

# Performance Study of the Rate-Based Pendulation Control System

VERSION 1.0

Dr. H. Schaub

*Virginia Polytechnic Institute and State University*

March 15, 2006

# Table of Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Sensor Error Models</b>	<b>4</b>
2.1	POS/MV Sensor Error Mode . . . . .	4
2.1.1	Numerical Model . . . . .	4
2.1.2	POS/MV Error Model Cases . . . . .	8
2.2	Accelerometer and Rate Gyro Sensor Errors . . . . .	13
2.2.1	Sensor Bias . . . . .	13
2.2.2	Sensor Value Scaling . . . . .	14
2.2.3	Gaussian Noise Model . . . . .	14
<b>3</b>	<b>Performance Study</b>	<b>16</b>
3.1	Simulation Setup . . . . .	16
3.2	Results of 3D Numerical Simulation . . . . .	17
3.2.1	General Description . . . . .	17
3.2.1.1	Cases 1 and 2 . . . . .	19
3.2.1.2	Cases 3 and 4 . . . . .	19
3.2.1.3	Cases 5 and 6 . . . . .	19
3.2.1.4	Comparison of Cases . . . . .	20
3.3	Influence of Sensor Bias . . . . .	21
3.4	2D-Model Sensitivity Study . . . . .	21
<b>4</b>	<b>Conclusion</b>	<b>25</b>

# Chapter 1

## Introduction

The U.S. Navy is expanding the capabilities of the boom cranes on the T-ACS vessel to be able to operate at higher sea states. These ships have several large boom cranes which can off-load cargo from container vessels to smaller lighter vessels as shown in Figure 1.1. A single crane on the Flickertail State was equipped with a Pendulation Control System developed by Sandia National Laboratories.<sup>1</sup> By installing ship motion and cargo pendulation sensors, as well as a closed-loop crane servo system, the PCS has demonstrated that it is capable of avoiding the onset of swing to due ship motion or operator commanded cargo motion, and reject any existing swing due to lift-off transients or wind disturbances.



**Figure 1.1:** Cargo Ship Off-Loading Containers Onto Lighter Vessels.

The current control solution of the PCS required the inertial position and attitude measurements of the ship. A commercial sensor called the POS/MV is used to obtain these states. This sensor uses a dual-GPS system to determine heading information, as well as three-dimensional positions. Note, the differen-

tial GPS mode is not used here. The reason for this is that the vessel must operate at arbitrary locations on Earth where such differential correction signals would typically not be available. An inertial measurement unit (IMU) is also included to provide roll and pitch angles. The accelerometer and rate gyro information of the IMU are then used within an extended Kalman filter to provide smooth estimates of all 6 degrees of freedom on the ship (latitude, longitude, height, heading, pitch and roll). This system works quite well for its intended purpose of providing accurate ship motion and location information. To use this sensor along with the PCS, additional sensor processing had to be done to use the ship state information in the control strategy. In particular, the POS/MV provides very smooth ship position estimates, a requirement of the PCS strategy. However, the un-aided GPS states are only accurate to about  $\pm 10$  meters. The POS/MV does filter these position errors using the IMU sensor information. However, some residual position errors will always exist. Depending on the severity of the GPS position errors, these long term drifts (in the order of dozens of seconds to minutes) can be less than a meter, and up to levels of multiple meters. The PCS performs additional filtering to remove this drift. However, this process is never perfect or without consequences. As such, the payload position with the current PCS strategy will wander by a small amount due to the POS/MV sensor errors. This erroneous motion is typically rather small and easily controlled by the operator driving the crane. Further, the crane servo system required crane

joint speed commands. To obtain these, the position-based PCS solution is numerically differentiated. Differentiating sensed states is a difficult task. Unaided, even small noise can become hugely amplified through the numerical differentiation process. Thus this differentiation is performed simultaneously with a low-pass filtering process. This has been shown to work reasonably well with the Flickertail installed sensor. However, adding this filter also introduces some performance penalty through the introduction of additional phase lag.

The accompanying report entitled “Rate-Based Pendulation Control System Study” presents two new control strategies for the Navy Pendulation Control System (PCS). Instead of measuring the inertial ship position and attitudes, it is assumed that the ship sensor will measure the ship acceleration and rotation rate (rate gyro), along with the inertial ship pitch and roll angles. Such sensors systems are much cheaper than the more sophisticated POS/MV sensor system. The accelerometer and rate gyro information is measured unfiltered and processed inside the new PCS control strategy. Carefully filtering and integrating the accelerometer information, the ship velocity and positions can be estimated. These steps are outlined in Reference 2. The filtered accelerometer-based ship positions are equivalent to the filtered GPS-based position. Note that neither filtered position measurements are the absolute ship motion. Rather, only the ship motion which will cause significant cargo pendulation is retained. The advantage of using accelerometers and rate gyros to estimate surge, sway, heave and heading is that the sensor technology is much cheaper to implement. The down side is that the numerical integration from accelerometer states to position states is complicated by the presence of sensor noise and bias. If a constant sensor bias is present in the accelerometer measurement, then the double integration will result in a quadratically growing error. The filtering and integration strategy presented in Reference 2 is able to provide a stable integration method.

Given the newly estimated ship motion posi-

tion, one proposed PCS solution is to use the existing position-based control strategy. This will involve less changes to the PCS software to implement. Another advantage is that the deck-tracking mode of the PCS will continue to function as is. The ship motion sensor processing software component will have to be replaced with new routines. This report studies the performance of this new PCS strategy and compared it to the performance of the position-based PCS strategy. Note that both of these methods end up numerically differentiating position-states to obtain crane velocity servo commands.

The second new PCS control strategy also measures the motion through accelerometer and rate gyro sensors, and filters-integrates the sensors to obtain velocities and position estimates. However, here the control algorithm of the PCS is replaced with a new algorithm which directly computes commanded crane servo rates. This avoids the numerical differentiation process of the position-based strategy. Sensing a certain ship motion rate, a corresponding crane rate is directly computed.<sup>2</sup> Un-aided, any pure velocity-based control strategy will be unstable in the presence of unmodeled system behaviors. This new control strategy is thus stabilized by using the position-based control solution.

This report will study and compare the performance of all three control strategies. An advanced crane simulation software package called CraneSim is used. Sensor communication, discretization, and noise behaviors are modeled. The crane hydraulic drive system is modeled through using a transfer function with a lag behavior which reflects that of the actual hardware. Rate and acceleration limits are also imposed. ***Note that this work is not intended to provide actual payload motion predictions.*** Rather, the differential performance of the crane control strategies is of interest. The three different simulations only differ in how they sense the ship motion, and how they compute a control strategy. The same crane drive system and swing sensor models will be used through-out. To compare to the position-based PCS strategy, realistic POS/MV sensor error

models are used. This are discussed in detail in the following chapter. The accelerometer and rate gyro sensor error type of concern is the bias. Small measured noise levels will be automatically smoothed out through the integration process. Thus, the accelerometer and rate sensors errors are modeled as a constant bias in this study.

## Chapter 2

# Sensor Error Models

### 2.1 POS/MV Sensor Error Mode

This section outlines how the POS/MV sensor errors are to be modeled. The three control strategies being considered will only differ significantly in their performance when sensor corruption are introduced. To provide a fair comparison, incorporating realistic POS/MV is important. The following issues are of concern. 1) The degree of multi-path severity will cause some slow acting drift (periods often ten times slower than the pendulation frequency) in the horizontal translation and vertical heave measurement, as well as some amount of increased drift in the heading measurement. This drift manifests itself as a non-Gaussian error component. 2) The measured ship states are clipped to values with a specific resolution. The POS/MV sensor model will need to duplicate this resolution. 3) A certain amount of Gaussian noise is included if needed. The position measurements of the POS/MV are very smooth and don't show any gaussian noise characteristics. Only the resolution clipping is present. However, the orientation measurements do exhibit some small amount of gaussian noise behavior. This will cause noise-amplification issues when the position-based control strategies numerically differentiate crane states to compute velocity servo commands.

