Autonomous Blimp

1/c Morris, 1/c Ulloa, 1/c White

Project Advisor: LCDR Godfrey

Projects in Electrical & Computer Engineering
United States Coast Guard Academy
28 April 2006

ABSTRACT

The goal of our senior design project was to design and implement an autonomous control system for an indoor blimp. The Coast Guard Academy Electrical and Computer Engineering section is sponsoring this effort to develop a model Unmanned Aerial Vehicle (UAV) to study the development and implementation of this technology. The information gained from this study will be used to further knowledge of UAV applications in the Coast Guard.

The 2005 fall semester was dedicated to developing three critical blimp systems - wireless communication, movement sensing, and manual computer control. In order to begin designing our digital controller in the spring semester, the group completed these three critical blimp systems during the fall semester. We set up an analog output card to send voltages from our computer into the blimp's remote controller circuit board, allowing users to control the blimp's movement via a computer. In order to design a digital controller we needed to measure the blimp's movement. Movement was measured using a µNav® MNAV100CA digital sensor package which detects acceleration in the x, y, and z directions, as well as angular acceleration.

All communication with the blimp was made wireless using two frequency modulated transceivers and a Bluetooth wireless transmission device. Transmitters were secured inside the blimp's gondola, and receivers attached to the computer. The group wanted to wirelessly send the blimp's acceleration in the x, y, and z directions as well as roll, pitch, and yaw from the sensor package to a computer interface. This was accomplished by implementing the sensor package in conjunction with two transceivers

During the spring semester, we examined new methods to receive positioning and movement information from the blimp to supplement the sensor data. This was accomplished by implementing a heading and acoustic sensors on the blimp. Next we developed system identification equations with the help of mechanical engineering cadets. We applied these equations to create a working digital controller for the blimp at the end of the semester.

INTRODUCTION

Unmanned Aerial Vehicles (UAV's) are, as the name implies, craft that perform aerial operations without the aid of an onboard pilot. UAV's provide for more efficient and cost effective operations without jeopardizing human life. Removing human operators results in less fuel consumption, and therefore UAVs can be deployed far longer than any manned vehicle- up to 80% longer [1]. The Coast Guard plans on implementing operational UAVs in 2006 [1].

The US Coast Guard (USCG) plans on using UAV technology as part of the Integrated Deepwater System [1]. Depicted in Figure 1, is the Eagle Eye, a UAV currently under development by the United States Coast Guard and the Bell Helicopter Company [1]. The Coast Guard plans on having the Eagle Eye, a vertical take-off and landing UAV, deployable from the new deepwater national security cutters and offshore patrol cutters [1].

Figure 1 - Eagle Eye unmanned aerial vehicle

Deepwater's goal is to optimize operations including search and rescue (SAR) and drug interdiction. Further

development would allow application of this technology in other Coast Guard missions as well. Rear Admiral Patrick M. Stillman, Deepwater Program Executive, stated that "the Integrated Deepwater System [IDS] and its employment of unmanned aerial vehicles are at the center of gravity of our solution to provide this nation's maritime security" [2].

The Coast Guard Academy Electrical and Computer Engineering section is sponsoring an effort to develop a model UAV to study the development and implementation of this technology. The goal of the project is to build a controller that will automate the blimp's movement, as predetermined by the user. The model used during this phase (academic year 2005-2006) of the project is an 11-foot blimp with four radio-controlled servo motors.

The system design process is divided into three stages. The first stage is developing the system identification for the blimp. System identification consists of recording the blimp's response to movement and commands from the computer or radio controlled (RC) controller. The data acquired from the system identification can then be used to derive the system equations. The system equations describe the blimp's response to forces exerted on the blimp, and are vital in the design of the system's digital controller.

The second stage is developing a sensor array which collects data from the blimp and transmits this data to the digital controller's feedback loop. Specifically, the sensor package determines the acceleration, velocity, and displacement of the blimp in three dimensions, as well as sensing changes in bearing, roll, and pitch. The acoustic sensors gather ranging information which can then be used to determine the blimp's position relative to the room it is in. The heading sensor provides the blimp's current heading using a gyrocompass. All of this information is then sent wirelessly to the computer so that the controller can determine what output commands should be, given the current and past movement of the blimp. Feedback will allow the controller to determine how the blimp will need to respond - the controller's output is subtracted from the next input, and the controller determines what adjustments need to be made.

