VARIABLE BUOYANCY CONTROL FOR A BOTTOM SKIMMING AUTONOMOUS UNDERWATER VEHICLE

A THESIS SUBMITTED TO THE GRADUATE DIVISION OF THE UNIVERSITY OF HAWAI’I AT MĀNOA IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

IN

MECHANICAL ENGINEERING

DECEMBER 2014

By

Anthony Sylvester III

Thesis Committee:

Brian Bingham, Chairperson
A Zachary Trimble
Reza Ghorbani
We certify that we have read this thesis and that, in our opinion, it is satisfactory in scope and quality as a thesis for the degree of Master of Science in Mechanical Engineering.

THESIS COMMITTEE

______________________________
Dr. Brian Bingham, Chairperson

______________________________
Dr. A Zachary Trimble

______________________________
Dr. Reza Ghorbani
“When you have exhausted all possibilities, remember this - you haven't.”

- Thomas A. Edison

“Life is a storm, my young friend. You will bask in the sunlight one moment, be shattered on the rocks the next. What makes you a man is what you do when that storm comes.”

- Alexandre Dumas
Acknowledgements

I would like to thank my family and friends who have guided me to this point in my life. I am specifically grateful for the encouragement from my mom and dad. They are the ones whom instilled in me a technical interest at a very young age and pushed me to explore my potential. I would also like to thank my girlfriend Melissa, and her family who have always been there for me over my college years. Thank you to Professor Reza Ghorbani and Professor A Zachary Trimble for your involvement in my academic journey, and for joining my thesis committee. Last but not least, I would like to thank Professor Brian Bingham. Thank you for being my mentor, advisor, teacher, colleague, and friend. I have grown and learned more than I could have ever hoped for over the past few years working with you and everyone in the lab. I greatly appreciate the unique and fulfilling opportunities you have presented me with, and it has been a real privilege.
Abstract

A variable buoyancy system (VBS) is a critical component that is widely used to achieve controlled operation of an autonomous underwater vehicle (AUV). In this thesis two on/off feedback controllers are presented that utilize data averaging and model-based estimation to offset the effects of sensor noise and achieve precise control of the VBS developed for a prototype AUV. Operation of the prototype bottom skimming AUV requires a constant reaction force between the seabed and the vehicle. While performing a mission, variable seafloor topography and a changing payload weight requires the use of a VBS to regulate the reaction force. One trait that made development of the VBS system a challenging problem is the presence of sensor noise that could not be mitigated with hardware solutions. The software solutions implemented here needed to minimize any time delays in their response that could cause destabilization when coupled with the system’s fast on/off actuation.

It was discovered that both of the presented controllers function under these conditions, but the model-based controller provides more precise control of the system. The data averaging controller was able to regulate the buoyancy of the system to within 5.0 pounds of the commanded setpoint, while the model based controller was able to regulate the buoyancy to within 1.25 pounds of the desired setpoint, an improvement in performance by a factor of 4. Presented here is the development and comparison of these two control algorithms based on both simulation results and field experiments conducted in a coastal environment.
# Table of Contents

Acknowledgements........................................................................................................................................ v

Abstract.......................................................................................................................................................... vi

Table of Contents.......................................................................................................................................... vii

Chapter 1 – Introduction............................................................................................................................... 1
  1.1 Overview/Motivation................................................................................................................................. 1
  1.2 Contributions ........................................................................................................................................ 4

Chapter 2 – Background................................................................................................................................. 5
  2.1 Literature Review .................................................................................................................................... 5
    2.1.1 Closely Related Work ...................................................................................................................... 5
  2.2 Variable Buoyancy Control ..................................................................................................................... 6
    2.2.1 Types of VBS Implementations ....................................................................................................... 7
    2.2.2 Description of VBS Chosen for this System .................................................................................... 7

Chapter 3 – Vehicle Project Platform............................................................................................................ 8
  3.1 Hardware Overview ................................................................................................................................. 8
  3.2 Vehicle Software ................................................................................................................................... 9

Chapter 4 – System Model.............................................................................................................................. 13
  4.1 VBS Model .......................................................................................................................................... 13
  4.2 Depth Effects ....................................................................................................................................... 14
  4.3 Loss of Payload Weight ......................................................................................................................... 15

Chapter 5 – Control Approach ...................................................................................................................... 16
  5.1 Filter/Controller Combination ............................................................................................................... 16
    5.1.1 Bang-Bang (On/Off) Control ......................................................................................................... 17
    5.1.2 System Limitations - Fast Flow Rates Lead to Overshoot .............................................................. 17
  5.2 Moving Average Filter ........................................................................................................................... 18
    5.2.1 Moving Average Model ............................................................................................................... 18
    5.2.2 Averaging Time Delay ................................................................................................................. 19
  5.3 Kalman Filter ....................................................................................................................................... 19
    5.3.1 Filter Model ................................................................................................................................. 19
<table>
<thead>
<tr>
<th>Chapter 6 – Results</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.1 Testing Scenarios</td>
<td>22</td>
</tr>
<tr>
<td>6.1.1 Bang-Bang Control: Moving Average</td>
<td>23</td>
</tr>
<tr>
<td>6.1.2 Bang-Bang Control: Kalman Filter</td>
<td>23</td>
</tr>
<tr>
<td>6.2 Simulation Results</td>
<td>24</td>
</tr>
<tr>
<td>6.2.1 Closed-Loop Simulation</td>
<td>24</td>
</tr>
<tr>
<td>6.2.2 Open-Loop Simulation</td>
<td>26</td>
</tr>
<tr>
<td>6.3 Experimental Results</td>
<td>28</td>
</tr>
<tr>
<td>6.3.1 Closed-Loop Field Experiments: Moving Average</td>
<td>28</td>
</tr>
<tr>
<td>6.3.2 Closed-Loop Field Experiments: Kalman Filter</td>
<td>31</td>
</tr>
<tr>
<td>6.4 Results Summary</td>
<td>33</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chapter 7 – Conclusions</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.1 Summary</td>
<td>36</td>
</tr>
<tr>
<td>7.2 Contribution</td>
<td>36</td>
</tr>
<tr>
<td>7.3 Future Work</td>
<td>36</td>
</tr>
<tr>
<td>Bibliography</td>
<td>38</td>
</tr>
<tr>
<td>Appendix A: Reference Figures</td>
<td>40</td>
</tr>
</tbody>
</table>
Chapter 1 – Introduction

1.1 Overview/Motivation

The goal of this work is to implement a variable buoyancy system (VBS) for a bottom skimming autonomous underwater vehicle (AUV) currently being developed through a collaboration between Makai Ocean Engineering and the University of Hawaii. During operation the AUV transits the sea floor while maintaining constant contact with the seabed. In order to successfully complete its mission it is crucial that the AUV does not lose contact with the seabed. This condition is met by regulating the net wet weight of the AUV using a VBS. Furthermore the VBS should regulate the net wet weight of the AUV to a negatively buoyant state so that the sediment exerts a constant, precise normal force on the AUV despite changes in depth as well as changes in the mass of the AUV during a mission.

