

**Work Plan**

**Harvey Mudd College Engineering Clinic**

Project Team

**Fall Semester**

Robert Cyprus (S)

Aishvarya Korde (S)

Zayra Lobo (J)

Paige Rinnert (Fall TL)

Project Advisor: Professor Christopher Clark

Project Liaisons: Ken Dreshfield ‘80

Ian Jimenez ‘11

Ron P. Smith ‘83

Stephanie Tsuei (Lead)

December 2016

ABSTRACT

[Insert abstract here]

*An abstract is similar to a short executive summary. In about 100-150 words, it should summarize the report and the key results in a form accessible to the general reader (e.g. a junior engineering major). Common mistakes in an abstract are to use generalities and to omit the most important information.*

TABLE OF CONTENTS

ABSTRACT ii

1.1. Northrop Grumman 3

1.2. Project Statement 3

1.2.1. Objectives 3

1.2.2. Constraints 3

1.2.3. Functions 3

1.3. Deliverables 3

1.4. Project status 5

1.4.1. Simulation 6

1.4.2. Hardware implementation 6

1.5. Impact 6

2. Background 7

3. design alternatives 9

3.1. States of the quadrotor 9

3.2. Sensors 9

3.3. SSE algorithm 9

3.3.1. Convex Optimization 9

3.3.2. Satisfiability Modulo Theory for nonlinear systems 10

3.4. Hardware 11

3.4.1. Quadrotor 11

3.4.2. Flight controller and computer 11

4. Detailed design 12

4.1. Simulation 13

4.1.1. Dynamic Model 13

4.1.2. Closed-loop simulation 13

4.1.3. Closed-loop simulation with SSE 13

4.2. Hardware implementation 13

4.2.1. Onboard computer 13

4.2.2. Sensors 13

4.2.3. Quadrotor 13

4.2.4. Controller 13

4.2.5. RC receiver 13

4.2.6. Power grid 13

5. Results 14

5.1. Dynamic model 14

5.2. Closed-loop simulation 14

5.3. SSE algorithm 14

5.4. Sensor data 14

5.4.1. 9DOF sensor 14

5.4.2. Laser scanner 14

6. Project Management 15

6.1. Fall progress 15

6.2. Spring overview 16

6.3. Work breakdown structure 16

6.4. Schedule 17

6.5. Division of labor 17

7. References 18

introduction

Northrop Grumman (NG) is sponsoring a 2016-2017 Clinic Project to test sensor robustness from cyber-attacks on autonomous vehicles, specifically aircraft. This section describes NG, presents the project statement, and defines the deliverables for the project.

## Northrop Grumman

Northrop Grumman is a leading global security company that innovates in the fields of autonomous vehicles, cyber security and communications, and air and space systems. NG is committed to preserving freedom and advancing human discovery. This Clinic Project is sponsored by NG’s Aerospace Systems sector, NGAS. NGAS is based in Redondo Beach, CA and provides next-generation solutions for military aircraft and land vehicles, autonomous systems, and space systems for its customers worldwide. NG holds world records in many fields, with technology integrated in products ranging from cell phones to space satellites. NGAS’s Research and Development department is sponsoring this project to test sensor robustness in autonomous aircraft.

## Project Statement

The Northrop Grumman Clinic Team will implement a secure state estimator (SSE) in software and then experimentally validate the estimator in hardware. The team will demonstrate that the addition of the SSE algorithm to the control loop reduces the real-time tracking error of a quadrotor when using measurements from compromised onboard sensors.

### Objectives

* Demonstrate quadrotor altitude tracking control with minimal tracking error [m]
* Maximize the number of compromised sensors for which the SSE can be tested
* Minimize cost

### Constraints

* SSE must reduce tracking error when sensors are compromised, as seen in Figure 1.