The third stage is to create a digital controller. The digital controller will be developed by applying system equations calculated by the project team. Once the system equations are known, PD and state space controller models will be tested until an optimal controller is designed. The digital controller will output commands to the blimp based upon input data from previous blimp movement. The code for the digital controller will be written in MATLAB®, and will interface directly with the blimp via wireless transceivers.

This report will summarize the design and desired outcome of the project. The group intends to produce and implement a working digital control system for the blimp. The project will entail an analysis of the system identification of the blimp and the constraints encountered. Furthermore, we present an explanation of the data collection process, as well as the communication system from the computer to the blimp. The Digital Controller section describes the operation of the controller and the reasoning behind choosing a state-space controller.

OBJECTIVES

The overall goal of this project is to successfully design a controller for an indoor model of a UAV which can be applied to the needs of the Coast Guard. The model we are using is an 11 foot long, 3-foot diameter, helium-filled, indoor blimp (figure 2).

Figure 2 - The project blimp

Developing and implementing UAV technology to fulfill the Coast Guard's missions is an elaborate process. We hope developing a digital controller for a UAV system will allow us to understand the technical challenges faced by designers. We will then attempt to solve such problems in the design of our UAV. Before we can fly the blimp autonomously, there are a number of other objectives which must first be accomplished. The group's project plan provides more detailed information on the project's goals, purpose, and technical challenges (Appendix A).

Motors and RC Controller

An objective early in the fall semester was to understand how the blimp's motors operate in conjunction with the Futaba® Skysport 6 RC controller. The blimp's power supply and radio frequency (RF) receiver are located inside of a detachable gondola shown below in Figure 3. Also shown in this figure are the two propellers located on the outside of the gondola.

Two fans powered by electric motors are attached to an axel which can be rotated 180º in either direction using the RC controller depicted in Figure 4. One may also command forward or reverse thrust, while also determining the rotational angle of the fans to move the blimp up, down, forwards, and backwards.

Figure 3- Head on view of gondola with rotational motors

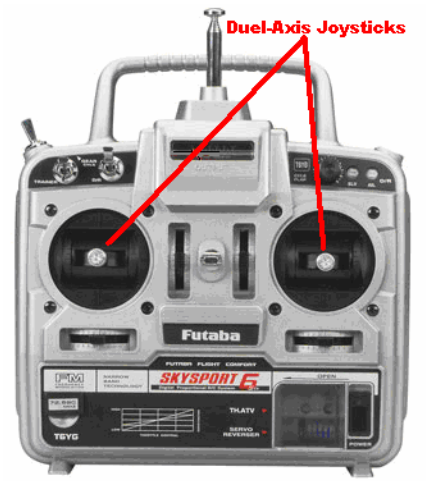

Figure 4- Futaba® Skysport 6 Controller

Another electric motor powers the blimp's tail fan, which is depicted in Figure 5. The direction of thrust on this fan allows the blimp to rotate in a clockwise or counter-clockwise direction about the horizontal plane.

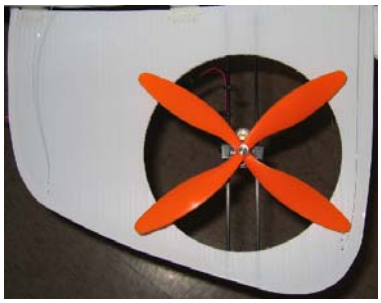

Figure 5– Blimp's tailfin

The Futaba® Skysport 6 controller's primary interfaces are two duel-axis joysticks located at the center of the controller. These joysticks control the thrust of the engines, as well as their orientation, and are used to drive the blimp.

Digital Controller

Our digital controller will take inputs from sensors on the blimp and decide what combination of tailfin and gondola motor angles and motor thrusts are needed to make the blimp move as desired. This is accomplished by determining the system equations, which describe the blimp's movement when thrust is applied with the motors. In order to determine these equations, we must analyze the blimp's response to thrust commands. Once determined, the system equations will be used to develop the digital controller. If successful, our controller will have the ability to calculate the blimp's response to different levels of thrust.