The VBS consists of a variable volume ballast chamber and a sensor to measure the normal force between the seabed and the AUV. The volume of the air-filled ballast chamber is changed by adding air from fixed volume compressed air cylinders or venting air from the chamber to seawater. This actuation of the VBS system is accomplished using two on/off solenoid valves commanded via a digital signal to be in either a fully open or fully closed state. The reaction force between the AUV and ocean floor is obtained indirectly from the output of a load cell situated to measure the net upward force provided by the ballast chamber (see Fig. 1).

The VBS is implemented with feedback control to actively change the volume of the ballast chamber based on the load cell measurements. It was discovered that one fatal characteristic of the control system is the presence of sensor noise on the load cell signal. The analog voltage output from the load cell is highly susceptible to electromagnetic noise from nearby electric motors, despite the inclusion of twisted-shielded-pair wiring and low-pass filters. This noise is illustrated in Fig. 2. One way to address this added noise is to introduce software filtering such as a moving average filter or a Kalman
filter. However, the addition of a moving average filter can introduce significant time delays in the feedback loop which can be destabilizing. The rate of volume change produced by the solenoid valves is high relative to the moving average filter time delay which is a further destabilizing influence on the system. The Kalman filter approach utilizes a mathematical model of the control system. This model incorporates known control actuation commands and combines this with previous sensor input values to generate predicted state estimate values. If implemented correctly these state estimate values can overcome the sensor noise and provide estimates that are closer to the true quantity being measured by

Figure 1: a) AUV conceptual image in the X-Z plane. FB is the buoyant force from the ballast chamber, FW is the wet weight of the vehicle, and FR is the reaction force between the vehicle and the sea floor. b) Venting of the VBS variable volume ballast chamber during AUV field trials in Hawaii.
the sensor. This study will focus on implementing the software filters mentioned above and evaluating their effectiveness in controlling the buoyancy of the AUV.

One alternative solution to the software filter approach would be to reduce the electrical noise from the system as much as possible. Completely eliminating the noise from the system would be ideal, but this scenario is not attainable. This alternative solution was also not possible given the pre-existing nature of the prototype AUV. The results reported here consider the design of the VBS system as fixed whereas no physical changes can be made to the existing system. Additionally, dealing with noisy sensors and fast on/off actuators is a relatively common situation for underwater robots and similar electromechanical systems.

Figure 2: Raw normal force measurements from the VBS load cell (blue) after passing through a hardware low-pass filter. It is assumed that the remainder of the noise is broadband. The true normal force is constant, but the sensor signal shows a standard deviation of ±2.19 lbs due to noise. To meet the system requirements a standard deviation of ±1 lb is necessary. The moving average (red) accomplishes this.
1.2 Contributions

The first contribution of this work is the development of a software filter/controller combination to reliably control a system with on/off actuators and input sensor measurement noise. This is first accomplished through the application of a moving average filter coupled with a bang-bang controller. A linear Kalman filter was then developed and coupled with a bang-bang controller.

The next contribution is the results from both computer simulations and a number of field trials to compare the above attempts and to assess their effectiveness. These field trials were conducted using a prototype AUV in the coastal waters around the island of Oahu.
Chapter 2 – Background

2.1 Literature Review

A number of different approaches to variable buoyancy control have been used to successfully regulate the buoyancy of AUVs [1]. A review of the literature shows that classic PI control, sliding mode control, and fuzzy logic control have all been applied to the control of underwater vehicle buoyancy and depth regulating systems. In [3] sliding mode control of vehicle pitch is designed to compensate for a net positive vehicle buoyancy, and actuation is provided by continuously articulated elevator fins. Riedel et al. utilize a linear-quadratic regulator in their ballast controller, which actuates ballast pumps with variable flow rates located fore and aft in their AUV to control pitch and depth, in addition to vertical thrusters [4]. In [5], DeBitetto proposes a fuzzy logic controller for an AUV with a similar mechanical arrangement to [4], utilizing variable flow rate ballast pumps. However, the reviewed buoyancy control techniques are not applicable to the binary on/off actuators in this type of system.

Other work was investigated where on/off solenoids are used in controlling air ventilation [6] and teleoperated systems [7]. In [6] the on/off solenoids are controlled using pulse width modulation (PWM) signals, whereas [7] makes use of sliding mode control to actuate a number of on/off solenoids in the system. Both sliding mode control and PWM control are not feasible for the as-built AUV due to existing mechanical and instrumentation limitations. These limitations will be discussed in further sections. It was therefore necessary to develop the moving average and model-based methods presented here. A survey of model predictive control methods is presented in [8], and the model-based controller developed in this paper is similar to the simplified Kalman filter described therein.

2.1.1 Closely Related Work

Some attempts to solve similar control issues have been documented in [11] and [12]. A generalized predictive control (GPC) technique is used in [11] that incorporates a system model and
known control signals to arrive at a predicted output. GPC can be used incrementally or over a sequence of inputs and control signals. One unique feature of GPC is the use of prediction horizons, which allow for more active control and can be scaled to more complex systems. At the very top level the GPC process can be described as the minimization of predicted system errors using known input and output data, where some constraints are placed on the projected control signals through the use of prediction horizons. An extension of the work in [11] can be found in [12] where GPC is tested further and analyzed for broader applications. GPC and the Kalman filter approach presented here share some similarities in that both methods utilize a model to formulate a future prediction, and this is done incrementally instead of over a sequence of collected measurements. One difference between these approaches is that the Kalman filter utilizes a Kalman gain value dependent upon a covariance prediction to alter the state prediction and arrive at a state estimate at the end of an iteration cycle.

Kalman filters have been used to reduce noise in varying capacities. Some applications range from noisy speech recognition [13] to noise reduction in gray image processing [14]. In [15] a Kalman filter is used to cope with quantization noise on a bilateral teleoperated system resulting from insufficient sensor resolution. An extended Kalman filter is also used in [16] to produce estimates of a signal with the presence of noise on the sensor input power voltage.

