Design and Integration of an Unmanned Aerial Vehicle Navigation System

A Thesis
Presented to
The Academic Faculty

by

Joerg S. Dittrich

In Partial Fulfillment
of the Requirements for the Degree
Master of Science in Aerospace Engineering

School of Aerospace Engineering
Georgia Institute of Technology
May 2002

Design and Integration of an
Unmanned Aerial Vehicle Navigation System

Approved:

_____

Dr. Eric N. Johnson, Committee Co-Chair

_____

Dr. J.V.R. Prasad, Committee Co-Chair

_____

Dr. Daniel P. Schrage

Date Approved _____

ACKNOWLEDGEMENTS

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ix

LIST OF SYMBOLS

Symbol	Description
a	accelration
a_{IMU}	IMU acceleration measurement
$\bar{a}$	corrected acceleration measurement
A	linearized system matrix
$\hat{b}_a$	IMU accelerometer bias estimate
$\hat{b}_\omega$	IMU gyro bias estimate
C	linearized system output matrix
F	process model
g	gravity vector
G	output model
h_{SONAR}	sonar altitude measurement
$\bar{h}_{son}$	corrected sonar altitude measurement
$\hat{h}_t$	terrain height estimate
K	Kalman Gain matrix
P	state estimate covariance matrix
$\hat{q}$	attitude quaternion estimate
Q	process noise matrix
r_{SENSOR}	sensor mounting position
R	sensor noise covariance matrix
$T_{b \rightarrow i}$	body to inertial frame coordinate transformation matrix
$\hat{v}$	velocity estimate
v_{GPS}	GPS velocity measurement
$\bar{v}$	corrected velocity measurement

xi

Symbol	Description
$\hat{x}$	position estimate
x_{GPS}	GPS position measurement
$\bar{x}$	corrected position measurement
X	state
$\hat{X}$	state estimate
$\bar{z}$	sensor measurement
ω	angular rate
ω_{IMU}	IMU angular rate measurement
$\bar{\omega}$	corrected angular rate measurement
σ	standard deviation

GLOSSARY

802.11b	IEEE Wireless LAN standard
AGL	Above Ground Level
AHRS	Attitude Heading Referance System
DGPS	Differential Global Positioning System
EMI	Electromagnetic Interference
FTP	File Transfer Protocol
GCS	Ground Control Station
GUI	Graphical User Interface
GPS	Global Positioning System
HF	High Frequency
HITL	Hardware-In-the-Loop
IMU	Inertial Measurement Unit
LAN	Local Area Network
LED	Light Emitting Diode
LORAN	Long Range Navigation System
MSL	Mean Sea Level
NDB	Non-Directional Beacon
PC/104	SBC form factor
RS-232	Serial data transfer standard
SBC	Single Board Computer
SITL	Software-In-the-Loop
TCP/IP	Transmission Control Protocol / Internet Protocol
UAV	Unmanned Aerial Vehicle
YACS	Yamaha Attitude Control System

YAS   Yamaha Attitude System (IMU)

YCS   Yamaha Control System

YRD   Yamaha Receiver Data

VHF   Very High Frequency (30-300 Mhz)

VOR   VHF Omni-directional Radio-Range

VTOL   Vertical Take-Off and Landing

CHAPTER I

INTRODUCTION

This document introduces the applied design and integration methodology used to develop an avionics system that provides navigational and terrain data to the flight computer of a Vertical Take-Off and Landing Unmanned Aerial Vehicle (VTOL UAV). The process includes both the design of flight hardware and a software package that interprets the sensor data. An overview of a specific design is also provided, placing special emphasis on the interaction between hardware and software development.

1.1 Design Requirements

The Defense Advanced Research Projects Agency (DARPA) Software Enabled Control program requires a VTOL UAV that has the capability to execute extreme performance maneuvers as well as to perform during complex and changing mission scenarios. Earlier flight tests and simulations in this program were conducted on an X-Cell radio-controlled helicopter UAV, depicted in figure 1, with a basic avionics system that allowed autonomous flight [6]. Future demonstrations are to utilize a more sophisticated testbed. This new UAV is based on a Yamaha R-Max industrial helicopter, which has six times the avionics payload capability of the X-Cell.

The most important design criterias for the new testbed are good performance and maximum system flexibility, i.e. ability to reconfigure. Good navigation system performance (high rate, high accuracy) allows for higher bandwidth control, which is required for extreme performance manuevers. While many rotorcraft UAV avionic designs have the system weight as their main design constraint, it is secondary to the system flexibility in the Georgia Tech design. With current commercial-of-the-shelf components it is relatively easy to design a very effective system within the payload allowance of the R-Max aircraft. Hence, efforts were concentrated to ease hardware reconfiguration and allow for considerable future system

Figure 1: Georgia Tech HB-1 UAV based on an X-Cell 60 helicopter

growth. This implies modularity, very good accessibilty of all components, and common interface standards.

1.2   Design Process

Autonomous Unmanned Aerial Vehicles (UAVs) require avionics systems that enable them to maintain a stable attitude and follow desired trajectories. Such an avionics package is comprised of sensor, computer, and data link hardware as well as software to guide, navigate, and control the vehicle. Therefore, the development of such a system for a specific vehicle is a multidisciplinary design and integration effort.

Figure 2 depicts the realization flow for a complete UAV avionics system. Hardware selection is a critical design trade-off in deciding which sensor and supporting electronics are needed to fulfill the design specification. The selection of the avionics components and the integration of that hardware into the R-Max airframe are discussed in chapter 2.

Sensor data from multiple sources can be filtered and integrated into a navigation solution. The navigation software will provide that functionality, interfacing with the sensor systems. The other two main tasks are the development of the flight control software for the vehicle and the implementation of a simulation model of the aircraft. However, this thesis will not focus on either of these, but will instead focus on the development and testing of the navigation hardware/software system. Before actual flight tests can take place, sensor interface software and navigation algorithms need to be validated to ensure proper functioning. The algorithms were first tested by using software emulated sensor data and
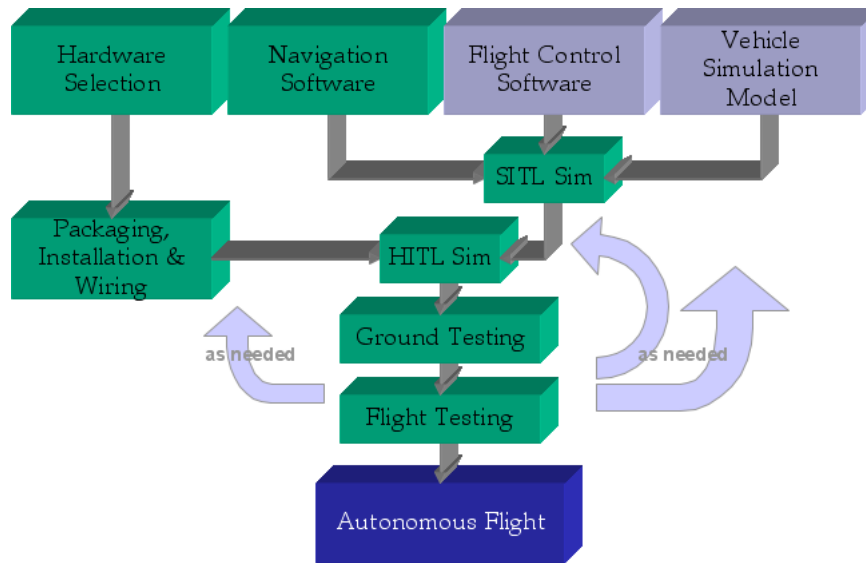
Figure 2: Design and integration process

a simulation model of the aircraft (software-in-the-loop). The implementation of the navigation software as well as the development of the simulation and ground control station interface are outlined in chapter 3.

Independent from these software design efforts, the hardware needs to be packaged and interfaced (hardware wiring) placing attention to vibration isolation, electromagnetic interference, and accessibility.

Once all these tasks have been completed, the simulation of the UAV system can be taken to the next level by integrating hardware components into the simulation loop (hardware-in-the-loop). In this scenario, simulated sensor data from a computer, that is running a simulation model of the vehicle and sensor behaviour, is being fed into the flight computer.

Chapter 4 shows how the performance and possible errors of the avionics system can be evaluated during intensive ground and remotely piloted flight tests. Ground and flight-test data showing sensor and navigation system performance are being recorded and compared to simulation results. Improvements within the previous steps can and should be undertaken at this point, until a configuration is reached that promises satisfactory results for autonomous flight experiments.

CHAPTER II

HARDWARE SELECTION AND INTEGRATION

The synthesis of the avionics hardware is a trade-off evaluation like any other design process. An optimal design solution is sought, by finding the best compromise to satisfy the design requirements.

The factors considered for hardware selection and integration include:

Performance of the navigation system sets the capabilities of the vehicle in terms of which maneuvers can be performed.

Weight requires a trade-off between available payload and desired on-board capabilities.

Electromagnetic interference (EMI) can be a serious problem when electronic systems operate in close vincinity to each other and radio transmitters or receivers. Proper shielding methods must be considered.

Power has to be supplied to the avionics. Usually batteries and/or generators serve as a source. If batteries are required, it will affect the system weight and endurance.

Vibration produced by small aircraft piston engines and rotors can damage the avionics or make sensor data unusable.

Hardware integration issues are physical aspects as form factors and mounting possibilities of the desired hardware as well as the electronic interfaces that provide communication between the components.

Flexibility is beneficial if configuration changes are common or required, it includes the possibility to replace components easily.

Redundancy allows for specific parts of the avionics to fail without causing hazard. Depending on the size and cost of an aircraft this may be more or less important.

Maintainability is a factor when easy access to the flight hardware is neccessary. The functionality of the chassis design directly affects the required effort to inspect or replace subcomponents.

Growth potential may be needed for future system extensions in terms of payload, volume and interfaces.

Cost is a limiting factor for achievable performance.

2.1   Flight Computer

If an aircraft is required to fly autonomously, sensor information has to be processed into actuator commands. This includes computation of the attitude and position as well as the computation of flight control commands. Flight computers and/or processors execute the needed algorithms.

For a UAV design an important consideration whether or not that computing power should be located on-board or on the ground. Off-board sensor interpretation as in [4] always relies on working data links and can result in significant weight savings. on-board computers can process the flight algorithms at a higher rate and are less susceptible to signal latency problems.

2.2   Sensors

If a UAV is to fly autonomously or needs stability augmentation in remote controlled flight, its flight control algorithms need information about its state. Depending on the vehicle type and its mission these requirements vary. For the purpose of this project, main sensor types have been split into attitude, position and altitude sensors.

2.2.1   Attitude Sensors

As helicopters are inherently unstable in hover, information about the UAV's attitude is needed in order to allow the inner loop of the flight controller to stabilize the vehicle.

Most common attitude sensors are based on gyros, that can be either mechanical, piezo-electric, or optical. A three axis gyro platform measures angular rates along all axes of the vehicle and is contained in an Inertial Measurement Unit (IMU), which also provides accelerometers that can be used in position estimation. Attitude and position can be integrated from IMU measurements, preferably in a state estimator. An alternative to an IMU or individual gyros, is a self-contained Attitude Heading and Reference System (AHRS) which can directly deliver vehicle attitude, supressing error growth internally.

In a supporting role, magnetometers can be used to determine attitude of a vehicle by measuring the earth's magnetic field. Ambiguity becomes an issue with this sensor type as there is an axis of attitude ambiguity parallel with lines of the Earth's magnetic flux. Also, vicinity to magnetic fields induced by currents and magnetic bodies will decrease the accuracy. While easily applied as a heading reference for navigation filters in AHRS units or custom navigation systems [5],[10], it can also deliver information about the other axes.

Besides inertial and magnetic sensors which can be found in many attitude systems, a couple of less common approaches are available to UAV's. The attitude of unmanned helicopters in flight have been successfully augmented by a carrier-phase GPS based system [2], which utilized multiple antennas to compute position and attitude of a vehicle. Vision-based attitude indicators could rely on scanning the horizon to keep an aircraft level, terrain permitting. An unusual idea was proposed in [7], utilizing an inclinometer to stabilize a small helicopter in hover.

2.2.2 Positioning Sensors

Basic to a navigation system is the ability to determine the current position of a vehicle with respect to a known point. Radio-based navigation methods that are being used in manned aviation are usually based on terrestrial non- and omni-directional beacons (NDB, VOR) and/or measurement of distances (DME, LORAN). The high accuracy, simplicity and availability of the Global Positioning System (GPS), however, makes it the emerging standard positioning system for unmanned aircraft as well for general and commercial aviation. If navigation is only needed in a limited area, the application of vision-based systems

can also be a feasible option. A rotorcraft UAV described in [1] utilizes a visual odometer and can propagate the position relative to the origin of the flight.

As most UAVs use GPS as a primary navigation sensor, more focus shall be put on its limitations and how to utilize it in a the best possible way. While single point GPS operation (subject to atmospheric errors, etc.) may be acceptable for many applications, the use of differential corrections from a second fixed-position GPS receiver is occasionally required to remove atmospheric and similar errors. Achievable accuracies in differential mode are approximately within 3 feet or better if the carrier phases of the GPS signals are analyzed. It is important to point out that a differential GPS system always needs some sort of data link in order to receive correction data from the reference station. A simple GPS equipped vehicle can technically operate totally autonomously (save the GPS satellite constellation) without any neccessary link.

Depending on the quality of the receiver, the achievable accuracy and the update rate varies. A sensor with an update rate of once a second can work fine, but the controller bandwidth may be limited. Besides obtaining a receiver with an improved update rate, a common way of solving that problem is to use inertial information and state estimation as described in section 3.2.

2.2.3   Altitude Sensors

Altitude sensors measure with reference to sea level (MSL, mean sea level) or the local ground (AGL, above ground level). The kind of data is needed in order to control the altitude of the aircraft depends on the type of vehicle and the modes and location of operation. Operations within ground vicinity such as landings usually require absolute AGL measurement or a very accurate terrain database. Available sensors include:

- Sonar (AGL)

- Radar (AGL)

- Laser/Lidar (AGL)

- GPS (MSL)

- Barometric (MSL)

Important issues for the selection of a specific sensor type, besides the type of measurement, are its accuracy and range.

2.3   Data Links

Wireless data links are used for unmanned vehicles to send commands and receive telemetry or payload data and can be divided into digital and analog links. An example for an analog link is a UHF video signal transmission. Digital links provide a way of communicating between ground and vehicle-mounted computers. The frequency band a data modem operates on affects its data rate. Typically, the higher the frequency, the higher the data rate. The frequency also affects the range of the data link. Lower frequencies typically offer a greater range than high frequencies. Furthermore, the higher the frequency the greater the Line-of-Sight problem, i.e. the ability to penetrate obstacles like buildings. Common data links in the 2.4 Ghz band are more easily blocked than VHF frequencies. Also, for unmanned vehicle operation a remote pilot data link is often used to steer the vehicle manually for some phases of the flight.