The blimp must be able to find its heading to within ±5 degrees, its velocity to within ±0.1 mph, and its height to within ±2 inches. These specifications are not requirements from the project sponsor; rather they are reasonable specifications as determined by the group based on what we think we can accomplish by the end of the spring semester. Further information regarding project requirements are found in Appendix B, our requirements specification document. We proposed these specifications based upon what the project group hoped to accomplish in terms of the controller's capabilities.

A constraint for our design is to stay within the weight limits of the blimp. The blimp can hold a total of fourteen ounces extra weight. This means that we can not use large sensors or put the laptop directly onto the blimp itself.

The end result of the project will be a wirelessly operated blimp with sensors that can communicate to a computer. The computer will then perform the digital controller function, sending commands back to the blimp, the blimp will then respond to these commands. The group plans on having a fully functional digital controller operational by the completion of our project in May as reflected in our project plan (Appendix A).

Sensor Package

The next objective is to send positional and movement data from the blimp to the personal computer (PC) so that they can be integrated into the digital controller. These data will come from a sensor package installed on the blimp.

The sensor package needs to be able to measure the blimp's change in height, heading, and acceleration because they are important inputs to the controller. These data will then be transmitted and sent into the

controller which determines instructions to be sent to the blimp's electric motors. The primary motors on the sides of the gondola will be given angle and thrust commands to control forward, backward, upward, and downward movement. Simultaneously, the tailfin motor will be given commands to rotate the blimp in a clockwise or counterclockwise direction. The group selected and implemented a sensor package able to communicate wirelessly with MATLAB by the conclusion of the fall semester.

Communication

To send and receive information, we must establish communication between the blimp and PC. This will be accomplished by sending voltage values to the Futaba® control board which then sends the values to the blimp. The blimp's motors respond after receiving this signal from the Futaba® control board. The graphical user interface (GUI) which sends these values will be essential in sending voltage values so that the person operating the blimp can easily see the data that is being sent. The user will also be able to control the commands sent to the blimp with precision using the GUI, because specific amounts of thrust are able to be entered. The group created an operational user interface by the conclusion of the fall semester.

The second half of the communication objective is to wirelessly send information from the sensor package to the PC. Although not specified in the project requirements, making all blimp communications completely wireless became an objective specified by the group. This will ensure the final product is more user-friendly because users will not need to worry about long transmission lines affecting the blimp. We will use wireless transmission devices capable of interfacing with the sensor package, as well as the PC. We must also ensure that the wireless transmission devices do not operate on the same frequencies as other devices in the area. Specifically we must exercise caution if transmitting should not operate around the 2.4GHz range, because the wireless routers in the engineering department operate at this frequency. The group had two transceivers integrated to the project design by the conclusion of the fall semester, but replaced them with Bluetooth transceivers during the spring semester because of they offer more flexibility.

SYSTEM DESIGN

The system design for the blimp is broken down into four components as follows:

1) Communication
2) System identification
3) Sensor array
4) Acoustic sensors
5) Digital controller.

We broke the system into these four parts in order to be able to manage the complexity of designing and testing the system.

Before discussing the four components of our system design in detail, it should be noted that the group decided to use MATLAB® as our primary programming language. We contemplated using C++ because of its high portability and flexibility. However, the amount of knowledge of C++ known by the group members was mediocre, and it would take much more time to develop an effective GUI using C++. We determined that it would not be the best platform to use because it would be very difficult to create an integrated GUI based on our C++ experience, and most functions for what we need to control the blimp are already included with MATLAB®.

Communication

In order to communicate with the blimp we pursued three different ideas. We needed to take electric motor command data from MATLAB®, and find a way to convert these data into a pulse code modulated signal. This signal must then be sent to the receiver on the blimp, and decoded such that it operates the correct servos.

Our first idea was to make our own frequency modulator and use the serial output from our program to communicate to the blimp. This idea was rejected because it would take too much time to create our own modulator. A modulator may be incorporated in the future; however our first priority is to complete a digital controller for the blimp.

The second idea was to use the trainer cable connection on the Futaba® Skysport 6 controller. The purpose of the trainer cable is to allow a more experienced operator to override and take control of another person's controller. This is done through attaching two controllers together via a trainer cable. However, we found out that the trainer connection offers no direct manipulation of the six channels.

Therefore we decided that we needed to use the internal board of the Futaba® Skysport 6 controller. Inspecting the controller, we determined that the joysticks are connected to potentiometers which pass different voltages depending on the angle of the joystick. The voltage passed by each joystick results in RC output to the blimp's motors. If we input voltages directly into

the internal board, we could bypass the manual joystick control.

