PRELIMINARY PROJECT PLAN
FOR DESIGN OF A LIFESAVING-
AERIAL-LIFE-VEST-DELIVERY-
SYSTEM (LALVDS) SYST 490:
OCT 25, 2015
Andrew Hardy¹, Mohammed Rajeh², Lahari Venuthurupalli³, Gang Xiang⁴

Department of Systems Engineering and Operations Research
George Mason University
4400 University Dr.
Fairfax, VA 22030-4444
United States

¹ahardy9@gmu.edu
²mrajeh@gmu.edu
³lvenuthu@gmu.edu
⁴gxiang@gmu.edu
TABLE OF CONTENTS

I. Context Analysis .......................................................................................................................... 3
   A. Rip currents .............................................................................................................................. 3
   B. Beach Incidents ......................................................................................................................... 3
   C. Process of a Lifeguard Rescuing a Victim .............................................................................. 6

II. Stakeholder Analysis .................................................................................................................. 6
   A. Lifeguarding Associations ........................................................................................................ 6
   B. Lifeguards ................................................................................................................................. 6
   C. Beach Goers ............................................................................................................................. 6
   D. Manufacturers .......................................................................................................................... 7
   E. Municipalities .......................................................................................................................... 7
   F. Liability issues ......................................................................................................................... 7
   G. Stakeholder Tensions .............................................................................................................. 8

III. Gap ........................................................................................................................................ 8

IV. Problem Statement .................................................................................................................. 8

V. Need Statement ........................................................................................................................ 9

VI. Win-Win ................................................................................................................................ 9

VII. Concept of Operations .......................................................................................................... 9
   A. Function Block Diagram ......................................................................................................... 9
   B. Operation Scenario ................................................................................................................ 10
   C. Drone Location ...................................................................................................................... 10

VIII. Requirements ....................................................................................................................... 11
   A. Mission Requirements ........................................................................................................... 11
   B. Functional Requirements ....................................................................................................... 11
   C. Design Requirements ............................................................................................................ 11
   D. Usability ................................................................................................................................ 11
   E. Availability ............................................................................................................................. 11
   F. Reliability ............................................................................................................................... 11

IX. Method of Analysis .................................................................................................................. 12
   G. Power and Thrust ................................................................................................................... 13
   H. Thrust Model ........................................................................................................................... 14
   I. Drag Model ............................................................................................................................... 15
   J. Main Force Model ................................................................................................................... 15
   K. Life vest Model ....................................................................................................................... 15
   L. Tether Model .......................................................................................................................... 16
   M. Rotational Dynamics Model ................................................................................................ 16
Appendix A: Simulation Codes (Prelim)

X. References
   A. Main Model
   B. Rotational Dynamics
   C. Main Model

XII. Project Risk Mitigation
   A. Critical Tasks
   B. Schedule
   C. Earned Value

XIII. Design of Experiment
   A. Motor Parameter DoE
   B. Motor Parameter Equations
   C. Motor Parameter DoE

XIV. Work Breakdown
   A. Work Breakdown Structure
   B. Critical Tasks
   C. Schedule
   D. Earned Value

XV. Appendix A: Simulation Codes (Prelim)
   A. Linear Dynamics
   B. Rotational Dynamics
   C. Main Model
I. CONTEXT ANALYSIS

A. Rip currents

In the United States, over three hundred million people go to the beach for different purposes. They go to the beach for swimming, surfing, scuba, and to just walk around the beach. For those people who go to the beach to swim, they have to be careful of rip currents as they are very dangerous. Rip currents are strong powerful currents of water that flow away from shore. They form when waves break near the shoreline, they can occur at any surf beach in the United States that has breaking waves. If anyone is caught in a rip current, it is dangerous for them if they try to fight it and tries to swim vertical to the rip current, they will not be able to escape back to shore safely. The swimmer has to swim parallel to the rip current in order to escape back to shore.

There are a few different measurements for rip currents [1]. First, the width of the rip current. It can be very narrow with a width from ten to twenty feet or it can be very wide with a width up to one-hundred and fifty feet. Second, the speed of rip current can varies; it has an average speed from one to two feet per second. This is not considered dangerous for strong swimmers, but it is dangerous for the weak swimmers. If it reaches up to eight feet per second, this is considered to be dangerous for all swimmers. Third, the length of rip currents, it can extend hundreds feet away from shore.

B. Beach Incidents

According to the United States Lifesaving Association, there are different incidents that happen at the beach every year. We have considered statistical data from the year 2003 to 2013. That data is: the unguarded drowning deaths, guarded drowning deaths, and rescues. For those incidents we have included rip currents and the total rip currents with swift, surf and scuba. In addition, according to the United States Lifesaving Association, the primary cause of rescue are rip currents. With that in mind we have considered statistical data from 2003 to 2012.

Please note, some beach agencies only report totals for fatalities and rescues, but not the specific causes of rescues. For example, just because a graph says there is only two rip current rescues that year, does not mean there was only two rip current rescues that year for all agencies. It means only two rescues for the group of agencies that do report that subcategory. We show these graphs to reveal the general trends and general numbers regarding rescues and fatalities.

![Figure 1 Total Unguarded Drowning Deaths](image1.png)

![Figure 2 Unguarded Drowning Deaths From Rip Currents](image2.png)
In Figure 2, you can see the unguarded drowning deaths from rip currents from 2003 to 2013. You can see from the trend line, when there wasn’t a lifeguard, the drowning deaths from rip currents is slightly increasing over the years. It is increasing by a rate of 0.6 per year. In Figure 1, you can see the total unguarded drowning deaths from 2003 to 2013. The total includes rip currents with swift, surf and scuba. You can see from the trend line, when there wasn’t a lifeguard, the total drowning deaths is not increasing or decreasing over the years; it is stable with a rate of 0.054.

In Figure 4, you can see the guarded drowning deaths from rip currents from 2003 to 2013. From the trend line, you can see that when there was a lifeguard, the drowning deaths from rip currents are increasing over the years from 2003 until 2013. It is increasing by a rate of 0.19 per year. Figure 3 shows the total guarded drowning deaths from 2003 to 2013. The total includes rip currents with swift, surf and scuba. From the trend line, you can see that when there was a lifeguard, the total drowning deaths is increasing over the years from 2003 to 2013. It is increasing by a rate of 0.67 per year.

Figure 6 shows the rescues from rip currents from 2003 to 2013. When looking at the trend line, you can see the rescues from rip currents are increasing over the years by a rate of 889.41 per year. Figure 5 shows the total rescues from 2003 to 2013. The total includes rip currents with swift, surf and scuba. By looking at the trend line you can see that the total rescues are increasing over the years by an increasing rate of 1163.4 per year.
Figure 8 shows that rip currents are the primary cause of rescues in the United States from 2003 to 2012. You can see that rip currents account for 81% of the rescues. From 2003 to 2012, the total number of people rescued from rip currents was 334,184. Almost 0% of the rescues are from surf with 2,459 people rescued. Just 1% of the rescues were from scuba with 2,538 people rescued. Also, 18% of the rescues were surfers with 73,670 people rescued. By comparing the rescues from rip currents with the rescues from surf, swift, and scuba, you can see that rip currents are the primary cause of rescue from 2003 to 2012. In addition, according to the National Oceanic and Atmospheric Administration the 10-year average of annual rip current fatalities is 51 [2]. Figure 7 is a sample for 2012 to show that rip currents are the primary cause of rescue. You can see that the rescue from rip currents in 2012 is 82%. In 2012, 35,935 people were rescued from rip currents. By comparing the rescues from surf, swift and scuba, you can see that rip currents are the primary cause of rescue. The rescue from rip currents is more than the half of the total rescues.
C. Process of a Lifeguard Rescuing a Victim

A lifeguard chair is placed every couple meters or miles, depending on how big the beach is. The beach is separated into multiple zones with one lifeguard chair in the top most center of each zone. For an example, if a drowning victim is spotted in zone A, the lifeguard will radio the control room, whistle at the adjacent lifeguards to cover their zone and point in the direction of the victim [4]. The lifeguard then will start running towards the shoreline with either their right hand up in the air, which means assistance is needed, or pointing and tapping on top of their head, which means assistance is NOT needed [4]. The lifeguard then swims to the victim, rescues the victim, guides the victim back to the shoreline, and then drags them out of the water. Medical care is provided.