2.4   Hardware Selection Example

2.4.1   Vehicle

Yamaha's R-Max (see figure 3) is an industrial remote controlled helicopter, which has been developed as an agricultural spraying aircraft for small fields. Its specifications are outlined in appendix A. The most important features for the modification to a UAV are its payload, endurance and power system. A 12V DC generator is included in the airframe and powers all systems through a buffer battery. This gives it the capability to power the avionics as long as fuel is available, as opposed to many smaller UAV's have to carry heavy battery packs to power the avionics.

Yamaha equips the helicopter with a stability augmentation and control system, the

Figure 3: Yamaha R-Max

YACS (Yamaha Attitude Control System) which has its own IMU comprised of 3 laser-gyros and 3 piezo accelerometers. Georgia Tech's aircraft has been retrofitted with a more powerful generator, a longer landing gear, and a modified version of the YACS that supports telemetry and IMU readouts from a computer. More importantly, it also allows for an on-board computer to control the actuators of the vehicle. Three RS-232 serial ports, which can be accessed on the back of the control system box, comprise the interface to this external hardware.

Port 1 Yamaha Attitude System (YAS), IMU raw data.

Port 2 Yamaha Control System (YCS), provides telemetry and accepts external servo commands.

Port 3 Yamaha Receiver Data (YRD), shows which servo commands are received from the remote pilot transmitter.

2.4.2 Selected Hardware

The new test-bed developed at the Georgia Tech UAV Research facility is based on an R-Max helicopter which requires a reliable avionics package. Similar sized Yamaha helicopters (R-50 and R-Max) have been instrumented for autonomous flight before [1], [9], [8]. The Georgia Tech design, however, uses multiple sensors to implement an avionics system that has an emphasis on redundancy and a significant payload capability for later extensions.

Only two different bus interfaces are used to interconnect the components: RS-232 serial bus for sensor data and 100 Mbit Ethernet bus for communication between computers. The selected components are:

- Flight Computer:

    JUMPtec Adastra VNS-786 embedded PC, 266 Mhz Pentium processor, 64MB RAM, 4x RS-232, Ethernet, 32MB solid-state harddrive, executes navigation and flight control code, communicates with sensors and actuators, records flight data and commits telemetry to the ground station. Recorded flight data is saved in the system RAM and is downloaded to the ground station using an FTP server.

    Diamond Systems Emerald-MM PC/104 Serial Port Extension Board, 4x RS-232, additional interfaces for the flight computer.

- Sensors:

    Inertial Science ISIS-IMU Inertial Measurement Unit (IMU), provides acceleration and rates along all three axes, primary sensor.

    Yamaha YACS IMU built-in, provides acceleration and rates along all three axes, backup sensor.

    NovAtel Millenium RT-2 Differential GPS receiver, high accuracy position and velocity data, requires error correction data from a ground-reference GPS-receiver for high accuracy operation.

    Honeywell HMR-2300 Magnetometer, measures the earth magnetic field for attitude initialization and updates of the navigation filter.

    Custom-built Sonar Altimeter, Polaroid 6500 ranging modules, measures the height above ground up to 35 feet.

    Roke Manor MRA Mk IV Radar Altimeter, altitude above ground level up to 2300 feet, 5 inch resolution.

Yamaha Vehicle Telemetry built-in, provides rotor RPM, battery level, vehicle status flags (control system status, reveiver error, faulty sensor readout, etc), remote pilot commands

- Data Links:

  Aironet 4800 11 Mbps Wireless Ethernet Data Link, IEEE 802.11b standard, 2.4 GHz, high data rate, supports a network with multiple wireless clients, TCP/IP protocol

  FreeWave DGR-115 Wireless Serial Data Link, 900MHz, high range, proprietory protocol

  D-Link DSS-5+ 10/100MBit Ethernet Switch, 5 ports, establishes link between Aironet client and on-board computers.

  Yamaha Remote Control 72 Mhz, allows a safety pilot to manually pilot the helicopter as needed.

- Ground Control Station:

  NovAtel Millenium RT-2 PowerPac GPS Reference Station, creates differential correction data to be sent to the vehicle GPS receiver.

  PC Laptop transmits GPS differential data to the vehicle via both wireless data links. Visualization of the vehicle state, low rate data recording, uplinks user commands to the vehicle, user interface.

All airborne elements are powered by an airframe-mounted generator, buffered by an on-board battery. In order to achieve a good system bandwidth the computer (i.e. autopilot) is installed on the airframe, also enabling the vehicle to operate independently from a ground station if required or in case of a data link loss. The equipment manufacturers are listed in appendix C.

2.5   R-Max Hardware Integration

Flexibility has been defined to be a key feature of the R-Max testbed UAV. Hence, one has to focus on an intelligent integration of the system components to achieve that goal. Similar existing UAV systems ([10], [9]) are usually outfitted with a single equipment enclosure mounted underneath the fuselage, which contains most of the sensor and computing hardware. The single-box design is easy to design when all sizes of the electronics are known and it is possible to install the hardware components within minimal space. Drawbacks of this approach are usually poor acessibility and maintainability. The box may have to be removed from the airframe in order to reach system components, or hardware has to be removed to gain access to a certain piece of equipment. If system reconfiguration is expected like in the R-Max design, this approach reveals an additional disadvantage as it may be impossible to replace certain components when the replacements have different form factors and interfacing requirements.

Placing attention to modularity is a common solution to growth potential and reconfiguration requirements. However, many helicopter design approaches (like [6]) reduce the word modularity to hardware interface compatibility (serial ports, power, etc). Therefore, it was decided to take the R-Max system modularity to the next level by introducing a modular enclosure system. The modules themselves are not to be directly connected by a custom interface bus, but are interconnected with patch cables in order to preserve maximum flexibility. By using this approach of indepedent metal enclosures for the avionics components, it also becomes easier to ensure sufficient EMI supression between the components, as the enclosures bring in some shielding capability. A commercial-off-the-shelf rack-mountable enclosure system (Uni-Pac from LMB/Heeger) has been selected to house all avionics components except the magnetometer, sonar, power system, and the antennas.

This way, the avionics can be installed into an airframe-mounted rack, which allows easy removal in case work needs to be conducted on a single module. The custom designed rack allows 4 full-size (3.5 inch) modules or 8 half-size (1.75 inch) modules respectively. It is a light weight construction that utilizes the structure of the avionics modules to achieve its rigidity.

Figure 4: Shock-mounted avionics rack on the airframe, front outer box cover removed, from left to right: Data Link Module, GPS module, IMU/Radar Module, Flight Computer Module, Auxiliary Module (uncompleted)

Figure 4 shows the modules in the rack. The selected hardware has been split up into 5 modules as follows:

Flight Computer Module: hosts the Pentium-PC and the serial port extension board, totaling eight RS-232 interfaces, full-size.

GPS Module: consists of the GPS receiver board with two RS-232 ports and an antenna connection, half-size.

Data Link Module: provides serial and Ethernet communication to the ground station and interfaces computers. Ethernet wireless link is connected to the Ethernet hub which provides 4 networking ports for on-board hardware, half-size.

IMU/Radar Module: Inertial measurement unit and radar altimeter with one RS-232 interface each, full-size.

Auxiliary Module: space can be used for additional capabilities, e.g. a second computer or a video/vision system, full-size or 2x half-size.

Modules outside of the rack are:

Sonar/Magnetometer Assemblies: these modules are mounted on the tail boom and are equipped with powered RS-232 ports.

Power Panel: distributes on-board or external power (12V DC) to the consuming modules. All power outlets are 12V DC and are secured by one separate circuit breaker each.

Each module has self-contained power regulation, ventilation, and EMI shielding, which allows for configuration changes without the need for redesigning the overall system.

The avionics rack is installed inside a custom aluminum enclosure that is hard-mounted on the airframe below the fuselage. The power panel is mounted on the back of this box. Special attention has been placed on the accessibility of the hardware interfaces, such that the planned hardware-in-the-loop simulations (see 3.4) and other changes to the setup can be performed with minimum effort. Therefore, the side panels of the avionics box can be removed easily to provide access to the front and back panels of the equipment modules. The front panels of the modules comprise the user interface to the system with external equipment interfaces (monitor, floppy drive, laptop) and status LED's covered by a transparent lexan side plate, while the back panels host the communication and power ports.

A tail boom mount, similar in shape to a horizontal stabilizer fin, provides installation points for the sonar assembly, the magnetometer, and the GPS receiver antenna. It is attached by using a velcro strap, providing a firm fit and leaving the structure of the boom unchanged. Figure 5 shows the assembled UAV with all avionics installed.

2.5.1 Hardware Interfacing and Wiring

A schematic wiring diagram of the vehicle-mounted avionics is depicted in figure 6. All interfaces are located on the back-sides of each module (figure 7), where they are interconnected with patch cables. This design allows for easy configuration changes, if neccessary. Examples include the possible routing of the differential corrections from the serial data link to the GPS which can either be processed through the flight computer or directly fed

Figure 5: R-Max UAV with main avionics box and boom mount

into the second GPS serial port. The flight computer ports can be directly connected to an off-board computer for a hardware-in-the-loop simulation while the computer is mounted on the airframe.

All components are interconnected and powered by an aviation-quality wiring harness which features individual shielding of all signal lines to ensure maximum operational safety.

Figure 8 shows the opened data link module, representative for the other modules that were designed in a similar manner. The wires inside the module are tied to several mounting points and to each other in order to prevent any shaving of the insulation.

2.5.2 Power System

As discussed earlier, all modules and assemblies require a 12V DC power connection. Power can be supplied from the on-board battery/generator system or an external power source for ground operations. Depicted in figure 9, the power panel holds two main power switches: the main/external power and on-board power switch. By operating the "on-board power" switch, power sources can be swapped on the fly, which is made frequent use of during field test operations, whenever the helicopter engine is not running.

Power is distributed to each module through a set of circuit breakers, one for each line. This way, electrical failures and short circuits will only affect one module at a time instead
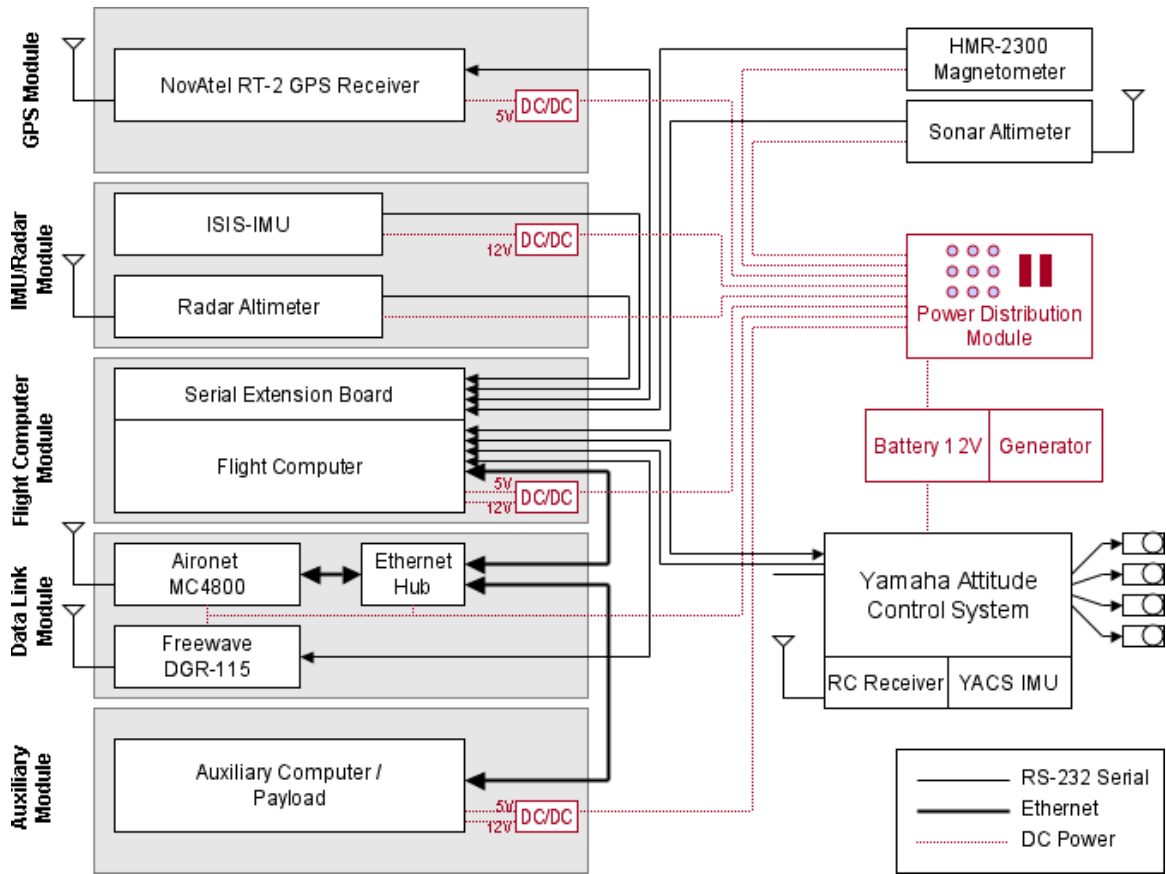
Figure 6: R-Max module schematics and wiring diagram, YAS interface is not connected in this configuration

of shutting down the whole avionics system. Furthermore, the circuit breakers allow one to power up parts of the system only.

In the current configuration, flight computers and GPS need a 5V DC power supply. Therefore, the corresponding modules are equipped with DATEL DC/DC converters, such that these modules can be hooked up to the 12V power system. Since the on-board power bus is unfiltered, the IMU is powered through a 12V regulator to ensure that this critical piece of equipment always receives the correct voltage and to isolate it from the rest of the electric system to prevent interference. The DC/DC converters can be found in the system modules schematics in figure 6.

Figure 7: Module back panels with wiring harness

2.5.3 Electromagnetic Interference Shielding

Ideally, each module is equipped with sufficient EMI shielding such that it cannot interfere with equipment in other modules. All aluminum parts of each enclosure are electrically connected and grounded. Additionally, the inside of the enclosures has been coated with anti-EMI conductive spray paint. Further attention needed to be paid to the data link module, as it contains two different radio modems which must not interfere with each other. Aluminum frames covered with copper foil appear to provide sufficient component isolation, which can be distinguished in figure 8. The shielding harness on top of the FreeWave modem can be seen in the lower left corner, which is like the other two components mounted on a copper covered chassis plate.