To send signals to the blimp we decided to input specific voltages into the circuit board of the Futaba® Skysport 6 controller. Specifically, we need a device compatible with MATLAB® which can output adjustable analog voltages to specified nodes on the controller's circuit board. Our solution to this problem was the 'PC-CARD-DAC08' analog output PC card from Measurement Computing™, shown below in Figure 6.

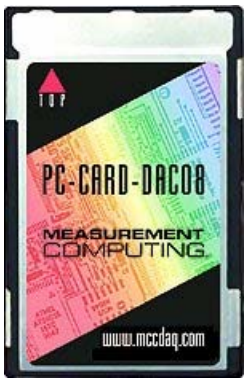

Figure 6 – PC-CARD-DAC08 - Analog output PC card

By dissecting the controller and exposing the circuit board, we determined that potentiometers attached to the joysticks change voltage signals within the controller. These signals are then transmitted from the controller's antenna, and received by the blimp. The intact controller before dissection is shown in Figure 4. Figure 7 below shows the internal components of the Futaba® Skysport 6 controller.

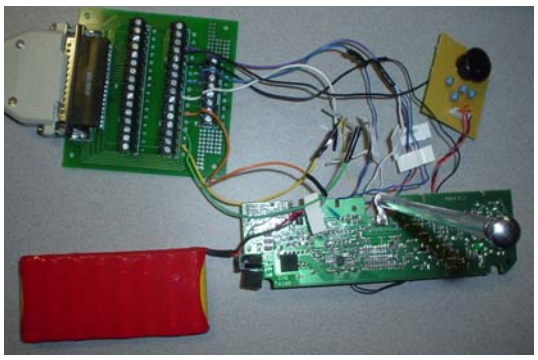

Figure 7 - Measurement Computing Output (top left) with Futaba 6 Controller internal board (bottom right)

Replacing the joysticks with a Measurement Computing™ analog output board will allow us to precisely control the voltage level within MATLAB®. With normal operation using the joysticks there is human error involved because we cannot precisely move the joysticks. Using MATLAB® along with the analog output card will allow users to input and send precise signals to the blimp.

The MATLAB® software code is a GUI program which allows the user to precisely control the blimp and read data coming from the sensors to determine the blimps current movement. The software was originally going to be a test bed to see if the different functions of the blimp (communication out, communication in, and digital filter) worked. However, it has developed into the primary blimp program that the operator of the blimp will use to administer commands, receive data, and manipulate the blimp. A printout of this code is included in Appendix D.

The second communications objective is to establish a wireless connection between the sensor package and the PC. This was accomplished using two Intelligent FM transceivers capable of sending RS232 serial data at a frequency of 433MHz. The FM Transceiver Module is depicted below in Figure 8. The two transceivers will act transparently, meaning they will form a wireless bridge. Thus it will be as if the data was being transmitted through a normal RS-232 cable.

Figure 8 - FM Transceiver Module

During the spring semester the group was able to implement wireless Bluetooth technology. Using Bluetooth allowed us to transmit and receive at the same time. We were able to operate several deices without infringing on each other due to separate channels of operation. Through testing the devices, we found that the signal strength of the Bluetooth devices was strong enough to overcome that of the wireless routers also operating at 2.4GHz. The Bluetooth devices were very useful for their fast speed and single point connectivity to the computer using a Universal Serial Bus (USB) connection.

System Identification

The system identification component of the design consists of determining the mathematical equations that model the blimp's movement. These equations are used to describe dynamic characteristics of the blimp. The air pressure in the blimp, as well as its unusual shape makes the blimp's behavior somewhat unpredictable and nonlinear. To combat the nonlinearity, the blimp's overall response is segmented into modes. One mode concentrates only on the blimp's rotational movement, such as turning 45 degrees to the left. Another mode for the blimp would be hovering in place or traveling a desired distance in a straight line. Each mode has one or more associated equations that will be derived in the spring semester.

The equations for the system identification are based on information gathered experimentally. This information will be gathered through applying varying amounts of thrust to all of the motors, and recording the blimp's response to the thrust.

To obtain the necessary data, the blimp will undergo a series of tests. This procedure is a test plan - a step by step guide on how to run each test as well as environmental considerations, affecting the blimp's response throughout the testing (as outlined in the support plan (Appendix C)). The purpose of the test plan is to gather as much information as possible in order to formulate the most accurate system equations for the controller.