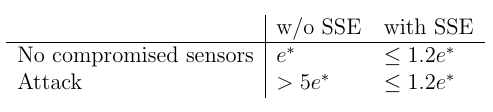


Figure | Specs for a successful SSE

* Final implementation of the SSE must be in ROS
* Quadrotor must fly for 10 minutes on a single battery charge
* Must estimate states at a frequency of at least 25 Hz
* Equipment and supplies < $7,000

### Functions

* Quadrotor should complete pre-planned altitude profile with compromised sensors
* Quadrotor should perform real time state estimation and correction
* The SSE must be configurable to simulate hacking
* The SSE should estimate the state of the quadrotor (position [m], velocity [m/sec], orientation [°], angular velocity [°/sec])

## Deliverables

By the end of the fall semester, the team will:

* Implement and simulate a state estimator in Matlab with
  + All sensors functioning.
  + Some sensors compromised.
* Successfully log sensor data.
* Produce a functioning quadrotor with onboard computer, controller, and sensors.
* Project documentation and presentations including:
  + Work Plan
  + Midyear Report
  + Design Review at Northrop Grumman
  + Three internal Clinic presentations

By the end of the spring semester, the team will:

* Test state estimator on real sensor data with
  + Some corrupted measurements.
  + Offline and in real time.
* Demonstrate quadrotor flight with real time data logging and state estimation.
* Project documentation and presentations including:
  + Final Report (including engineering drawings)
  + Spring semester presentation at HMC
  + Projects Day presentation
  + Final Presentation at Northrop Grumman

The fall deliverables in the Work Plan had called for a functioning quadrotor with an onboard computer, controller, and sensors. Due to the lead time in getting the quadrotor to Harvey Mudd, the HMC team did not have enough time to assemble all components onto the quadrotor. This will be done at the beginning of the Spring semester. The team expects to be able to complete the spring deliverables as initially planned.

## Project status

This section succinctly summarizes what the team has achieved and the status of the deliverables. The section is broken into two categories: simulation project status and hardware implementation hardware status. The simulation subsection focuses on progress with the dynamic modeling of the quadrotor, the coding of the secure state estimator in Matlab / Simulink, closing a control loop, and other simulation aspects of the project. The hardware implantation subsection focuses on the progress of the onboard computer, controllers, sensors, power, and connectivity. The Optitrack system, used for position and orientation verification, quadrotor frame, and netting are also discussed.

The background research about secure state estimation is outlined in Section 2. Section 3 discusses the design alternatives and the selection process for the system states, onboard sensors, SSE algorithm, and onboard hardware (computer and controller). Section 4 gives a detailed design of all simulations, which include the dynamic model and control loop with SSE, and hardware implementations, including all sensors, onboard computer, onboard, controller, quadrotor frame/motors, RC transmitter/receiver, and power components. The current results of these designs will be stated in Section 5. This report will conclude with an overview of tasks and work breakdown for the Spring semester, outlined in Section 6.

### Simulation

### Hardware implementation

## Impact

*The HMC mission statement calls for preparing you to be ready to assume leadership in your fields with a clear understanding of the impact of your work on society.*

*This section addresses the significance of your project and its implications. Why does the sponsor want the deliverables? If the project could eventually lead to a new product, what is the size of the market? Who are the potential users and how would they benefit from the project? How will you personally benefit from your experience on the project? What are the implications for society? Benefits? Costs? If your technology became widely adopted, how would it impact the world? Are there any controversies related to the technology you are working with? Impacts on health or quality of life? The environment? Privacy? Security? Energy? Entertainment? Our understanding of the universe? If the impacts seem trivial or obvious at first glance, take the time to probe more deeply.*

*Most projects have both positive and negative potential implications for society. Consider ways in which the negative implications could be avoided or alleviated. Word this section honestly but tactfully because your sponsor is one of your important audiences.*

# Background

Quadrotors are beginning to be used in a variety of applications from product delivery to agricultural maintenance to humanitarian aid. Even though quadrotors may appear in a wide range of environments, each of these contexts has something in common: the quadrotor must have a robust control system to follow the user’s desired trajectory as accurately as possible. Most control systems use feedback control, where a controller calculates necessary inputs based on the desired trajectory and previous system outputs. The controller then feeds these inputs back into the system to help the quadrotor stay on the correct path. However, a control system is only as good as the states fed into the controller.

Estimating the states of a real system, like position and orientation, is a challenge given that the exact state cannot be determined simply by observation. Instead, the state must be estimated given a set of measurable parameters of the system. The goal of state estimation is to provide an estimate of the system state given the inputs and outputs of the real system; this state can then be fed into a standard controller to maneuver the vehicle.