All serial connection cables inside the modules and the patch cables on the outside are made from shielded, aviation-grade, three conductor wire. All of these cable shieldings are grounded individually.

2.5.4 Vibration Isolation

Electronic circuits and sensors can be affected by harmful vibration from the engine and rotors. In particular, the IMU, GPS, and the sonar altimeter are likely to produce faulty
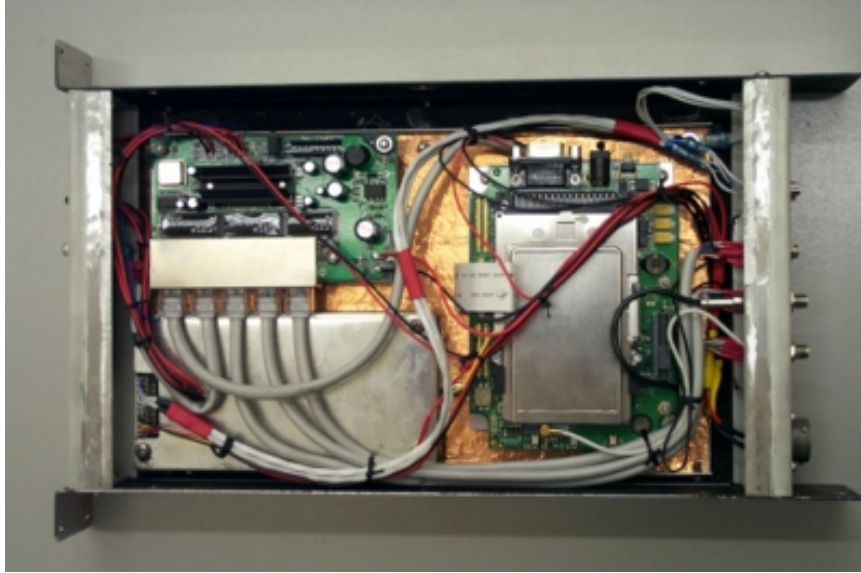
Figure 8: Data link module with removed side panel

readings with inadequate vibration insolation.

The avionics rack is supported inside of the main enclosure by four elastomeric shock absorbers at its corners, which can be seen in figure 4. In order to prevent rotational movement due to translation, the shock-mounts are installed, such that their tops are aligned with the center of gravity of the avionics rack. The separation of translation and rotation in terms of avionics movement should contribute to improved IMU data.

The critical frequency of the shock mounted rack must not be close to any produced by the aircraft at its operation point. With the selected Shock-Tech FM10-S01 vibration elements, and an assumed rack weight of 30 lbs, a natural frequency of 9 Hz can be expected. This frequency is far enough from the closest frequency of the system, the rotor-induced oscillations at about 22 Hz, to prevent any adverse effects.

As the sonar is mounted beneath the tail rotor boom it needs separate protection from harmful vibration. By using short pieces of rubber to connect the sonar to the mounting bracket, it should be effectively decoupled from the tail boom vibrations.

2.6   Overall System Setup

The UAV system is designed to allow for several system setups. The most common setup for ground and flight tests is shown in figure 10. The wireless data links are setup redundantly,

Figure 9: Power panel on the back of the avionics box

(i.e. the same data is sent and received by both). If one link fails the other one continues to exchange messages. Higher data rates and the use of the TCP/IP protocol, offer additional benefits to the 802.11b Aironet link. The operating system on the flight computer has been equipped with an FTP-Server which enables the ground operator to download recorded flight data while the vehicle is still in flight. Also a VxWorks target server connection between the development environment and the on-board computer is run to update and execute the flight code remotely via that TCP/IP connection.

In order to conduct a flight test, the vehicle avionics are initially powered by an external power source as long as the engine is not running to avoid draining the on-board battery. Compiled flight code is uploaded from the ground station onto the flight computer and executed remotely. The ground control station (GCS) software connects to the vehicle and displays status. After an IMU warm-up period of approximately three minutes, the navigation filter is initialized by a ground station operator command. A more detailed description of the initialization process can be found in section 3.2.4. When everything and everybody on the test team is set up for the flight the engine is started and the avionics power source is switched from external to on-board power, allowing flight for up to approximately one hour, limited by on-board fuel capacity.

Computer controlled flight can be enabled by toggling a switch on the safety pilot
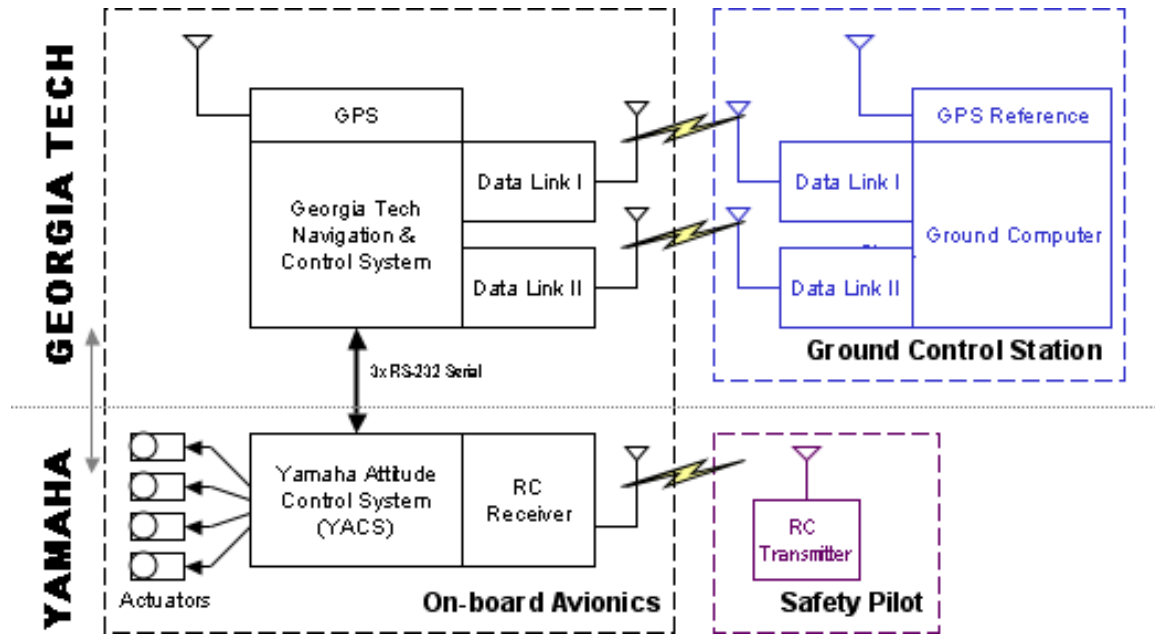
Figure 10: System flight configuration

transmitter and setting a register value in the servo control command structure. That way, the UAV can only be switched into automatic control when both the safety pilot and the GCS operator agree to do so. In case of a safety pilot transmitter failure, the computer will switch to computer control when the computer is up and ready, otherwise the Yamaha control system will activate failsafe mode.

High rate on-board data recording is independent from data logging on the GCS. It can be started and stopped by the GCS operator and will generate a file in the volatile flight computer RAM. This file has to be copied to the on-board flash drive or, more conveniently, downloaded to the GCS or another computer in the LAN via FTP. Data is not currently written to the flash drive while the flight code is running because of the additional system load that would be generated.

CHAPTER III

NAVIGATION SOFTWARE AND SIMULATION

Due to its multiple sensors, the software to provide sensor fusion (navigation) for a UAV avionics system has many elements. The sensor drivers and navigation algorithms, that provide the state estimate to the flight controller, need to be verified and tested thoroughly. Although experiments with ground vehicles as in [7] have been successful, numerous developments in the VTOL UAV field like [5] show that the use of simulation promises the best results in catching errors early. Therefore, a simulation model of the aircraft and its sensors is used to conduct the necessary tests before actual flight tests. The sensor emulators are designed with an error model with parameters for noise and bias. Models of the aircraft actuators are also included in the simulation. Once this capability is achieved, several simulation configurations are possible, e.g. only in software or including elements of the flight hardware, such as the flight computer executing its navigation functions.

3.1 Software Architecture

Software for the flight hardware and the ground control station needs to be implemented. As discussed above, the tool of simulation shall be used to test the system. This requires that the flight code and the ground control code should also be able to run in the simulation environment. The simulation tool ESim is used as a basis for all code developed within the project. ESim has a graphical user interface (GUI) with numerous visualization functions. All data structures of the developed project can be viewed and reinitialized in a real-time data browser window. A 3-D representation of the simulated vehicle can be displayed in a computer generated scene. Every variable in the simulation can be plotted over time or any other variable and saved as a Matlab data file for further evaluation. Furthermore, ESim has a scripting language to simplify configuration and operation.

As the graphics output is based on the OpenGL standard and communication routines

are implemented for Windows NT and POSIX based systems, ESim can run on a variety of desktop and laptop computers. ESim can be operated under Windows 2000 or Linux.

3.1.1   Flight Code

Navigation filter algorithms, flight control routines, and sensor and actuator drivers comprise the flight software. Unlike the desktop or laptop computers for simulation and ground interface, the flight computer is operated under the Windriver VxWorks real-time operating system. Navigation filter updates are triggered by new IMU data at 100 Hz, while the flight control code and the actuators are updated at every other time step (50 Hz). The code can be compiled and executed as a stand-alone VxWorks application for the flight hardware or it can be integrated into the ESim simulation environment.

In its current configuration the flight code only has one main loop that is executed at the forementioned 100 Hz. At every loop each sensor port is polled for new data, to ensure that the latest data is always used in the following filtering algorithm.

3.1.2   Simulation Environment

ESim can be set up for a piloted simulation of the aircraft, a software-in-the-loop or hardware-in-the-loop simulation. ESim features many data plotting and recording functions for any variable values within the simulation.

3.1.3   Ground Control Station Software

The ground control station software provides the main user interface for the operator, which requires similar visualization capabilities as within a simulation of the system. Therefore ESim can also be configured to perform these tasks.

The ground station screen (figure 11) shows a 3-D representation of the vehicle in computer generated scenery, that supports a trajectory and actual flight path plotting.

An annunciator panel summarizes the system status and warns in case component failure. A command prompt gives the operator the ability to execute command scripts to trigger on-board functions including navigation system initialization, data recording, and
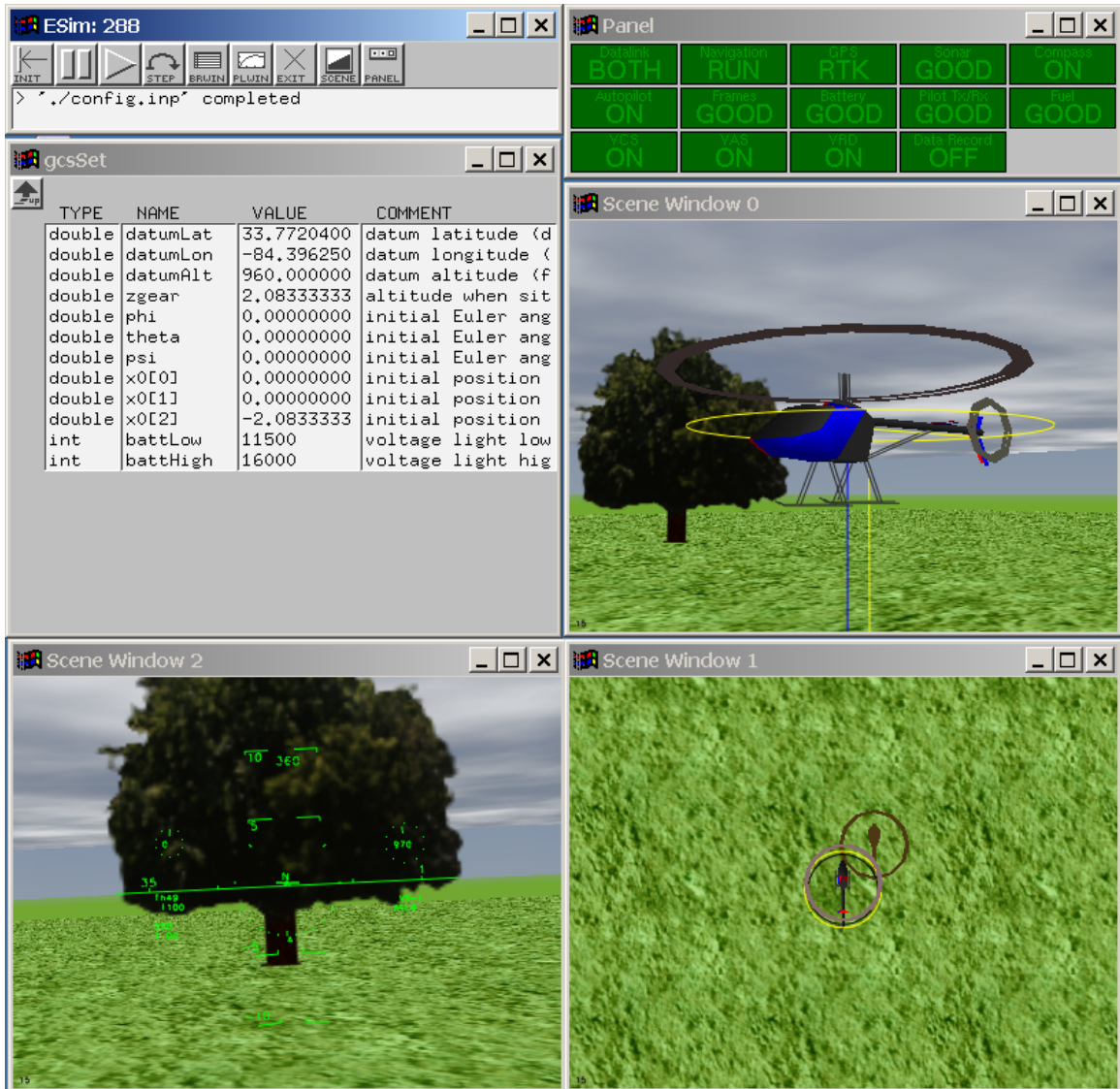
Figure 11: Sample ESim GCS screenshot

flight maneuvers. An important feature derived from the ESim data structure management, is the request of data structures from the aircraft. A "get" command enables the ground crew to retrieve any data structure from the helicopter while in flight, display it in the ESim browser window, modify it if needed, and eventually upload it again with a "send" command.

3.2   Sensor Fusion

Data from the flight sensors is fused into one navigation solution in a state estimator. An Extended Kalman Filtering approach is used to integrate data from the navigation sensor equipment. Georgia Tech's R-Max platform adds two altimeters with different range and accuracy and a magnetometer to a DGPS-aided Inertial Navigation System as shown in figure 12. Propagated IMU-Data (treated as continuous) is fused with discrete updates from the other sensors. Table 1 shows the update rates of the filter inputs. It should be noted, that the integration of the Yamaha YACS IMU data into the navigation filter is not considered in this document.

