications of the plant model that we have implemented on the ACE test bed, namely PID gain tuning and automatic in-flight trimming.

## Nomenclature

$a_i(t)$	= acceleration in the direction corresponding to channel $i$
$c_i$	= proportionality constant relating acceleration to control voltage on channel $i$
$d_i$	= proportionality constant relating force to control voltage on channel $i$
$e_i(t)$	= position error in the direction corresponding to channel $i$
$E_i(s)$	= Laplace Transform of $e_i(t)$
$F_i(t)$	= net force in the direction corresponding to channel $i$
$i$	= controller channel corresponding to thrust, roll, pitch, or yaw
$k$	= generic PID gain
$k_{iP}$	= gain for proportional term of PID controller on channel $i$
$k_{iI}$	= gain for integral term of PID controller on channel $i$
$k_{iD}$	= gain for derivative term of PID controller on channel $i$
$L_i(s)$	= open loop transfer function for helicopter and PID controller
$R^2$	= correlation coefficient for line fits
$t_d$	= time delay between control voltage $V_i(t)$ and helicopter kinematics $a_i(t)$ or $F_i(t)$
$T_i(s)$	= closed loop transfer function for helicopter and PID controller
$V_{0i}$	= reference voltage for trimmed flight on channel $i$
$V_i(t)$	= control input on channel $i$ expressed as a voltage set in the RC helicopter's transmitter
$V_i(s)$	= Laplace Transform of $V_i(t)$
$V_{trim,i}$	= value of $V_i(t)$ when there is no acceleration
$x$	= Cartesian body frame coordinate corresponding to the pitch direction
$y$	= Cartesian body frame coordinate corresponding to the roll direction
$Y_i(s)$	= Laplace Transform of the output on channel $i$ , namely a yaw angle or position
$Y_{di}$	= Laplace Transform of the desired output on channel $i$
$z$	= Cartesian body frame coordinate corresponding to the thrust direction

<sup>1</sup> Graduate Student, Applied Physics, 450 Church St., Ann Arbor, MI, Student Member.

<sup>2</sup> Assistant Professor, Aerospace Engineering, 1320 Beal Ave, Ann Arbor, MI, Member.

## I. Introduction

The University of Michigan Aerospace Robotics and Controls Laboratory (ARCLab) maintains an extensive indoor test bed to develop cooperative control algorithms for small rotorcraft and unmanned ground vehicles. In addition to testing cooperative control algorithms, the group continually upgrades its low-level control algorithms for individual vehicles, especially for the radio-controlled helicopters. These efforts aim at improving flight performance, adding capabilities, and addressing the needs of higher level control algorithms under development. The project described herein is an attempt at a simple, time-domain system identification technique supporting the improvement of the helicopters' low-level controllers.

### A. Motivation

The ARCLab test bed uses a low level controller for helicopter flight that is based on the proportional-integral-derivative (PID) technique. The gains used in this controller have until now been manually tuned. Additionally, the lab does not have an environment that simulates low-level control of its small rotorcraft, so any helicopter control algorithms must be tested on the actual vehicles. While this ensures that control algorithms developed by the lab work in-flight, it also leads to excessive wear-and-tear on the aircraft as new algorithms are polished into finished products. With these considerations in mind, the group sought techniques to analytically determine PID gains and develop a low-level control simulation for the rotorcraft. In particular, we wanted to try to use existing flight data to identify the system rather than set up and conduct focused system identification experiments. For every flight the group's low-level helicopter controller generates a log that records time stamped position, attitude, waypoint, and control input data. We developed the techniques below to make use of the information that we already had available.

### B. Prior Work & Present Contribution

Other researchers have done extensive work modeling and identifying aircraft. For instance, detailed models of small, coaxial helicopters have been presented.<sup>1</sup> Numerous researchers have conducted system identification experiments on small-scale rotorcraft.<sup>2,3,4</sup> In particular, Ref. 5 provides a very detailed and thorough procedure for identifying miniature rotorcraft, including design considerations. Such techniques require dedicated system identification flight tests and, for some of the cited experiments, a special setup that limits the helicopter's motion. Similarly, the references had to design careful control inputs for the aircraft during the initial data collection, whether flown by hand or by computer. While the references give impressive results, we instead sought identification algorithms that could utilize the working setup we already have and the logs that we already generate during every flight. As such, the methods below required no more than a normal, automated flight of our helicopter where the aircraft hovered at a specified location and the controller logged inputs and kinematic data as usual.