Standard state estimation techniques are sufficient when all sensors are functioning as expected, but these methods break down when sensors become compromised. If a sensor starts to output incorrect or nonsensical values, a standard state estimator will not produce an accurate measure of the system state. Therefore, the focus of this project is to implement a *secure* state estimator on a quadrotor that accurately measures states of the quadrotor in the presence of cyber-physical attacks. The category of cyber-physical attacks includes both cyber attacks, like a malicious virus introduced to a system or a signal that is intercepted and altered by a hacker, and physical attacks on the hardware. Both cyber and physical attacks could cause a state estimator to fail to produce an accurate state of the system if they alter the sensor outputs. These types of attacks can also affect the behavior of the controller and even the actuators that provide control input, but given the scope of the project, the team is focusing only on attacks on sensors, in the form of broken sensors, compromised signals, or noisy outputs.

One of the main techniques employed to write a secure state estimator that is robust against attacks is convex optimization. (Another option is satisfiability modulo theory, but this approach will not be discussed in detail.) Convex optimization is a special class of nonlinear optimization problems that includes least squares programming (LP) and quadratic programming (QP). Many different programs exist to translate a mathematical problem into a useful form for optimization and to find the solution. Convex optimization is used in a wide array of fields, including control, circuit design, economics, and machine learning. Recent advances have drastically reduced solution times, which makes it possible to use a convex optimization solver in real time on a physical system like a quadrotor [1]. In this project, the team is using the CVX solver to implement the secure state estimator because it is compatible with Matlab and Simulink. The team’s goal is not to improve the state estimation algorithm; the aim is to implement a pre-existing algorithm that has only been tested in simulation and valid the algorithm in hardware.

# design alternatives

The team considered many options for the states of interest, the sensors to measure these states, the algorithm to perform state estimation, and the type of quadrotor and other flight hardware to purchase. This section details the team’s process for selecting each of these components and the final implementation decision.

## States of the quadrotor

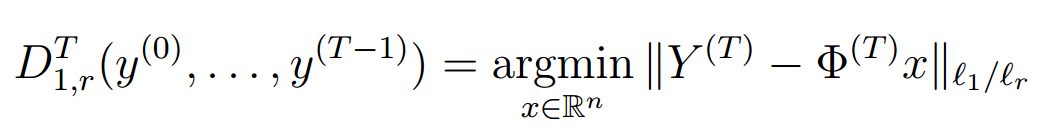
## Sensors

## SSE algorithm

The team considered two main approaches for the secure state estimation algorithm, based on literature suggested by the liaisons. The first algorithm uses concepts from compressed sensing and error correction over the reals to pose the estimation problem as a convex optimization problem [2]. The other algorithm uses Satisfiability Modulo Theory to perform state estimation on nonlinear systems [3]. The team decided to use the convex algorithm because of previous coursework and familiarity with state space control, but both algorithms will still be described here.

### Convex Optimization

This secure state estimation algorithm is presented in as a convex optimization problem:

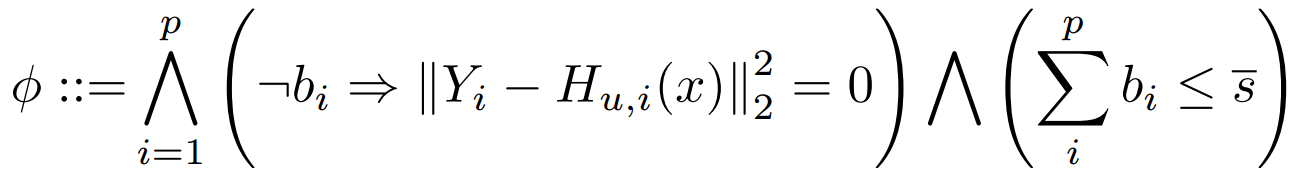


Equation | Convex optimization algorithm.

where is a decoder whose inputs are the sensor output column vectors (1 by the number of sensors); *argmin* for a vector of real values denotes the states for which the argument is minimized; contains the matrix of outputs (number of time steps by number of sensors) added to the control inputs , where is one of the system matrices; is the linear transformation that maps to using the system matrices and ; and the notation is the sum of the norms (also known as a -norm) of the rows of matrix [2].