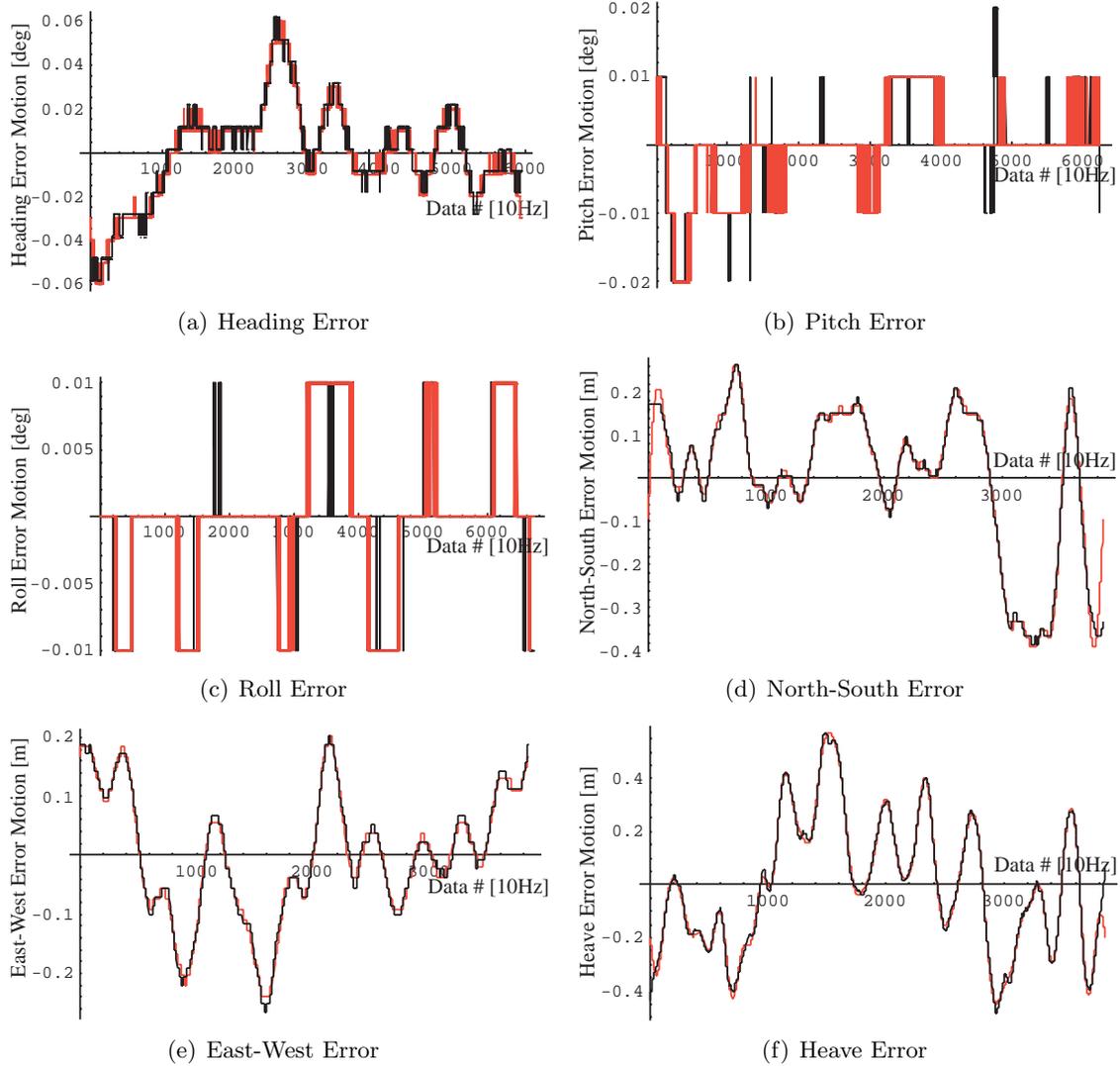
Dynamic POS/MV tests conducted at Sandia National Labs found the sensor to be very accurate and no perceived sensor lag was found in the rates (even within the high accuracy of the benchmark units). Thus no sensor lag is incor-

porated into the POS/MV sensor model. The static drifts discussed earlier will introduce by themselves some small amount of rate errors. However, since these drifts are relatively slow acting, the modeled rate errors will be small, as required.

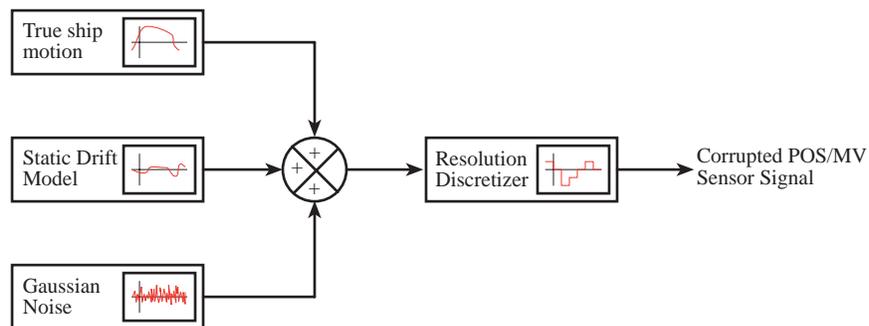
#### 2.1.1 Numerical Model

Figure 3.3 shows some sample static drift behavior of the actual POS/MV unit. The results shown are for a low multi-path environment. The amount and frequency content of the translational and heading drift depends on how severe the multi-path environment is. To model different multi-path situation, three cases are considered. Case 1 is a best case scenario with a very low multi-path environment. Both the static drift amplitudes and dominant drift frequencies will be small. Case 2 is a medium severity case of sensor errors. The frequency content of the error drifts is still low, but the amplitudes are somewhat larger. Case 3 is a worst case scenario modeling static drifts that were encountered with a high multi-path environment. Note that the roll and pitch errors are not affected at all by the degree of GPS multi-path severity. Thus, the roll and attitude error models are identical for all three cases.

The general concept of the POS/MV sensor model is to mathematically capture the static sensor drifts, as well as any Gaussian noise components, and to superimpose this error on top of the true ship motion. The summed signal is then clipped to the appropriate resolution. This concept is illustrated in Figure 2.2. The thus produced POS/MV sensor signal will cause



**Figure 2.1:** Static POSMV error comparison of the numerical POSMV error model (red) and the measured POSMV static errors (black). A low multi-path scenario is shown.



**Figure 2.2:** Illustration of the POS/MV sensor Error Model

the PCS an equivalent amount of differentiation and sensor error issues as the true POS/MV sensor signal will.

To most challenging aspect of modeling the POS/MV sensor errors is capturing a realistic static drift behavior. The main cause of the "random-walk" like drift of the sensor signal is due to the multi-path environment. There are routines available to generate bounded random walk behavior, which are similar to the routines which generate true gaussian noise. However, these random walk routines don't provide a high enough fidelity match with the measured POS/MV sensor drifts. To mathematically model the translational sensor signal drifts, a model of the POS/MV Kalman filter with its multi-path rejection logic would need to be used. Because the POS/MV is a proprietary commercial unit, obtaining such information would not be trivial. Further, to program such a model would be a very complex, involved and expensive development.