### C. Paper Organization

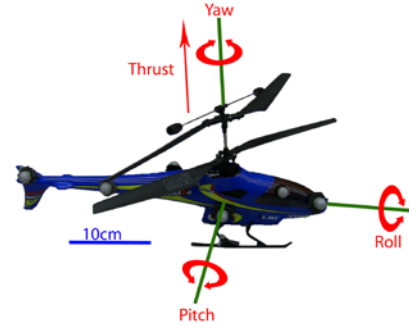
We will begin the paper by describing the test bed setup and data collection. We will then describe the fitting techniques that we used to develop correlations between the helicopter kinematics and the control inputs. Finally, we will discuss two applications of the curve fitting: tuning PID gains and tuning the control inputs' reference values. The applications will go over both the analytical framework used and some experimental results.

## II. Experimental Setup & Data Collection

### A. System Architecture & Low-Level Control

The Autonomous Control Environment (ACE) is an indoor test bed that through a central database networks a Vicon motion capture system and unmanned vehicles with high level controllers and graphical user interfaces (GUI). A detailed description can be found in Ref. 6. The Vicon system measures position and attitude in real time by detecting reflections off of small markers affixed to the vehicles. Low-level controllers receive vehicle state measurements from the Vicon system as well as waypoints and commands from the database. High-level controllers and GUIs can access the same Vicon state information and use it to update waypoints and commands in the database. In this manner the low-level control system has been separated from high-level algorithms. Any computer can connect to the database to issue commands and waypoints to the low-level controller.

ARCLab currently flies Esky LAMA V4 helicopters, as seen in Fig. 1. In the past we have also used LAMA V3 and Co-Comanche helicopters, and the techniques below were originally developed for the latter. These are all inexpensive, coaxial, 4-channel, 7.2 MHz or 2.4 GHz, radio-controlled (RC) helicopters. The LAMA V4 weighs approximately 230 g, is 408 mm long, and can fly for about 7 minutes on a 7.4 V 800 mAh Li-polymer battery. The low-level controller interfaces with the RC helicopters through a serial port and a commercially available Esky RC helicopter hand-held transmitter. The transmitters have been rewired so that the joysticks are bypassed; instead of the joysticks setting a voltage in the controller, the serial port sets the same voltage directly. This rewiring has been done on all four channels, so the computer can issue pitch, roll, yaw, and thrust commands without the help of a human operator's thumbs. The radio communication between the transmitters and the helicopters has not been altered.



**Figure 1. Esky LAMA V4 Helicopter.** An Esky LAMA V4 helicopter used in the ARCLab. Note the attached reflective markers tracked by the Vicon camera system.

As we mentioned earlier, the low-level algorithm uses a classical PID controller. For each channel  $i$ , corresponding to roll, pitch, yaw, and thrust, the control voltage is calculated as:

$$V_i(t) = k_{iP}e_i(t) + k_{iD}\frac{de_i}{dt} + k_{iI}\int e_i dt + V_{0i} \quad (1)$$

where  $V_i(t)$  is the voltage set in the transmitter on channel  $i$ ,  $e_i(t)$  is the position error in the reference frame, and the  $k$  values are gains that depend on the channel and are scheduled based on the helicopter's position error.  $V_{0i}$  is reference voltage for trimmed flight. The voltages on each channel range from 0 V to 5 V with a resolution of 0.1 V. The  $k$  and  $V_{0i}$  values have been manually tuned and in Section III we will describe techniques to try to calculate them from the model.

## B. Data Collection

Before beginning this system identification project, our low-level controller already generated logs during every flight. The log recorded the 3-dimensional position and attitude, control inputs on all four channels, and waypoints for each aircraft at a user-specified time interval. Additionally, our controller could keep a helicopter hovering with a mean position within 5 cm of a desired waypoint and could stably move between waypoints. We designed our data collection to take advantage of these pre-existing capabilities. We conducted experiments where, using a GUI connected to the database as described earlier, we commanded the helicopter to hover at various waypoints for roughly 10s each. The controller would generate a flight log in spreadsheet format. In this manner we collected flight data from more than 20 hovering events spanning several flights of multiple helicopters at different altitudes. For this project we focused exclusively on hovering, but using the same techniques we could consider other phases of flight.