Utilized states in the navigation filter include: position, velocity, attitude (quaternion), accelerometer and gyro biases, and the terrain height, totaling 17 states.

$$\hat{X} = \begin{bmatrix} \hat{x}^T & \hat{v}^T & \hat{q}^T & \hat{b}_a^T & \hat{b}_\omega^T & \hat{h}_t \end{bmatrix}^T \tag{1}$$

The basic design for the navigation filter is described in [3]. Modifications to that approach include the addition of the terrain height state and improved integration of the magnetometer. GPS data is converted into a cartesian north-east-down coordinate system as a flat earth is assumed. IMU propagation is based on rigid-body dynamics.

The IMU propagation equation $\dot{X} = F(X, a, \omega)$ (see equation 3) already contains correction for the IMU mounting position r_{IMU} and corrects for the IMU biases $\hat{b}_a$ and $\hat{b}_\omega$ through equation 2.

$$\begin{aligned} \bar{a} &= a_{IMU} - \hat{b}_a \\ \bar{\omega} &= \omega_{IMU} - \hat{b}_\omega \end{aligned} \tag{2}$$

$$\dot{\hat{x}} = \hat{v},$$

$$\dot{\hat{v}} = \hat{T}_{b \to i} \left(\bar{a} - \dot{\bar{\omega}} \times r_{IMU} - \bar{\omega} \times (\bar{\omega} \times r_{IMU}) \right) + g,$$

$$\begin{bmatrix} \dot{\hat{q}}_0 \\ \dot{\hat{q}}_1 \\ \dot{\hat{q}}_2 \\ \dot{\hat{q}}_3 \end{bmatrix} = \begin{bmatrix} 0 & -\bar{\omega}_x & -\bar{\omega}_y & -\bar{\omega}_z \\ \bar{\omega}_x & 0 & \bar{\omega}_z & -\bar{\omega}_y \\ \bar{\omega}_y & -\bar{\omega}_z & 0 & \bar{\omega}_x \\ \bar{\omega}_z & \bar{\omega}_y & -\bar{\omega}_x & 0 \end{bmatrix} \begin{bmatrix} \hat{q}_0 \\ \hat{q}_1 \\ \hat{q}_2 \\ \hat{q}_3 \end{bmatrix}, \tag{3}$$

$$\dot{\hat{b}}_a = 0,$$

$$\dot{\hat{b}}_\omega = 0,$$

$$\dot{\hat{h}}_t = 0.$$

The covariance propagation is:

$$\dot{P} = A(\hat{X}, \bar{\omega})P + PA^T(\hat{X}, \bar{\omega}) + Q. \tag{4}$$

where A is the derivative of the state vector estimation equation with respect to the states:

$$A = \frac{\partial F}{\partial X} \Big|_{X = \hat{X}}. \tag{5}$$

The effects of process noise are added through the assumed matrix Q.

$$Q_{mn} = 0 \quad \forall \quad m \neq n$$

$$Q_{11} = 0; \quad Q_{22} = 0; \quad Q_{33} = 0$$

$$Q_{44} = \sigma_{a_x}^2; \quad Q_{55} = \sigma_{a_y}^2; \quad Q_{66} = \sigma_{a_z}^2$$

$$Q_{7-10,7-10} = \sigma_\omega^2; \tag{6}$$

$$Q_{11-13,11-13} = \sigma_{b_a}^2$$

$$Q_{14-16,14-16} = \sigma_{b_\omega}^2$$

$$Q_{17,17} = f(\hat{v})$$

The noise variance values that are currently being used are listed in table 2. Those values were once assumed and then modified in accordance with the results from the first static engine run test as described in 4.2.3. Due to computation power constraints, the
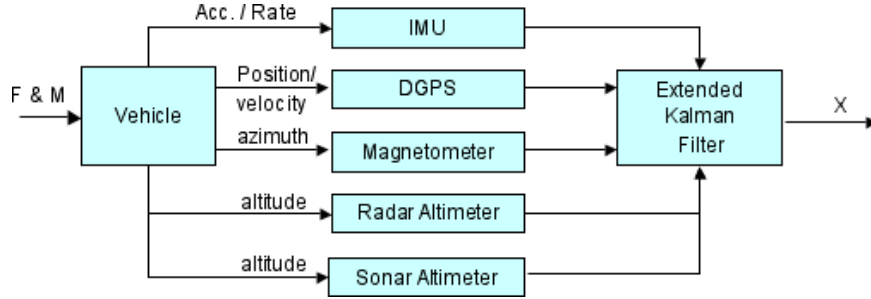
Figure 12: Extended Kalman Filter

Sensor	Rate
ISIS-IMU	100 Hz
YACS-IMU	200 Hz
DGPS	5 Hz
Sonar	10 Hz
Radar	variable
Magnetometer	20 Hz

Table 1: Sensor update rates

17x17 covariance matrix P is only updated at 20 Hz as opposed to the full 100 Hz of the state propagation, assuming that changes in P are relatively slow.

Whenever, new data is available from one of the supporting sensors, an optimal discrete update is applied to the state estimate and the covariance matrix. Equation 7 computes the Kalman Gain matrix K, which describes the weight of the new set of data, where C contains the partial derivatives of the output equation $Y = G(X)$, and $\bar{z}$ is the actual measurement. The state and the covariance matrix are then updated by equations 9 and 10.

$$K = PC^T \left[CPC^T + R \right]^{-1} \tag{7}$$

$$C = \left. \frac{\partial G(X)}{\partial X} \right|_{X=\hat{X}} \tag{8}$$

$$\hat{X}^+ = \hat{X} + K \left[\bar{z} - G(\hat{X}) \right] \tag{9}$$

$$P^+ = P - KCP \tag{10}$$

3.2.1   GPS

Sensor data from the GPS needs to be corrected for their mounting positions with respect to the center of gravity. As GPS measurements have a latency, that needs to be compensated.

IMU	$\sigma^2_{a_x}$	0.01
	$\sigma^2_{a_y}$	0.01
	$\sigma^2_{a_y}$	0.01
	σ^2_{ω}	0.0001
	$\sigma^2_{b_a}$	0.0001
	$\sigma^2_{b_\omega}$	0.000001
GPS (RTK Mode)	$\sigma^2_{GPS_x}$	25.0
	$\sigma^2_{GPS_y}$	25.0
	$\sigma^2_{GPS_z}$	49.0
	$\sigma^2_{GPS_{vx}}$	25.0
	$\sigma^2_{GPS_{vy}}$	25.0
	$\sigma^2_{GPS_{vz}}$	49.0
Magnetometer	σ^2_{ψ}	1
Sonar	$\sigma^2_{h_{sonar}}$	$f(h_{SONAR})$

Table 2: Assumed sensor variances

The discrete update needs to be based on the older state estimate corresponding to sensor latency, but is applied on the current state $\hat{X}$. State updates for position and velocity are applied independently, as the measurements of these values have a different signal latency.

$$\bar{x} = x_{GPS} - \hat{T}_{b \to i} r_{GPS} \tag{11}$$

$$\bar{v} = v_{GPS} - \hat{T}_{b \to i} \bar{\omega} \times r_{GPS} \tag{12}$$

$$C_{GPSpos} = \begin{bmatrix} I_{3x3} & | & 0_{14x3} \end{bmatrix} \tag{13}$$

$$C_{GPSvel} = \begin{bmatrix} 0_{3x3} & | & I_{3x3} & | & 0_{11x3} \end{bmatrix} \tag{14}$$

$$R_{GPSpos} = \begin{bmatrix} \sigma^2_{GPS_x} & 0 & 0 \\ 0 & \sigma^2_{GPS_y} & 0 \\ 0 & 0 & \sigma^2_{GPS_z} \end{bmatrix} \tag{15}$$

$$R_{GPSvel} = \begin{bmatrix} \sigma^2_{GPS_{vx}} & 0 & 0 \\ 0 & \sigma^2_{GPS_{vy}} & 0 \\ 0 & 0 & \sigma^2_{GPS_{vz}} \end{bmatrix} \tag{16}$$

3.2.2 Magnetometer

Data from the magnetometer is evaluated along all three axes and described as a normalized field line vector. However, the data is only applied to correct for heading drift. This is done by projecting the magnetic field vector on the North-East plane, using the last vehicle attitude estimate $\hat{q}$ to eliminate the effects of magnetic dip. The C matrix only allows a change in the state quaternions such that only rotations around the Down axis can be made.

$$C_{Mag} = \begin{bmatrix} 0_{6x1} & | & Cq0 & Cq1 & Cq2 & Cq3 & | & 0_{7x1} \end{bmatrix} \tag{17}$$

$$R_{Mag} = \begin{bmatrix} \sigma_\psi^2 \end{bmatrix} \tag{18}$$

3.2.3 Sonar and Radar Altimeter

The sonar and radar altimeters are used to do discrete updates on the vertical position and the terrain height. Raw sensor data needs to be corrected for the mounting point with respect to the vehicle center of gravity. It is assumed that the sensor has enough beam width to identify the closest distance to the ground which is ideally straight down. Therefore, no additional corrections are applied when the vehicle is not in level attitude.

$$\bar{h}_{son} = h_{SONAR} - \hat{T}_{b \rightarrow i}[3] r_{SONAR} + \hat{h}_t \tag{19}$$

$$C_{Sonar} = \begin{bmatrix} 0 & 0 & -1 & | & 0_{13x1} & | & 1 \end{bmatrix} \tag{20}$$

$$R_{Sonar} = \begin{bmatrix} \sigma_{h_{sonar}}^2 \end{bmatrix} = f(\bar{h}_{son}) \tag{21}$$

3.2.4 Filter Initialization

During the initialization of the navigation filter, sensor data is averaged over a few seconds while the helicopter is sitting on the ground and then applied as follows. The initial position is based solely on GPS data with the velocity being set to zero. Initial IMU data is used to determine the roll and pitch attitude, while magnetometer data is used to identify the initial heading of the aircraft. IMU accelerometer biases are set to zero. IMU angular velocity
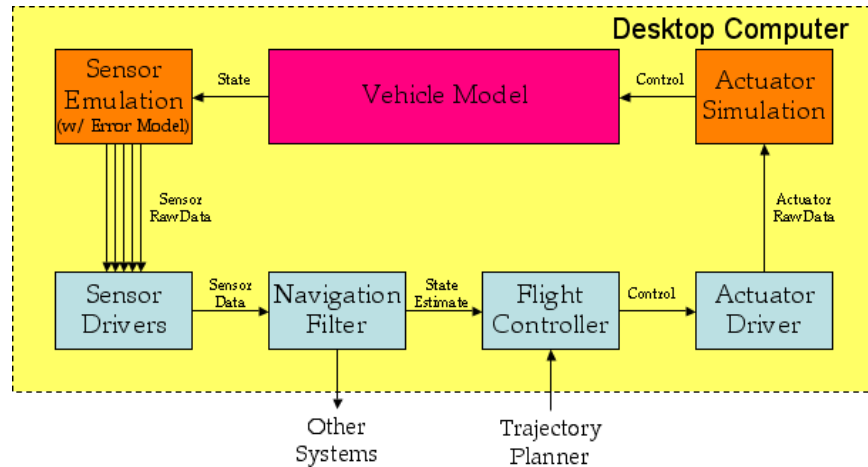
Figure 13: Software-in-the-Loop simulation

biases are measured. The terrain height is based on the GPS altitude compared with the sonar.

3.2.5   Tuning the Filter Settings

The initial navigation filter setup is heavily based on assumed values. It is necessary to tune the filter by adjusting these values based on the actual sensor characteristics of the avionics system. Ground and flight testing was used to determine those characteristics. The adjustable values within the filter are the sensor variances as in table 2 and the latency of the each sensor. The selection of these parameters is discussed in sections 4.2.1 and 4.2.3.

3.3   Software-In-The-Loop (SITL) Simulation

The first test of the algorithms is performed within the simulation. Figure 13 shows a possible setup to test navigation, flight controls, and sensor and actuator drivers in one step, without the neccessity of employing actual flight hardware. Configuration changes are only performed within the software. Additionally, this configuration enables multiple developers, as there is no need to run software tests on a dedicated hardware that is only accessible by one developer at a time. This of course implies that it is also possible to test new code at times when the on-board hardware is being disassembled or even non-existent.

The output of the vehicle model is assumed to be the state of a real helicopter. The
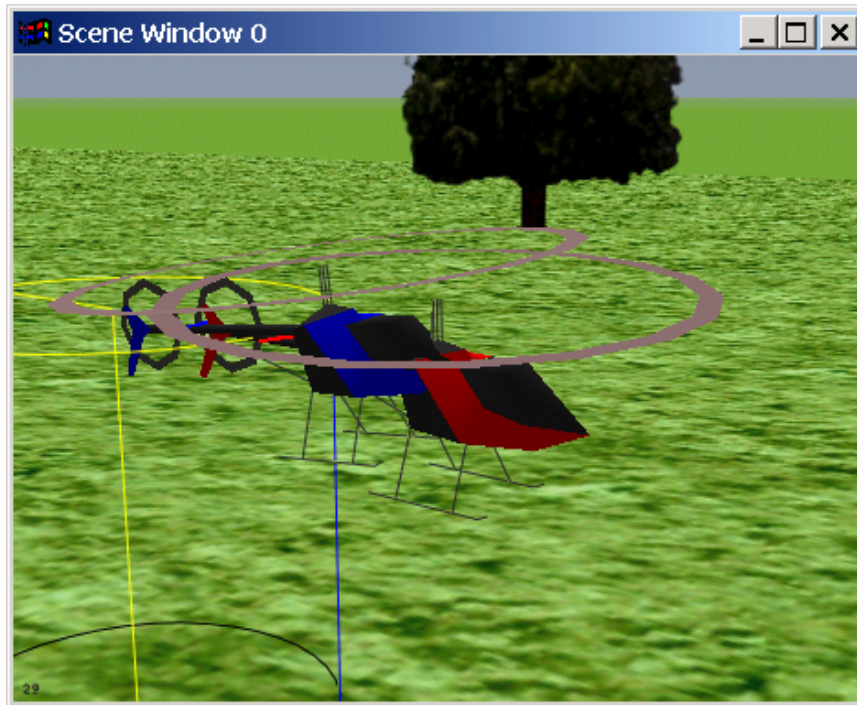
Figure 14: Navigation solution following the simulated "truth"

goal of the navigation system is to estimate the vehicle state as accurately as possible. If the errors in the sensor and actuator models are set to zero, the estimate of the navigation filter should exactly equal the state of the model. When noise and biases are introduced and increased the navigation solution will deteriorate. A filter should have a reasonable tracking error within the estimated errors for the sensors. When the first flight test data is available, the simulation can be repeated with more accurate information about actual sensor performance. Additionally, a sensor data playback functionality is built into the simulator, such that prerecorded sensor data from actual flight can be fed into the navigation filter. This proves to be very useful to test the function of the navigation filter.