A simpler and sufficiently effective modeling method is chosen instead. The spectral decomposition or Fast-Fourier-Transform (FFT) of the sensor drift provides a means to "fingerprint" the drift signal. To create a time dependent function which models the desired drift behavior, we only use the dominant FFT components to reconstruct the signal. The mathematical details of this idea are outlined next. Let  $h(t)$  be the original sensor signal which we are attempting to model. The signal is sampled  $N$  times at discrete time intervals  $t_k$  with a sampling time  $\Delta$ .

$$t_k = k\Delta \quad \text{for } k = 0, 1, \dots, N-1 \quad (2.1)$$

Let  $h_k$  be the sampled signal  $h(t)$  at a particular time  $t_k$ .

$$h_k = h(t_k) \quad (2.2)$$

Performing the FFT of a time based signal  $h(t)$  we obtain an equivalent frequency based signal  $H(f)$ . Doing a discrete FFT on  $N$  samples  $h_k$  we obtain  $N$  complex variables  $H_k$ . Each  $H_k$  parameter provides the amplitude and phase response of the signal  $h(t)$  at a particular frequency  $f_k$ . If  $h(t)$  has a dominant sinusoidal

motion at a frequency  $f_k$ , then the magnitude of the complex variable  $H_k$  would be much larger than the remaining complex FFT coefficients. The frequency  $f_n$  is defined as

$$f_n = \frac{n}{N\Delta} = \frac{n}{T} \quad (2.3)$$

for  $n = -\frac{N}{2}, \dots, 0, \dots, \frac{N}{2}$

where  $T$  is the total time that the signal  $h(t)$  was sampled. Due to Nyquist's sampling theorem, the highest frequency content that can be detected with a sampling period of  $\Delta$  is  $f_{\max} = 1/(2\Delta)$ . The FFT produces the signal response due to both positive and negative frequencies. The first  $N/2$  samples of the FFT output  $H_k$  correspond to the positive frequencies. The second half of  $H_k$  outputs correspond to the negative frequencies. If the signal  $h(t)$  is real, then  $H(-f) = H(f)^*$ , with  $()^*$  being the complex conjugate operator. The FFT bode magnitude plot will thus be symmetric about the  $N/2$  sampling point. To consider real signals  $h(t)$ , we can focus on the real frequencies  $f_n$  with  $n = 0, 1, \dots, N/2$ . The angular velocity  $\omega_n$  is defined as

$$\omega_n = 2\pi f_n = 2\pi \frac{n}{N\Delta} = 2\pi \frac{n}{T} \quad (2.4)$$

At this point we are ready to look at the mathematical details of the discrete FFT transformation itself. Let us define the discrete FFT of a set of signal samples  $h_k$  through

$$H_n = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} h_k e^{2\pi i k n / N}$$

$$= \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} h_k e^{i\omega_n t_k} \quad (2.5)$$

This definition of the discrete FFT is standard except for the scaling parameter  $\sqrt{N}$ . In some definitions this is set to 1. However, using the scaling in Eq. (2.5), the inverse discrete FFT has the same functional form except for the sign

of the exponential.

$$h_k = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} H_n e^{-2\pi i k n / N}$$

$$= \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} H_n e^{-i\omega_n t_k} \quad (2.6)$$

Given the  $H_n$  coefficients, to reconstruct the original time based signal  $h(t)$  one would use

$$h(t) = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} H_n e^{-i\omega_n t} \quad (2.7)$$

However, it is not necessary to use all  $N$  of the complex  $H_n$  coefficients to reproduce the signal  $h(t)$  to a reasonable accuracy. Assume that  $h(t)$  was sampled at 10 Hz, then  $\Delta$  would be 0.1 seconds. Using Nyquist's sampling theorem, the highest frequency which can be detected is 5 Hz or  $\Delta = 0.2$  seconds. However, the signal often only has a frequency content which is a small subset of the total frequency content that is measured. To reproduce an approximate signal  $\hat{h}(t)$  we can ignore the higher frequencies and only use the dominant  $H_k$  components in the inverse FFT. For the case of modeling the POS/MV sensor drift, we assume that the drift will occur about a zero static offset. Any non-zero offset, due to either the sensor itself or its mounting, can be incorporated later with the calibration procedure. If the drift is occurring about a nominal zero mean point, then  $H_0$  (corresponds to a zero frequency) will be zero. Thus we start the counter in the approximate inverse FFT with 1 instead of zero. Further, assume that only the first  $M$  FFT coefficients are significant. The integer  $M$  must be less than  $N/2$  to only use the  $H_n$  corresponding to positive frequencies. The approximate signal  $\hat{h}(t)$  is then expressed as the truncated series

$$\hat{h}(t) \approx \frac{1}{\alpha} \sum_{n=1}^M \text{Re} (H_n e^{-i\omega_n t}) \quad (2.8)$$

The scaling factor is changed here to  $\alpha = 2\sqrt{N}$  since only half of the frequency spectrum is considered (thus only half of the  $H_n$  coefficients).

Whereas Eq. (2.7) will return a real function  $h(t)$  if the original function is real, Eq. (2.8) will not return a real function. Due to the truncation, some imaginary components will remain. This is why the  $\text{Re}()$  operator has been added to the  $\hat{h}(t)$  evaluation. Let the complex coefficient  $H_n$  be expressed in real and imaginary components as

$$H_n = H_{n_r} + iH_{n_i} \quad (2.9)$$

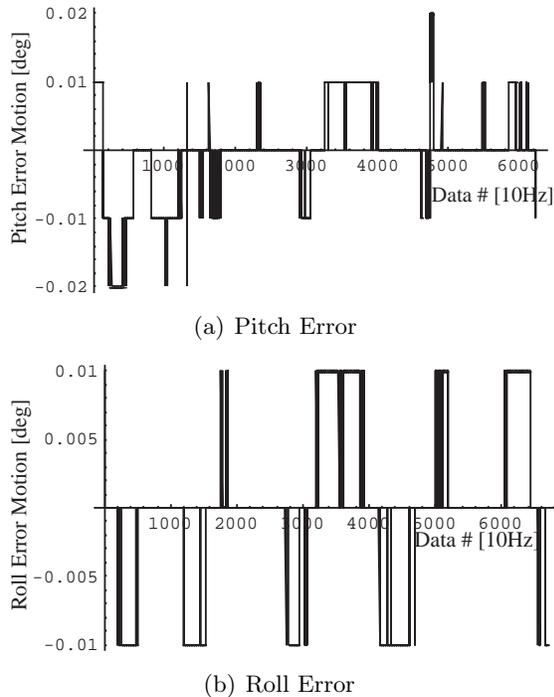
Using  $e^{i\theta} = \cos\theta + i\sin\theta$ , Eq. (2.8) is then rewritten as

$$\hat{h}(t) \approx \frac{1}{\alpha} \sum_{n=1}^M \left( H_{n_r} \cos(\omega_n t) - H_{n_i} \sin(\omega_n t) \right) \quad (2.10)$$

The advantage of this form is that no complex algebra must be programmed. All variables in Eq. (2.10) are real variables. To model the general drift behavior of the POS/MV signal, typically 15–25  $H_n$  coefficients are used.

Note that a myriad of other curve fitting methods could have been used. For example, one consideration was to model the sensor drift by using an interpolation scheme such as cubic splines or Hilbert polynomials. However, the drawback of this method is that there is no smooth sway to loop the drift data once the end of the measurements are reached. Using the FFT coefficients we are left with a naturally cyclic drift function that smoothly will transition at the data end to the beginning of the data. Further, using the data interpolation method would require reading in large amounts of data. Using the FFT coefficients provided a faster and more efficient implementation.

The resolution of the POS/MV serial communication is specified by the TSS specifications. What remains is the modelling of the gaussian random noise component. As was observed earlier, the position states of the POS/MV were very smooth. No Gaussian noise was observed on the signal, only the communication discretization. The attitude measurements did show some small amount of gaussian noise. Figures 2.1(b) and 2.1(c) show the attitude static



**Figure 2.3:** Illustration of typical pitch and roll static attitude errors

errors as black lines. Note that the errors are overall very small and of the order of the communication resolution level.

Figure 2.3 shows only the POS/MV sensor error for the pitch and roll angles. This illustration is simpler to view than Figure 3.3 because no approximated  $\hat{h}$  signal are super imposed. If the attitude errors were solely due to gaussian noise, then the signal errors would be composed of a series of sharp spikes. Instead, a small drift-like behavior is observed. This is why the attitude errors are also modeled using the  $\hat{h}(t)$  function mentioned earlier. However, note the black box-like areas in Figure 2.3. For example, consider the black box near sample point 2000 in the pitch plot. The true value of the pitch error is probably near -0.005 degrees. This is precisely between the 0 and -0.01 degrees that are obtained by the pitch communication resolution. A small amount of gaussian noise would cause the POS/MV pitch signal to either jump to 0 or -0.01 degrees. A region where the gaussian noise has no effect is around data point 3500–4000. Here the pitch error is essen-

tially steady at 0.01 degrees. The actual pitch angle must be close to 0.01 degrees in this zone. Adding a small amount of gaussian noise is not sufficient to tip the POS/MV pitch output to either 0 or 0.02 degrees. Looking at the modeled pitch and roll error behavior in Figures 2.1(b) and 2.1(c), we see that the combination of modeling the overall drift through the FFT coefficients and then adding some small gaussian noise before clipping the angle to the communication resolution does produce a signal whose qualities are equivalent to the original POS/MV signal. In particular, differentiating the modeled attitude signal would be equally difficult to differentiating the true POS/MV sensor signal.