### 2.2 Variable Buoyancy Control

A buoyancy regulating system is a feature present on many AUVs. Achieving precise regulation of a vehicle’s buoyancy can have many operational benefits. Proper buoyancy adjustments can decrease the load on thrusters and propulsion systems resulting in more efficient use of power. Buoyancy control can also contribute to the maneuverability of an underwater vehicle. For the specific vehicle discussed in this thesis, buoyancy control is crucial to complete the vehicle’s mission. The vehicle needs to exert a
constant normal force on the seabed despite loss of payload weight. This is an impossible task without the use of a VBS.

2.2.1 Types of VBS Implementations

Within the AUV and remotely operated vehicle (ROV) community there are a number of variable buoyancy systems in use. A VBS will usually fall into one of three categories: pumped water, oil displacement, and mass discharge systems [1]. The pumped water VBS utilizes a rigid fixed volume buoyancy chamber which water can be pumped in to reduce buoyancy. Water will then be pumped out in order to increase buoyancy. An oil displacement system is completely isolated from contact with sea water. This functions by pumping oil from within a fixed volume chamber out into a flexible bladder. As the flexible bladder expands it displaces sea water. The lower density of the oil now displacing the sea water will result in an increase in buoyancy. This evolution can be reversed to reduce the buoyancy of the oil displacement system. A mass discharge system will shed mass with a higher density than water to increase buoyancy. Alternatively it will discharge a gas such as air with a lower density than water to decrease buoyancy. The VBS used on this AUV is considered a mass discharge system.

2.2.2 Description of VBS Chosen for this System

The VBS chosen for this vehicle is one that utilizes a flexible variable volume ballast chamber in which a gas can be filled into and vented from. This system utilizes a flexible ballast chamber that allows buoyancy adjustment in both the positive and negative directions. The filling and venting of gas into the chamber is accomplished through the use of on/off solenoids. The gas used to fill the ballast chamber is from high pressure compressed air tanks attached to the vehicle. The air is then vented to seawater from the ballast chamber as needed to reduce buoyancy. The vehicle only possesses a finite amount of compressed air. Therefore, it is important that the VBS does not encounter over actuation situations where the compressed air is wasted.
Chapter 3 – Vehicle Project Platform

3.1 Hardware Overview

The AUV that this VBS is being developed for is unique in both function and appearance. The AUV resembles a cylinder and is approximately fifteen feet long with a diameter of two feet (see Fig. 3). The vehicle operates in the vertical position with one side in contact with the sea floor. The buoyancy chamber is attached at the opposite end and streaming toward the ocean surface.

The VBS system consists of a buoyancy chamber, two on/off solenoids, two scuba tanks, a load cell, various electrical wiring, and swagelock tubing routed throughout the system. The buoyancy bag has the capability to supply up to 300 pounds of positive buoyancy. The on/off solenoids are the normally closed type and can be commanded into either the fully opened (on) or fully closed (off) states. The current instrumentation build does not have the capability for PWM control of the solenoids thus...
limiting the possible control schemes. One of the on/off solenoids is used to fill air into the buoyancy bag and is connected to two scuba tanks that contain compressed air. The second solenoid is used to vent air from the buoyancy chamber to seawater. The load cell is fixed to the vehicle on one side with the opposite side attached to the buoyancy chamber. The load cell is the only link between the vehicle and the buoyancy chamber to ensure proper monitoring of the buoyant force resulting from the buoyancy chamber. The load cell is rated to measure up to 500 lbs of force and outputs an analog voltage signal within the range of 0.0-20.0 mV. The load cell is wired to an electronics package that contains a power system, analog to digital converter (16 bit) and processor (single-board computer). The processor takes care of data acquisition from the load cell as well as running the vehicle software and the filtering software developed for this study.

3.2 Vehicle Software

The software used on the vehicle was developed in house by the Field Robotics Laboratory (FRL). The software suite is called the FRL Vehicle Software (FVS) [10]. A flowchart of the FVS structure is shown in Fig. 4. FVS consists of three main parts: hardware drivers, real-time control modules, and simulation tools that enable rapid prototyping of AUV operations. The hardware device drivers are typically written in C code and interface directly with instruments that communicate through RS232 serial, analog, as well as digital signals. The real-time control modules and simulation tools are written in Python. The three primary functions of the software are offline simulation, real-time simulation, and real-time control. Both simulation modes allow testing and development of control algorithms without the use of hardware. The difference between the two simulation modes is that the offline simulation occurs as quickly as the process can be computed. For example a ten minute mission may only take ten seconds to run in the offline simulation. The real-time simulation operates using the same time scale as the mission. The same ten minute mission will take ten minutes to complete in the real-time mode. The real-time operation mode also utilizes actual message passing between software modules so it is
effective at testing software development as a whole, whereas the offline simulation mode can only be
used to verify algorithm functionality. When operating on the AUV the software works in the real-time
control mode. During this control mode the software is interfacing with actual sensors and actuators on
the AUV in contact with the operational environment.

To pass messages between the hardware drivers and other control software modules, FVS uses
the Lightweight Communications and Marshalling (LCM) libraries [9]. LCM operates on the
publish/subscribe message passing functionality and is similar to the Robot Operating System (ROS).
LCM was chosen for use with FVS for a number of reasons discussed in [10]. LCM was developed by MIT
for the DARPA Urban Challenge. LCM communicates over UDP/IP protocol and is multicast capable,
allowing decentralized communication. This decentralized property is particularly useful in robotic
development. In Fig. 4 the LCM presence is noted by types in red and channels in blue. Channels can be
subscribed to or published to within Python and other software scripts. The LCM channels are described

Figure 4: AUV software flowchart. On the far left all of the sensors are represented by grey boxes with rounded edges. Their respective drivers are then shown in the blue rectangles. The control software is detailed by the green rectangles, most of these modules are written in Python. The lcm channels are shown in blue, with their corresponding types shown in red.
by their type, where multiple LCM channels can utilize the same type yet still be subscribed to independently of each other.