ESim provides a visualization interface as follows. Two 3-D visualizations of the helicopter are displayed in the same window using different colors, one for the "truth" (the model state), and one for the navigation estimate as shown in the screenshot in figure 14. Discrepancies between the two solutions can be easily visualized in that way. Of course, there is the additional option of plotting the desired values for comparison.

3.3.1 Sensor Emulation and Error Models

The aforementioned sensor emulation can be described as follows. State data from the dynamic model is transformed into measurements of those values each sensor actually measures. Coordinate transformations and shifts as they are mentioned in section 3.2 are applied in reverse. Then, noise and biases are added to those pseudo measurements. Eventually the simulated values are used to fill the binary data structures of the sensors communication protocol and can be sent to the sensor drivers in the simulation.

3.4 Hardware-In-The-Loop (HITL) Simulation

During the HITL simulation, the flight software runs on the actual on-board computer while all simulations run on a separate desktop or laptop computer (see figure 15). Instead of the sensors, I/O ports of the Simulation Computer are plugged into the flight computer (RS-232 Null-modem cables for the R-Max system). The same goes for the actuators in this example. The result of this setup is that the on-board computer effectively "thinks" it is flying the vehicle, as all of its configuration/data flow is identical to an autonomous flight setup. Most importantly, this simulation will show if the selected on-board hardware is capable of executing the flight software in real-time, i.e. it shows if all components can actually run fast enough.

Using the software architecture approach described above, the same software as for the SITL simulation is used. The only difference is that the on-board part of the code is compiled for the flight operating system and uploaded onto the flight computer. The simulation interface with the visualization and data recording stays totally identical. High bandwidth data recording, however, has to be done on the flight computer, controlled by the GCS software (section 3.1.3). It is possible to do a complete HITL simulation on three separate computers: the flight computer, the simulation computer, and GCS computer. For practicality reasons, the simulation and GCS code is usually executed on the same machine. This is possible beacause the data structures for these tasks are independently organized under the ESim environment.
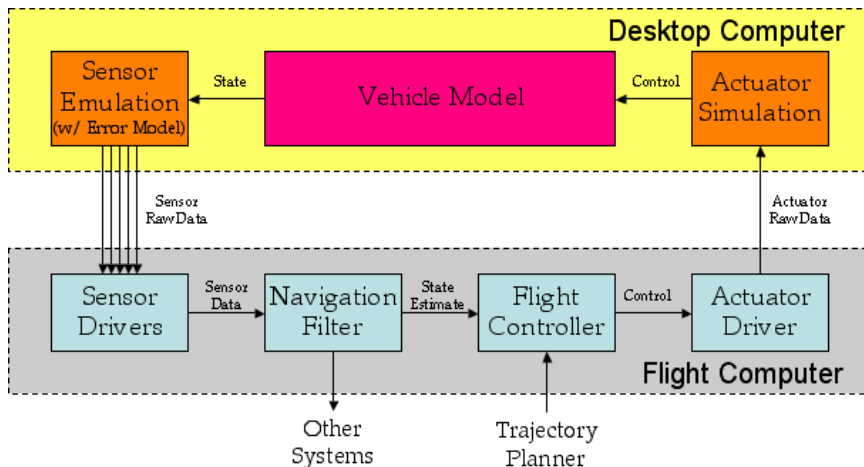
Figure 15: Hardware-in-the-Loop simulation

As the back of the avionics box on the helicopter is easily accessible, the serial connections on the flight computer can be easily unplugged as required and connected to the simulation computer. A "partial" HITL simulation is also possible. For example, generated GPS data can be fed into the flight computer for tests in flight configuration without GPS availability, e.g. when the aircraft is located indoors. This flexibility was important because of the R-Max UAV's mission as a research vehicle.

Alternatively to the direct serial port hookup, a communication protocol has been implemented that allows serial data to be sent via Ethernet. This is done, by emulating serial port behaviour and then rerouting the data stream over a TCP/IP socket connection. Hence, HITL simulations can be conducted while the avionics are wired for flight, which is especially useful for last minute function checks before a flight test. However, experience with this method showed a bandwith limitation when applied to replace all serial hardware links to the simulation computer.

CHAPTER IV

TEST RESULTS

As the development of the UAV system nears its end, it needs to be tested on the ground and in the air before autonomous flight can be safely attempted. This chapter describes the order and method that was used to evaluate the system.

At the time of this testing series, the radar altimeter was inoperable and could not be integrated into the system. As discussed later in the chapter the operation of the sonar altimeter did not deliver usable data, although the same design worked properly on the helicopter UAV described in [6]. Accordingly, no altimeter data, except for the GPS height, was available during the tests.

4.1 Simulation Results

The navigation system was tested within the discussed simulation architecture (SITL and HITL). Tests included automatic hover, high speed maneuvers and pirouettes, as well as remotely piloted flight (with a human pilot in the simulation loop). A sample position and attitude plot from that series is shown in figure 16, where the helicopter takes off to an altitude of 10 feet, hovers, and then translates 100 ft forward into a new hover position. Special attention was placed to the system behaviour in case of GPS signal loss, step changes in IMU biases, and in- and out-of-range situtations of the sonar altimeter. Two of these cases will be explained in the next paragraphs.

Figure 17 shows the effects of a GPS malfunction on the position estimate. The aircraft model was bounded to a fixed position to counter unwanted flight controller corrections for that test. The first plot in the figure shows one of the position channels of the GPS receiver as a reference for the sensor outage. The other plots show the position estimate, which starts drifting in the North-East Plane, while no drift occurs in the Down direction. Sensor noise and errors from the IMU cause this drift as there is no other means of getting position

information. The altitude (z axis) however is still being updated by incoming sonar data. When GPS data is being reaquired, the position is discretely updated immediatly. The small jump in the altitude when GPS is enabled again is relatively small.

Because the sonar altimeter has a very limited range, the behaviour of the navigation filter needs to be analyzed to ensure a consitent altitude and terrain height estimate with intermittent sonar data. A possible problem scenario for the filter would be a flight right at an altitude which is very close to the maximum sonar range. In figure 18, the aircraft is hovering 30 feet off the ground and starts a climb to 50 feet, which exceeds the range of the sonar, and descends back to 30 feet. The altitude estimate does not show any significant edges which could influence the altitude control loops. The terrain height estimate (which is set to constant zero in the simulation) only shows fluctuations within 0.2 feet.

4.2   Ground Tests

4.2.1   Navigation System Test

The R-Max avionics package has been successfully tested in a basic configuration, while mounted on a pickup truck (figure 19). A state vector of the vehicle was determined and transferred back to the Ground Control Station (GCS) via wireless data links. The first position plot is depicted in figure 20. The GPS was operated in single-point mode with an update rate of 1 Hz in this test. This represents the results prior to vibration testing (see section 4.2.3) and the improvements discussed in section 4.3.1.

Further ground tests of the navigation system with enabled differential GPS have been conducted to experimentally identify the latency of the GPS receiver and update the corresponding values in the navigation filter settings. When set properly, the GPS corrections should be relatively small during a dynamic response.

4.2.2   EMI Test

In order to test the EMI shielding of all system components, the assembled UAV is placed on the flight field. When all systems were powered up on external and on-board (with engine running) power everything worked as expected. No malfunctions were noted. Special

attention needs to be placed to the function of the data links, especially as the safety pilot link is essential for safe operation in case of any system failures during the tests.

A link quality test of the remote control transmitter is performed with removed antenna. Its range was checked with powered external (non-Yamaha) avionics and compared to a clean configuration. The resulting range was sufficient to ensure flight safety. However, it was found that the range of the remote control link can be slightly affected when the helicopter is close to the ground station FreeWave antenna, although the data modems operate on a completely different frequency band (72 Mhz vs. 900 Mhz).

4.2.3   Vibration Test

A first engine run test with the avionics mounted on the airframe indicated sufficient vibration isolation. All on-board electronics worked properly while sensor data was recorded at 100 Hz. GPS and magnetometer data is almost identical to engine-off measurements while the IMU raw data (figure 21) shows significant vibration. However, when integrating that data it delivers smooth and stable position, velocity, and attitude solutions. This basically means, that there is no significant aliasing of high frequency IMU input, (i.e. the unit's resolution lies above the vibration frequencies of the avionics box). Thus, it is possible to propagate the IMU data and use it for navigation.

The ground vibration test is also the first chance to analyze the raw sensor data for the magnitude of sensor noise, which has to be known for optimal navigation filter function. First values, that can be determined in this test, can immediatly replace initially assumed navigation filter settings and can be used to improve the sensor models in the simulation for more realistic SITL and HITL results.

4.3   Flight Tests

4.3.1   Navigation System Test

A couple of flights were conducted to test the performance of the navigation system and to record sensor data in flight. Raw sensor data plots from one of the flight can be found in appendix B.

The first flight test showed problems with the sonar altimeter which delivered inconsistent altitudes while in flight. This problem is probably caused by vibrations of the airframe that trigger the sensitive ultra-sonic transducers in the unit prematurely. Better vibration isolation for this component might solve that issue in the future.

Although the navigation performance recorded on the first flight test day was poor, the recorded sensor raw data was used to solve these problems. By using the data playback feature of the simulator, it is possible to rerun the sensor data through a modified navigation filter. Based on that flight data, the process noise of the terrain height $\sigma_{h_t} = f(\hat{v})$ was lowered. Additionally, it was discovered that bias errors in the pitch and roll of the magnetometer were causing erroneous accelerometer bias states. Therefore, the magnetometer data processing was changed from a three axis attitude update to the present one channel heading correction, as described in section 3.2. Comparing figure 22 with figure 23, the plot gets much smoother after the navigation filter has been updated.

4.3.2 Flight Controller Testing

When the navigation system was optimized to the point where it would output acceptable state data, first tests of the flight controller were conducted. Although the flight controller was not discussed in this document, its testing was supported by the overall development concept presented here.

The controller was tested by closing only one loop at a time. While the heading and altitude hold modes worked right away, the control loops for pitch and roll did not have enough authority to stabilize the helicopter in hover. By using the on-board data recording functionality of the flight software, the flight could be reconstructed and the controller behaviour analyzed and improved for the next flight test. Similar to the tuning of the navigation filter, the data playback in the simulation was a key feature to improve system behaviour without having to have additional flight tests.
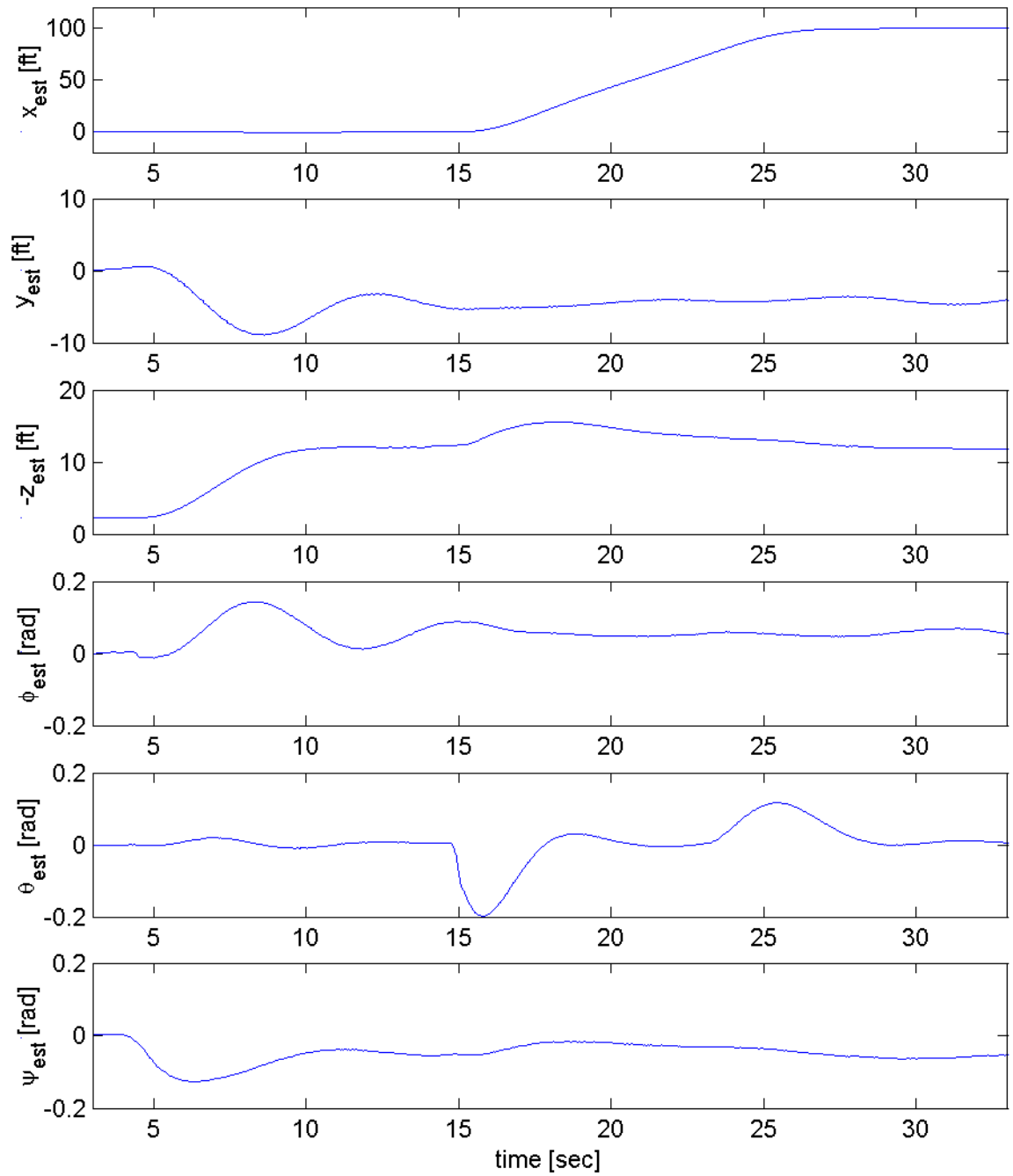
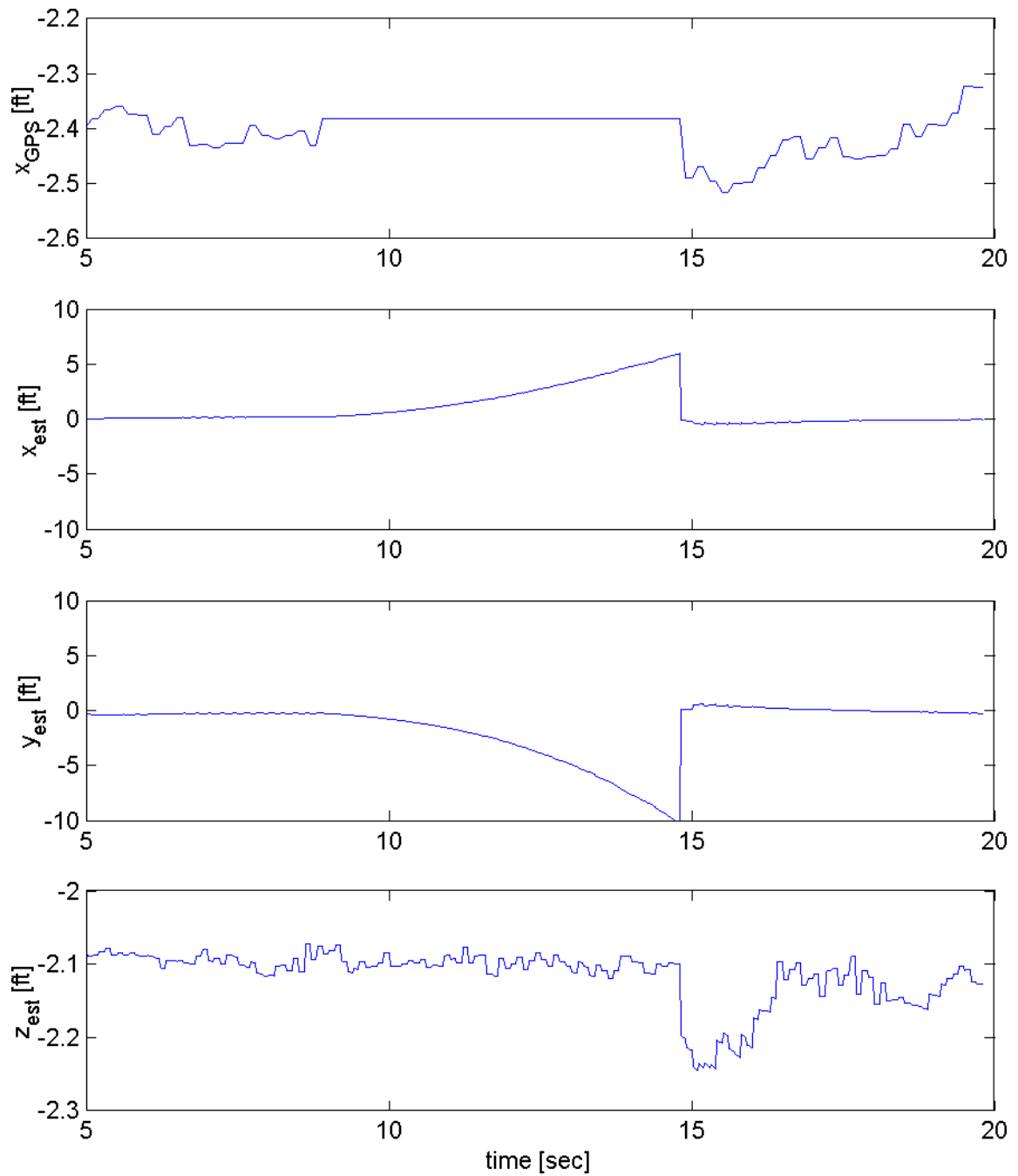Figure 16: Simulated flight, take-off, 100 ft forward, 30 sec

