Title

Kevin Nastasi[*], Dylan Thomas[†], Kristen Tetreault[‡],
Ian Elliott[§], Robert Schieble[§] and Jonathan Black [¶]

Virginia Polytechnic Institute and State University, Blacksburg, Virginia, 24061, USA

Nomenclature

A	State Transition Matrix
B	Control Input Model
u	Control Vector
P	State Covariance Matrix
Q	Process Noise Covariance
H	Measurement Map Matrix
Δt	Time Step, seconds
t	Time, seconds
X	State
$I(N)$	Identity Matrix of size NxN

I. Introduction

A. Previous Work

1. Background

Setting up the scenario

II. Model

A. ODTK

B. GVE

C. HCW

HCW Proxops

1. Adaptive, objective mpc

fancy words

2. Measurement Simulation and Filtering

A brief study of different possible sensor systems was conducted. Ultimately a combination of camera tracking system and a laser range-finder are used to find direction and range, respectively. A phased-array

[*]Ph.D. Candidate, Aerospace and Ocean Engineering, and more
[†]Ph.D. Candidate, Aerospace and Ocean Engineering, and more
[‡]Ph.D. Candidate, Biomedical Engineering and Mechanics, and more
[§]Undergraduate, Aerospace and Ocean Engineering, Address, and AIAA Member Grade.
[¶]Associate Professor, Aerospace and Ocean Engineering, Address, and AIAA Member Grade.

radar system was considered, and may be explored in later research; however, the size of current systems is prohibitive.

The laser-rangefinder the MATLAB simulation uses is a FLIR MLR1001. The data sheet cites a range from near 0 cm to over 100 m, and a resolution under 0.2 m1. For the purposes of the MATLAB simulation, the measured range is found by taking the true range, adding a random Gaussian-distributed error with a standard deviation of 0.0255, and rounding to the closest 0.2 m increment. The standard deviation of 0.0255 provides a 95

A camera tracking system would be used to find the direction of the target. This simulation is very simplistic; the camera can track the target in all conditions, and does not simulate target acquisition. The tracking camera is assumed to have an order of magnitude more error than the star tracker; 60 arc seconds 1-sigma bore sight accuracy, and 400 arc seconds 1-sigma roll axis accuracy. These values are based off of the Blue Canyon Nano Star Tracker2.

A Kalman filter is used to reduce the error and estimate the position. The equations for a Kalman filter can be found in section 3.3.1 of Crassidis and Junkins. The state is the relative position and relative velocity.

$$X = \begin{bmatrix} x \\ y \\ z \\ \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} \tag{1}$$

The state transition matrix is a simple propagation using Euler's method. As the simulation itself propagates with Euler's method, a more robust system model causes unnecessary drift.

$$A = \begin{bmatrix} 1 & 0 & 0 & \Delta t & 0 & 0 \\ 0 & 1 & 0 & 0 & \Delta t & 0 \\ 0 & 0 & 1 & 0 & 0 & \Delta t \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{2}$$

The acceleration is accounted for as a control-input u, containing both the thrust of the spacecraft and the calculated HCW accelerations. The control-input model uses Eulers method to update the state, so position is not affected by acceleration.

$$B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ \Delta t & 0 & 0 \\ 0 & \Delta t & 0 \\ 0 & 0 & \Delta t \end{bmatrix} \tag{3}$$

$$u = \begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} \tag{4}$$

A Montecarlo simulation was used to find information on the performance of the Kalman filter. The simulation output the root mean squared error of the estimated position, giving a single number to tune the filter with. The state covariance matrix P, process covariance matrix Q, and measurement covariance matrix R, are identity matrices multiplied by a single value. The filter was tuned by testing six possibilities for each value, ranging from 10-1 to 10-6, by order of magnitude. This resulted in the following matrices, with a root mean squared error of 44.7854 over 2000 data points.

$$P = 10^{-4} * I(6) \tag{5}$$

$$Q = 10^{-3} * I(3) \quad (6)$$

$$R = 10^{-1} * I(3) \quad (7)$$

III. Results

A. odtk

more results

B. rend/proxops

more results

C. Measurement Simulation and Filtering

The results for the position error in a Montecarlo simulation of 1000 runs can be seen in Fig: 1, 2, 3. The Kalman filter clearly reduces error across the board; however, there is an initial discrepancy in the z-position due to assuming an incorrect initial z velocity (Fig. 3). The maximum 95% confidence bound for any estimate error is about 0.08 meters. As the simulation has a safety buffer of 0.5 meters, this is well within the acceptable range. Even without using a filter at all, the maximum measurement error is 0.1 meters, leaving a factor of safety of 5. The filter reduces the error to about 0.02 meters for x, about 0.001 meters for y, and about 0.001 meters for z as well.