The software filters discussed here were developed in the python programming language. Within the software flowchart the filters are called separately by the state estimator module in green. The filtered outputs are then used by the controller to pass control commands to the actuator module based on criteria set within the controller module. A screenshot of the Kalman filter Python code can be seen in Fig. 6. This algorithm is also shown using a simplified pseudocode form in Fig. 5.

```
SET A matrix to scalar value 1
SET B matrix values equal to model rates of change multiplied by change in time
SET initial state prediction to previous state estimate
SET initial covariance matrix to previous covariance estimate
SET the measurement matrix to scalar value 1 since we are only using one sensor input
SET the process noise matrix this is the tuning knob for the filter
SET the measurement noise matrix based on known system noise

COMPUTE state prediction by summing the initial state prediction with the product of the B matrix and control commands

COMPUTE the measurement prediction by multiplying the measurement matrix and state prediction

COMPUTE residual by subtracting the sensor measurement from measurement prediction

COMPUTE predicted state prediction covariance by adding the process noise matrix to the product of the initial covariance matrix and A matrix

COMPUTE measurement prediction covariance by multiplying by multiplying the predicted state prediction covariance with the measurement matrix and adding the measurement noise matrix

COMPUTE Kalman gain by multiplying the predicted state prediction covariance with the measurement matrix then multiply the result by the inverse of the measurement prediction covariance

COMPUTE the state estimate by adding the state prediction to the product of the Kalman gain and the residual

COMPUTE the updated state covariance estimate by multiplying the measurement prediction covariance with the square of the Kalman gain then subtract this product from the predicted state prediction covariance

RETURN the computed state estimate
RETURN the computed state covariance estimate
END ITERATION
```

Figure 5: Pseudocode representation of the Kalman filter algorithm. At the end of each iteration the state estimate and state covariance estimate are computed and returned. These returned values will then be used as the initial values in the next iteration.
```python
class KalmanLinear(object):
    """A linear Kalman Filter class""
    def execute(self, Z, Xe, Pe, u, dt):
        """inputs:
        Z = measurement
        Xe = initial/previous state estimate
        Pe = initial/previous covariance estimate
        u = control input [slowfil, vent]
        dt = timestep
        """
        outputs:
        Xest = state estimate advanced by dt
        Pest = state covariance estimate advanced by dt
        A = 1
        FR = 4.25 # fill rate in lb/s
        VR = 1.75 # vent rate in lb/s
        FFR = 25.0 # fast fill rate in lb/s
        B = [FR*dt, VR*dt, FFR*dt]
        Xp0 = Xe # initial state prediction
        Pp0 = Pe # initial covariance matrix (in standard deviations squared)
        H = 1 # measurement matrix
        Q = 0.001 #0.0005 #0.01 #4.79 #0.0001 #0.005 # process noise matrix
        R = 4.79 # measurement-noise matrix
        Xp = min(Xp,500.0) # out bag limits the buoyancy.
        Zp = H*Xp # measurement prediction
        Re = Z-Zp # measurement residual
        Pp = A*Pp0*A+Q # state prediction covariance
        S = H*Pp*H+R # measurement prediction covariance
        K = Pp*H*(S**-1) # Kalman Filter gain
        Xest = Xp+K*Re # updated state estimate
        Pest = Pp-K*S*K # updated state covariance
        return Xest, Pest
```

Figure 6: Screenshot of the linear Kalman filter developed in Python. The model rates of change for the system are stored in the B matrix. This matrix is then multiplied by the timestep and the control commands, then summed to arrive at the next incremental state prediction.
Chapter 4 – System Model

4.1 VBS Model

In order to model the physical VBS system a free body diagram was developed. A simplified version of the free body diagram is shown in Fig. 1 where the buoyancy force due to the ballast chamber and the vehicle wet weight are summed to determine the vehicle reaction force with the sea floor. The VBS system as illustrated in Fig. 1, is modeled based on the assumptions that air is treated as an ideal gas, and the air within the ballast chamber is both isothermal and of homogeneous temperature. The resulting model captures the rate of change of the system buoyancy based on the control inputs and the characteristics of the system. The model is then applied to the Kalman filter algorithm to estimate the actual buoyant force on the system resulting from the VBS.

\[ F_R = F_W - F_B \] (1)

\[ F_W = g[m_A - m_V - (\Delta m_p * \Delta x)] \] (2)

\[ V = \frac{nRT}{P} = \frac{mRT}{M} \] (3)

\[ \frac{dF_B}{dt} = \rho g \left[ \frac{mRT}{(M)(P)} \right] \left[ q_f(u_f) - q_v(u_v) \right] \] (4)

\[ q_f(u_f) = \begin{cases} 
C_f & u_f = 1 \\
0 & u_f = 0 
\end{cases} \] (5)

\[ q_v(u_v) = \begin{cases} 
C_v & u_v = 1 \\
0 & u_v = 0 
\end{cases} \] (6)

The reaction force resulting from the free body diagram is given by (1) where \( F_R \) is the reaction force, \( F_W \) is the wet weight of the vehicle, and \( F_B \) is the buoyancy force. Equations (2) and (4) can then be substituted into (1) to model the vehicles reaction force with the seafloor. The vehicle wet weight is
found using (2) where $m_a$ is the mass of the vehicle in air, $m_v$ is the total mass of the water displaced by the vehicle, and $\Delta m_p$ is the rate of change in mass of the payload multiplied by a change in distance $\Delta x$.

The buoyancy force is obtained by finding the volume of the ballast chamber to determine the volume of water displaced. The volume of the ballast chamber can be determined from (3) where $R$ is the ideal gas constant, $T$ is the air temperature, $m$ is the mass of air, $M$ is the molar mass of air, and $P$ is the ambient pressure. The volume calculated from (3) can then be substituted into (4) to obtain the change in buoyancy force with respect to time. In (4) $\rho$ is sea water density, $g$ is acceleration due to gravity, $q_f$ is the mass flow rate of the fill valve, $q_v$ is the mass flow rate of the vent valve, $u_f$ is the fill control signal, and $u_v$ is the vent control signal. The fill rate $q_f$ in (5) is dependent upon $u_f$ the fill control signal, similarly the vent rate $q_v$ in (6) is dependent upon the vent control signal $u_v$. The fill and vent control signals $u_f$ and $u_v$ can only exist as on/off corresponding to a numeric value of 1 or 0. When $u_f$ is equal to 1 then $q_f$ is equal to the constant fill rate $C_f$ and when $u_v$ is equal to 1 then $q_v$ is equal to a constant vent rate $C_v$.

Both fill and vent rates $C_f$ and $C_v$ respectively are dependent upon the ambient pressure and the compressed air supply pressure. For this exercise the fill and vent rates were experimentally determined at a fixed depth.