II. Stakeholder Analysis

There are five main stakeholders: 1) lifeguarding associations, 2) Lifeguards, 3) Beach goers, 4) Manufacturers, and 5) Municipalities.

A. Lifeguarding Associations

Lifeguarding associations are professional lifesaving associations that train dedicated beach lifeguards and open water rescuers. These associations certify lifeguards and make sure they are up to date with all four requirements: 1) Yearly licensing, 2) Four hours of in-service training, 3) Monthly training meetings and, 4) Red ball test. If the individual meets all these requirements each year they will be certified as a lifeguard.

B. Lifeguards

Lifeguards are strong swimmers who supervise the safety and rescue of swimmers, surfers, and other water sports participants. We were able to interview three certified lifeguards to help with our research on rip currents, the lifeguard process, liability issues, and the use of a drone.

C. Beach Goers

Beach goers are the people who go to the beach and use its services. They want the least
restrictions and the instructions given by lifeguards are of no concern. However, they want the lifeguards to be 100% effective at their jobs. In other words, the beach goers want the lifeguards to be alert, well trained, and to save everyone in time.

D. Manufacturers
Manufacturers are companies that produce equipment for the lifeguards. Some major companies include Swimoutlet and Marine Rescue Products. A lifeguarding association can also manufacture rescue products. For an example, the Jeff Ellis & Associates, an international lifeguard trainings, builds their own products such as a rescue board, a lifeguard buoy, a ring buoy and life jackets that lifeguards must use when they rescue a victim.

E. Municipalities
The municipality is broken down into owner and operator. The owner either can be the county, city, or state. The operator documents the lifeguards current status of certification and requirements met. They are the ones who pay the lifeguarding associations, certified agencies like Jeff Ellis, and lifeguards. Both the owner and operator are usually the same. However there are some operator businesses that concentrate on the documentation and financial aspect of lifeguarding that the owner may hire.

F. Liability issues
Between these five stakeholders there are liability issues, the body of water, in this case a beach, is broken down into municipalities which can be looked at as the owner and operator. The operator is where the lifeguard certification history and updates are all documented and financial status is monitored. This is decomposed into lifeguard associations, which are certified lifeguard agencies. Then these associations train lifeguards who are hired by the owner to supervise the safety of the beach goers [4].

Lifeguards and certification agencies are always liable if a drowning victim sues a lifeguard for any injuries during the rescue process. However, if the lifeguards use only the designated, branded equipment the agencies provide and the agencies have proof and correctly the document, train, license, there is a very low chance of a successful lawsuit. Most cases would be settled out of court [4].
G. Stakeholder Tensions

This diagram is a summary of all the interactions, liabilities, and tensions among all the stakeholders. Ideally there are tensions between all the stakeholders. Whether it is the lifeguards not doing their jobs correctly and not following what the agencies have taught them or the municipalities not having proper documentations stored or the beach goers not obeying the rules and regulations [4]. The Life guarding associations train and certify lifeguards and inspect the Beach Operator. They provide legal assistance. The lifeguards protect beach goers and provide rescue services. The beach goers hold the lifeguard liable if injured, who hold the operator liable, and then the operator holds the owner liable. In the end the beach owner is always liable. The catastrophic insurance umbrella protects the beach owner, beach operator, and lifeguards if a lawsuit is unsuccessful.

![Stakeholder Tensions Diagram](image)

**FIGURE 11 Stakeholder Tensions**

III. GAP

There is a gap between the victim survival time and the time needed for a lifeguard to reach the victim.

IV. PROBLEM STATEMENT

Rip currents are, on average, 80% of annual beach rescues, 80% of annual beach fatalities, and cause 51 annual deaths [2] [3] [5]. Lifeguards are very good at their job and can reach victims in an average time of X seconds [6], however some victims cannot survive that long and have survival times as low as X seconds. By raising the average victim survival time and reducing the variance of the victim's survival time by X, we can reduce drowning deaths by X%.
V. Need Statement

There is a need for a system that can close the gap between a victim’s survival time and a lifeguard’s rescue time. Specially, there is a need for a system that can reach and assist the victim before they drown. That can increase the victim’s survival time to the time it takes a lifeguard to reach said victim, in order to increase the victim’s survival time.

VI. Win-Win

- Beachgoers
  a. Increase safety of beaches without added regulation
  b. Decrease rip current-related deaths
- Municipalities
  a. Increased safety leads to more beach goers, which lead to more beach services used
- Manufacturers
  a. Allow system to use any life ring/lifesaving device
  b. Let beaches use the devices they want within a certain weight limit (no switching manufacturers)
- Lifeguarding Operators
  a. Reduce legal actions and save lives.
- Lifeguards
  a. Improve rescue process and save lives.

VII. Concept of Operations

A. Function Block Diagram

The lifeguard process and the drone system are independent. If the lifeguard for some reason is not able to reach the victim the drone will still continue and drop the ring buoy. If the drone is not able to drop the ring buoy the lifeguard will reach the victim. After the drone leaves the home point to the given coordinates, there is a decision mark that is if the drone reaches the victim on time it will continue the process of dropping the ring buoy near the drowning victim, releasing the tether when the victim grabs it, and then the drone going back to its home point. This will modify the victim’s survival time. However, if the drone does not reach the victim on time, it will directly go back to its home point.
B. Operation Scenario

**Precondition:** Lifeguard has detected a victim in a rip current. Lifeguard is prepared for rescue process. Lifeguard informs controller the section # or victim’s general direction. Victim is currently somewhere on the rip current and is attempting an escape method. Drone is ready to deploy. Ring buoy is stocked on drone.

**Primary:**

1. Controller is informed by lifeguard of section # of drowning victim.
2. Controller confirms victim location through received section #.
3. The system takes off to a height of X meters.
4. The system accelerates to X m/s towards the section given.
5. Controller confirms specific location in that section through camera or eyesight.
6. The system maintains X m/s towards the victim’s location.
7. Once the system is within 4m of the victim’s location, the system decelerates to victim’s speed and position. At the same time, the system will reduce height until the ring buoy is just above the water (confirmed by controller).
8. The system drops the ring buoy and positions it by the victim.
9. Once victim has a firm grasp on ring buoy, system detaches the tether.
10. The system maintains a X m hover over victim until lifeguard has reached the victim.
11. Once lifeguard has reached the victim and starts pulling them to shore, system will be flown back to the home point.
12. System lands on home point.

**Post-Condition:** Lifeguard is enacting the rest of rescue process starting with pulling the victim back to shore. Drone has landed back at the home point and awaits restocking of ring buoy. Victim is being helped by the lifeguard.

C. Drone Location

There are two options for where the drone could be located. First, the drone could be located by the lifeguard towers. There could be a drone at every tower or it might be spaced out every few lifeguard towers. From here the drone would be able to reach the victim faster. The controller would be able to see both the victim and the drone by eyesight. However there would need to be electrical cables to charge the drone and storage for the drone to protect it from damage.

The second option is for the drone to be stationed at some main lifeguard building. Here there would be charging stations and spare batteries for the drone as well as an additional stock of ring buoys. There would be no need to worry about protecting it from the elements. However, the drone would most likely be farther away from the location of the victim, leading to an increased travel time. The drone would need a camera on it to provide a video feed for the controller to fly the drone to the victim.
VIII. REQUIREMENTS

A. Mission Requirements
   MR.1: The system shall reduce the average annual number of rip current deaths by a minimum of X%.

B. Functional Requirements
   F.1: The system shall hover at a minimum altitude of 2m above the ground.
      F.1.1: The system shall hover at an altitude of 2m with a minimum payload of 2kg.
   F.2: The system shall be operable within Xm of the home point.
   F.3: The system shall reach a victim within X seconds.
      F.3.1: The system shall increase the victim survival time by an average of X seconds, if the system does reach the victim.
   F.4: The system shall be able to restock its payload within X seconds.
   F.5: The system shall be able to deploy its payload within X seconds.
   F.6: The system shall do the entire rescue process at a maximum time of X seconds.

C. Design Requirements
   DR.1 The system shall make use of a drone that is within the Expert Drones inventory of drones.
   DR.2 The system shall attach the lifesaving device to the drone through a tether.
      DR.2.1 The system may have a disconnect method to cut or release the tether in order to deliver the lifesaving device.
      DR.2.2 The system shall have a tether release system that weighs under Xkg.
      DR.2.3 The system shall be able to release the tether within X seconds of the request to release.
   DR.3 The system shall have a camera system pointing downward.
   DR.4 The system may have a speaker system that is connected to an input device.