Our solution to the digital controller design is a state space design approach for the controller. We believe state space is the best choice because we have multiple inputs going to multiple outputs, and state space allows the designer to more easily apply the equations and make connections to the different modes. Our controller should separate the blimp's possible movement into different components - height, heading, and velocity. This will make for simpler models of a system which overall is very complex.

After gathering all the necessary data through testing, the next step was to use the appropriate mathematical approach to generate the system response equations. Because there are many variables associated with the blimp's movement, we are considering using state space analysis. State space analysis consists of gathering multiple pieces of information from the blimp (inputs) such as its response to a degree of motor thrust, and then producing an output matrix. The state of a system is a minimum set of numbers (called state variables) which contain sufficient information about the history of the system to allow computation of future behavior [3]. Our system will attempt to predict the future based on the past, thus our controller will be able to determine how the blimp needs to be maneuvered. State space produces reliable response equations that can be integrated into a controller.

During the Spring semester, three Mechanical Engineers and one Operations Research major were assigned to the blimp project. Their role in the project was to run tests on the blimp to determine the mathematical system identification equations of the blimps response. Once these equations were derived, the electrical engineers will then implement them in the blimps digital controller.

µNav™ Sensor Array

The sensor package we have selected is µNav™'s MNAV100CA Navigation and Servo Control Board. This sensor package is depicted in Figure 9 below.

Figure 9 - µNav™'s MNAV100CA Navigation and Servo Control Board

The µNav™ MNAV100CA digital sensor package meets the requirements of our system's sensor array. The sensor array should be able to find the blimp's heading within ±0.5 degrees, its height within ±2 inches, and its speed to within ±0.1 mph [4]. The lightweight (33 gram) sensor package contains accelerometers, angular rater sensors, and magnetometers, as well as a barometer, and a thermometer [4]. From these devices on the sensor package, we know the blimp's acceleration in three dimensions, speed, displacement, roll, pitch, and bearing change. The sensor package requires a DC power source of 3.7 to 16 volts for operation, which we will be able to power using our battery pack which operate at 8.94 volts. The sensor package is extremely sensitive to very slight movement and can read acceleration at a maximum of ±2g's of force [4].

The MNAV100CA sends updated sensor data at a rate of up to 100Hz through a serial cable [4]. The serial

data is sent in packets to the Bluetooth transceiver which puts the data into new packets for wireless transmission to the USB hub on the laptop. The USB hub takes these packets, extracts the serial data, and then puts them onto a virtual COM port as if the sensor was directly connected to the computer. The Bluetooth transceivers are not limited in their data rate, and the computer can poll the data as fast as it can.

The group received the sensor package on November 17th, 2005, and it was fully operational on the blimp by the end of the fall semester. In its fully operational state, it was fixed in the blimp's gondola with a power source, and able to wirelessly send sensor data to the computer.

Heading Sensors

In the spring semester the group received Honeywell HMR3300 digital magnetic compass sensors as shown in Figure 10 below. These sensors output data to a serial device via ASCII text which makes it extremely easy to import into Matlab®. The data are heading in degrees magnetic, pitch, and roll. Pitch can be measured to a maximum of 60° and roll at a maximum of 40°. These sensors are useful for the blimp because they enable us to determine magnetic heading at all times so that the blimp can be controlled to follow a desired heading.

Figure 10 – HMR3300 Digital Compass

Acoustic Sensors

The µNav™ sensor package is only useful in measuring short term and sudden movement of the blimp due to the need to integrate acceleration to achieve position or velocity. For a long term and more precise solution, acoustic sensors will be used in conjunction with the µNav™ sensor. Acoustic sensors will provide an approximate position of the blimp relative to the room. When this positional information is filtered with the information from the inertial sensor array, the accuracy will exceed that of either sensing system. Unfortunately, due to difficulties in the procurement of the acoustic sensors, the group was unable to provide acoustic positioning as one of the blimp's systems.

RESULTS AND ANALYSIS

Sensor Package & Communication

The group had the MNAV100CA sensor package configured and sending data soon after its arrival. We initially tested the sensor package using the Microview® software which came with it. This software produced graphic displays of changes in acceleration, velocity, movement, roll, pitch, temperature, and pressure. Analysis of the sensor package's capabilities showed that it was sensitive enough to show very slight changes in movement, therefore it should be able to accurately represent the relatively slow movement of the blimp.