### Satisfiability Modulo Theory for nonlinear systems

The other state estimation algorithm is formulated in as a Boolean equality:



Equation | SAT algorithm.

where is a pair that satisfies the following Boolean expression; is the number of sensors; is a Boolean value indicating whether sensor is compromised ( indicates that the sensor has been attacked); is a column vector of sensor measurements from the sensor where the first element is the measurement from the beginning of the window (at time ) and the final element is at the end of the window (at time ); is the system transfer function; the notation is the norm of the matrix squared; and the upper bound on the number of compromised sensors [3].

## Hardware

### Quadrotor

### Flight controller and computer

# Detailed design

*This section is often the heart of a Midyear Report. It should fully document the design. Common figures and tables in this type of section include block diagrams, detailed drawings, schematics, printed circuit board layouts, and a bill of materials (BOM). The section should explain why design decisions were made, such as how tolerances were selected or why certain component values were used.*

*A bill of materials should list everything that the reader would need to order the components. Component designations should match those in the detailed drawing or schematic to assist assembly. An example is shown in Table 1.*

Table 1 Bill of Materials

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Component** | **Description** | **Supplier** | **Supplier Part #** | **Unit Price** | **Quantity** | **Total** |
| R1-R7 | 1 k resistor | DigiKey | 1.0KQBK-ND | $0.01 | 7 | $0.07 |
| C1 | 0.1 F capacitor | DigiKey | P4525-ND | $0.18 | 1 | $0.18 |
| U1 | Spartan XCS3400-4TQ144C FPGA | Nu Horizons | XCS400-4TQ144C | $19.10 | 1 | $19.10 |
| D1 | Common anode 7-segment display | Jameco | 24715 | $1.26 | 1 | $1.26 |
| … |  |  |  |  |  |  |

*If the design includes source code, the body should give an overview of the design and operation, but the code should be placed in an appendix.*

## Simulation

### Dynamic Model

### Control Feedback Simulation

### Control Feedback Simulation with SSE

## Hardware implementation

This section discusses the hardware implementation of the components in this project, both onboard and off-board the quadrotor. This includes the onboard computer, the on and off-board sensors, the quadrotor itself, the onboard controller, the RC receiver, and the power to all necessary components.

### Onboard Computer

The onboard computer is the Jetson TK1, where all processing is done aboard the quadrotor. The Jetson has a separate wired connection to each onboard sensor (outlined in Section 4.2.2), and is continually receiving data in real time. The ROS environment is used to manage all the asynchronous processing, including simultaneously receiving data from multiple sensors, processing that data with an SSE algorithm, and sending signals to the Pixhawk controller.

#### ROS Environment

#### SSE Algorithm

#### Data Logging

Data logging on the Jetson is somewhat straight forward. Since the team uses ROS onboard the Jetson, data sets from the sensors are published as “rostopics”, accessible by other “nodes” in the ROS environment.

However, the published data in rostopics are only accessible while the programs are running. Therefore, if the team wants access to the received sensor data for offline analysis, the rostopic data needs to be copied to another location. This can be accomplished by executing simple scripts through the Jetson’s command-line interface to copy data to an offline directory.

### Sensors

#### 9DOF Sensor

#### Laser Scanner

#### Optitrack vision system

### Quadrotor

### Controller

### RC Receiver

### Power Grid

# Results

## Dynamic model

## Closed-loop simulation

## SSE algorithm

## Sensor data

### 9DOF sensor

### Laser scanner

# Project Management

*By this point, you will likely have obtained new information not available at the time of the Work Plan that causes the plan for the project to shift in minor or major ways. You also are likely to have a much better sense of which parts of the project are easy and which parts are hard. This section is a good place to critically reflect on how the fall semester actually proceeded in comparison to your plan and to use the experience to create an effective spring plan.*

[Introduction here, you should not have two headings in a row without text between.]