To generate gaussian noise we use the following numerical procedure. Assume that  $x$  is a random number with a value between 0 and 1. This number by itself will not have a gaussian distribution. Let  $y$  be the desired gaussian random number which is to have a standard deviation of  $\sigma$ . This value is obtained by computing  $2N$   $x_i$  values:

$$y = \sigma \left( \sum_{i=1}^{2N} x_i - N \right) \quad (2.11)$$

The random  $y$  value distribution will resemble the standard bell-shaped gaussian distribution out to  $N\sigma$ . The parameter  $N$  is given a value of 6 in the current simulations.

### 2.1.2 POS/MV Error Model Cases

As mentioned earlier, depending on the local GPS conditions, the level of random-walk behavior observed with the POS/MV can differ substantially. The POS/MV sensor errors are modeled for three different situations:

- *Case 1:* Best GPS conditions encountered. This usually meant very low multi-path problems with the GPS receivers.
- *Case 2:* Medium case. This meant moderate GPS multi-path issues.
- *Case 3:* Worst case. This illustrates some of the worst POS/MV random walk issues

encountered. Unfortunately, the environment on the ship tended to favor case 3.

The parameters required to model the POS/MV signal behavior of each of the sensed 6 degrees of freedom are stored to an ASCII data file `POSMVx.dat`. The letter `x` corresponds to the sensor error model case number. Case 1 is the best situation with the least amount of drift, while case 3 is the worst situation with a large amount of drift and higher drift frequencies. The order in which the signal behavior parameters are stored is:

1. North-South motion (to be replaced with latitude information)
2. East-West motion (to be replaced with longitude information)
3. Heave motion
4. Roll angle
5. Pitch angle
6. Heading angle

For each DOF, the first line contains these parameters:

$$M \quad \alpha \quad T \quad \sigma \quad \delta$$

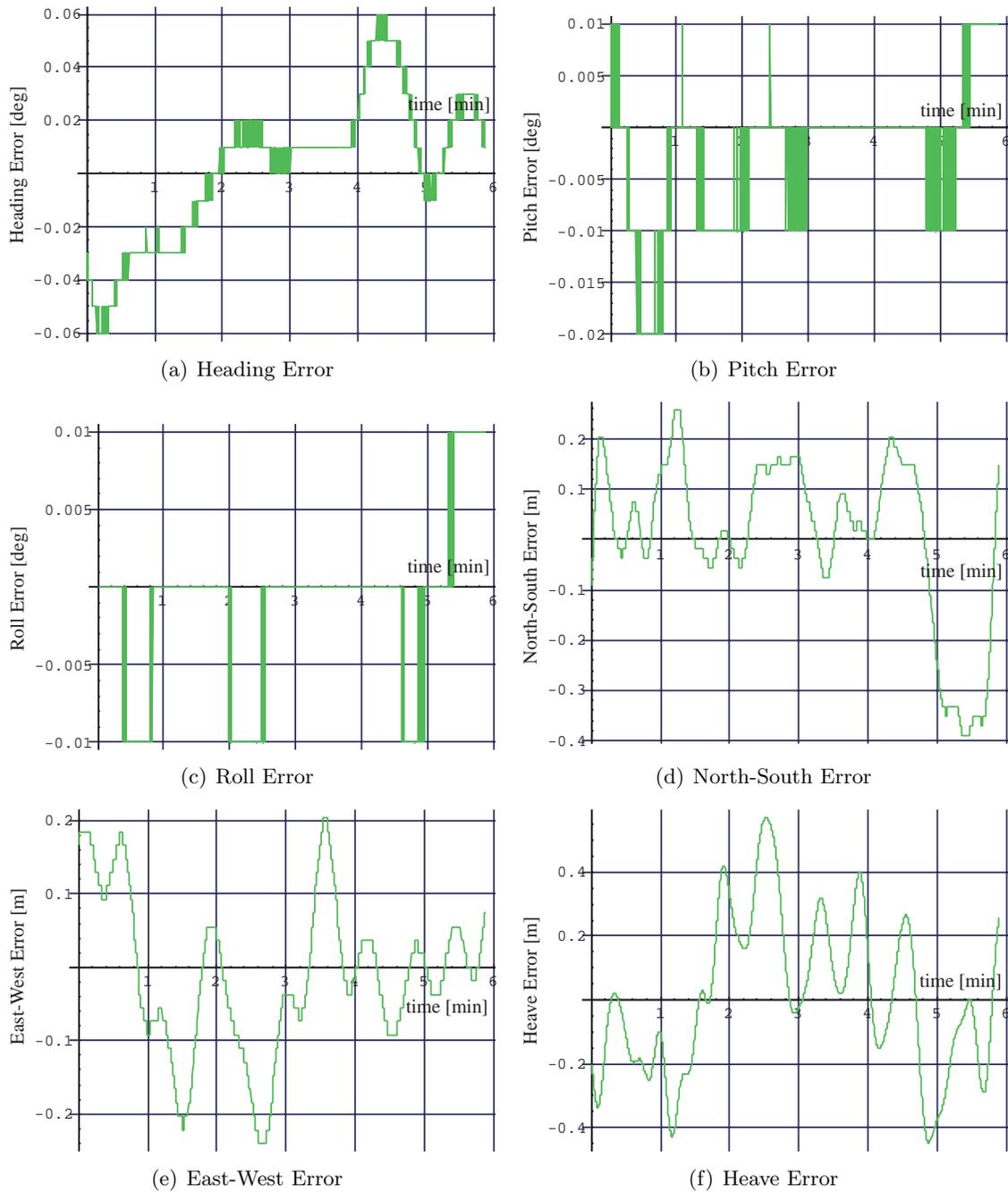
where  $M$  is the number of  $H_n$  coefficients used,  $\alpha$  is the inverse FFT scaling factor,  $T$  is the total signal duration of the original signal (in seconds, used in Eq. (2.4) to compute  $\omega_n$ ),  $\sigma$  is the standard deviation of the gaussian noise component (in radians) and  $\delta$  is the communication resolution (in radians). The following  $M$  lines then contain the parameters