The next step of interfacing the sensor package with MATLAB® proved to be a bit more complicated. The MATLAB® code for this interface is included in Appendix D. The sensor package sends data in either voltage or scaled mode. We desire to use scaled mode because that provides us with the actual data in G's, degrees per second, and applies the calibration settings to the data as well. Reading in the scaled data allowed us to plot the data in a MATLAB® figure. This allows us to have a real time analysis of the sensor data. Further development of the sensor package interface in the GUI was undertaken in the spring semester. The data is displayed within the GUI and contains many useful error checking and speed enhancements.

Now that the sensor package and MATLAB® were communicating, the communication needed to be made wireless. This wireless communication via the two transceivers proved to be very challenging to set up correctly. In order to get the sensor package working with the wireless transceivers the firmware of the sensor package needed to be changed. The firmware is the program that the sensor package runs when it is powered. We needed to make the sensor package start up with a packet rate of 10Hz and a data rate of 9600 bits per second because that is the maximum transmission rate of the transceivers. After editing the firmware, the transceivers relayed the data to the computer as if it was directly connected to the sensor package. With the sensor package communicating wirelessly with the computer, we could now install and test the sensor package on the blimp.

Using the new Bluetooth modules we were able to increase the data rate as much as we wanted due to the increased bandwidth of the Bluetooth protocol. The

Bluetooth devices also allows us to have more sensors on the blimp without creating conflicts between them.

Wireless communications to the blimp have also been successfully implemented. The group has successfully designed and programmed a MATLAB® GUI interfacing with the analog output card, which allows a user to manually control the blimp via PC.

System Identification

The early stages of the system identification were based on the response time of the blimp. The tests done in the fall semester consisted of operating the blimp and observing its behavior with the naked eye and a stop watch. There were no defined guidelines as to what to calculate save for the time it took the blimp to settle. One of the test consisted of observing the blimp's reaction to a gravitational force slightly greater than the buoyant force due to the helium. We measured the time it took the blimp to accelerate downwards after a shot of thrust upwards from the main rotors. This proved to be inefficient and inaccurate. A stopwatch is not a precise timing device and the naked eye is not sufficient to determine the blimp's state. Figure 11 shows the generic test flowchart created to manage the physical test.

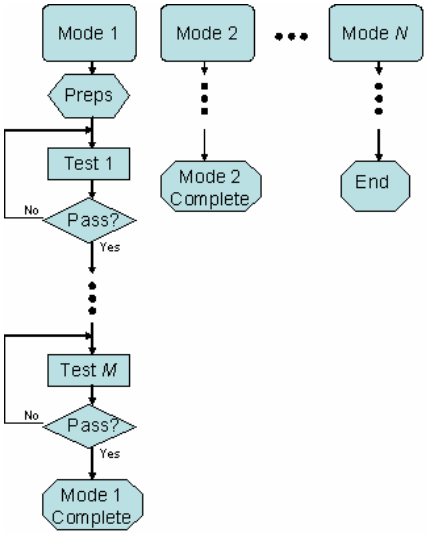

Figure 11- Test Plan flowchart.

It was necessary to create a testing rubric given the poor data collection processes. With this approach, it is imperative to have sufficient data to compensate for lack of instrumentation. A Mode, or the specific blimp response being tested, involves dozens of iterations containing time of reaction as that shown in Table 1.

Table 1- Thrust and response from blimp

Duration (s)	Power (%)	Descent Start Time (s)	Ground Time (s)
1	8	0	2
1	16	0	2.5
1	25	0.5	2.5
1	50	1	3
1	75	1	3.5
1	100	1.5	4
2	8	1	3
2	16	1	3.5
2	25	1.5	4
2	50	2	4.5
2	75	2.5	5.5
2	100	3	6

To improve the dynamic information for our system identification, we ran tests using methods commonly practiced in the mechanical engineering section. The tests were more specific and were meant to gather data that can be interpolated to find the best fit line for the system equations. The tests were run using LoggerPro® ranging devices and software. This allowed for precise positioning and timing in one of three translational movements at a time.