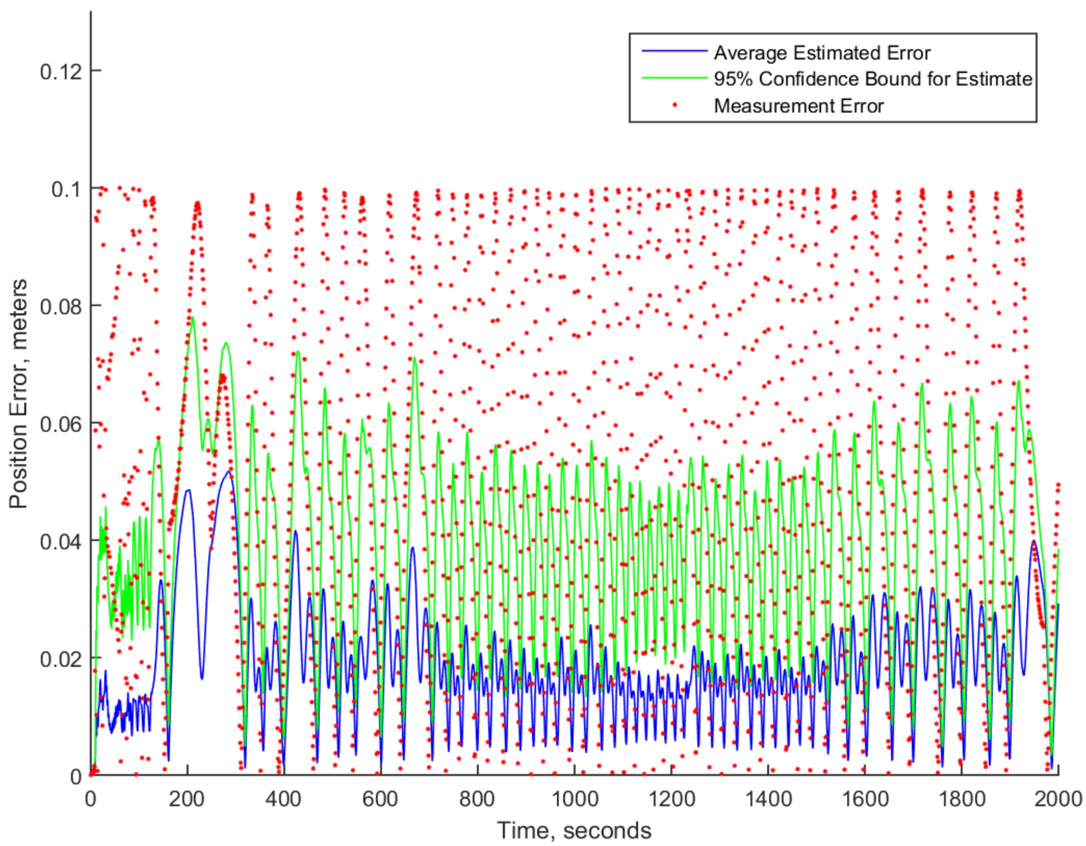

Figure 1. x Position Error

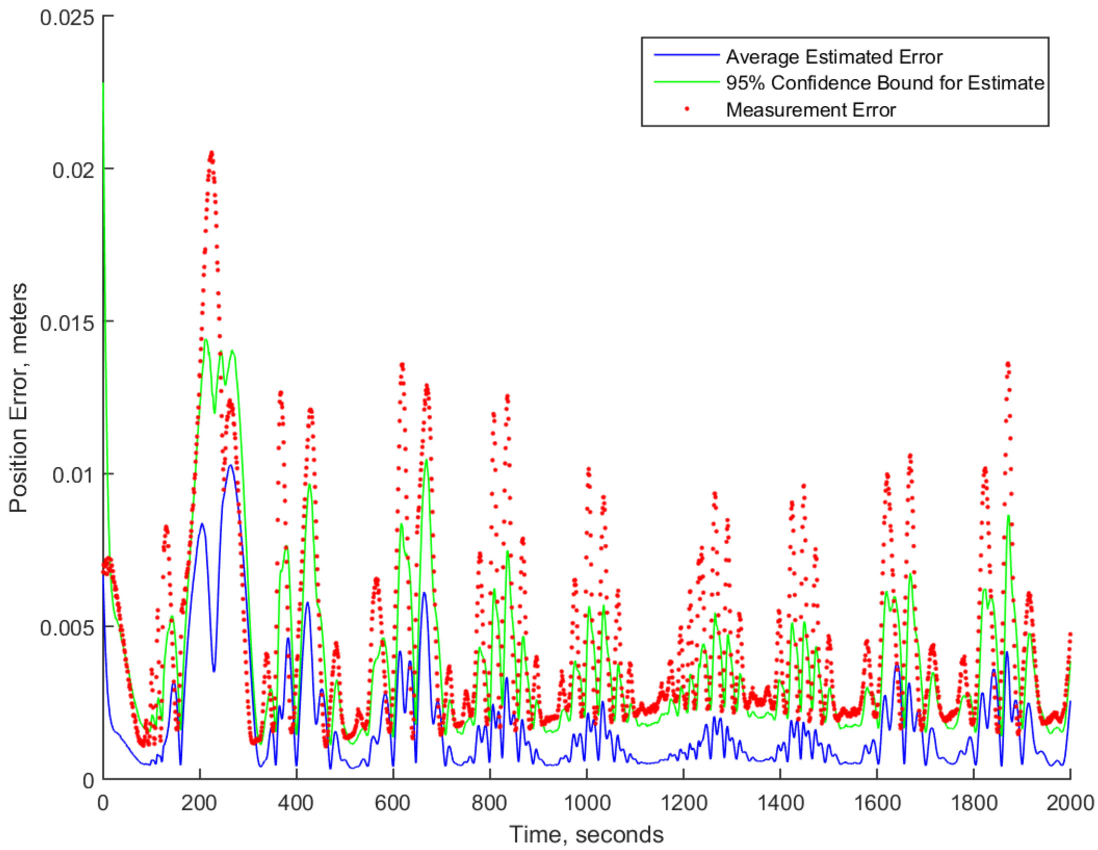

Figure 2. y Position Error

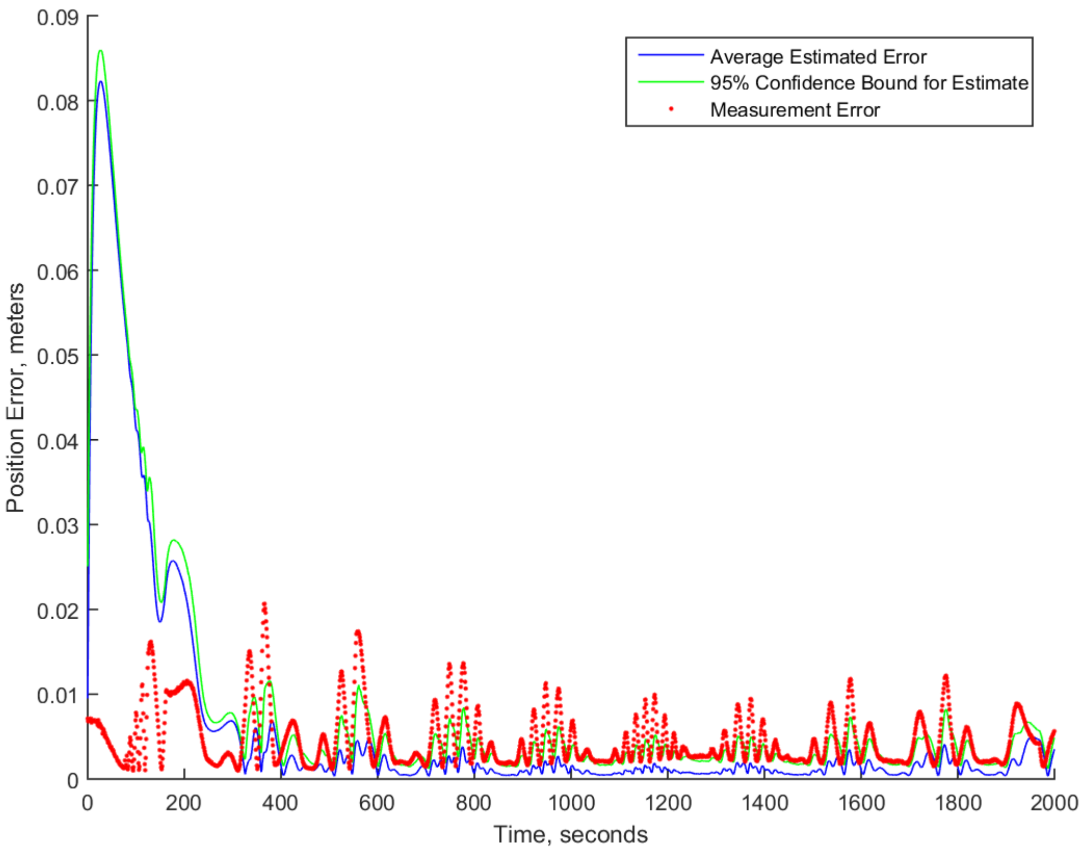

Figure 3. z Position Error

IV. Conclusion

After much typing, the paper can now conclude. Four rocks were next to the channel. This caused a few standing waves during the rip that one could ride on the way in or jump on the way out.

type the conclusion

A. future work

how to list this...

1. better eoms

adfgdsg

2. integration of rel. od into control

asdgadg

3. attitude d&c

asdgadg

4. cad model for more realistic 6u design

asdgadg

Appendix

An appendix, if needed, should appear before the acknowledgments. Use the 'starred' version of the `\section` commands to avoid section numbering.

Acknowledgments

A place to recognize others.

References

[1]Crassidis, John L., and Junkins, John L., Optimal Estimation of Dynamic Systems, $2^{nd} ed., CRC Press, New York, 2012, Chap. 3.3.1.$