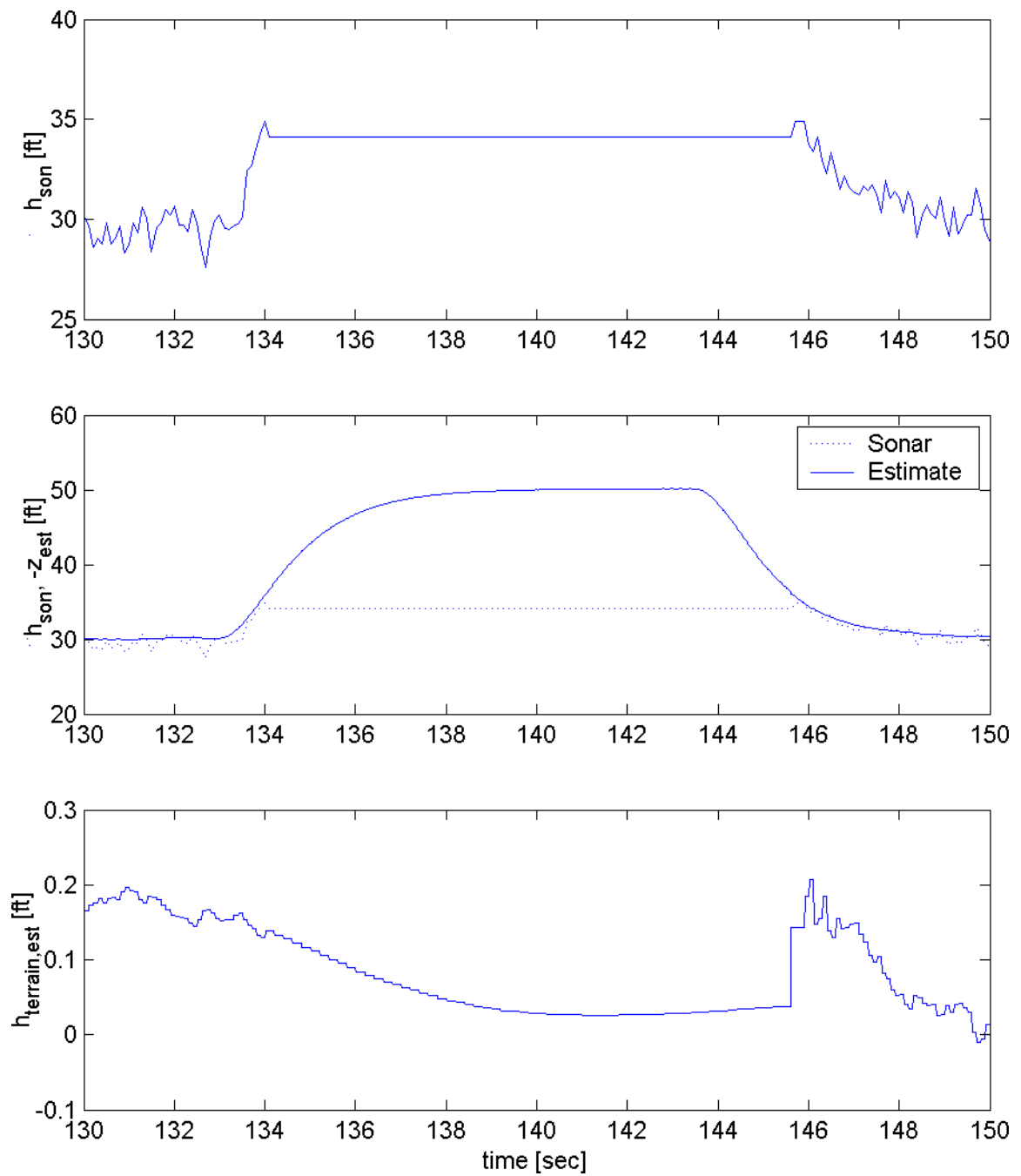Figure 17: Simulated GPS malfunction, on the ground, 15 sec

Figure 18: Sonar out-of-range problem, simulated flight, 20 sec

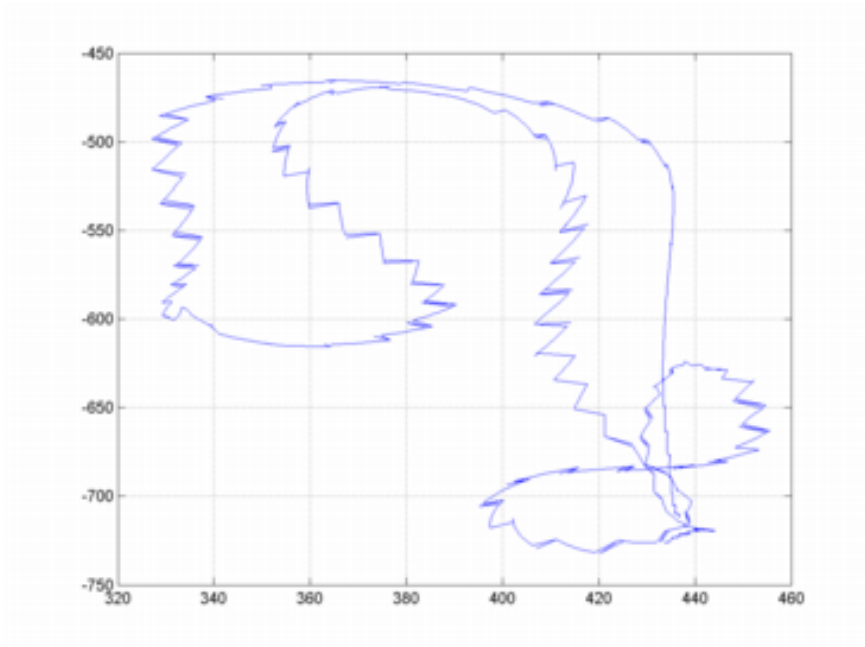Figure 19: R-Max avionics on test vehicle
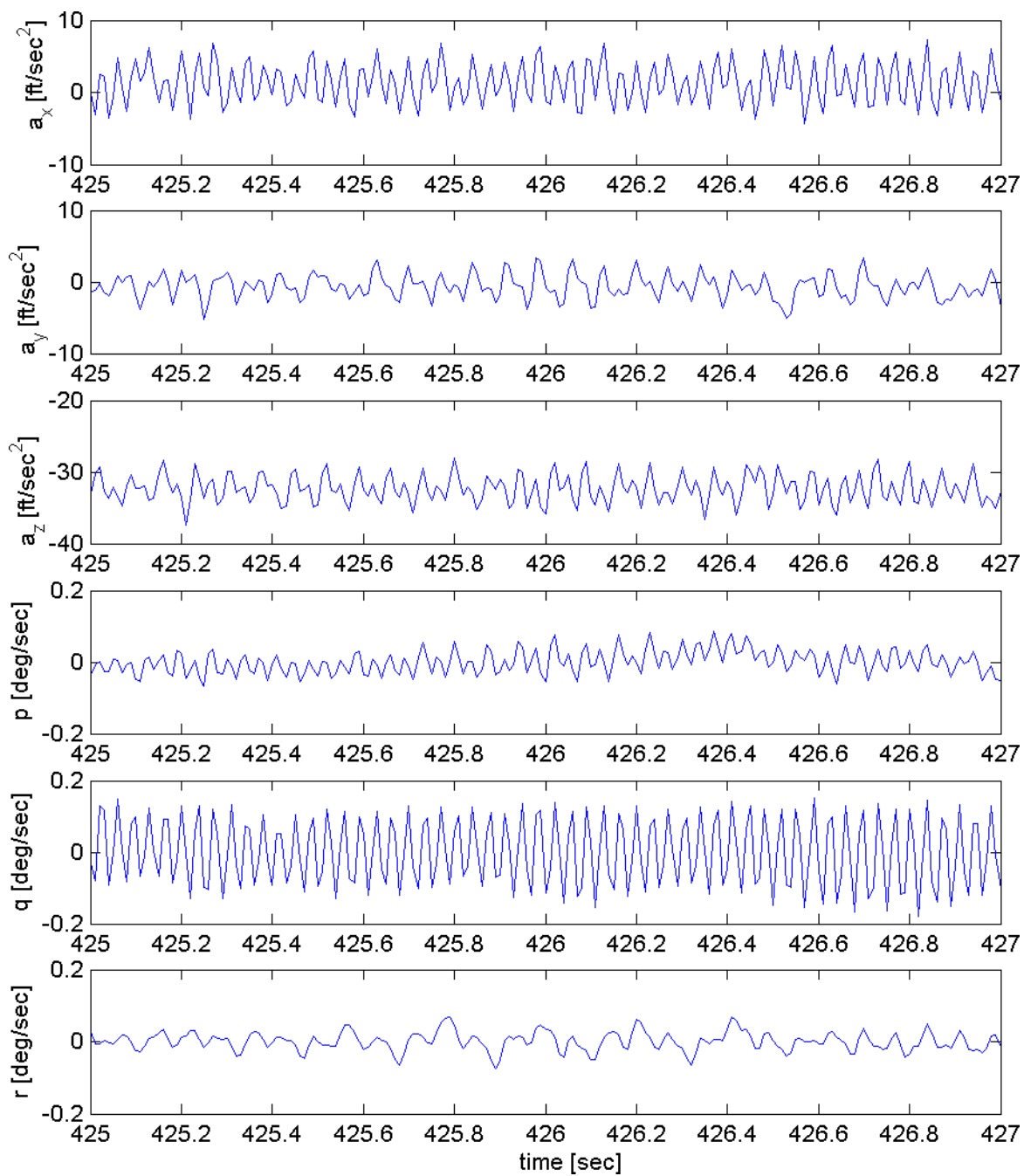