$$H_{n_r} \quad H_{n_i}$$

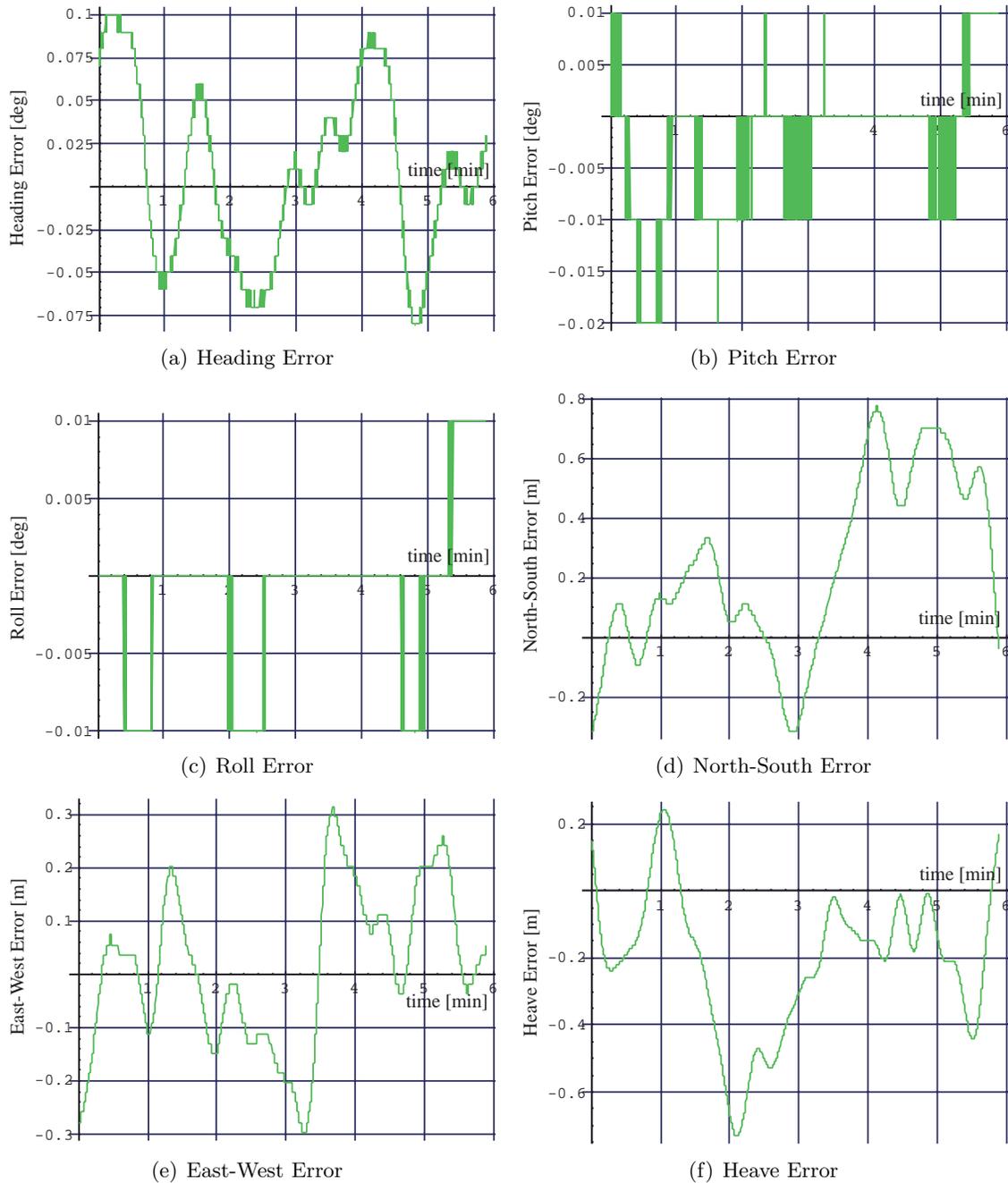
This process is repeated for each POS/MV degree of freedom.

Figures 2.4 through 2.6 show the static POS/MV errors that are modeled for each case. In case 1, the position drifts are very small with values less than 0.5 meters. In case 2, the position drifts increase some to values of about 1 meter. The worst case, case 3, has position

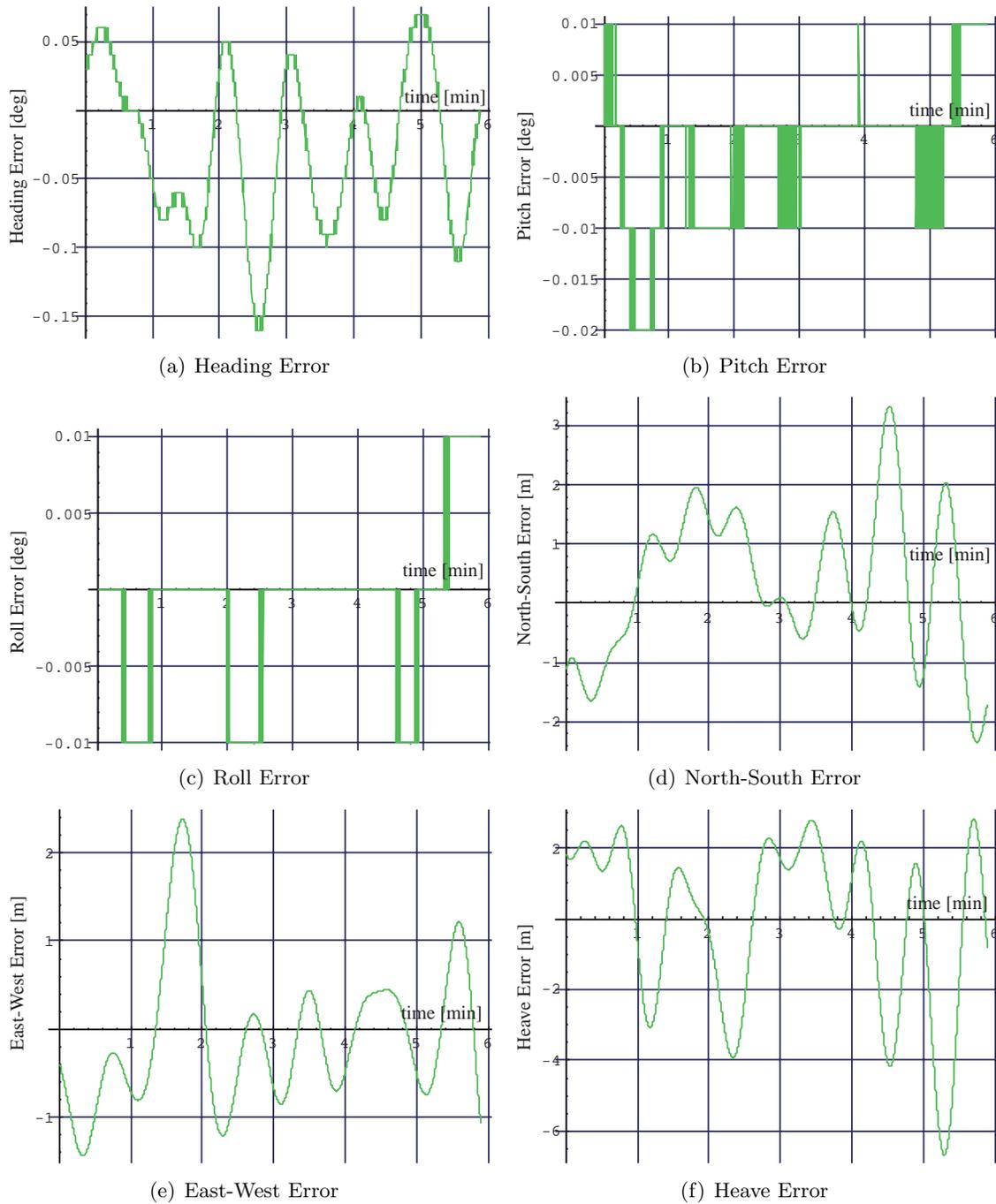
drifts up to several meters. Further, recall that since the roll and pitch angle errors were essentially invariant to the effects of the GPS multi-path induced errors, the same roll and pitch error models are used for all three cases.



**Figure 2.4:** Case 1 (best) static POSMV static error drift model. (low amplitudes and low frequency content found during a low multi-path environment)



**Figure 2.5:** Case 2 (medium) static POSMV static error drift model. (medium amplitudes and low frequency content found when during a medium multi-path environment)



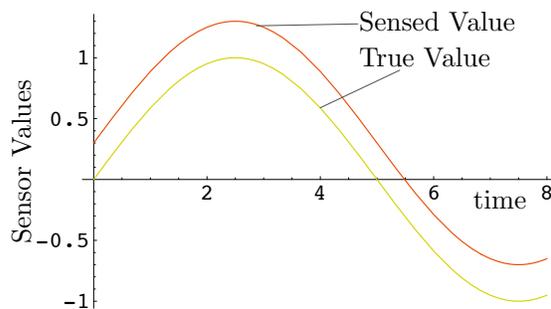
**Figure 2.6:** Case 3 (worst) static POSMV static error drift model. (larger amplitudes and higher frequency content found when during a high multi-path environment)

## 2.2 Accelerometer and Rate Gyro Sensor Errors

Fortunately, the accelerometer and rate gyro sensor errors required for this crane control performance study are far simpler to model than the POS/MV sensor errors. The POS/MV errors have to be modeled to a high fidelity because the drifts will directly cause payload wander, and the sensor noise can be amplified through the numerical differentiation process. The accelerometer and rate gyro sensor corruption types considered include gaussian noise, static bias, and an erroneous sensor measurement scaling value. Thus, no random walk behavior is modeled for the modeled IMU sensor measurements.