D. Usability
   U.1 The system shall be usable by a person that has less than 12 hours of training.
   U.2 The system shall be usable in X rescues a day if properly charged.

E. Availability
   A.1 The system shall be available to at least any beach on U.S territory
      A.1.1 The system shall comply with all federal drone regulations.
   A.2 The system shall be available for rescues over 95% of the time.

F. Reliability
   RR.1 The system will have MTBF of X months
   RR.2 The system will have MTTR of X days
IX. Method of Analysis

In order to analyze the feasibility of the system, we shall use a MATLAB simulation to test a drone's ability to fly to a moving target (victim in a rip current).

In order to do that, let us first detail the equations and notations we are using. We first define the motor number, and spin in Figure 13. The prelim simulation will only deal with a quadcopter, however in the future we will most likely go to a hexacopter and/or octocopter drone system instead. We define odd-numbered motors to be spinning clockwise and even-numbered motors to be counterclockwise. We define the axis of the body-frame drone as the following: the front side as the positive X axis, the left side as the positive Y-axis, and the top side as the positive Z-axis (the Z axis is point out of the screen in Figure 13).

We will define the position of the drone in X-Y-Z coordinates. The orientation of the drone will be in the Euler angles of $\phi$-$\theta$-$\psi$ or known as roll-pitch-yaw.

Drones have rotation in three axes. They can rotate around their X-axis (roll), the Y-axis (pitch), and the Z-axis (yaw) as shown in Figure X.2.

http://theboredengineers.com/2012/05/the-quadcopter-basics/

Figure 14 The Axes of Rotation
In order to transform body-frame forces into inertia-frame forces, we will use transformational matrices. For example, force inertia = R*force body. There is respective transforms for φ-θ-ψ (roll-pitch-yaw). We only need to multiply them together in order to a full transform using the three values of φ-θ-ψ.

\[ R_φ = \begin{bmatrix} \cos φ & \sin φ & 0 \\ -\sin φ & \cos φ & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (1) \]

\[ R_θ = \begin{bmatrix} \cos θ & 0 & -\sin θ \\ 0 & 1 & 0 \\ \sin θ & 0 & \cos θ \end{bmatrix} \quad (2) \]

\[ R_ψ = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos ψ & \sin ψ \\ 0 & -\sin ψ & \cos ψ \end{bmatrix} \quad (3) \]

\[ R = R_φ R_θ R_ψ = \begin{bmatrix} \cos θ \cos ψ & \sin ψ \cos ψ + \sin φ \sin θ \cos ψ & \sin φ \sin ψ - \cos φ \sin θ \cos ψ \\ -\cos θ \sin ψ & \cos φ \cos ψ - \sin φ \sin θ \sin ψ & \sin φ \sin ψ + \cos φ \sin θ \cos ψ \\ \sin θ & -\sin φ \cos ψ & \cos φ \cos ψ \end{bmatrix} \quad (4) \]

Additionally there is a transform matrix to convert derivatives of Euler angles to angular velocities in body-frame. Shown below.

\[ \begin{bmatrix} \dot{ω}_x \\ \dot{ω}_y \\ \dot{ω}_z \end{bmatrix} = \begin{bmatrix} 1 & 0 & -\sin θ \\ 0 & \cos φ & \sin φ \cos θ \\ 0 & -\sin φ & \cos φ \cos θ \end{bmatrix} \begin{bmatrix} \dot{φ} \\ \dot{θ} \\ \dot{ψ} \end{bmatrix} \quad (5) \]

G. Power and Thrust

We have three equations to voltage and power. We first assume that the rotor torque is proportional to the current difference. Show below:

\[ τ = K_t (I - I_0) \quad (6) \]

Where τ is rotor torque, I is the current, I₀ is the no-load current when the rotor is not spinning, and Kt as the proportionality constant. We assume I₀ is negligible. We also assume that the voltage difference of a rotor is the sum of the voltage drop across internal resistance and a term that involves the angular velocity of the propellers.

\[ V = IR_m + K_v \omega \quad (7) \]

Where V is voltage, I is the current, Rm is the internal resistance, Kv is the proportionality constant, and ω is the angular velocity of the rotor’s propeller. For the simulation, we assume Rm is negligible. Also note that rotor manufacturers list Kv publicly, but that is the inverse of the Kv we are using. Their equation for Kv is Kv*V = ω, while we have V = Kv*ω. We simply need to do 1/Kv of the manufacturer’s value to get our version of Kv.
From the two equations before (6) (7), we gain terms for current and voltage, thus we can find power.

\[ P \approx \frac{K_v}{K_t} \tau \omega \]  

(8)

We now assume that rotor torque is proportional to the rotor’s thrust, thus,

\[ \tau = K_t T \]  

(9)

\[ P \approx \frac{K_v K_t}{K_t} T \omega \]  

(10)

Where \( \tau \) is the rotor torque, \( K_t \) is a proportionality constant, \( T \) is thrust, \( P \) is power generated, \( K_v \) and \( K_t \) are proportionality constants, and \( \omega \) is rotor propeller angular velocity.

<table>
<thead>
<tr>
<th>Variable 1</th>
<th>Relation</th>
<th>Variable 2</th>
<th>Proportionality Constant</th>
<th>Equation</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \tau )</td>
<td>( \propto )</td>
<td>( I )</td>
<td>( K_t )</td>
<td>( \tau = K_t * I )</td>
</tr>
<tr>
<td>( \tau )</td>
<td>( \propto )</td>
<td>( T )</td>
<td>( K_t )</td>
<td>( \tau = K_t * T )</td>
</tr>
<tr>
<td>( V )</td>
<td>( \propto )</td>
<td>( \omega )</td>
<td>( K_v )</td>
<td>( V = K_v * \omega )</td>
</tr>
</tbody>
</table>

**Table 1 Summary of Proportionality Constants**

**H. Thrust Model**

Each rotor has a propeller. Each propeller spins, pushing air downwards. The air in turn push upwards. This is the general idea of how rotors provide an upward force. The thrust equation we use is listed below:

\[ T = \frac{\pi}{2} D^2 \rho P^2 \left[ \frac{1}{3} \right] \]  

(11)

Where \( T \) is thrust, \( D \) is diameter of the propeller, \( \rho \) is the air density, and \( P \) is power. Notice that we have a power term (10), thus we can use our earlier power equation to substitute into the thrust equation (11). The new thrust equation is listed below.

\[ T = \frac{\pi}{2} \rho D^2 \left( \frac{K_v K_t}{K_t} \right) \omega^2 \]  

(12)
Where $K_v$, $K_t$, and $K_t$ are proportionality constants, and $\omega$ is propeller angular velocity. In the body frame, the drone only provides thrust upwards. Moving is possible when the drone is titled, but to the drone it always provide thrust force in the positive $Z$-axis. Also each rotor follows the thrust equation, and so the rotor’s thrusts can be added together. The follow shows that idea.

$$T_{body} = \frac{\pi}{2} \rho D^2 \left( \frac{K_v}{K_t} \right)^* \begin{bmatrix} 0 \\ 0 \\ \sum \omega_i^2 \end{bmatrix}$$ (13)

Where $\omega_i$ is the $i$th rotor.

I. Drag Model

We will use the follow drag equations.

$$F_D = \frac{1}{2} C_D \rho v^2 A$$ (14)

$$P_D = \frac{1}{2} C_D \rho v^3 A$$ (15)

Where $F_D$ is force due to drag, $P_D$ is tthe power due to drag, $C_D$ is the coefficient of drag of the drone, $\rho$ is the air density, $v$ is the velocity of the drone, and $A$ is the cross sectional area of the drone.

J. Main Force Model

With the force of drag(14), force of thrust (13), force of gravity, force due to the tether rope, and force due to the life vest, this will lead up to the equation of

$$ma = \sum F_i$$ (16)

Where $m$ is the mass of the drone system, $a$ is the acceleration, and $F$ is the various forces acting on the drone. Expanding on this equation, we have:

$$m \begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix} + RT_B + F_D + F_{rope} + F_{lifevest}$$ (17)

Where $x,y,z$ is the coordinate position of the drone, $g$ is the gravitational constant (9.81 m/s/s), $R$ is the rotational matrix, $T_B$ is the thrust in body-frame, $F_D$ is the force of drag, $F_{rope}$ is the force due to the tether, and $F_{lifevest}$ is the force due to the life vest.