Figure 20: X-Y Plot of the first ground test

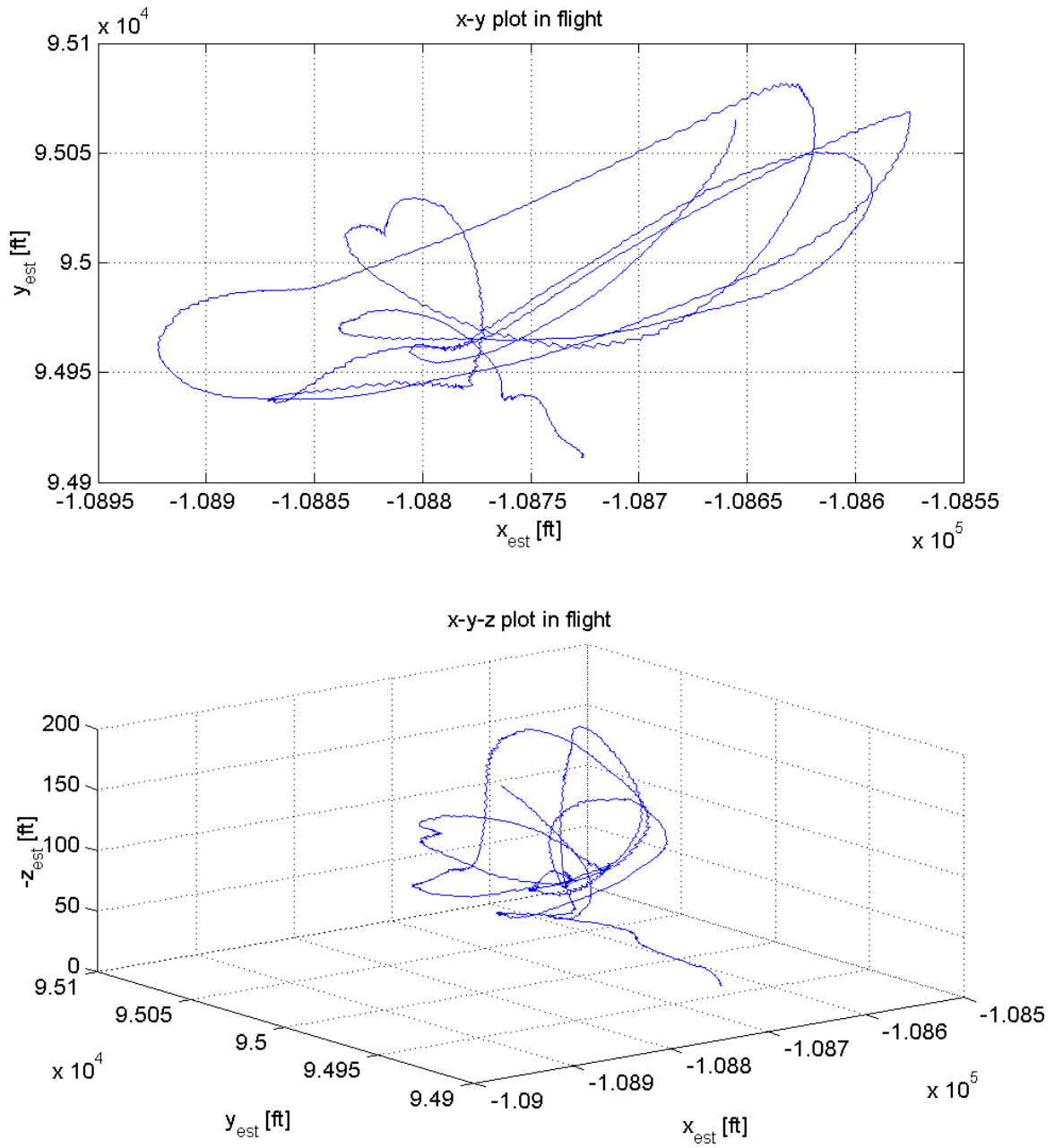Figure 21: IMU raw data on the ground, engine running, 2 sec

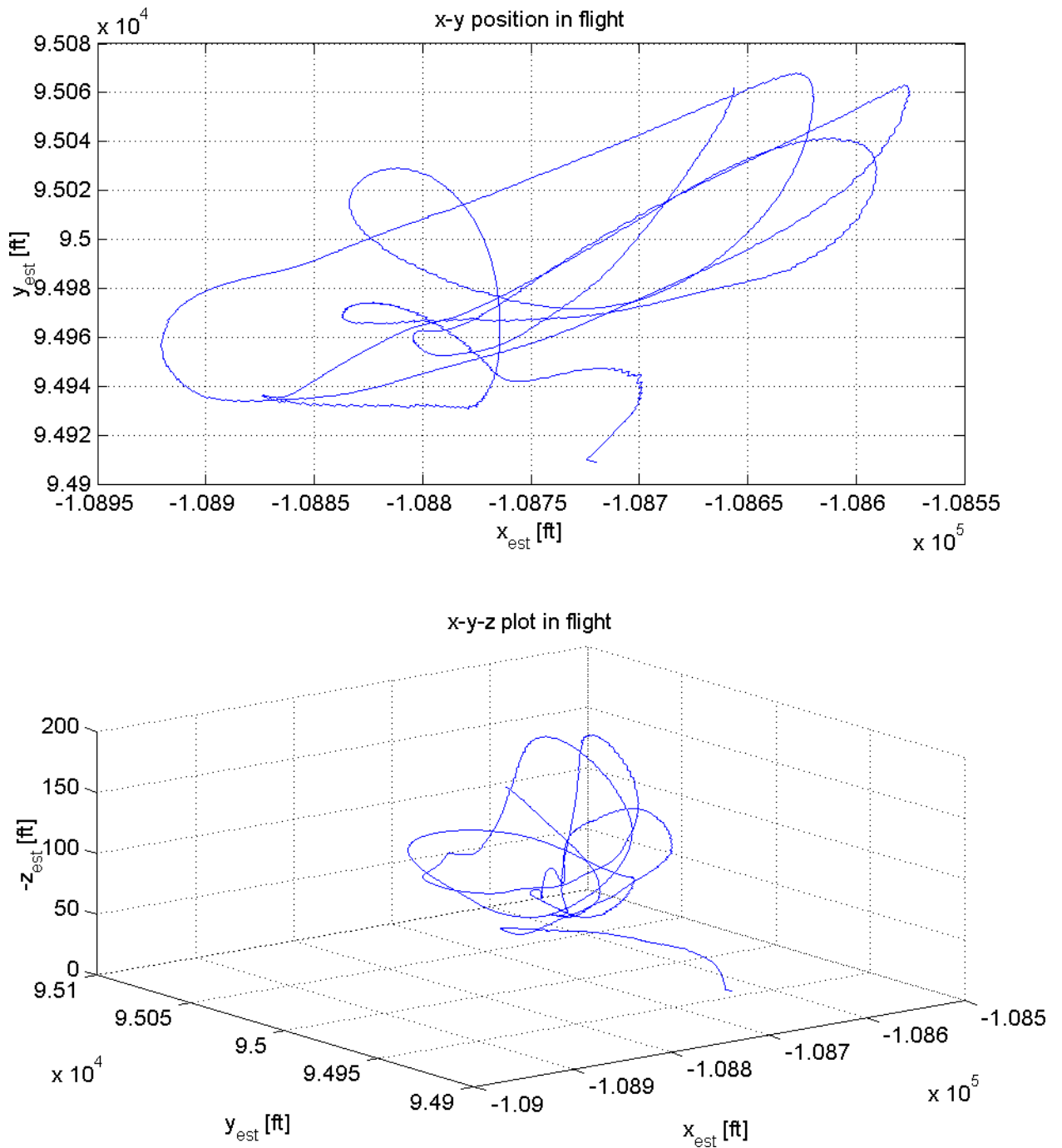Figure 22: Estimated flight trajectory, 110 sec, recorded in flight

Figure 23: Estimated flight trajectory, 110 sec, sensor data playback with an improved navigation filter

CHAPTER V

CONCLUSIONS

The navigation system of the R-Max UAV yielded very good performance, close to the predictions from simulation. By using simulation and sensor data playback, expensive flight test time to develop the navigation system could be reduced. The utilization of a software architecture, which is identical for SITL/HITL simulation, flight code and Ground Control Station, also contributed to significant time savings in the system development.

Besides the software benefits, the hardware integration approach gives the R-Max UAV a great extendibility, which is supported by its unique reconfiguration flexibility. It is the combination of hardware and software integration which makes the Georgia Tech UAV testbed an ideal platform for advanced navigation and flight control research.

5.1   Future Plans and Improvements

The validation of the R-Max navigation system will continue, with the near term goals of getting the sonar, the radar, and the Yamaha YACS-IMU operational, and including them into the navigation filter. The R-Max UAV will then be well prepared for its role as a test platform and can be used as a research and demonstration vehicle for flight control systems.

Future research may also be conducted on a modified or reconfigured navigation system, or computer based vision systems might be included into the avionics package.

5.1.1   Alternate Setup Proposal

An example of a different experimentation setup of the R-Max UAV platform, could be a redundant system, where most components are allowed to fail. Such a configuration could be realized as follows. By adding a second, smaller and simpler GPS receiver, as well as a second flight computer. The first computer could be connected to the ISIS-IMU, the RT-2 GPS, the Sonar and the FreeWave data link. That would leave the YACS IMU, the second

GPS and the Radar, and Ethernet data link to connect to the alternate flight computer. That way, there would be two independent navigation and control systems. Whenever one component of a system fails, the other system could take over. However, the bottleneck of the helicopter flight control interface cannot be made redundant, and would have to be via some kind of switching board to both flight computers.

On the hardware side, the only changes neccessary for this example would be to equip the auxiliary module with a flight computer and an additional GPS receiver and to change the patch connections on the back side of the modules. The new configuration with a switching logic, could be implemented and tested in simulation in the same way that it has been for the basic configuration described in this document.

APPENDIX A

AIRCRAFT SPECIFICATIONS

Yamaha R-Max industrial helicopter

| Dimension	metric	imperial
Total Length	3630 mm	142.9 in
Rotor Diameter	3115 mm	122.6 in
Tail Rotor Diameter	545 mm	21.5 in
Fuselage Length	2750 mm	108.3 in
Fuselage Width	720 mm	28.3 in
Fuselage Height	1080 mm	42.5 in
Empty Weight	58 kg	127.6 lb
Gross Weight	93 kg	204.6 lb
Payload	30 kg	66 lb
Engine		
Type	Gasoline 2 cycle	
Configuration	2 cyl. hor. opposed	
Cooling	water cooled	
Displacement	246 ccm	15 cin
Engine RPM (Hover)	6350 min^{-1}	
Power Output	15.4 kW	21 hp
Max. Torque	25.5 Nm	3.36 lb-ft
Fuel Capacity	6 l	1.6 gallons
Endurance	60 min	

APPENDIX B

SENSOR RAW DATA
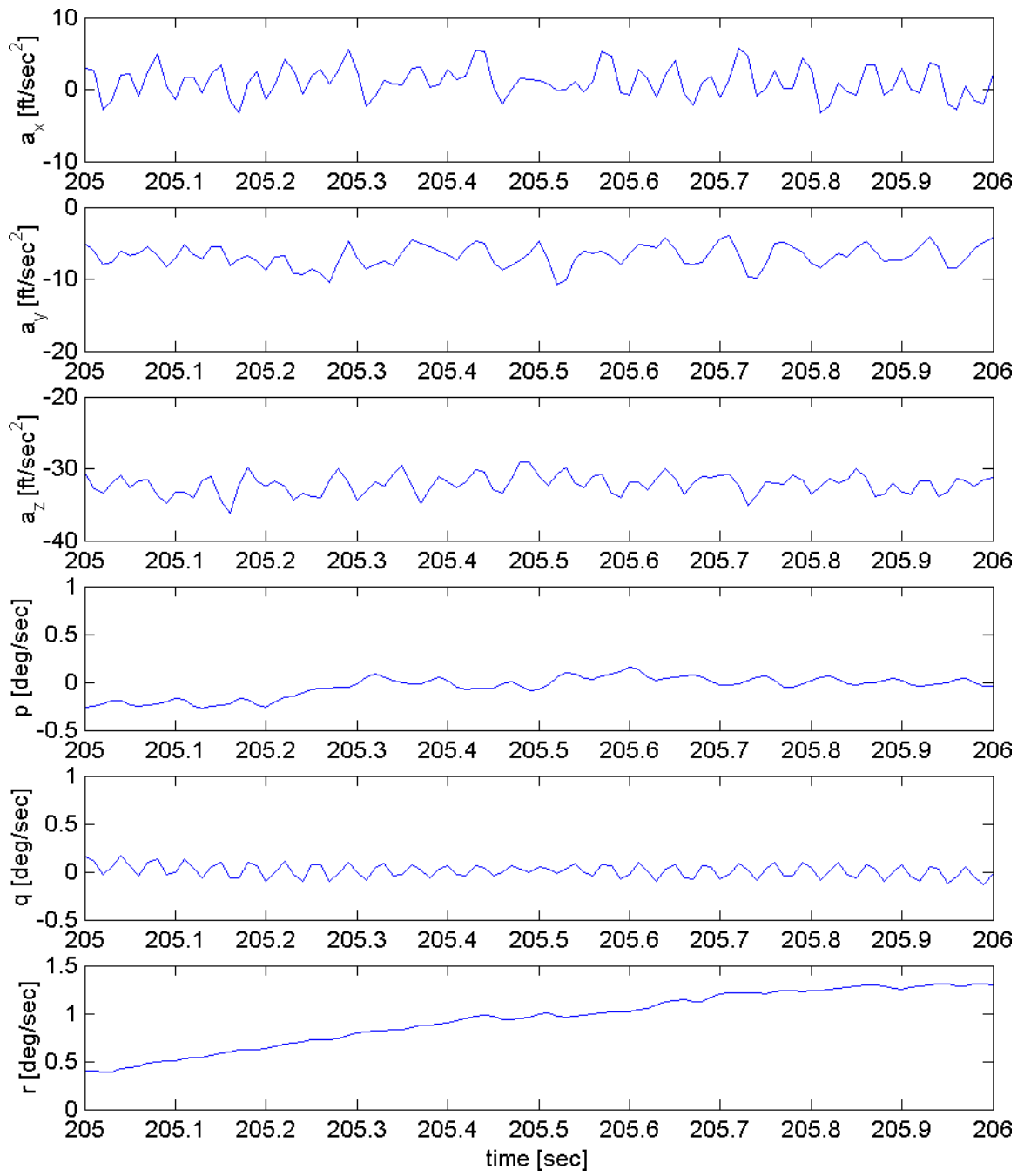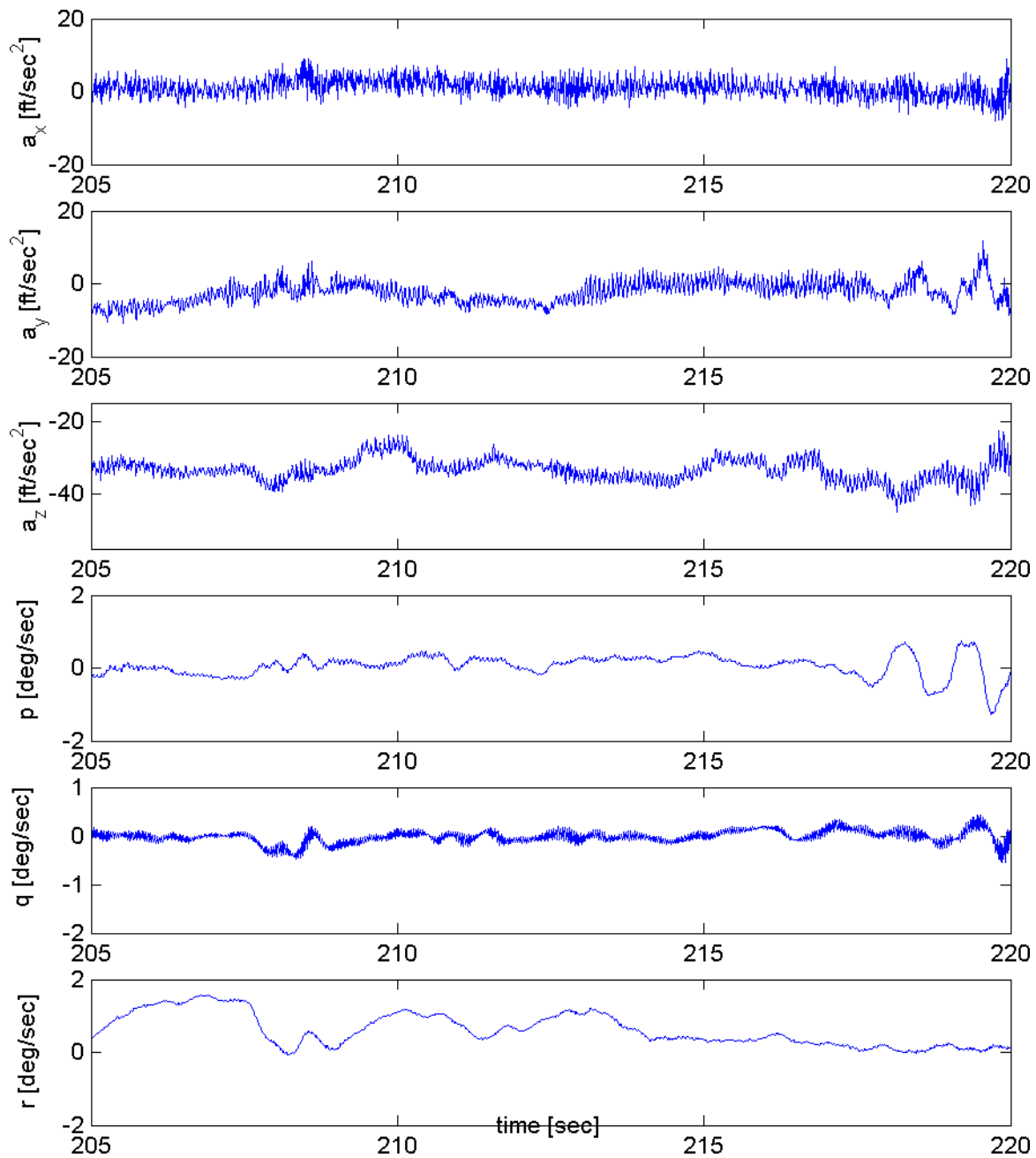