## III. Correlations

Newton's second law and a simple model of the forces of flight motivated our attempts at correlating the control inputs and helicopter kinematics. We assume that when hovering, the control input voltage set by the serial port in the transmitter is proportional to the net force in the corresponding body frame direction. In other words, we assume that the voltage set on the thrust channel of the transmitter relates linearly to the vertical component of the net force on the helicopter, the  $z$ -component, that the voltage set on the pitch channel relates linearly to the longitudinal component of the net force, the  $x$ -component, and that the voltage set on the roll channel relates linearly to the lateral component of the net force, the  $y$ -component. As an equation, we can say that for channel  $i$  corresponding to roll, pitch, and thrust:

$$V_i(t) = d_i \cdot F_i(t + t_d) + V_{trim,i} \quad (2)$$

where  $V_i(t)$  is again the control voltage on channel  $i$  ranging from 0 V to 5 V,  $F_i(t)$  is the net force in the direction corresponding to channel  $i$ ,  $d_i$  is a proportionality constant with units V/N, and  $t_d$  represents a time delay between the control input and the helicopter's reaction.  $V_{trim,i}$  is, by definition, the value of  $V_i(t)$  when the net force goes to zero.

We assume that  $V_{trim,i}$  is constant and we will see later that it is a good candidate to use as the reference value  $V_{0i}$  described in the PID controller.

Note that on the thrust channel,  $F_i$  is not the thrust but rather the difference between the thrust and the weight. The portion of the thrust that counteracts the helicopter's weight, which should be constant on a battery-powered helicopter, is absorbed into the constant  $V_{trim,thrust}$ . Also note that our low-level controller assumes level flight, namely that the roll and pitch angles remain negligibly small. In making this assumption, the controller does not take into account any coupling between inputs on the four different channels. Notwithstanding movements of the helicopter's swashplate, this level flight assumption is reasonable when hovering because the standard deviation of the roll and pitch angles in our setup is around  $1^\circ$ - $2^\circ$ . Finally, note that we attempted to correlate both the first and second time derivatives of the yaw angle with the yaw control input, but we could not generate scatter plots with anything resembling a linear trend. Therefore we limit our discussion to the thrust, pitch, and roll channels.

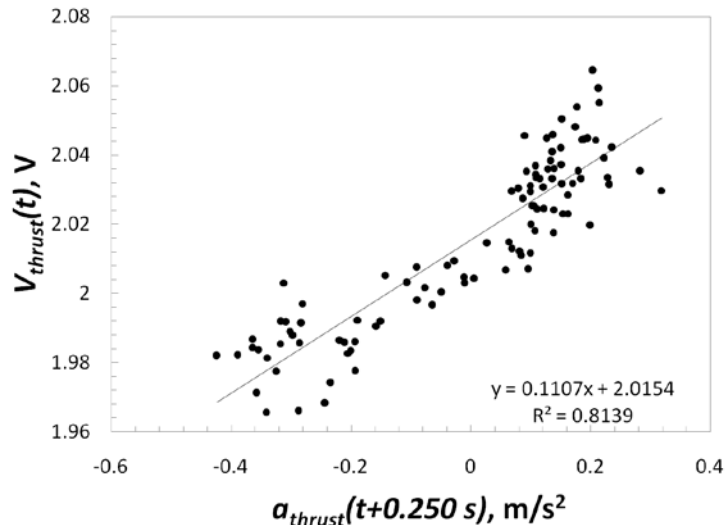
With our first order approximation in mind, we then consider Newton's second law, that the net force is proportional to acceleration, so each component of our thrust should be proportional to the corresponding accelerations. Putting the two together we get a linear relationship between the accelerations and control voltages. For each channel  $i$  corresponding to roll, pitch, and thrust, the control voltage on that channel relates to the acceleration in the corresponding direction as follows:

$$V_i(t) = c_i \cdot a_i(t + t_d) + V_{trim,i} \quad (3)$$

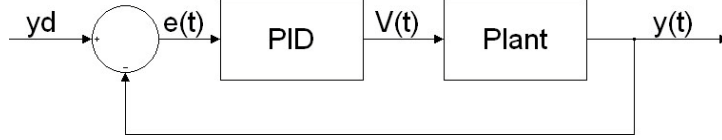
where  $a_i(t)$  is the acceleration in the direction corresponding to channel  $i$ ,  $V_{trim,i}$  is from now on defined as the value of  $V_i(t)$  when there is no acceleration, and  $c_i$  is another proportionality constant, essentially  $d_i$  multiplied by the mass of the helicopter.