Note that all control strategies assume that the ship pitch and roll angles are measured directly. With an IMU (accelerometer and rate gyro combination) this can easily be done. The position-based control strategy requires a nominal heading motion to be estimated. This is obtained by taking the rate gyro sensor information (i.e. the body angular velocity vector  $\omega$ ) and computing first the corresponding heading rate through a kinematic transformation. Note that this calculation only requires the absolute ship roll and pitch angles, which are assumed to be measured directly. The periodic ship heading motion for a vessel anchored at sea is a relatively minor motion compared ship roll and pitch motion. Most of the motion is typically the ship roll motion excited through the ocean waves. As with the ship translational motion, the true inertial heading information is not required. Only the heading motion with a frequency content close to that of the cargo pendulation frequency is required.

The following sensor corruption types are thus applied to the translational accelerometer states  $\ddot{x}$ ,  $\ddot{y}$  and  $\ddot{z}$ , as well as to the rate gyro states  $\omega_1$ ,  $\omega_2$  and  $\omega_3$ . These corrupted measurements are then used to estimate the current ship velocity and position vectors, as well as the



**Figure 2.7:** Illustration of Sensor Bias

ship heading angle.

### 2.2.1 Sensor Bias

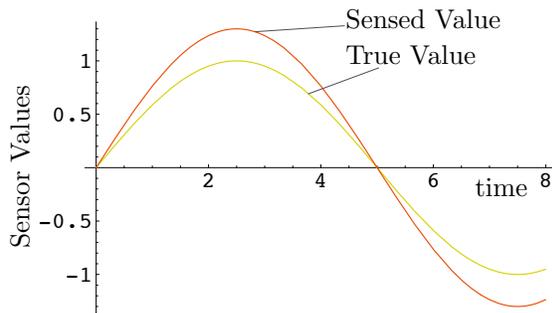
Every sensor contains a bias error. Instead of measuring the true acceleration, for example, the sensed value is offset by a bias factor  $b$ . This is typically due to minor miss-calibration issues. A sample bias error is illustrated in Figure 2.7. This bias can vary slowly over time as the electronics warm up. However, over short time periods, it is reasonable to model the bias as a constant value. It takes less than 10 minutes for the crane to pick up cargo and deposit it at a new location. The bias is assumed to not vary significantly over this short time period. Thus, the 3D crane simulation uses a constant bias value when modeling the accelerometer and rate gyro sensor information.

The Litton LN200 is a very common IMU used to measure inertial motion.<sup>3</sup> The current POSMV sensor contains an IMU of this class. The LN200 specifications list an accelerometer bias which can vary between 300–3000  $\mu\text{g}$ . The simulation uses a value of 1500  $\mu\text{g}$ . The LN200 rate gyro bias specification lists a possible range of 1–10 deg/h. The simulation uses the upper bound of 10 deg/h. The performance results shown are representative of using an IMU of this grade and quality.

When using accelerometer information to estimate inertial motion, the sensor bias is the primary source for integration instabilities. For any position of velocity based control strategies, the effect of the sensor bias on the cargo stabi-

lization performance is of primary concern.

For actual IMU sensors the measurement bias levels are not constant. Rather, these biases change rather slowly as the system is running for while and the electronics temperature increases. Note that the employed estimation and filtering techniques do not require precise bias levels which are then compensated for in software. Instead, frequency-based filtering techniques are employed which provide much more robust results. As a result, the shown performance is not expected to vary by any significant amount if the bias is replaced with a slowly varying bias. The only requirement is that the bias does not vary significantly over the time period that it takes to pick up and land cargo.



**Figure 2.8:** Illustration of Sensor Scaling

### 2.2.2 Sensor Value Scaling

Besides having a sensor bias, the next relevant sensor error is the erroneous scaling value. This concept is illustrated in Figure 2.8. The actual sensor value is typically obtained through a scaling process using sensor-internal signals. If this step is not perfectly calibrated, then the perceived motion will always be some percent too large or too small. Note the conceptual difference to having a bias  $b$ . The bias adds a fixed error to the sensor signal, where the scaling error add a percentage error signal. The larger the motion, the more significant the effect of the scaling error will be.

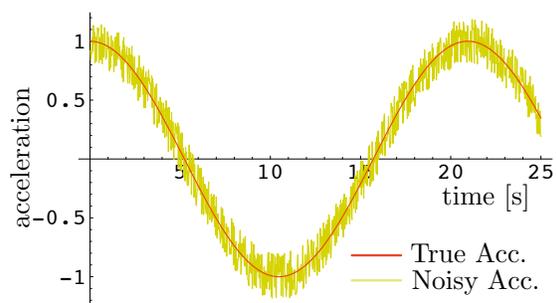
The Litton LN200 IMU list a scaling error range of 0.03–0.5% for the accelerometer information, and a range of 0.01–0.05% for the rate

gyro measurements.<sup>3</sup> The 3D simulation uses very conservative values of 0.5% scaling error for both the accelerometer and rate gyro measurements.

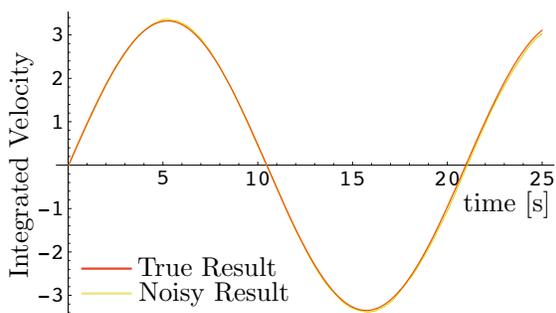
Note that this scaling error is expected to have the most severe impact on the vertical heave motion estimation. The accelerometers measured the total inertial acceleration. This includes the gravitational acceleration. A nominal gravity acceleration of 9.81 m/s is assumed. However, this value can vary slightly on different location on the Earth. The ship motion estimation algorithm determines the gravity acceleration vector using the ship yaw and pitch angles, and then removes this component from the measured ship acceleration. Thus, if the ship is not translating relative to the Earth, the effective measured acceleration will be zero, as expected. However, if the sensor has scaling errors, then the measured gravitational acceleration will be slightly larger or smaller, depending on the scaling error value. As a result, it is impossible to perfectly cancel the gravitational bias of the acceleration. This introduces a relatively large net bias in the ship vertical direction. The filters will try to remove, or at least minimize, the effect of this bias. In essence, the sensor perceives the ship to be slowing rising or sinking. The filtering and estimation strategy used for the updated PCS strategy will stabilize the effect of this scaling induced bias.

### 2.2.3 Gaussian Noise Model

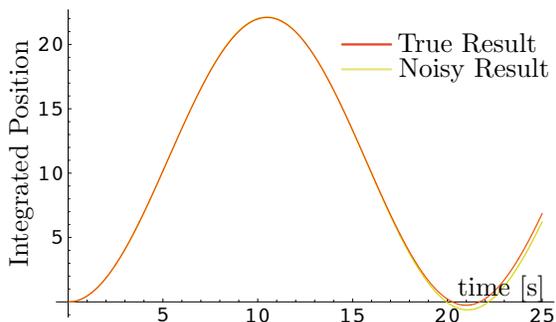
The sensor gaussian noise levels have a negligible effect on the crane performance. The accelerometer and rate gyro information is filtered and integrated to yield nominal velocity and position estimates of the ship motion. This process is illustrated through a numerical example in Figure 2.9. Here a very noisy sinusoidal signal (with  $\omega = 0.3$  rad/s) of a one-dimensional accelerator is used as a sample sensor input. After simply integrating the sensed signal without performing any filtering, an indirect measurement is obtained of the velocity state. After another integration step the indirect measurement of the position state is ob-



(a) Acceleration Measurement



(b) Integrated Velocity States



(c) Integrated Position States

**Figure 2.9:** Illustration of Gaussian Noise Smoothing through Numerical Integration

tained. The digital sampling rate used is 40 Hz (same as PCS digital sampling rate). Note that the noise level of the computed velocity state is already drastically improved over the signal to noise level of the accelerometer information. The position states are even smoother. As expected, after some time the position measurements will begin to deviate slowly from the true positions. Without further aid, integrating unfiltered accelerometer data is never stable. The actual integration process used in the

updated PCS strategy is described in detail in Reference 2. By carefully applying a series of bandpass-filters, the integration process is stabilized and the ship motion with a frequency content close to the natural frequency of the cargo pendulation is retained.