### 4.2 Depth Effects

For this study it is assumed the air flow rates of the system are constant when the solenoid valves are in their open state. Field trials were only conducted at a constant water depth of 20 ft. to obtain the data presented here; therefore as the depth increases the effects on the VBS due to increased pressure are neglected. Due to the ability to compress the variable volume ballast chamber, the air flow rates into and out of the buoyancy chamber will vary with changes in depth.
4.3 Loss of Payload Weight

As the AUV performs a mission it will shed its payload at a known rate proportional to the distance traveled. This rate can be multiplied by the current measured distance traveled to determine the change in wet weight of the AUV. In (2) this change in payload weight is denoted by \( \Delta m_p \cdot \Delta x \) where \( \Delta m_p \) is the known rate of payload loss with respect to distance and \( \Delta x \) is the change in distance. The value \( \Delta x \) is directly obtained from a sensor on the AUV.
Chapter 5 – Control Approach

5.1 Filter/Controller Combination

The goal of this research is to implement two different estimation and control algorithms capable of addressing the challenges described above and to quantify the trade-offs between the two approaches using both computer simulation and field experiment results. The first control approach is a moving average filter coupled with a bang-bang controller and is shown in Fig. 7. The second approach is a model-based controller, comprised of a Kalman filter estimator coupled with a bang-bang controller and is shown in Fig. 8. For the remainder of the paper the model-based controller will also be referred to as the Kalman filter controller, and these two terms will be used interchangeably.

A bang-bang type controller is known alternatively as an on/off controller and is commonly used in thermostats to control temperature. This controller method matches well with the Boolean on/off nature of the actuator solenoid valves. However, the utility of bang-bang control is limited by the size of

![Figure 7: Moving average and bang-bang controller combination. Here the raw load cell sensor inputs are processed through a moving average before being passed to the controller.](image)

![Figure 8: Kalman filter and bang-bang controller combination. The Kalman filter uses both the raw load cell sensor measurements and the previous increment control commands to produce an estimate that is passed to the bang-bang controller.](image)
the operational deadband set by the developer. Without a deadband the bang-bang controller would switch rapidly between states when approaching the commanded setpoint. When used in a real-world environment the input values to the bang-bang controller are susceptible to noise. The presence of noise requires a large deadband for the controller to function without causing large unwanted overshoot and oscillations in response to changes in sensed buoyancy. On the AUV this noise is believed to originate from conductive coupling, as well as radiative coupling from electromagnetic motors.

5.1.1 Bang-Bang (On/Off) Control

A bang-bang controller is also commonly known as an on/off controller and the two names will be used interchangeably throughout. This control method is very simple and is suitable for systems which utilize actuators that operate in either the on or off state. An example of this type of system is in the control of the temperature of a room. A user will set the desired room temperature using a thermostat. The thermostat then regulates the temperature as close to the commanded set point as the control deadband will allow. The deadband can be defined as an acceptable bounds around the desired set point. For example the temperature in a room may be commanded to 25 degrees Celsius with a dead band of +/- 1 degree. As the room heats up to within one degree of the 25 degrees set point, the heater will shut off and no actuation will occur until the temperature exceeds the deadband around the set point value. Without commanding a set point with a deadband we will encounter an infinite switching problem where the system will approach equilibrium at the setpoint and constantly switch the actuators on and off in an attempt to perfectly achieve the set point.

5.1.2 System Limitations - Fast Flow Rates Lead to Overshoot

One major limitation of on/off control systems is the potential for runaway situations. As discussed in the previous section an on/off controller will utilize a boundary or deadband to regulate a quantity. However, in the presence of fast actuators relative to response times it is possible to exceed
the control deadband. In the event that this happens the controlled value will constantly overshoot the desired deadband bounds and result in instability and continual oscillations about the desired setpoint. Over actuation can be a major problem in systems with limited resources such as battery power or compressed air. Actuators can sometimes draw a large amount of power to be energized thus wasting energy if they are continually cycling.

5.2 Moving Average Filter

5.2.1 Moving Average Model

To address the noise in the load cell signal and decrease the size of the deadband for more precise control, a moving average filter is applied to the sensor input to estimate the true normal force. The moving average filter was applied over 20 samples. As can be seen in Fig. 9, the standard deviation

![Figure 9: Raw normal force measurements from the VBS load cell (blue) after passing through a hardware low-pass filter. It is assumed that the remainder of the noise is broadband. The true normal force is constant, but the sensor signal shows a standard deviation of ±2.19 lbs due to noise. To meet the system requirements a standard deviation of ±1 lb is necessary. The moving average (red) accomplishes this.](image-url)
of the averaged sample is decreased by 56.3% from the standard deviation of the raw data samples. However, it is expected that the standard deviation of the averaged sample would decrease by 77.6% from the raw data standard deviation based on the variance of the sample mean for a sample size of 20 units.

5.2.2 Averaging Time Delay

The use of a moving average does introduce a time delay. For the case where 20 samples are averaged with a sample frequency of 10 Hz, a significant delay of 0.95 seconds is introduced. The delay is defined as

\[
Delay = \left( \frac{N-1}{2} \right) \times \left( \frac{1}{f_s} \right)
\]

where \( N \) is the number of samples and \( f_s \) is the sample frequency. Within the timescale of operations on this vehicle, a delay of 0.95 seconds is considered large.

5.3 Kalman Filter

5.3.1 Filter Model

The Kalman filter controller was developed from a simplified version of the discrete Kalman filter. This is a well-known algorithm with many successful applications. The equations below represent a general deployment of the algorithm.

\[
X_{P[k+1]} = A_{[k]}X_{E[k]} + B_{[k]}u_{[k]}
\]

\[
P_{P[k+1]} = A_{[k]}P_{E[k]}A_{[k]}^T + Q_{[k]}
\]

\[
Z_{P[k+1]} = H_{[k]}X_{P[k+1]}
\]

\[
Re_{[k+1]} = Z_{[k+1]} - Z_{P[k+1]}
\]
In (8) the state prediction $X_{P[k+1]}$ is found from the state transition matrix $A_{[k]}$, the previous state estimate $X_{E[k]}$, the control-input model matrix $B_{[k]}$, and the control command matrix $u_{[k]}$. The predicted state covariance $P_{P[k+1]}$ is obtained from (9) where $P_{E[k]}$ is the previous covariance estimate and $Q_{[k]}$ is the process noise covariance matrix. Using (10) the measurement prediction $Z_{P[k+1]}$ is determined from the measurement matrix $H_{[k]}$ and the state prediction $X_{P[k+1]}$ from (8). The residual $Re_{[k+1]}$ is then calculated in (11) by subtracting the measurement prediction $Z_{P[k+1]}$ from the sensor measurement $Z_{[k+1]}$. In (12) the measurement prediction covariance $S_{[k+1]}$ is obtained from $H_{[k]}$, the predicted state covariance $P_{P[k+1]}$ in (9), and the residual $Re_{[k+1]}$ from (11). The Kalman gain $K_{[k+1]}$ is found using (13) where the predicted state covariance $P_{P[k+1]}$, the transpose of $H_{[k+1]}$, and the inverse of $S_{[k+1]}$ are multiplied. The state estimate $X_{E[k+1]}$ is calculated in (14) by substituting in (8), (13), and (11). Finally, the state covariance estimate $P_{E[k+1]}$ is obtained from by substituting (9), (13), and (12) into (15).