K. Life vest Model

For the life vest, we will model the force of gravity and the force of drag on the lifesaving device, the force of gravity is of course $F = mg$, while the force of drag is from the equation of drag (14).
L. Tether Model
We are currently reconsidering the tether from the home point to the drone. Since rip current
distances can extend hundreds of meters, we have found that, even the thinnest of tethers, can be 2kg
or above of weight on the drone. For even an octocopter, this weight is massive and is over 25% of its
possible payload alone. This weight could be used for another life ring, or to allow the drone to
accelerate faster. We believe that there is enough safety precautions from the drone manufacturers
that allow a drone to return to the home point to not need a tether from the home point to the drone
(for example, when battery is low, drones automatically return to home point regardless of user
directions).

M. Rotational Dynamics Model
The torque on the body of the drone is from the rotors. Each rotor has spinning propellers that
push air horizontally, but the air also pushes back. This creates a force at a distance away from the
center of the drone, which is the definition of torque. The torque is given by
\[ \tau_D = r \times F_D \] (18)

Where \( r \) is the distance between the rotor and the drone’s center of mass. We already know the
force of drag from above. We also know that velocity = radius * angular velocity, thus we can derive a
torque equation based on angular velocity. However we also note that torque = moment of inertia *
angular acceleration. However, it is a small enough value that we can later consider negligible in
equation (21).

\[ \tau_D = \frac{1}{2} r \rho C_D A(r)^2 (\omega)^2 \] (19)

\[ b = \frac{1}{2} r \rho C_D A(r)^2 \] (20)

\[ \tau_Z = b \omega^2 + I_Z \dot{\omega} \] (21)

Where \( \tau_Z \) is the torque around the body Z-axis, \( I_z \) is the moment of inertia about the body Z-axis.
Now that we know torques are related to the angular velocity squared, we can find the torques about
the X-Y-Z axes.
For roll, the torque is given as torque = force * distance. The distance is the distance from the rotor to the center of mass. We assume the drone is symmetrical, thus the distance is the same for all rotors. When rolling, drones usually increase the thrust of one motor while decreasing the other rotor in the pair in order to maintain the same body thrust while also creating a torque form that difference of thrust. Thus, the torque from one rotor minus the torque from the opposite rotor creates the body’s rolling torque.

For pitch, it is the same as roll, however it involves the other pair of rotors on the drone.

For yaw, we add up all the rotor’s torques (21). Since two rotors spin counterclockwise and two spin clockwise, two of the terms must be negative. We denote the even numbered rotors contributing to clockwise spin, thus the terms involving even numbered rotors are negative.

\[
\tau_B = \begin{bmatrix}
\tau_{\text{roll}} \\
\tau_{\text{pitch}} \\
\tau_{\text{yaw}}
\end{bmatrix} = \begin{bmatrix}
Lk(\omega_1^2 - \omega_3^2) \\
Lk(\omega_2^2 - \omega_4^2) \\
b(\omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2)
\end{bmatrix}
\]  (22)

Where \( L \) is the distance from a rotor to the center of mass. We will be using Euler’s equation for torque in order to get the angular acceleration of the system.

\[
\tau = I \omega + \omega \times (I \omega)
\]  (23)

Where \( \tau \) is the body’s torque, \( I \) is the moment of inertia, and \( \omega \) is the angular velocity of the drone. This equation does not directly give us roll, pitch, or yaw, thus we need to expand on it. Since we assume the drone is symmetric, the moment of inertia about the X-Y-Z axes are independent of each other, shown below in that equation. If we modify the equation to solve for angular acceleration, we get the other equation below.

\[
I = \begin{bmatrix}
I_{xx} & 0 & 0 \\
0 & I_{yy} & 0 \\
0 & 0 & I_{zz}
\end{bmatrix}
\]  (24)

\[
\dot{\omega} = \begin{bmatrix}
\dot{\omega}_x \\
\dot{\omega}_y \\
\dot{\omega}_z
\end{bmatrix} = I^{-1}(\tau - \omega \times (I \omega))
\]  (25)

\[
\dot{\omega} = \begin{bmatrix}
\tau_{r} I_{xx}^{-1} \\
\tau_{r} I_{yy}^{-1} \\
\tau_{r} I_{zz}^{-1}
\end{bmatrix}
\begin{bmatrix}
I_{yy} - I_{zz} & \omega_z & -\omega_y \\
I_{zz} - I_{xx} & \omega_x & -\omega_z \\
I_{xx} - I_{yy} & \omega_y & -\omega_x
\end{bmatrix}^{-1}
\]  (26)
Where $\tau_\phi$, $\tau_\theta$, and $\tau_\psi$ are the torques in their respective rotations (22), $I_{xx}$, $I_{yy}$, $I_{zz}$ is the moment of inertia about their respective axes, and $\omega_x$, $\omega_y$, $\omega_z$ are the angular velocities about their respective axes. Once we have the angular accelerations, we can use the transformation matrix to get the Euler angle derivatives (5). Integrate those to get the Euler angles.

**N. Simulation Design**

The simulation will be done on MATLAB & Simulink. We will be simulation the rescue process with and without the drone system in order to compare the drone’s effectiveness at increasing the victim’s survival time and at decreasing the percent fatalities.

The three processes we will simulate are:

1. Rip current influences on victim’s state
2. Drone rotational/translational dynamics
3. Lifeguards reaching the victim in the rescue process

**O. Rip Current Influences on Victim State**

![Figure 15 Simulation Diagram of rip current effects on victim state](image)

This simulation will calculate the position of the victim over time. There is three methods of escape methods (float, swim parallel to shore, swim toward shore) that will be randomly chosen for each simulation run. Each method would be represented as a velocity. For example, swimming toward the shore will be represented as an $X$ m/s velocity in the negative-x direction. The rip current velocity will be randomly chosen between its minimum and maximum possible velocity. The output is the victim’s position, which will later act as the waypoint input for the drone and for the lifeguard.
P. Drone Rotational/Translational Dynamics

In order to test the drone’s abilities to fly and accelerate when carrying a life ring, we must simulate the electrical and mechanical properties of the drone under flight. The simulation will follow the following diagrams.
The translational and rotational dynamic models were derived from Andrew Gibiansky’s work [7]. The actuator will be simulated as manual flight. This means that we must consider the pilot’s abilities into the equation later on as delays and randomness in the control actuator.

The waypoint is the position of the victim. Using its position and various errors calculated, we can control the drone to be the same position and velocity as the waypoint. Once it reaches that waypoint and stays at it for a set period of time, we record the time.

Q. Lifeguard Rescue Process

This simulation will calculate the lifeguard position over time. The lifeguard simulation will be inputted the victim position as a waypoint. The victim position will be randomly modified to be at a
position at a later time due to the delay of lifeguards identifying victims. It will also input the rip current velocity (because lifeguards use the rip current to get to the victim quickly). Once the lifeguard has reached the victim, we will add a delay time to simulate the lifeguard helping the victim and grabbing hold of the victim.

The rescue process after the lifeguard reaches the victim is outside the scope of the drone system. It is up to the lifeguard from here on out to help the victim, thus we will not simulate the rescue process beyond this point.

R. Post MATLAB & Simulink Analysis

After the simulation of the processes is over, we generate a random victim survival time. Depending on the drone simulation and if it deployed the life ring on time at the right position, the victim survival time will be modified. Next we see if the lifeguard reached the victim and finished rescue process in less time than the victim survival time. Repeat for several trials to generate survival rate for with-drone and without-drone processes, along with average time to reach victim, average drone velocity, average power used. From average power used and average time to reach victim, we can calculate power, work, and amount of battery used per rescue.
X. SIMULATION REQUIREMENTS

A. Functional Requirements
   SR.1 The system shall be able to fly towards a waypoint and maintain position within 0.5m of the waypoint.
   SR.2 The system shall have one run simulated under 1 minute.
   SR.3 The system shall simulate wind and weight interactions with the drone.
   SR.4 The system shall model drone rotational and translational dynamics.
   SR.5 The system shall model the lifeguard-victim rescue process until the lifeguard reaches the victim.
      SR.5.1 The system shall model the three victim escape methods (swim parallel to neck, swim against the neck, and float)
      SR.5.2 The system shall model riptides of length 100/200/300/400/500 meters.
      SR.5.3 The system shall model the lifeguard speed on land and on water as an average velocity of X m/s and Y m/s respectively.