Equation 3 shows us that identifying the portions of interest of our system is simply a matter of line fitting  $V_i(t)$  versus  $a_i(t+t_d)$ . Using the data collected on the ACE test bed we fitted lines onto plots of control voltage versus acceleration with different time delays. We used the line fit's correlation coefficient ( $R^2$ ) as a measure of the accuracy of the fit. We varied  $t_d$  for all three channels during every hovering event. Because the controller generated log entries at 0.125 s intervals we could only try time delays that were multiples of that interval. We found that a  $t_d$  of 0.375 s maximized  $R^2$  about half the time and a  $t_d$  of 0.250 s maximized  $R^2$  about 40% of the time. We subsequently reduced the controller's data collection interval to 0.0625 s and found that a time delay of 0.3125 s maximized  $R^2$  for almost every data set.

Once we had picked the best time delays for each data set, we observed that the  $R^2$  values varied greatly from hover-to-hover and channel-to-channel. In general the thrust channel gave the best line fits, with a median  $R^2$  of 75%, while the roll channel gave the weakest fits, with a median  $R^2$  of 21%. Figure 2 shows an example of one of these plots. While the fact that we could correlate voltages with accelerations showed promise for our analytical framework, the fit parameters  $c_i$  varied substantially. For example,  $c_{thrust}$  ranged from 0.062 to 0.115  $\text{Vs}^2/\text{m}$  while  $c_{roll}$  ranged from -0.149 to -0.357  $\text{Vs}^2/\text{m}$ . This likely means that our first order approximations do not sufficiently describe the system, or that we didn't adequately control other experimental variables while collecting data. For example, flying at different altitudes means the helicopter experienced different amounts of ground effect. Additionally, perhaps each LAMA V4 helicopter has different  $c_i$  parameters, or after repairs perhaps the same helicopter will have different values of  $c_i$ . Similarly, perhaps  $c_i$  depends on the battery charge. Lastly, we noted that the accelerations tended to be smoother curves than the control voltages, adding an additional complication to the fits.



**Figure 2. Acceleration vs. Thrust Voltage.** Plot of the thrust control input against the acceleration along the z-axis with a 0.250 s time delay. The trend line, as well as its equation and  $R^2$ , are shown.



**Figure 3. Block Diagram of the low-level controller and helicopter.**

Nevertheless, despite the varying values of  $R^2$  and  $c_i$ ,  $V_{trim,i}$  remained relatively constant on each channel from flight-to-flight. For example  $V_{trim,thrust}$  ranged from 1.98 V to 2.02 V and  $V_{trim,roll}$  had the largest range from 2.32 V to 2.58 V. The consistent values of  $V_{trim,i}$  suggest that it is the most useful of our calculated parameters to employ in low-level controllers.

## IV. Applications of the Model

### A. PID Gain Tuning

With a simple model constructed for how the helicopter responds to control inputs, we first attempted to design appropriate PID gains for hovering. We had fitted lines onto scatter plots of  $V_i(t)$  versus  $a_i(t+t_d)$  to find the parameters in Eq. 3. We were faced with the problem of having different, in some cases very different, values of the line fit parameters from flight-to-flight. In an attempt to generalize what the parameters  $c_i$  and  $V_{trim,i}$  should be, we computed weighted averages for each parameter on each channel using the  $R^2$  values of the line fits as the weights. That way the best line fits would get the most credence in the generalized values but we would still consider the full set.

With the line fitting complete, we computed the closed-loop transfer function  $Y/Y_d$  for our plant and PID controller. Figure 3 shows a diagram of the system. Ignoring the time delay and reference voltages  $V_{0i}$  and  $V_{trim,i}$ , taking the Laplace transform of Eqs. 1 and 3 yields:

$$V_i(s) = k_{iP}E_i(s) + k_{iD}sE_i(s) + \frac{k_{iI}E_i(s)}{s} \quad (4)$$

$$V_i(s) = c_i s^2 Y_i(s) \quad (5)$$

where the error is defined as

$$E_i(s) = Y_{di} - Y_i(s) \quad (6)$$

$Y_i(s)$  is the output on channel  $i$  and  $Y_{di}$  is the desired output on channel  $i$ . So for a channel  $i$  corresponding to roll, pitch, thrust, and yaw the closed loop transfer function  $T_i(s)$  is:

$$T_i(s) = \frac{L_i(s)}{1 + L_i(s)} \quad (7)$$

$$L_i(s) = \frac{k_{iP} + k_{iD}s + \frac{k_{iI}}{s}}{c_i s^2} \quad (8)$$

In this transfer function  $c_i$  is known on each channel and we must choose the various  $k$  values. We tried several techniques for tuning the gains in this transfer function, including a Ziegler-Nichols based approach and a root locus based approach. In the root locus approach we started with the manually tuned values of  $k_{iP}$  and proceeded to place transfer function poles by choosing  $k_{iD}$  and  $k_{iI}$ . We chose  $k_{iD}$  by setting  $k_{iI}$  to zero and rearranging  $T_i(s)$  as follows:

$$T_i(s) = \frac{\frac{k_{iP} + k_{iD}s}{k_{iP} + c_i s^2}}{1 + \frac{k_{iD}s}{k_{iP} + c_i s^2}} \quad (9)$$

Now  $k_{iD}$  appears in the denominator as the equivalent of a forward path gain. We can use the denominator of this version of  $T_i(s)$  for root locus plots that vary  $k_{iD}$ .<sup>7</sup> We typically chose the value of  $k_{iD}$  that yielded critical damping.

With a value for  $k_{iD}$  set we considered  $k_{iI}$ . No longer ignoring  $k_{iI}$ , we rearranged  $T_i(s)$  again to get the equation

$$T_i(s) = \frac{\frac{k_{iI} + k_{iP}s + k_{iD}s^2}{k_{iP}s + k_{iD}s^2 + c_i s^3}}{1 + \frac{k_{iI}}{k_{iP}s + k_{iD}s^2 + c_i s^3}} \quad (10)$$

Here  $k_{iI}$ 's position in the denominator lends itself to root locus. We typically chose the largest value of  $k_{iI}$  for which the poles remained real. As an alternative approach, we also tried choosing  $k_{iD}$  to place the poles a little more than 45° off of the imaginary axis and then chose  $k_{iI}$  to place the poles exactly 45° off of the imaginary axis. We set the resulting  $k_{iD}$  and  $k_{iI}$  values in the hovering schedule of our low-level controller.

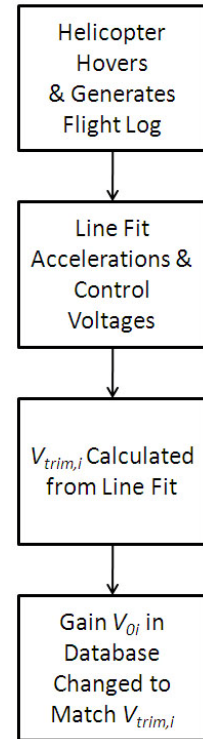
In all cases the resulting gains were stable according to the model and unstable when used in-flight. In particular, the values of  $k_{iD}$  and  $k_{iI}$  were larger relative to  $k_{iP}$  in the analytical results than they were when manually tuned. In our experiments, the helicopter would move between waypoints as designed using the manually tuned traveling gain schedules. However, when the helicopter got very close to a waypoint it would switch to the analytically determined hovering gains, which would generate unreasonably large control inputs and kick the helicopter away from its desired waypoint. This result leaves room for improvement either in our model or in our methods of placing the closed loop pole to give better analytically determined gains.

## B. Automatic Trimming

In Eq. 3 the voltage  $V_{trim,i}$  is the voltage that corresponds to zero acceleration in the flight logs. By definition that makes it a voltage corresponding to trimmed flight. We also noted that the line fits described earlier vary from flight-to-flight. With this in mind, we decided that instead of generalizing the line fit parameters we should consider using this system identification technique in-flight to generate a plant model specific to a particular helicopter under particular operating conditions. We ultimately developed an algorithm that could, in flight, automatically conduct the line fitting described earlier and replace the manually determined reference voltages  $V_{oi}$  with the voltages  $V_{trim,i}$  on the roll, pitch, and thrust channels. Unfortunately, as mentioned earlier, we still have not succeeded in correlating yaw control inputs and kinematics, so we cannot yet automatically trim the yaw channel.

This algorithm utilizes the flexibility of the ACE environment; it runs in MATLAB while the helicopters' low-level controller, a separate application, runs in the background. The automatic trimming algorithm gathers the position and control information from the low-level controller's flight log, performs the line fitting on a hovering event as described in Section III, and resets the  $V_{oi}$  values in the ACE database all while the helicopter is in the air. The low-level controller regularly checks the database for new gain values and begins using the new reference voltages immediately. Figure 4 summarizes the procedure.