Thus, *no gaussian noise levels were considered in the 3D crane simulations*. The code is equipped to include such noise. However, for IMU sensors such as the LN200 used in the POSMV sensor, these noise levels are very small and don't introduce any significant corruptions.

# Chapter 3

## Performance Study

### 3.1 Simulation Setup

The full three-dimensional simulation of the cargo pendulation, crane motion and ship motion is used to study the performance of the new PCS strategy. Note that this simulation also simulates communication lags and discretization issues, as well as ship and swing sensor corruptions. A simple hydraulic drive system model is employed which models the actual TG3637 crane’s hydraulic performance before any servo enhancements were made. The drive system contains a linear transfer function to model the frequency dependent response, as well as saturation behaviors as the crane speed and acceleration limits are reached. However, note that the resulting crane performance tests did not hit the existing cranes velocity and acceleration limits. Doing so would only complicate this analysis even further without providing any meaningful insight into the new PCS performance.

The goal of this 3D simulation is to compare the cargo stabilization performance of the various new PCS strategies to that of the existing position-based PCS implementation. Thus, great effort is made to provide illustrative and meaningful *relative performance* comparisons, not just overwhelm the reader with lots of plots with little relevance. Please note that this report does not make any claims on the actual final performance of the crane. This will depend greatly on the final sensor choice, and very importantly, on how well the crane servo drive system operates. Instead, the position-based PCS performance is computed for a representative

set of crane joint states, and a particular ship motion. The operator joystick commands are all set to zero during this study. Thus, the cargo is to ideally maintain a fixed position while the ship is translating and rotating. The resulting performance is used as a yard-stick to measure the performance of the various new PCS strategies against.

Including a drive system in study is important to have a feel of how much these improvements will really mean in the final hardware application. Using a perfect drive system, it would be easy to generate results which could claim great-sounding percentage improvements in tracking. However, the drive system lags will already result in some residual cargo motion. The cargo motion errors due to the sensors issues should be kept in perspective relative to this drive system induced cargo error motion.

**Table 3.1:** Relevant Parameters of the High Fidelity Crane Simulation

Mode	Value
Slew	45.0°
Luff	52.2°
Hoist	35 m
Roll Amp.	1.5°
Roll Period	11 sec

The essential simulation parameters are listed in Table 3.1. This crane configuration attempts to keep the cargo just off the port side of ship at about deck level, a common situation when picking up cargo or landing it on another ves-

sel. The ship motion is prescribed to be a pure sinusoidal motion about the ship center with a period of 11 seconds. The actual ship period could vary between 10-15 seconds. The faster the period, the more difficult it will be for the crane drive system to keep up. Thus, a conservative value of 11 seconds was chosen. Also, note that the ship natural period can vary with the ship loading. The code has a ship roll period estimation algorithm implemented which will estimate the actual, current ship roll period. Thus the PCS will know the roll period to within less than a second. Even though the ship motion is a simple sinusoidal motion, the IMU sensor is located away from the center of rotation (same location as the current POSMV sensor on the Flickertail State vessel). This results in the ship sensor having to estimate both the resulting translation and rotation of the ship at this sensor location.

**Table 3.2:** Litton LN200 IMU Sensor Error Values

Error	Value
Accelerometer Bias	1500 $\mu\text{g}$
Gyro Bias	10 $^\circ$ /h
Sensor Scaling	+0.5%
Noise	Unmodeled

The IMU sensor corruption levels are listed in Table 3.2. These are very conservative IMU sensor corruption levels that might be experienced with the Litton LN200 IMU, a common IMU used for inertial navigation applications.

**Table 3.3:** New PCS Strategies Considered

Case Num.	Ship Motion Sensing	Control Method	Acc. Filter
1	POSMV (1)	Pos.	N/A
2	POSMV (3)	Pos.	N/A
3	IMU	Pos.	No
4	IMU	Pos.	Yes
5	IMU	Vel.	No
6	IMU	Vel.	Yes

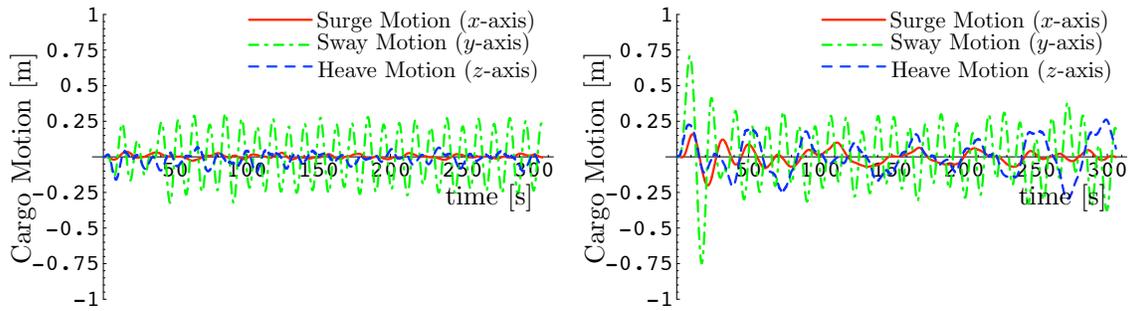
There are six different PCS strategies that are considered here as shown in Table 3.3. Cases 1 and 2 are the benchmark cases where the existing position-based PCS strategy is used. Both cases introduce expected POSMV random walk behaviors. Case 1 uses best case POSMV error values, while case 2 uses the worst case values modeled. The next four cases all assume the ship motion is measured using an IMU. Cases 3 and 4 directly estimate the ship positions relative to the slowly wandering inertia-prime frame (see Reference 2), and then use the existing position-based pendulation control strategy. Cases 3 and 4 differ in the number of applied filters to the ship accelerometer information. Each integration process to obtain velocities from acceleration, or to obtain positions from velocities, is done simultaneously with a band-pass filter. Further, the code can optionally apply a band-pass filter to the accelerometer measurements themselves. This will further help reduce the effect of the sensor biases, but will make the system system also more sensitive to non-sinusoidal ship motion. The final two cases measure the ship motion the same way as cases 3 and 4. However, here a new velocity-based pendulation control strategy is employed.<sup>2</sup>

Note that the same swing damping gains, and crane servo position-loop (to stabilize the crane servo velocity commands) are used in all these simulations. Also, all band-pass filter settings are equivalent to those of the existing PCS installation. This is done to assure a fair and reasonable relative performance comparison.