B. Input Requirements
   IR.1 The system shall inputted a victim swimming method. It will pick between floating, swimming parallel against the shore, and swimming parallel to shore.
      IR.1.1 The system shall model the swimming methods as velocities. The choice will be based on a discrete random distribution.
      IR.1.2 The system shall be inputted a random victim survival time based on the swimming method chosen.
   IR.2 The system shall be inputted a random rip current speed. The speed will be chosen by a random X distribution with mean X and variance X.

C. Output Requirements
   OR.1 The system shall output the victim position over time.
   OR.2 The system shall output the lifeguard position over time.
   OR.3 The system shall output the drone position over time.
   OR.4 The system shall output 1 or 0 depending if the lifeguard rescue time is under victim survival time.
      OR.4.1 The system shall detect if the drone reached the victim before the lifeguard and increase victim survival time by X seconds.
XI. PROPOSED DESIGN SOLUTIONS AND ALTERNATIVES

A. Proposed Design Solution

We will design a system that will increase the victim survive time during the rip current rescue process. This will be done by delivering a life-saving-device to the victim while they are in the water. The delivery will be done by the use of an aerial drone. The life-saving-device will be delivered before the lifeguard reaches the victim.

B. Drone Alternatives

Table 2 shows a comparison of drones of potential drones to use as a basis for our design that are sold by our sponsor Expert Drones. The Spreading Wings Series has the largest payload capacity and size with a middle of the road cost and flight time. We have agreed with our sponsor to use an S900 for testing purposes. [11]

<table>
<thead>
<tr>
<th>Drones</th>
<th>Cost ($)</th>
<th>Payload Weight (Kg)</th>
<th>Size</th>
<th>Flight Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>3DR X8+</td>
<td>1,350</td>
<td>1</td>
<td>350 x 510 x 200 mm</td>
<td>15 minutes</td>
</tr>
<tr>
<td>DJI Phantom 3</td>
<td></td>
<td></td>
<td>Diagonal size 590mm</td>
<td>23 minutes</td>
</tr>
<tr>
<td>DJI Inspire 1</td>
<td>2,899</td>
<td>0.465</td>
<td>438x451x301 mm</td>
<td>18 minutes</td>
</tr>
<tr>
<td>DJI Spreading Wings S1000+</td>
<td>1,900</td>
<td>6.6</td>
<td>Diagonal Wheelbase 1045mm</td>
<td>15 minutes</td>
</tr>
<tr>
<td>DJI Spreading Wings S900</td>
<td>1,400</td>
<td>4.9</td>
<td>Diagonal Wheelbase 900mm</td>
<td>18 minutes</td>
</tr>
</tbody>
</table>

TABLE 2 COMPARISON OF EXPERT DRONES AVAILABLE DRONES [8] [9] [10] [11] [12] [13]
C. Life Saving Device Alternatives

Alternatives for the life saving device are listed in Table 3. It compares the three most popular and reasonable floating devices in order to conclude the best alternative for the drone to carry and deliver to the drowning victim. The quantitative factors being compared are weight, dimensions, buoyancy, and cost [14]. The most reasonable one is the ring buoy as it is cost effective and the victim has a better chance of grasping it. The effectiveness and durability are the ratings from a professional lifeguard which is purely opinion based (TBA).

<table>
<thead>
<tr>
<th>Floatation Device</th>
<th>Weight (lbs.)</th>
<th>Dimensions (inches)</th>
<th>Buoyance</th>
<th>Cost</th>
<th>Effectiveness (5 star rating)</th>
<th>Durability (5 star rating)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ring Buoy</td>
<td>3</td>
<td>20</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lifeguard can</td>
<td>4</td>
<td>29.5x9.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lifejackets</td>
<td>1.5</td>
<td>17x10x2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Table 3 Life Saving Device Alternatives (in progress)**

XII. Design of Experiment

A. Force of Drag DoE

The purpose of this experiment is to determine the force of drag acting upon the Spreading Wings S900 while the drone is in flight. This is needed to accurately create a simulation of said drone. To determine the force of drag we need to perform a flight test of the drone and collect some of its telemetry for the flight.

The force of drag has been broken down into two components vertical and horizontal. Each of these tests will be performed at different speeds to determine the change in the coefficient of drag with the change in velocity of the drone.

In order to find the coefficient of drag of the drone, we need to collect from this experiment: pitch, roll, velocity of the wind, and velocity of the drone. We would also like to have recorded how far above the ground and how far from home the drone is. All of this data would be required to be timestamped. The velocity, altitude, and distance of the drone would also be used to verify that the flight went as expected.
## Table 4 Force of Drag DOE

<table>
<thead>
<tr>
<th></th>
<th>Inputs</th>
<th>Outputs</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Pitch</td>
</tr>
<tr>
<td>Procedure</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Horizontal $C_0$ test</td>
<td>fly 30 m north</td>
<td># 1</td>
</tr>
<tr>
<td></td>
<td>fly 30 m east</td>
<td># 2</td>
</tr>
<tr>
<td></td>
<td>fly 30 m south</td>
<td># 3</td>
</tr>
<tr>
<td></td>
<td>fly 30 m west</td>
<td># 4</td>
</tr>
<tr>
<td>Maintain constant 10 m</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Vertical $C_0$ test</td>
<td>Ascend 30 m</td>
<td># 1</td>
</tr>
<tr>
<td></td>
<td>Descend 30 m</td>
<td># 2</td>
</tr>
<tr>
<td></td>
<td>At least 10 m above ground</td>
<td></td>
</tr>
</tbody>
</table>

For each leg of the test there are three sections, a section where the drone accelerates to the required velocity, the section where the test will be performed, and a section where it decelerates and/or turns. The data will be gathered from the test section, where the drone will follow the flight procedure. All of the data must be ether timestamped or otherwise linked together. We would prefer the data in some giant data stream for the entire flight which we would parse later, but an excel file with data from the relevant sections would be acceptable.

### B. Force of Drag Equations

\[ F_L = F_T \cdot \cos(\theta) \]

\[ F_F = F_T \cdot \sin(\theta) \]

When the drone is in a state of forward flight maintaining a constant velocity and height, the drone is generating a force of thrust that is counteracting both the force of gravity and force of drag on the drone. This force of thrust consists of the force of lift, which is counteracting the force of gravity, and the forward force, which is counteracting the force of drag. Therefore if we know the pitch/roll of the drone at a specific velocity we can recompose the force of lift with the pitch/roll to find the force of drag. With the force of drag we can find the coefficient of drag.
\[ F_D = F_F \]
\[ F_D = F_T \times \sin(\theta) \quad ; \quad F_T = \frac{F_L}{\cos(\theta)} \]
\[ F_D = F_L \times \tan(\theta) \quad ; \quad F_L = m \times g \]
\[ F_D = m \times g \times \tan(\theta) \]
\[ C_D = \frac{2 \times F_D}{\rho \times v^2 \times A} \]

This would only net us the horizontal force of drag. To find the vertical force of drag if we consider the conservation of energy.

The force of drag is the loss of energy due to friction with the air divided by the distance traveled. This is equal to the potential energy at point 1 subtracted by the kinetic energy at point 2.

\[ F_D = \frac{(mgh_1 - \frac{1}{2}mv^2)}{h_1} \]
\[ C_D = \frac{2 \times F_D}{\rho \times v^2 \times A} \]

C. Motor Parameter DoE

The purpose of this experiment is to determine certain information about the brushless DC electric motors used on the Spreading Wings S900 for our simulation. Those parameters are: the no load current, the torque constant, and a way to relate torque to thrust. The properties of: the torque-speed curve [15], the power-speed curve [15], and the torque-current curve [16] of a DC electric motor will allow us to determine those values.