For controlling the VBS it was only necessary to estimate one value, the buoyant force. This resulted in a version of the filter where the matrices $A$ and $H$ are reduced to the scalar value 1. The matrices $Q$ and $R$ are also reduced to the scalar values $q$ and $r$ respectively. These reductions eliminated most of the matrix operations and allowed direct substitutions. By simplifying and combining the above equations, the algorithm itself was reduced to four equations. The simplified equations used are presented below.

$$S_{[k+1]} = H_{[k+1]}P_{P[k+1]}H'_{[k+1]} + Re_{[k+1]}$$  \hspace{1cm} (12)$$

$$K_{[k+1]} = P_{P[k+1]}H'_{[k+1]}S^{-1}_{[k+1]}$$  \hspace{1cm} (13)$$

$$X_{E[k+1]} = X_{P[k+1]} + K_{[k+1]}Re_{[k+1]}$$  \hspace{1cm} (14)$$

$$P_{E[k+1]} = P_{P[k+1]} - K_{[k+1]}S_{[k+1]}K'_{[k+1]}$$  \hspace{1cm} (15)$$

$$X_{P[k+1]} = X_{E[k]} + \left[ C_f (dt) \ C_v (dt) \right] [u_f[k]; u_v[k]]$$  \hspace{1cm} (16)$$
The predicted state is represented as $X_{P[k+1]}$ in (16), where $X_{E[k]}$ is the previous state estimate, $C_f$ is the air fill flow rate, $C_v$ is the air vent rate, $u_{f[k]}$ is the fill control input, and $u_{v[k]}$ is the vent control input. The Kalman gain $K_{[k+1]}$ is updated using (17) where $P_{E[k]}$ is the previous covariance estimate, $q$ is the process noise, and $r$ is the measurement noise. The state covariance is updated in (18). An updated state estimate is then achieved from (19) where the predicted state estimate $X_{P[k+1]}$ is summed with the product of the Kalman gain $K_{[k+1]}$ from (17) and the difference of the new sensor measurement $Z_{[k+1]}$ and the previous predicted state $X_{P[k+1]}$. This process is repeated for each new measurement and control input received, and then returns updated values for the state estimate and state estimate covariance.

On the prototype AUV system this cycle occurs at a frequency of 5 Hz, where at the beginning of an iteration the predicted covariance and predicted state are set equal to the previous covariance estimate and previous state estimate.
Chapter 6 – Results

A series of experiments were performed to validate and compare the two candidate control approaches, first in a computer simulation environment, then in the field with both controllers deployed on the AUV. The development and tuning of the controllers was done in a three step process. The first step is done in the simulation environment described in Section 3.2 where a closed-loop real-time simulation is performed. This is done to validate the functionality of the controller and to begin initial tuning. The second step is to evaluate the controller again in simulation using previously logged experimental data played back in real time. This is done in the open-loop format as the output control commands from the controllers will have no effect on the data being played back in real time. The use of this step is to observe the behavior of the controller when working with actual sensor data. Here, the effectiveness of the tuning done in step one can be evaluated and the controllers can be tuned further in preparation for field deployment. The final step is to deploy the controllers to the AUV in the field running in real-time control. The controllers can then be assessed and tuned based on feedback from the vehicle and with the presence of outside environmental disturbances.

6.1 Testing Scenarios

For the testing scenario used in each of the three steps described above, a step change in the reference signal (the goal of the VBS regulator) is provided as an input. To test these step changes in succession a square wave was generated using zero mean, additive, Gaussian white noise with a standard deviation of 2.19 lbs in order to simulate the random errors evident in the load cell measurements. This square wave is then processed through both filters to test their functionality and ability to produce a state estimate from a noisy signal. The stability and performance of the two algorithms is then evaluated for comparison.
6.1.1 Bang-Bang Control: Moving Average

In developing and tuning the moving average controller the goal was to achieve an output signal with as little noise and oscillations as possible. The tradeoffs were smoothing versus time delay of the response. The tuning was done by varying the sample size of the moving average. This was varied from between 2 and 100 samples. Taking an average of only 2 samples resulted in a time delay of only 0.095 seconds but provided very little smoothing of the signal. An average of 100 samples resulted in a response delay of 4.75 seconds. This provided a very smooth result but with too large of a time delay. If the fill valve were left open for the 4.75 second delay it would result in a buoyancy change of 20.19 lbs and this is unacceptable. An average size of 20 samples was found to be the best fit to maximize the tradeoff between signal smoothing and time delay. This sample size resulted in a delay of 0.95 second which corresponded to a maximum change in buoyancy of 4.04 lbs. While this is not ideal it provided the minimum amount of signal smoothing and the maximum amount of delay that was manageable.

6.1.2 Bang-Bang Control: Kalman Filter

The Kalman filter developed for controlling the VBS was first tuned in the simulator environment. Some of the challenges faced in tuning the moving average filter are shared in the Kalman filter tuning process. The same tradeoff of signal smoothing versus time delay must be considered. The tuning variables for the Kalman filter are the measurement noise and the process noise variables. The measurement noise $r$ was identified based on the measured variance of the load cell where $\sigma = 2.19$ lbs and $r = \sigma^2 = 4.80$ lbs$^2$ and this was held constant. With $r$ held constant, the process noise $q$ was varied from $0.0 \leq q \leq r$ until a desirable response was achieved. This is first done in closed-loop simulation, then open-loop simulation, then last on the vehicle during real-time control. A process noise value of $q = 0.001$ lbs$^2$ was used for this system and produced the best response with little to no time delay and an output signal that conformed to the actual input before the addition of sensor noise.
6.2 Simulation Results

During the simulation step two types of simulations were performed. The first simulation was closed loop operation using purely simulated input values from a random number generator. The second simulation replayed previously logged data and the controllers responded to this playback in the open loop configuration where their control outputs did not affect the apparent buoyancy. When replaying previously logged data it is only possible for the controllers to interact with them in the open loop configuration, their output control commands have no effect on existing data. While this was not ideal it did suffice to tune the Kalman filter via the process noise variable before deploying it to the actual vehicle for experimentation. Due to the limitations of the simulation processes noted above, it was necessary and useful to test the control software in the field.