## Fall progress

*Discuss what the team completed in the fall and how it matched the Work Plan in terms of deliverables, schedule, and division of labor. Include a comparison of the fall elements from the work breakdown structure with the actual work performed. The comparison will show new tasks added to the work beyond the original plan. Review the minutes from the team meetings. If the project deviated from the plan, why did the deviations occur? What can the team learn from the experience to create a more realistic plan to reach the spring deliverables?*

…

**Activity Planned Time (hours) Actual Time**

Background research *(already complete)*

Roadrunners

Diet 6 7

Habitat 6 5

Locomotion 6 3

Existing Acme products 6 8

Review footage of product failures 3 × 4 = 12 12

Conceptual Design

Research

Skates 12 8

Boulders 12 9

Anvils 12 7

Jet-Powered Anvils new 4

Brainstorming, sketching, component research

Skates 18 13

Boulders 18 15

Anvils 18 12

Jet-Powered Anvils new 9

Drawings

Skates 4 7

Boulders 4 8

Anvils 4 5

Jet-Powered Anvils new 9

Comparison of Alternatives 5 × 4 = 20 16

Detailed Design *(significant uncertainty in these initial estimates)*

Initial SolidWorks model 24 18

Finite Element analysis 48 60

(note: ideally this would be broken down further to better estimate time)

Component selection 20 30

Manufacturer selection 12 7

Send prototype for manufacturing 8 not done

Test Plan

Initial Test Plan 8 9

Team Meetings

Teleconferences 15 × 4 × 1 60

Internal Team Meetings 15 × 4 × 1 60

Tuesday Presentations 15 × 4 × 1 40

Team Leader Meetings 3 3

Planning 15 × 0.5 8

Logistics

Register to drive Clinic Van 2 1

Presentation and Preparation

Orientation Day 3 × 4 = 12 12

Fall Review #1 2 × 4 = 8 8

Fall Review #2 3 × 4 = 12 12

Fall Review #3 3 × 4 = 12 12

Fall Site Visit 12 × 4 = 48 55

Reports

Team Charter 5 6

Work Plan

Background 5

Design Alternatives 5

Project Management

Work Breakdown 4

Schedule 2

Division of Labor 1

Other sections 3

Writing Center review 2

2nd Draft n/a

3rd Draft n/a

Midyear Report 40

Total Time (Fall) (add up here) (add up here)

Figure Comparison of Planned and Actual Fall Work Breakdown Structure

## Spring overview

[Lay out the big picture of what the team plans to do in the spring, how the work will be divided, and what the major milestones are along the way.]

## Work breakdown structure

*Spring semester work breakdown structure with expected hours; see the work plan for an example from the fall.*

Figure 3 Spring Work Breakdown Structure

## Schedule

[See work plan. Focus on spring semester Gantt charts.]

Figure Spring Gantt chart

Figure 5 Detailed Spring Gantt chart

## Division of labor

[Similar to work plan, but for spring semester tasks]

**Activity Owner**

Detailed Design

Send prototype for manufacturing Coyote

Revised SolidWorks model Coyote

Revised analysis Bunny

Revised components Pig

Second prototype manufacturing Spring Junior

Test Plan Bunny

Testing Pig, Bunny

Presentation and Preparation All

Reports All

# References

*If your field has a standard format for references or your liaison or advisor prefers a format, follow that convention. Otherwise, use the IEEE format given below. Note that different sources such as books, conference papers, journal papers, and web sites have different forms to reflect the different information. A unifying principle is that a reader in the distant future should be able to track down the reference using the information you supplied. When doing your research, go to the library and check out books and papers rather than simply relying on Google; there is a vast amount of technical knowledge in the world that is still not available through Internet searches.*

*All the references in this section should be cited in the text.*

[1] S. P. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge, UK ; New York: Cambridge University Press, 2004.

[2] H. Fawzi, P. Tabuada, and S. Diggavi, “Secure estimation and control for cyber-physical systems under adversarial attacks,” *IEEE Transactions on Automatic Control*, vol. 59, no. 6, pp. 1454–1467, Jun. 2014.

[3] Y. Shoukry, P. Nuzzo, N. Bezzo, A. L. Sangiovanni-Vincentelli, S. A. Seshia, and P. Tabuada, “Secure state reconstruction in differentially flat systems under sensor attacks using satisfiability modulo theory solving,” in *2015 54th IEEE Conference on Decision and Control (CDC)*, 2015, pp. 3804–3809.