The torque-current curve is a line that relates the torque that the motor outputs with the current that the motor draws. It can be found by applying a fixed amount of torque to the motor and measuring the current that is being drawn. From the torque-current curve we will be able to determine the no-load current (a point on the line). The slope of the torque-current curve is the torque constant. [16]

To relate the torque to the thrust of the drone, we need to find both the power-speed curve, and the torque-speed curve. The thrust of the drone is related to the mechanical power output of the motor. The mechanical power output of the motor is for a given speed is the power-speed curve. [17]
The power-speed curve is derived from the torque-speed curve, the power of the motor at a given speed is the area of a square with one corner at the origin and another on the torque-speed curve. \[15\]

In order to find these curves we need to know the torque, current, and speed of the motor at a fixed voltage. The procedure we will follow is: \[17\]

1) Apply Fixed Voltage to motor, which is not to exceed motor rated voltage
2) Run the motor without a load applied to its shaft.
   a. The RPM of the motor will be recorded
      i. used in finding the torque-speed curve
   b. The current the motor is drawing will also be recorded
      i. used in finding the torque-current curve
3) Apply torque to the shaft of the motor
   a. Record the torque
      i. used in finding the torque-speed curve and the torque-current curve
   b. Record the current
      i. used in finding the torque-current curve

These steps will be repeated for additional voltages to determine if/how the data changes.

For this DOE we would require: 1) electric motor of the S900, 2) adjustable voltage supply, 3) ammeter, 4) ohm meter, 5) adjustable torque load, 6) non-contact tachometer.

<table>
<thead>
<tr>
<th>Inputs</th>
<th>Outputs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Voltage</td>
<td>Torque</td>
</tr>
<tr>
<td></td>
<td>Current</td>
</tr>
<tr>
<td></td>
<td>Speed (RPM)</td>
</tr>
</tbody>
</table>

\[TABLE 5\] MOTOR PARAMETER DOE

\[XIII. \] STATEMENT OF WORK

A. Scope of Work
1. The scope of the work we will do includes:
   a. Testing the feasibility (physical and economical) of lifesaving-device-delivery drones with varying drone and lifesaving-device properties.
      i. We will analyze current stakeholder and current rescue processes.
      ii. We will simulate drone delivery and other drone mechanics.
      iii. We will design a drone that can help lifeguards with the rescue process.
   b. Estimating life-cycle costs of such a system if it was employed.
2. The scope of work will NOT include:
a. Designed systems that help lifeguards but do not employ a drone. There is too many possibilities of modifying training, equipment, beach rules, submersibles, etc. that can be done or are being researched currently.

B. Schedule/Milestones

<table>
<thead>
<tr>
<th>Milestones</th>
<th>Dates</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fall Brief 1</td>
<td>9/21/2015</td>
</tr>
<tr>
<td>Fall Brief 2</td>
<td>10/5/2015</td>
</tr>
<tr>
<td>Preliminary Project Plan</td>
<td>10/21/2015</td>
</tr>
<tr>
<td>Fall Brief 3</td>
<td>10/26/2015</td>
</tr>
<tr>
<td>Fall Brief 4</td>
<td>11/09/2015</td>
</tr>
<tr>
<td>Faculty Presentation</td>
<td>11/20/2015</td>
</tr>
<tr>
<td>Proposal Final Reports</td>
<td>12/09/2015</td>
</tr>
<tr>
<td>Proposal Final Report Slides</td>
<td></td>
</tr>
<tr>
<td>Draft Conference Paper</td>
<td></td>
</tr>
<tr>
<td>Draft Poster</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Milestones (SYST 495 Spring 2016)</th>
<th>Dates</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spring Brief 1</td>
<td>2/2016</td>
</tr>
<tr>
<td>Spring Brief 2</td>
<td>2/2016</td>
</tr>
<tr>
<td>Spring Brief 3</td>
<td>3/2016</td>
</tr>
<tr>
<td>SIEDS Abstract Due</td>
<td>2/2016</td>
</tr>
<tr>
<td>SIEDS Notification</td>
<td>3/2016</td>
</tr>
<tr>
<td>SIEDS Manuscript Due</td>
<td>4/2016</td>
</tr>
<tr>
<td>SIEDS Conference</td>
<td>4/2016</td>
</tr>
<tr>
<td>Registration for Keith Memorial Capstone Conference</td>
<td>Before 4/2016</td>
</tr>
<tr>
<td>GDRKMC Conference</td>
<td>5/2016</td>
</tr>
</tbody>
</table>

**TABLE 6 SCHEDULE MILESTONES**
C. Type of Contract/Payment Schedule
1) The average salary of an Entry-Level Systems Engineer in Fairfax is 63,000 per year [18]. Assuming 50 work weeks and 40 hours a week, this is 27.50/hour.
2) We will round this up to $30 for simplicity.
3) Overhead will be a 2.0 multiplier (accord to Sherry)
4) Total charge is $60.00 per hour per person [18] [19]

XIV. WORK BREAKDOWN

A. Work Breakdown Structure

Of these tasks the stakeholder analysis, DOE, and especially the simulation are the heart of the project. It is crucial that we properly understand the stakeholders and what they want otherwise any design we come up with would never end up adopted as a life saving device. Without the simulation we will be unable to effectively trade off design alternatives. Similarly the DOE’s will be necessary to complete the simulation and verify its functionality.
B. Critical Tasks

We are considering a task as a critical task if it has a slack less than 2 days. This is a margin of error as in that time an unplanned for homework assignment or test could lead to slippage of the task for our project. The tasks we have found to be critical are: 1) Project File, 2) Tension and sequence diagrams, 3) Functional Requirements, 4) Evaluate Life Saving Device Alternatives, 5) Stakeholder: Lifeguard Research and Analysis, 6) Stakeholder Liability Research and Analysis, 7) Sensitivity Analysis, 8) Project Risk Analysis, 9) Simulation Risk Analysis, 10) Motor Experiment, 11) Program PID Control, 12) Create Rip Current Model, 13) Create GUI, 14) Simulation Testing, 15) Faculty Presentation, and 16) Final Reports and Stuff.
<table>
<thead>
<tr>
<th>WBS</th>
<th>Task Name</th>
<th>Duration</th>
<th>Start</th>
<th>Finish</th>
</tr>
</thead>
<tbody>
<tr>
<td>11.1</td>
<td>Start</td>
<td>0 days</td>
<td>8/31/2015</td>
<td>8/31/2015</td>
</tr>
<tr>
<td>8.1.4</td>
<td>Program PID Control</td>
<td>30.75 days</td>
<td>9/23/2015</td>
<td>12/7/2015</td>
</tr>
<tr>
<td>4.2</td>
<td>Functional Requirements</td>
<td>8.88 days?</td>
<td>10/22/2015</td>
<td>12/25/2015</td>
</tr>
<tr>
<td>9.2.1</td>
<td>Faculty Presentation Creation</td>
<td>5.13 days</td>
<td>11/14/2015</td>
<td>11/19/2015</td>
</tr>
<tr>
<td>9.2.2</td>
<td>Faculty Presentation</td>
<td>0 days</td>
<td>11/20/2015</td>
<td>11/20/2015</td>
</tr>
<tr>
<td>10.2.3</td>
<td>Final Reports and Stuff</td>
<td>14.25 days</td>
<td>11/23/2015</td>
<td>12/8/2015</td>
</tr>
<tr>
<td>6.4.3</td>
<td>Simulation Risk Analysis</td>
<td>8.5 days</td>
<td>12/7/2015</td>
<td>12/16/2015</td>
</tr>
<tr>
<td>10.1.2</td>
<td>Proposal Final Report</td>
<td>0 days</td>
<td>12/9/2015</td>
<td>12/9/2015</td>
</tr>
<tr>
<td>8.1.7</td>
<td>Create Rip Current Model</td>
<td>7 days?</td>
<td>2/17/2016</td>
<td>3/2/2016</td>
</tr>
<tr>
<td>9.4.3</td>
<td>Spring Brief 3</td>
<td>0 days</td>
<td>3/14/2016</td>
<td>3/14/2016</td>
</tr>
<tr>
<td>3.2.3</td>
<td>Sequence Diagrams</td>
<td>6.21 days?</td>
<td>3/29/2016</td>
<td>4/12/2016</td>
</tr>
<tr>
<td>3.2.2</td>
<td>Tension Diagrams</td>
<td>6.67 days?</td>
<td>4/2/2016</td>
<td>5/7/2016</td>
</tr>
<tr>
<td>11.3</td>
<td>Finish</td>
<td>0 days</td>
<td>5/14/2016</td>
<td>5/14/2016</td>
</tr>
</tbody>
</table>

**Table 7 Critical Path**
C. Schedule

Our project schedule dealing with stakeholders and building the simulation in all of 490 and first half of 495, Working on doe in 490 with some in early 495 and dealing with context and con-ops in 490.
D. Earned Value