6.2.1 Closed-Loop Simulation

The closed-loop real-time simulation results are presented in Fig. 10 and Fig. 11. It can be seen in Fig. 10 and Fig. 11 that the Kalman filter state estimate (green) conforms closely to the true values (black) in the closed-loop simulation. The simulated Kalman filter response is more desirable than the moving average filter response due to increased accuracy and the absence of the time delay. This is illustrated well in Fig. 11 where a close up view of one step input is provided. The Kalman filter output conforms closely to the known true values despite the presence of the simulated noise in red. The moving average in blue is also able to deal with the noise but exhibits some oscillation about the true values. The previously calculated time delay of 0.95 seconds for the sample size of 20 units is also visible. Based on this visual, the Kalman filter is clearly a more desirable bang-bang controller input than both the raw noisy values and the moving average values.
Figure 10: Moving average filter and Kalman filter tested in simulation. The measured values Z were generated using the true values X and zero mean, additive, Gaussian white noise with a standard deviation of 2.19 lbs. Note the delay in the moving average filter estimates (blue).

Figure 11: Close up view of the filter estimates for one step input, taken from (Fig. 9). The time delay on the moving average estimate is easily viewed on the rising and falling slopes of the step response.
6.2.2 Open-Loop Simulation

A second simulation test was performed using experimental data in order to observe the behavior of the moving average and Kalman filter estimates. Log files from previous experiments included both the raw load cell measurements and the solenoid valve commands. These log files were recorded using the lightweight communications and marshalling (LCM) library [9]. The LCM library includes a player to allow the log file to be broadcast to the network. This functionality was used to evaluate the two filters prior to implementing the controllers. The log file was played back in real time and both the moving average filter and Kalman filter were executed and produced estimates of the state of the load cell. This is shown in Fig. 12 and Fig. 13. While the noise on the load cell sensor had been previously characterized, it is noted that the logged data used in Figs. 12 and 13 appears to have a higher noise standard deviation. This higher presence of noise could be the result of different ocean conditions than during previous tests. Despite the added noise, both filters were tested with the logged data and found to be functional.

In Fig. 12 and Fig. 13 it can be observed that the Kalman filter estimator (cyan) produces a more consistent state prediction than the moving average response (blue) despite the large amount of sensor noise in the raw data (magenta). Both filtered responses were able to approach the goal (black) and stay within the +/- 7 lb deadband bounds shown in red and green. It is important to stress that these figures were generated in the open loop configuration. Both filters were only able to respond to the sensor data but were not able to react. Their control commands had no effect on the response. Therefore, it is expected that with closed loop control the response from both controllers would be superior to the response shown in Fig. 12 and Fig. 13.
Figure 12: Moving average filter and Kalman filter response tested in simulation using previously logged data from the AUV load cell.

Figure 13: Close up view of one step input response, taken from (Fig. 11). The Kalman filter estimate (cyan) produces a state estimate ahead of the moving average estimate (blue) and with smaller oscillations about the setpoint value.
6.3 Experimental Results

Upon obtaining satisfactory results from both filters in simulation, a series of field tests were conducted. For the tests, the vehicle was submerged to a depth of 20 ft. below sea level. The AUV was stationary, and the test site is located in an area protected from direct ocean waves. The AUV was programmed to perform step responses in buoyancy setpoint value represented by a 30 lb amplitude square wave over 315 seconds (Figs. 14-21). This experiment was then repeated with bang-bang controller deadband values of +/- 7 lbs, +/- 5 lbs, +/- 2.5 lbs, and +/- 1.25 lbs and for each filter running active VBS feedback control. The deadbands of +/- 5 lbs and +/- 1.25 lbs were noted as they appeared to capture the operating limits of the moving average filter and the Kalman filter respectively. In general, a smaller deadband value results in more precise control of the VBS. In the even numbered Figs. 14-20 the error (blue) is calculated between the buoyancy force estimate and the buoyancy force setpoint. The error is then plotted where the spikes in error (blue) occur as the setpoint is switched corresponding to the setpoint (black) in the bottom subplot of each figure.

6.3.1 Closed-Loop Field Experiments: Moving Average

It was determined that the smallest deadband the moving average filter would tolerate is +/- 5 lbs from the buoyancy setpoint. This is shown in Fig. 14 where there are some small noticeable oscillations in the state response, but the state estimate error still remains within the operational deadband. This behavior is not continuous for smaller deadband sizes as can be seen in Fig. 16 where the deadband is set to +/- 1.25 lbs. The moving average state estimate error continuously overshoots the setpoint resulting in unwanted oscillations. This type of behavior is not only destabilizing for the vehicle but also taxing on the solenoid valves as they are working to rapidly open and close. The rapid actuation of the fill and vent solenoids also uses an unnecessarily large amount of compressed air. The compressed air cylinders contain only enough air gas to complete a mission. Constant adjustment to the ballast chamber will result in wasted gas and premature expenditure of the compressed air cylinders.
Figure 14: State estimation error in response to a step input with a bang-bang controller deadband of +/- 5 lbs. This deadband appears to be the operational limit of the moving average filter estimator.

Figure 15: VBS response using the moving average controller and a deadband of +/- 5 lbs about the buoyancy setpoint goal.
Figure 16: State estimation error for a deadband of +/- 1.25 lbs. The moving average controller is unstable at this point with the presence of obvious oscillations and continual overshoot beyond the deadband bounds. This would result in rapid cycling of the solenoid valves and likely lead to their failure. Additionally, the system is unable to achieve a steady state buoyancy.

Figure 17: VBS response using the moving average and a deadband of +/- 1.25 lbs. Notice the oscillations about the deadband, the buoyancy never attains a controlled steady state.
6.3.2 Closed-Loop Field Experiments: Kalman Filter

With a deadband of +/- 5 lbs in Fig. 18, the Kalman filter state estimate error converged quickly to each deadband with little overshoot. This behavior is encouraging and suggests that the model-based controller can operate under smaller deadband conditions. The deadband was then reduced further in Fig. 20. With the deadband decreased to +/- 1.25 lbs, the state estimate error was still able to converge quickly to the bounds of the deadband and produce little to no oscillations and overshoot. The error between the buoyancy force estimate and buoyancy force setpoint quickly approaches zero in Fig. 20 suggesting that the actual vehicle buoyancy response is true to the commanded buoyancy. This result confirms that the model-based Kalman filter estimator controller can achieve more precise control of the VBS than the moving average controller, with a faster response. The Kalman filter controller was able to maintain stability beyond the operating limits of the moving average controller during actual field trials.