Results from automatic trimming experiments show some improvements in flight performance. For one, when an experiment is run repeatedly with the same helicopter under similar conditions, the algorithm very consistently trims the control surfaces in the same direction from flight-to-flight. Where automatic trimming seems to have the biggest impact is on the helicopter's acceleration. In general, after trimming the helicopter its average acceleration and the acceleration's standard deviation both tend to be smaller. In one set of trials where the helicopter was trimmed on all three channels simultaneously the mean



**Figure 4. Automatic Trimming Flow Chart.** Sequence of events in the automatic trimming algorithm.

longitudinal and lateral accelerations were reduced on all channels two thirds of the time, averaging a 50% reduction. In the same trials the standard deviation of the vertical acceleration was reduced two thirds of the time, averaging a 10% reduction. This effect on the acceleration, especially the effect on the mean acceleration, is expected since the trimming process chooses a steady-state offset that corresponds to zero acceleration. Another interesting result observed was that trimming only the pitch and roll channels made consistent improvements to the acceleration in the vertical direction. In a different set of trials, trimming only the roll and pitch channels reduced the mean vertical acceleration two thirds of the time and reduced the standard deviation of the vertical acceleration every time, averaging just over a 20% reduction in both cases. This suggests that the coupling between the control inputs may be substantial, even when hovering. Accounting for coupling either in the model or in the controller may be a good way to improve the techniques described in this paper.

## V. Conclusion

We have presented a time-domain technique for correlating control inputs and accelerations on small RC helicopters. We explained two applications we have tried using the resulting plant model. We succeeded in developing correlations between the control inputs and corresponding accelerations on the roll, pitch, and thrust channels, with the strongest correlations on the thrust channel. However the line fits, and especially the slopes of the line fits, vary considerably from flight-to-flight. We used the results of the line fits to try to generalize what our PID gains should be, but the calculated gains were unstable when implemented on our test bed. We also developed an algorithm that can perform the line fits automatically in-flight. We have used this algorithm to trim the helicopters automatically, which can effectively smooth the helicopters' acceleration. We feel that the model proposed should be reconsidered to account for the variability of the slopes of the line fits, but that the trim voltages generated by the current framework hold promise for future low-level control development.

## Acknowledgments

This work was supported in part by the Department of Defense under Air Force grant number FA 8650-07-2-3744 and through the National Defense Science & Engineering Graduate Fellowship Program. The authors would also like to acknowledge Jonathan White, an officer in the US Coast Guard and recent graduate of the University of Michigan, for his contributions to this work.

## References

- <sup>1</sup>Castillo, P., Lozano, R., and Dzul, A.E., *Modeling and Control of Mini-Flying Machines*, Springer-Verlag, London, 2005, Chap. 5.
- <sup>2</sup>Morris, J.C., van Nieuwstadt, M., and Bendotti, P., "Identification and Control of a Model Helicopter in Hover," *Proceedings of the American Control Conference*, American Automatic Control Council, Baltimore, MD, 1994.
- <sup>3</sup>Shim, D.H., Kim, H.J., and Sastry, S., "Control System Design for Rotorcraft-based Unmanned Aerial Vehicles Using Time-domain System Identification," *IEEE International Conference on Control Applications*, IEEE, Anchorage, AK, 2000.
- <sup>4</sup>Kim, S.K., Tilbury, D.M., "Mathematical Modeling and Experimental Identification of an Unmanned Helicopter Robot with Flybar Dynamics," *Journal of Robotic Systems*, Vol. 21, No. 3, 2004, pp. 95-116.
- <sup>5</sup>Mettler, Bernard, *Identification Modeling and Characteristics of Miniature Rotorcraft*, Kluwer Academic Publishers, Boston, 2003.
- <sup>6</sup>Hasan, Z., Faied, M., Klesh, A., and Girard, A., "Distributed Heterogeneous Control of Mini-Flying Machines and Ground Robots," *AIAA Guidance, Navigation, and Control Conference*, AIAA, Chicago, IL, 2009.
- <sup>7</sup>Nise, Norman S., *Control Systems Engineering*, 3<sup>rd</sup> ed., John Wiley & Sons, Inc., New York, 2000, Chap. 8.