Our earned value (BCWP) for the first few weeks of the project is an estimation of reality. It took us awhile to get our project file up and running so our planned and actual schedule is to the best of our recollection. We now have a baseline set and further BCWP will no longer be from memory. For the first 2 months our cost variance (CV) and schedule variance (SV) are both negative. Our actual cost (ACWP) has been consistently above our BCWP. Our BCWP was not that much different from our planned value (BCWS) but has been increasing. We have been consistently over budget. Our indices are hovering between 0.8 and 1. As of October 18th our ACWP is $36,642.68, our BCWP is $28,631.91. Our estimated cost at completion is $146,937.25.
# XV. **Project Risk Mitigation**

<table>
<thead>
<tr>
<th>Risk</th>
<th>Severity</th>
<th>Likelihood</th>
<th>Detectability</th>
<th>Score</th>
<th>Mitigation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simulation Control Structure is not done by Nov. 15th (2 weeks behind schedule)</td>
<td>8</td>
<td>10</td>
<td>1</td>
<td>80</td>
<td>Revert to old model with no control. Manually adjust voltages in order to get the right distance and other values needed.</td>
</tr>
<tr>
<td>Simulation Testing is Delayed by X days beyond scheduled due date</td>
<td>10</td>
<td>3</td>
<td>5</td>
<td>150</td>
<td>Do primary analysis of the drone’s effectiveness in reducing fatalities. Forgo all other simulation tests until we find time again.</td>
</tr>
<tr>
<td>Evaluate life saving device alternatives is not done, ring buoy information is wrong</td>
<td>9</td>
<td>3</td>
<td>5</td>
<td>135</td>
<td>Find other life saving devices with accurate information about them and evaluate them.</td>
</tr>
<tr>
<td>Gather accurate information about force, pitch, roll, yaw, velocity, height and wind speed</td>
<td>7</td>
<td>4</td>
<td>4</td>
<td>112</td>
<td>Perform the flight experiment again until we get accurate data. Use pocket money to get program and tablet that can watch the instruments.</td>
</tr>
<tr>
<td>Unable to acquire tools to perform experiments</td>
<td>10</td>
<td>6</td>
<td>1</td>
<td>60</td>
<td>Use the University’s lab experiments to get the tools.</td>
</tr>
</tbody>
</table>

*Table 8: Project Risks, RPN, and Mitigation*
XVI. References


A. Linear Dynamics

What comes into the system is the four voltages ($v_1, v_2, v_3, v_4$). They will be inputted into `volt2thrust` in order to get a value for the thrust in body form. In the linear motion model, we will use the equation above to calculate the acceleration. Note that mass of drone is separated outside, for sensitivity analysis purposes. After integrating the acceleration for position and velocity, these values will be used to calculate various other forces. Position will be used to calculate the tether force. Velocity will be used to calculate the drag forces on the drone and on the life vest.

Voltage to Thrust Conversion ($\text{volt2thrust}$)

```matlab
function \text{THRUST} = \text{volt2thrust}(v1,v2,v3,v4)
%\%codegen
%Function to convert voltage into the thrust force vector in inertia frame
%All units are metric unless specified

%constants
Kv = 1; %proportionality constant between propeller voltage and angular velocity
Kt = 1; %proportionality constant between propeller torque and current
KT = 1; %proportionality constant between propeller thrust and torque
D = 1; %propeller diameter. Adjusted by propeller properties
rho = 1; %density of air (humid or dry). Probably need to be a function

%V = Kv * w assuming internal resistance = 0
w1 = v1/Kv;
w2 = v2/Kv;
w3 = v3/Kv;
```
\[
\begin{align*}
w_4 &= \frac{v_4}{K_v}; \\
W &= \left[0;0;(w_1^2+w_2^2+w_3^2+w_4^2)\right]; \\
k &= \frac{\pi/2*\rho*D^2}{(K_v/K_t/K_T)}; \\
\text{%ThrustBody} &= \text{thrust vector in body frame} \\
\text{thrustBody} &= k \times W; \\
\text{%THRUST} &= \text{thrust vector in inertia frame} \\
\text{THRUST} &= \text{thrustBody}; \\
\end{align*}
\]

**Linear Motion Model (DroneState)**

```matlab
function [ax,ay,az] = DroneState(Fthrust,Frope,Fdrag,Flifevest,EULER,massDrone)
%#codegen
%--------------------------------------------------
%rotation matrix for converting body-frame thrust to inertia-frame thrust
roll=EULER(1);
pitch=EULER(2);
yaw=EULER(3);
Ryaw = [cos(yaw) sin(yaw) 0;-1*sin(yaw) cos(yaw) 0;0 0 1];
Rpitch = [cos(pitch) 0 -1*sin(pitch);0 1 0; sin(pitch) 0 cos(pitch)];
Rroll = [1 0 0;0 cos(roll) sin(roll);0 -1*sin(roll) cos(roll)];
RotMat = Ryaw*Rpitch*Rroll;

THRUST = RotMat*Fthrust;
%--------------------------------------------------

Fx = 0 + THRUST(1,1) + Frope(1,1) + Fdrag(1,1) + Flifevest(1,1);
Fy = 0 + THRUST(2,1) + Frope(2,1) + Fdrag(2,1) + Flifevest(2,1);
Fz = -massDrone*9.81 + THRUST(3,1) + Frope(3,1) + Fdrag(3,1) + Flifevest(3,1);

ax = Fx/massDrone;
ay = Fy/massDrone;
az = Fz/massDrone;
```

**Force of Rope (RopeState)**

```matlab
function Frope = RopeState(x,y,z)
%#codegen
h = 1; %vertical difference between ends
L = 1; %horizontal difference between ends
mu = 1; %weight/length density of rope
S = 1; %arc length of rope
```
%first find absolute distance between ends
LEN = sqrt(x^2+y^2+z^2);

S = 1.1*LEN; %assumption: arc length is proportional to absolute distance

lamda = sqrt(S^2-d^2)/(L^2);

Fg = [0;0;-mu*LEN];
Fh = 0;
%if lamda is <1, equation will not work, must approximate by straight length?
if lamda<=1
    theta = asin(sqrt(x^2+y^2)/LEN);
    Fh = Fg(3)*tan(theta);
else
    syms u
    Y = solve(sinh(u)/u == lamda);
    Fh = mu*L/(2*Y);
end
HorizV = Fh[-x;-y;0]/sqrt(x^2+y^2);
Frope = Fg+HorizV;

% Frope = [0;0;0];
end

Force of Drag (Fdrag)

function Fdrag = Fdrag(vx,vy,vz)
 %#codegen

%force of drag depending on its air resistance
%pulls drone BACKWARDS
Cd = 0.02; %coefficient of drag for the lifevest
rho = 1; %air density
A = 1; %cross sectional area of lifevest
%backwards = negative of current velocity vector
Vsqr = sqrt((vx^2)+(vy^2)+(vz^2)); %total velocity magnitude
%unit vector direction of velocity
mag = sqrt(vx^2+vy^2+vz^2); %magnitude of total velocity
if mag==0
    mag=1;
end
unitV = [vx;vy;vz]*-1/mag;

Fmag = Cd*rho*A*Vsqr^2; %magnitude of drag force
Fdrag = Fmag.*unitV;

% Fdrag=[0;0;0];
**Force of Lifevest (LifevestState)**

```matlab
function Flife = LifevestState(vx, vy, vz)
    %#codegen
    % force of drag depending on its air resistance
    % pulls drone BACKWARDS
    Cd = 1; % coefficient of drag for the lifevest
    rho = 1; % air density
    A = 1; % cross sectional area of lifevest
    % backwards = negative of current velocity vector
    Vsqr = sqrt((vx^2) + (vy^2) + (vz^2)); % total velocity magnitude
    % unit vector direction of velocity
    mag = sqrt(vx^2 + vy^2 + vz^2); % magnitude of total velocity
    if mag == 0
        mag = 1;
    end
    unitV = [vx; vy; vz] * -1/mag;

    Fdrag = Cd * rho * A * Vsqr^2;
    FDRAG = Fdrag .* unitV;

    % force of weight depending on mass
    vestMass = 1; % mass of the lifevest
    Fgrav = [0; 0; vestMass * -9.81]; % pulls drone DOWN

    Flife = FDRAG + Fgrav;
    % Flife = [0; 0; 0];
```

B. Rotational Dynamics

Similar to the linear motion model, we take in four voltages (v1, v2, v3, v4) to calculate the angular velocities of each rotor. The velocities and the derivative (acceleration) will be used to calculate the torques in the main rotational model (21). Since we need the Euler angles instead of angular velocity, we need to convert angular velocity to Euler angle derivatives. Then we integrate the Euler angles into roll, pitch, and yaw which we can use for the rotation matrices in the linear motion model (4)(17).