Figure 24: IMU raw data in flight, 1 sec

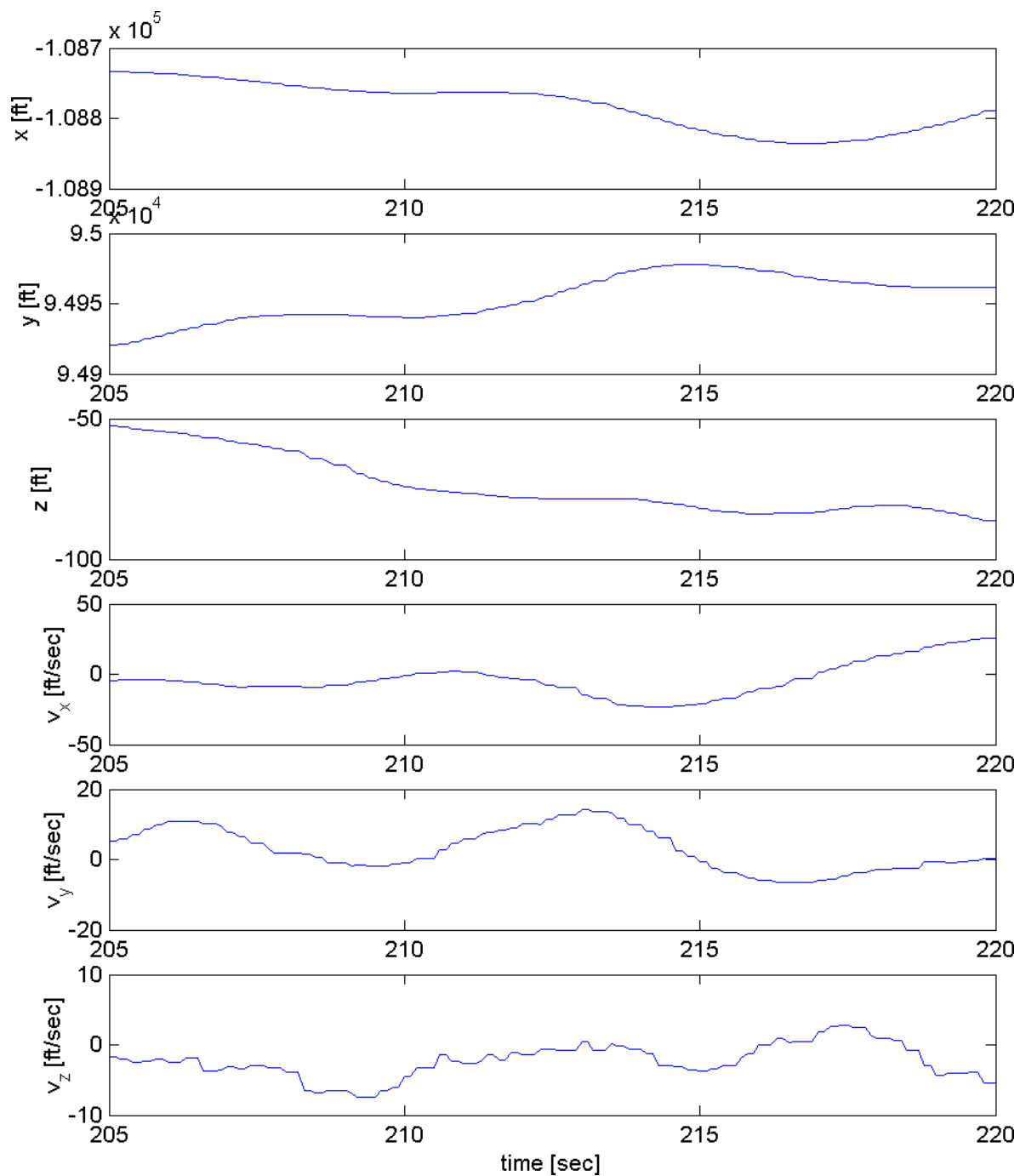Figure 25: IMU raw data in flight, 15 sec
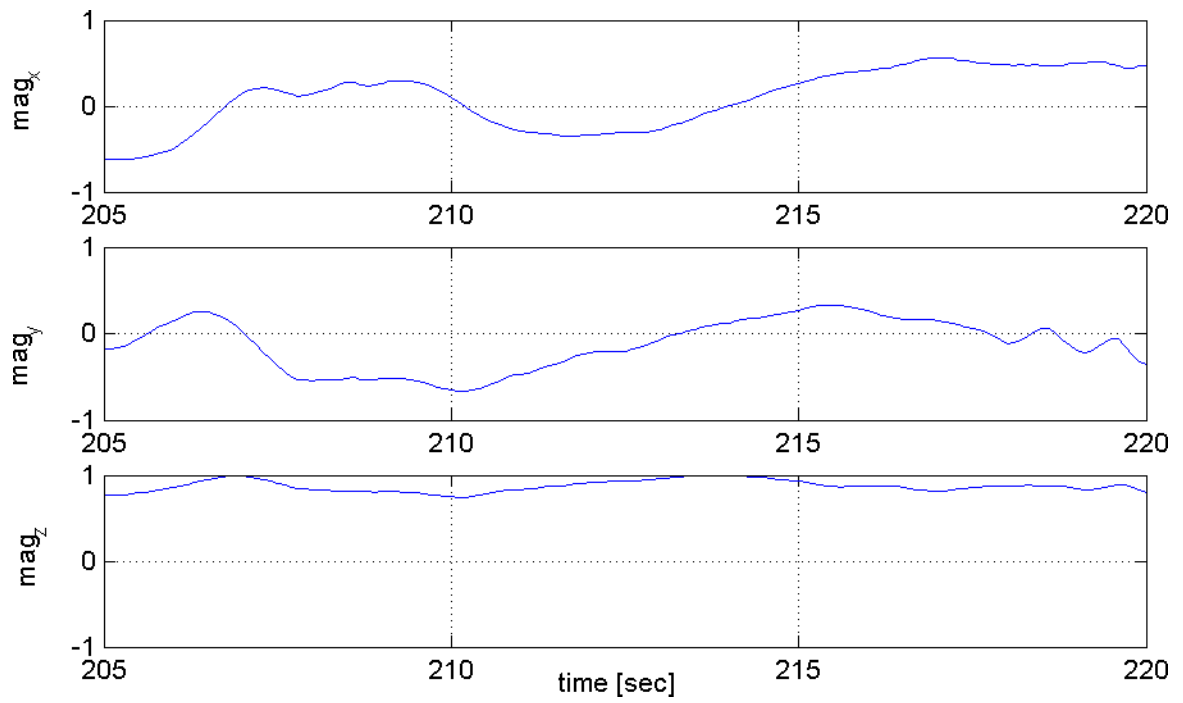
Figure 26: GPS raw data in flight, 15 sec

Figure 27: Magnetometer raw data in flight, 15 sec

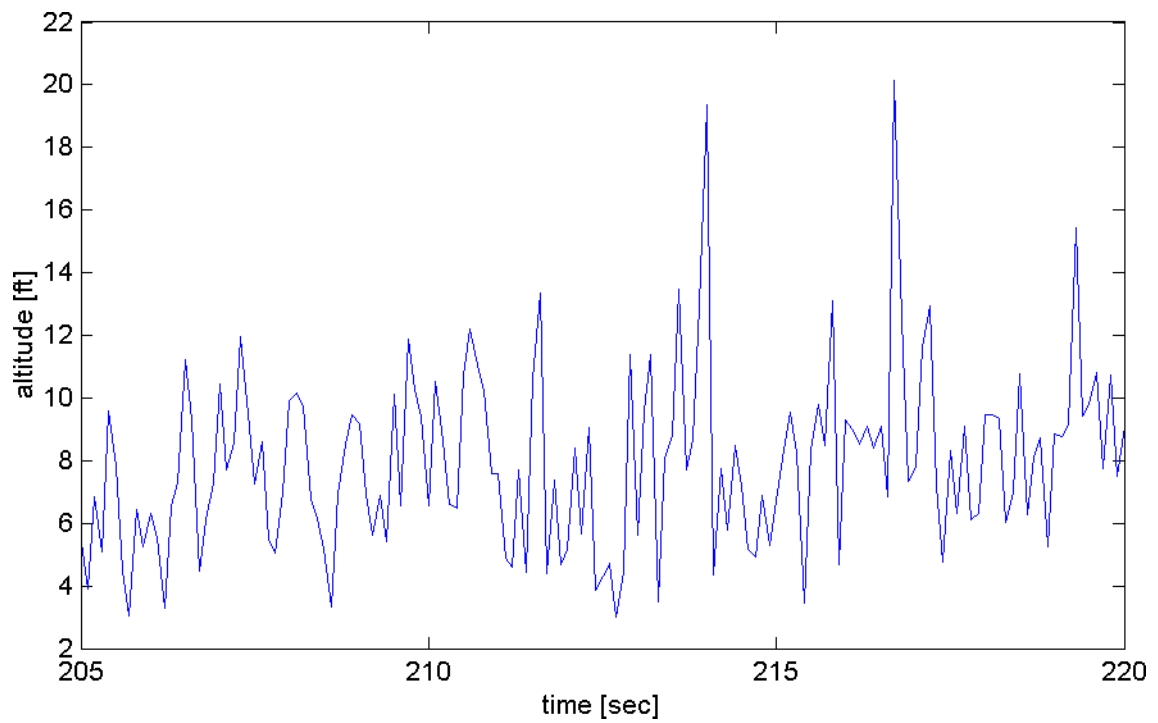Figure 28: Sonar raw data in flight, 15 sec

APPENDIX C

EQUIMENT MANUFACTURER WEBSITES

Cisco Systems, Inc. (former Aironet)	**www.cisco.com**
DATEL, Inc.	**www.datel.com**
Diamond Systems Corporation	**www.diamondsys.com**
D-Link Systems, Inc.	**www.d-link.com**
FreeWave Technologies	**www.freewave.com**
Honeywell International SSEC	**www.ssec.honeywell.com**
Inertial Science, Inc.	**www.inertialscience.com**
JUMPtec Adastra	**www.adastra.com**
LMB/Heeger, Inc.	**www.lmbheeger.com**
NovAtel Inc.	**www.novatel.com**
Roke Manor Research, Ltd.	**www.roke.co.uk**
Shock-Tech	**www.shocktech.com**
Wind River Systems, Inc.	**www.windriver.com**
Yamaha Motor Co., Ltd.	**www.yamaha-motor.co.jp**

REFERENCES

[1] O. Amidi, T. Kanade, and J. R. Miller. Autonomous helicopter research at carnegie mellon robotics institute. In *Proceedings of Heli Japan '98*, Apr. 1998.

[2] A. R. Conway. *Autonomous Control of an Unstable Model Helicopter Using Carrier Phase GPS Only'*. PhD thesis, Stanford University, Mar. 1995.

[3] Y. Devouassoux. Extended kalman filter for the navigation system of an autonomous helicopter. Special Problem Report, School of Aerospace Engineering, Georgia Institute of Technology, 2001.

[4] E. Hallberg, I. Kaminer, and A. Pascal. Development of a flight test system for unmanned aerial vehicles. *IEEE Control Systems Magazine*, 19(1):55–65, Feb. 1999.

[5] E. Johnson, P. DeBitetto, C. Trott, and M. Bosse. The 1996 mit/boston university/draper laboratory autonomous helicopter system. In *15th AIAA/IEEE Digital Avionics System Conference*, volume 1, 1996.

[6] A. D. Kahn. The design and development of a modular avionics system. Master's thesis, Georgia Institute of Technology, School of Aerospace Engineering, Atlanta, GA, Apr. 2001.

[7] H. Lamela, M. Ferreras, and A. Varo. Sensor and navigation system integration for autonomous unmanned aerial vehicle applications. In *IECON' 99 Proceedings. The 25th Annual Conference of the IEEE*, volume 2, 1999.

[8] A. Sato. Research, development and civil application of an autonomous, unmanned helicopter. Technical report, Aeronautic Operations, Yamaha Motor Co., Ltd., Shzuoka, Japan, 2000.

[9] D. H. Shim. *Hierarchical Control System Synthesis for Rotorcraft-based Unmanned Aerial Vehicles'*. PhD thesis, University of California, Berkeley, 2000.

[10] K. L. Sprague. Design and validation of an avionics system for a miniature acrobatic helicopter. Master's thesis, Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science, Cambridge, MA, Feb. 2002.