Figure 18: The Kalman filter estimator responds well to a step input with a deadband of +/- 5 lbs. Once the controlled response reaches a deadband it does not overshoot.
Figure 19: VBS response using the Kalman filter and a deadband of +/- 1.25 lbs. Note how the response quickly approaches the deadband and does not actuate past it.

Figure 20: Kalman filter state estimation response with a control deadband of +/- 1.25 lbs. The filter performs well at this range with no overshoot and quick conversion to a minimal amount of state error. At this range the Kalman filter estimator clearly outperforms the moving average filter.
6.4 Results Summary

A ranking of the performance of the controllers in the various exercises would be as follows: first the closed loop control with purely simulated values, second is the closed loop control deployed on the AUV, third is the open loop control using previously logged data. The reasoning for this outcome is because the closed loop pure simulation is ideal and represents the best case scenario that the controllers will experience. The closed-loop real-time control was done in the field in contact with the ocean environment so it is expected that there will be some externalities that prevent the controllers from performing as well as in pure simulation. The open loop scenario using previous data offered the controllers no feedback and no response to their apparent control commands, thus we wouldn’t expect this to be a good representation of actual field deployment.
The moving average controller and Kalman filter controller were both able to regulate the buoyancy, but at different deadband thresholds. It is desired that the buoyancy is regulated as close to the desired value as possible. Tables 1 and 2 below provide a qualitative analysis of the results from the field tests for the different deadband values.

Table 1: Moving average controller qualitative analysis

<table>
<thead>
<tr>
<th>Deadband Setting [lbs]</th>
<th>Overshoot</th>
<th>Oscillations</th>
<th>Analysis</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0</td>
<td>None</td>
<td>None</td>
<td>Acceptable</td>
</tr>
<tr>
<td>5.0</td>
<td>Yes</td>
<td>Yes</td>
<td>Acceptable</td>
</tr>
<tr>
<td>2.5</td>
<td>Yes, frequently</td>
<td>Yes</td>
<td>Unacceptable</td>
</tr>
<tr>
<td>1.25</td>
<td>Yes, frequently</td>
<td>Yes</td>
<td>Unacceptable</td>
</tr>
</tbody>
</table>

Table 2: Kalman filter controller qualitative analysis

<table>
<thead>
<tr>
<th>Deadband Setting [lbs]</th>
<th>Overshoot</th>
<th>Oscillations</th>
<th>Analysis</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0</td>
<td>None</td>
<td>None</td>
<td>Acceptable</td>
</tr>
<tr>
<td>5.0</td>
<td>None</td>
<td>None</td>
<td>Acceptable</td>
</tr>
<tr>
<td>2.5</td>
<td>None</td>
<td>None</td>
<td>Acceptable</td>
</tr>
<tr>
<td>1.25</td>
<td>None</td>
<td>None</td>
<td>Acceptable</td>
</tr>
</tbody>
</table>

One other metric used to compare the controllers with each other is the amount of controller effort. This can be quantified by the number of actuation cycles needed to adjust buoyancy. This is particular important for this system as there is only a finite amount of the buoyancy resource available (compressed air). The actuation commands using the moving average and a deadband of +/- 1.25 lbs is shown in Fig. 22. Similarly the actuation commands for the Kalman filter controller and a deadband of
+/- 1.25 pounds is shown in Fig. 23. It appears that the moving average controller actuates the VBS more frequently than the Kalman filter controller by a factor of approximately 6.

![VBS Actuation](image1)

**Figure 22:** Actuation commands for the moving average controller regulating the buoyancy using a deadband of +/-1.25 lbs.

![VBS Actuation](image2)

**Figure 23:** Kalman filter controller actuation given a control deadband of +/- 1.25 lbs.
Chapter 7 – Conclusions

7.1 Summary

Preliminary experimental results show that the moving average bang-bang controller approach is operational, but the performance and stability is limited. The model-based Kalman filter control algorithm delivers more precise regulation of the VBS, faster response to un-modeled disturbances and guaranteed stability. The Kalman filter controller allows smaller deadbands for the on/off actuation thus resulting in a more predictable VBS response and state estimates that are true to the actual vehicle state. The moving average controller was only able to achieve +/- 5 lbs regulation of the VBS while the Kalman filter controller was able to achieve +/- 1.25 lbs regulation.

While this approach is developed specifically for the challenges of the VBS of a bottom skimming AUV, the same control algorithm can be used to improve the performance of any system with noisy sensor inputs and fast on/off actuation.

7.2 Contribution

One contribution of this work is the application of a model-based filter to overcome sensor noise in an on/off controller system used to regulate the buoyancy of an AUV. Another contribution is the presentation of field trial results using the above model-based controller deployed on an AUV. A literature review did not turn up any similar work using a model-based controller developed to deal with sensor noise on an ocean robotic system.

7.3 Future Work

The progression of this study will move towards incorporating a strategy to cope with the effects of changes in depth on the VBS system. Other types of filters can also be explored such as the extended Kalman filter or GPC to name a couple. The application of these other filters can then be compared with...
the results found here. One other area of future investigation will be on the effects of quantization noise resulting from the analog to digital conversion going from the load cell sensor to the data acquisition unit on the micro-controller. A good first step in this direction will be to obtain a base value for the quantization noise on the system. Then this base value can be compared with existing external noise on the system to determine if it is a source of noise that needs to be addressed.
Bibliography


Appendix A: Reference Figures

Figure 24: The error in the state estimation using a deadband of +/- 7 lbs for the moving average controller. It is preferred that the error converge as close to zero as possible.

Figure 25: The actual system response is plotted here where the moving average estimate is shown in blue and the buoyancy goal is represented by the black dotted line.
Figure 26: Moving average filter applied with a moving average of +/- 2.5 lbs. The controller often overshoots the deadband in both the positive and negative directions.

Figure 27: The controlled buoyancy response using the moving average controller and a deadband of +/- 2.5 lbs. There is a bit of unwanted over oscillation occurring here.
Figure 28: Buoyancy control error using the Kalman filter controller and a deadband of +/- 7 lbs.

Figure 29: VBS response given a deadband of +/- 7 lbs using the Kalman filter controller
Figure 30: Control error for the Kalman filter controller using a deadband of +/- 2.5 lbs.

Figure 31: Response of the VBS using a control deadband of +/- 2.5 lbs with the Kalman filter controller.