**Voltage to Motor Angular Acceleration and Velocity**

```matlab
function Wmotor = volt2motorW(v1,v2,v3,v4)
    Kv = 1;  % proportionality constant between voltage and motor angular velocity
    w1=v1/Kv;
    w2=v2/Kv;
    w3=v3/Kv;
    w4=v4/Kv;
    Wmotor = [w1,w2,w3,w4];
end
```

**Motor Angular Velocity to Torque in Body Frame**

```matlab
function [Troll,Tpitch,Tyaw] = w2tau(W,Wdot)
    Kv=1;  % proportionality constant between motor voltage to angular velocity
    Kt = 1;  % proportionality constant between propeller torque and current
    KT = 1;  % proportionality constant between propeller thrust and torque
    L=1;  % distance between motor and center of body frame.
    D=1;  % diameter of propeller
    rho=1;  % density of air
end
```
Cd = 1; % coefficient of drag for propeller wings
A = 1; % propeller cross section
r = D/2; % radius of propeller. Do not change

Iz = 1; % moment of inertia for propellers

% equations for constants we will need for τ
k = pi/2*rho*D^2*(Kv*Kτ/KT);
b = 0.5*r*rho*Cd*A*r^2;

% assume w1 and w3 are right and left motors respectively.
% assume w2 and w4 are back and front motors respectively.
% grab angular motor velocities
w1 = W(1);
w2 = W(2);
w3 = W(3);
w4 = W(4);

% τ of roll pitch and yaw based on angular velocities
Troll = L*k*(w1^2 - w3^2);
Tpitch = L*k*(w2^2 - w4^2);
Tyaw = b*(w1^2 - w2^2 + w3^2 - w4^2) + Iz*(Wdot(1) - Wdot(2) + Wdot(3) - Wdot(4));

Rotational Motion Model

function [wdotx, wdoty, wdotz] = RotationMotion(Troll, Tpitch, Tyaw, eulerDot, euler)
% #codegen

% moment of inertias in the X/Y/Z-axis of the drone body frame
Ixx = 1;
Iyy = 1;
Izz = 1;
I = [Ixx 0 0; 0 Iyy 0; 0 0 Izz];

roll = euler(1);
pitch = euler(2);
yaw = euler(3);

rollDot = eulerDot(1);
pitchDot = eulerDot(2);
yawDot = eulerDot(3);

τ = [Troll; Tpitch; Tyaw];
% First generate rotation matrix for angular velocity and thetaDot
eulerDot2Ω = [1 0 -sin(pitch);
0 cos(roll) sin(roll)*cos(pitch);
0 -sin(roll) cos(roll)*cos(pitch)];

% convert the euler Dots to angular velocities in inertia frame
Ω = eulerDot2Ω*[rollDot; pitchDot; yawDot];
W = [Ω(1); Ω(2); Ω(3)];
%use rotational motion equations now
Wdot = inv(I)*(τ - cross(W,I*W));
wdotx=Wdot(1);
wdoty=Wdot(2);
wdotz=Wdot(3);

Convert body angular velocity to Euler angle velocity

function [eulerDotRoll,eulerDotPitch,eulerDotYaw] = w2eulerDot(wx,wy,wz, euler)
  %#codegen
  roll = euler(1);
  pitch = euler(2);
  yaw = euler(3);

  %rotational matrix for EULERDOT to W. need to inverse it to get
  %the relevant transformation matrix
  E2W = [1 0 -sin(pitch);
  0 cos(roll) sin(roll)*cos(pitch);
  0 -sin(roll) cos(roll)*cos(pitch)];

  eulerDot = E2W*[wx;wy;wz];

  eulerDotRoll = eulerDot(1);
  eulerDotPitch = eulerDot(2);
  eulerDotYaw = eulerDot(3);

Euler Angle PID Controller
Position PID Controller

C. Main Model

The model will take in the waypoint position (and velocity, but we do not use this) and calculate the PID errors based on current drone position. The position error will be used to calculate the wanted Euler angles. For example, if we want to accelerate in the X-direction and we want to go at max velocity, we need to a large magnitude of pitch. On the other hand, if we need to slow down, we need to pitch in the opposite direction. We assume the drone will have a max angle of roll-pitch-yaw. Many controls on drones set the max angles at 45 or 30 degrees. We chose 30 degrees as it is more common of a limit.

After calculating the wanted angle, we calculate the angle PID error. Both errors will be used in the actuator, which converts the error readings into voltage. We simply added the errors to the hover voltage (voltage needed to hover, stay in the same position), in order to get the next time step’s voltage. We must adjust the PID gains in order for voltage to make sense of the errors.
**set the desired Euler angle**

```matlab
function eulerWanted = setWantedAngles(Eposi)
    %#codegen

    Ex = Eposi(1);
    Ey = Eposi(2);
    %#no error for yaw currently!
    %---------------------------------------------------------------
    %#Now we set next iteration's wanted eulers based on error
    %#The angles we want
    Wroll = 0;
    Wpitch = 0;
    Wyaw = 0;

    if (Ex>0) Wpitch=-((pi/6 - exp(-0.05*Ex+log(pi/6)))); end
    if (Ex<0) Wpitch= (pi/6 - exp( 0.05*Ex+log(pi/6))); end
    if (Ey>0) Wroll= (pi/6 - exp(-Ey+log(pi/6))); end
    if (Ey<0) Wroll=-((pi/6 - exp( Ey+log(pi/6)))); end

    %#no current controls for yaw
    % eulerWanted = [Wroll*abs(Ey)/10;Wpitch*abs(Ex)/10;Wyaw];
    eulerWanted = [Wroll;Wpitch;Wyaw];
```

**Main Control Actuator**

```matlab
function [v1,v2,v3,v4] = PIDHumanControl( Eposi, Eeuler, mass)
    %#codegen

    maxVoltage = 10; %#Maximum voltage of the rotors
    %#voltage totally based on error.
    v1=0;
    v2=0;
    v3=0;
    v4=0;

    %#Set the angles we want to go for----------------------------------
    %The specific errors in the X,Y,Z axis of the inertia frame below.
    Ex = Eposi(1);
    Ey = Eposi(2);
    Ez = Eposi(3);

    Eroll = Eeuler(1);
    Epitch = Eeuler(2);
    Eyaw = Eeuler(3);

    %#if roll error is positive, we need to roll more in positive radians.
    And
    %#so on for the other angles
```
% how much thrust is needed to hover? (assuming 0 angles)
Kv = 1; % proportionality constant between propeller voltage and angular velocity
Kτ = 1; % proportionality constant between propeller torque and current
KT = 1; % proportionality constant between propeller thrust and torque
D = 1; % propeller diameter. Adjusted by propeller properties
rho = 1; % density of air (humid or dry). Probably need to be a function

k = pi/2*rho*D^2*(Kv/Kτ/KT);

vHover = Kv*sqrt(mass*9.81/(4*k));

% Ex+Ey+Ez is the throttle of the motors. More distance error, more thrust!
% The Eeulers is the turning force of the drone, more roll needed, more rolling we must adjust for

% version 1
v1 = vHover + Ez + Eroll + Eyaw;
v2 = vHover + Ez + Epitch - Eyaw;
v3 = vHover + Ez - Eroll + Eyaw;
v4 = vHover + Ez - Epitch - Eyaw;

% set the limits of voltage based on proportions
% first find the minimum and maximum raw voltage.
minV = min([v1, v2, v3, v4]);

% if minimum voltage is less than zero, add that minimum to every voltage so the minimum is now zero
if (minV<0)
    v1 = v1-minV;
    v2 = v2-minV;
    v3 = v3-minV;
    v4 = v4-minV;
end

% if the highest voltage is greater than the max voltage, divide everything
maxV = max([v1, v2, v3, v4]);
% so everything is still proportionnal
if (maxV>maxVoltage)
    v1 = v1*maxVoltage/maxV;
    v2 = v2*maxVoltage/maxV;
    v3 = v3*maxVoltage/maxV;
    v4 = v4*maxVoltage/maxV;
end