The end result was series of coefficients necessary for the proper derivation of dynamic equations. The latter is true for a simplified model of the blimp that does not account for roll and pitch. Our data was of better quality than the data reflected in Table 1. However, a full identification of a system as unpredictable as the blimp requires even more advanced testing tools. Absolutely describing the blimp's response would be far more plausible if the blimp's aerodynamic properties and propulsion were enhanced. Due to these constraints, we approximated the blimp's dynamics creating a linear model of the system as best as possible.

One of the main changes, resulting from our efforts during the second semester, was new system identification for the blimp based on reduced sensing capability. The new sensor package was downsized to account solely for the blimp's reaction to change in magnetic heading. The shift in focus from the μNAV™ sensors to the magnetic compass made the approach simpler. The blimp's dynamics were originally a problem with six degrees of freedom, that is, three translational movements (x, y, z direction) and three rotational (yaw, pitch and roll). Initial testing, performed in cooperation with our team of mechanical engineering students, formally excluded roll and pitch due to the dynamic properties of the actual blimp and for purposes of simplicity. The μNAV™ inertial

navigation capabilities proved insufficient to achieve a proper feedback system. The translational properties were also discarded in response to the lack of proper sensing capabilities. New hardware was installed to concentrate on the exchange of magnetic compass data.

The system identification became a simpler problem consisting of only one degree of freedom. We used the new approach to create a PD controller that uses proportional and derivative feedback loops. The controller was based on this approach and the proportional and derivative values were obtained from simulations in MATLAB®. At this point the controller is now embedded in the software code based on the values obtain through the tests. A state space version of the controller is also available, though its response is of lesser quality thus far.

The system identification was modeled in similarity to the attitude control of a satellite. The rotational motion of the blimp depends on the second derivative of the angle times the torque. Using the relationship between torque and moment of inertia, we came up with an approach that described the blimp's reaction to a thrust from the tail rotor. The percentage of thrust was tested to generate a linear approximation of the force (in Newtons), which is provided by the tail rotor. Thus, the variable of force became the output of the controller, which is then transmitted to the servomotor control onboard the blimp. To create the plant, we used the experimental values of moment of inertia and damping. The plant was used primarily to test the PD controller.

When first testing our digital controller we used P and D values produced by the Simulink® simulation. The result of this test is shown in Appendix E. We attempted to control the blimp to achieve a bearing of 251 degrees. The result of this test was that there initially was a lot of overshoot, and eventually the system fluctuated approximately 10 degrees above and below the target bearing. The system did not reach a steady state we were satisfied with, so in our next test, we tweaked the P and D values to optimize the controller's response.

Through tweaking, we determined the optimal P and D values to be 0.012 and 0.023 respectively. Appendix F is the plot of the data from our test using these P and D values. During this test we applied external rotation at two points in time to test if the blimp would regain a desired bearing of 180 degrees. From the plot we can see that the controller reaches the desired bearing quickly after the rotation is changed. The final system was also very stable, and was able to maintain a constant bearing. We were satisfied with the test and decided that the controller was functioning properly, while also meeting the system requirements (Appendix B). We were confident that our plant fairly represented the blimp's dynamics.

CONCLUSION

Over the course of the 2005-2006 school year, the group set several goals. We did not complete all of our goals that we set out to complete at the beginning of the project, but the majority of the harder goals have been completed. This leaves the project in a perfect position to be taken up again next year.

Our system design for the fall semester was successfully completed. We implemented a MATLAB® user interface to send commands manually and wirelessly to the blimp. We acquired a sensor package to meet our project needs, and were successful in meeting our goal of having it communicating information wirelessly into MATLAB®.
The advances made during the semester with the sensor, communication, and computer interface components facilitated the process of the system identification in the spring semester. The digital controller will be the end product, a result of the integration of the system equations. During the spring semester, we are primarily focusing on successfully implementing one or two control modes. A group of mechanical engineering students from The United States Coast Guard Academy have joined the team to help analyze the dynamics of the system and contribute equations that can better approximate the blimp's response. These contributions can then be applied to a state space control system. The electrical engineers in the group will work on creating a digital equivalent for the analog control and model it using computer software. The group plans on having a working digital controller for at least one mode of operation by the end of the spring semester.

It may seem that the constraints mentioned have come to set significant drawbacks on the project. However, identifying deficiencies in the approach is an accomplishment in itself. The end result is a system of equations that describes the blimp's reaction to an order heading and the torque produced by thrust of the tail rotor. This is a deliverable that follows our requirements specification and sets the groundwork for future efforts on the Autonomous Blimp project.

During the process of finding equations for the blimp the electrical engineers continued development of the sensor package systems and the GUI. The GUI interface is more robust and includes graphs of the sensor data, the ability to save sensor data, and

increased error detection. The mechanical engineers were able to apply our work with their work to develop the blimp's equations.

Finally, we combined both of our efforts to create the digital controller. The controller allowed us to control the blimp's heading using the digital compass as the sensor.

It is recommended that next semester's project engineers continue work where the group left off. The new project team should implement another mode into the digital controller to control altitude. The acoustic ranging sensors need to be implemented on the blimp to provide accurate positioning data as an input to the controller. Using Kalman filtering techniques, the ranging data from the acoustic sensors can be combined with the inertial data from the sensor package to develop the most accurate positioning system possible.

REFERENCES

Works Cited

[1] Alam, Carl. "Coast Guard Aviation Gets a New Look." The Naval Institute: Proceedings. August 2003. available from http://www.military.com/NewContent/0,13190,NI_CoastGuard_0803,00.html >. (accessed on Dec 21 05).

[2] "Integrated Deepwater System." US Department of Homeland Security Press Release. October 17, 2005.

[3] L. K. Timothy and B. E. Bona. "State Space Analysis: an Introduction." McGraw Hill. New York. 1968.

[4] "µNav Navigation & Servo Control Board Users Manual". Crossbow Technology Inc. San Jose California.

References Reviewed

Accelerometer Theory of Operation. Tech Note 402, Vol 8, Iss 3. pgs 48-53.

Aitenbichler, E; Robling, G; Trompler, C. *Intelligent Headsets for Supporting Digital Lecture Halls*. Department of Computer Science, Darmstandt University of Technology. 2003.

Aitenbichlet, E; Muhlahauser, M. *An IR Local Positioning System for Smart Items and Devices*. Telecooperation Group, Department of Computer Science, Darmstadt University of Technology. 2003.

Drake, K.; McVey, E.; Inigo, R. *Experimental position and ranging results for a mobile robot.* IEEE Journal of Robotics and Automation. Vol. 3, Iss. 1, Feb 1987. Pages: 31-42.

Favre-Bulle, B.; Prenninger, J.; Eitzinger, C. *Efficient tracking of 3D-robot positions by dynamic triangulation.* Instrumentation and Measurement Technology Conference. Conference Proceedings. IEEE, Vol.1, Iss., 18-21 May 1998. Pages: 446-449 vol 1.

Haapaoja, C. *PC-CARD-DAC08 Analog Outputs & Digital I/O User's Manuel*. March, 2001. Copyright 2001, Measurement Computing Corporation.

Ka C. Cheok, et. al. *An Ultra-Wideband RF Method for Localizing an Autonomous Mobile Robot.* Dec 2004. Available <http://www.stormingmedia.us/28/2882/A288234.html >

Maeda, M.; Ogawa, T.; Kiyokawa, K.; Takemura, H. *Tracking of user position and orientation by stereo measurement of infrared markers and orientation sensing.* Wearable Computers. 31 Oct 03 – 3 Nov 04. Pages 77-84.

Manuel for FUTABA SKYSPORT 6. FUTABA Corporation. 1998.

McGillem, C.D.; Rappaport, T.S. *A beacon navigation method for autonomous vehicles.* Vehicular Technology. IEEE Transactions on, Vol. 38, Iss. 3, Aug 1989
Pages: 132-139.

RPI. Laboratory Introduction to Embedded Control. Available <http://litec.rpi.edu/>. Accessed 01 Oct 05.

Talluri, R.; Aggarwal, J.K. *Mobile robot self-location using model-image feature correspondence.* IEEE Transactions on Robotics and Automation. Vol. 12, Iss. 1, Feb 1996. Pages: 63-77.

Tong, F. *A high precision ultrasonic docking system used for automatic guided vehicle*. Science Direct. October 9, 2003.

APPENDICES

A. Project Plan
B. Requirements Specification
C. Support Plan
D. BLIMP Code
E. Plot of GUI output values for heading controller with Simulink® P and D values
F. Plot of GUI output values for heading controller with P and D values after tweaking