## 3.2 Results of 3D Numerical Simulation

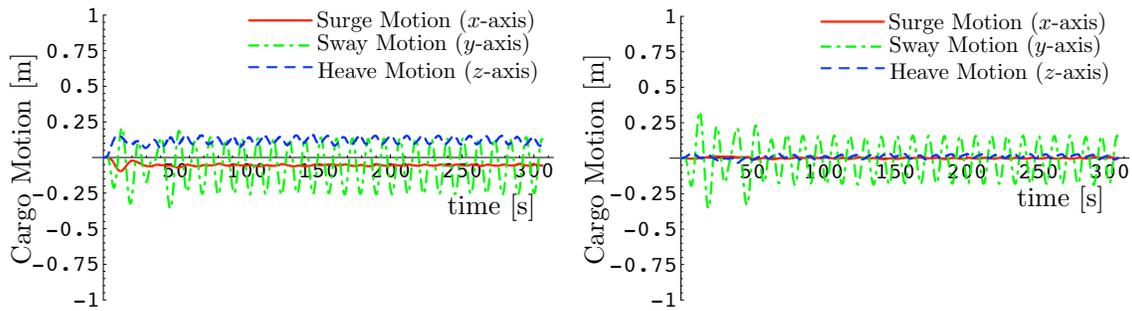
### 3.2.1 General Description

The time histories of the residual payload motion are shown in Figure 3.1. Each plot shows the motion of the cargo relative to the initial inertial position (time of control activation). The goal of the PCS strategy is not to place the cargo at a very specific location, but rather to



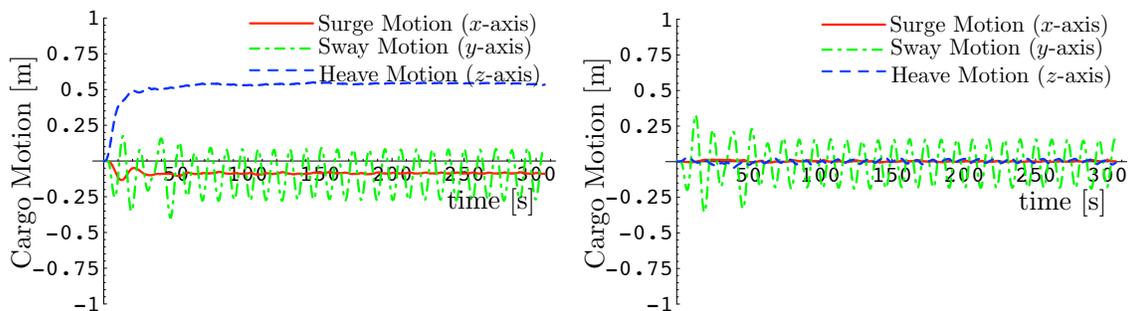
(a) POSMV Sensor Case 1

(b) POSMV Sensor Case 3



(c) IMU Sensor with Position-Based Control Strategy and No Bandpass Filter on the Accelerometer Measurements

(d) IMU Sensor with Position-Based Control Strategy and a Bandpass Filter on the Accelerometer Measurements



(e) IMU Sensor with Velocity-Based Control Strategy and No Bandpass Filter on the Accelerometer Measurements

(f) IMU Sensor with Velocity-Based Control Strategy and a Bandpass Filter on the Accelerometer Measurements

**Figure 3.1:** Illustration of Cargo Station-Keeping Performance of the Three Control Strategies Considered.

maintain the current location relative to the crane ship frame, whatever it might be. Thus, if the ship position has a constant 10 meter error, it will have no influence on the PCS performance. The operator in the loop will guide the payload to the desired location.

Further, please note that in the current implementation the ship motion filters are assumed to have converged to their steady-state values. Every filter will experience some start-up transient behavior before settling down to the desired performance. In the current TG3637 implementation, the code verifies that the PCS ship filter routine has received ship motion data for a certain amount of time before allowing the crane control to be engaged. This is not modeled in these simulations. The ship motion filters are started up at the same time as the crane control. Thus we will see some initial transient crane performance which is not representative of the actual control. The simulations are run for 5 minutes (300 seconds). The initial 100 seconds are ignored when computing steady-state performance errors.

### 3.2.1.1 Cases 1 and 2

Cases 1 and 2 illustrate the PCS performance with the existing POSMV ship motion sensor and the position-based control strategy. Case 1 assumes that the GPS-induced random walk behavior is very low, while Case 2 assumes the worst case random walk behavior. Note that the largest amount of payload motion is the ship  $y$ -axis direction. This is expected because the ship is performing a roll motion. Because the crane drive system will have some lag associated with it, the crane is not compensating perfectly for the ship motion and some small payload motion of 0.2–0.25 meters results. The payload vertical  $z$  motion is mostly due to the GPS-induced random walk behavior of the POSMV sensor. As the level of these position errors is increased in case 3, the erroneous cargo heave motion becomes 4 times worse. Note that the POSMV position errors also have an influence on the cargo surge and sway motion motion. However, this effect is relatively minor, cer-

tainly when compared to the drive system induced error motion.

### 3.2.1.2 Cases 3 and 4

Cases 3 and 4 assume that the ship motion is measured using an IMU, while the existing position-based control strategy is employed. Case 3 only applies the band-pass filter to the two integration steps, but not to the original accelerometer information. The bias and scaling issues of the sensed IMU ship motion do result in stable, bounded cargo payload motion. However, a noticeable cargo position bias is apparent with the LN200 level of IMU corruption. If the additional accelerometer band-pass filter is applied, then this bias is removed.

Even though case 3 shows a small bias, in particular in the cargo vertical motion, note that this bias is relatively steady. The payload motion about this steady-state offset is rather small. In fact, the nominal vertical payload motion about the steady-state values of case 3 is slightly less than the best POSMV ship motion measurement case 1. Recall that the operator actually commands the final cargo position. Thus, if the sensor bias is causing a steady-state bias in the ship position measurement, then the operator will easily be able to compensate by commanded the desired payload position. As the sensor bias or scaling errors change *slowly* with time, this will cause some additional payload motion. However, these sensor changes are expected to take place over the time frame of hours and days, not a few minutes.

### 3.2.1.3 Cases 5 and 6

Cases 5 and 6 show the crane performance if the ship motion is measured using an IMU, while the control strategy is replaced with the velocity-based kinematic solution. Note that the gains are used for the swing damping control and velocity-steering law feedback loop. For case 5 the sensor bias and scaling error causes a slightly larger payload position bias in the sway and surge motion. The payload heave motion, however, is noticeably larger. The bias present in the vertical acceleration mea-

**Table 3.4:** Steady-State RMS Error Comparison of the Cargo Station-Keeping Performance of the Three Control Strategies Considered

Ship Sensor Type	Control Method	Acc. Filter	RMS Surge Error [m]	RMS Sway Error [m]	RMS Heave Error [m]
POSMV (1)	Position-Based PCS	N/A	0.009	0.154	0.025
POSMV (3)	Position-Based PCS	N/A	0.027	0.162	0.100
IMU	Position-Based PCS	No	0.003	0.124	0.019
IMU	Position-Based PCS	Yes	0.003	0.107	0.009
IMU	Velocity-Based PCS	No	0.003	0.113	0.004
IMU	Velocity-Based PCS	Yes	0.003	0.107	0.009

surement, along with the scaling error causing the gravity components only partially removed, causes the velocity-based control to erroneously sense the ship as continuously raising or sinking. The position-loop of the velocity steering law of the PCS servo command generation stabilizes this error to finite values. Note, again, that the same steering law gains are used here as with the position loop. These gains could be increased to reduce the apparent payload offset. However, as noted for cases 3 and 4, a constant offset in the perceived ship motion does not cause any payload swing. The operator can trivially compensate for a less than 1 meter offset. If the additional accelerometer band-pass filter is employed, than these constant sensor biases are perfectly removed.

### 3.2.1.4 Comparison of Cases

Table 3.4 shows a direct comparison of the residual steady-state payload motion magnitudes along each ship frame axis. The errors are computed by removing the first 100 seconds of the data to discard the effects of the ship motion filter transients response. Next, the constant bias of the remaining payload motion is removed before computing the RMS value. This illustrates the effective payload error motion that would appear to the crane operator.

The pure roll motion of the ship causes the dominant payload error motion to be along the  $y$  axis (due to drive system lags in implementing the crane joint rate commands) and along the vertical  $z$  axis direction (due to the erroneous

ship sensor information). Note that the IMU based performance results are all slightly better than the POSMV based results. However, the improvements in the sway direction are not that significant compared to the drive system induced errors. Even a perfect IMU sensor would not improve these numbers very much. The drive system is really the more limiting factor. What this does show is that the POSMV sensor (a \$100,000 sensing system) could be replaced with a LN200 type IMU (about a \$20,000 sensor) without penalizing the  $(x, y)$  cargo control performance. In fact, the performance might even improve slightly.

The most noticeable difference between the POSMV and IMU-based cases is in the vertical cargo heave motion. The current POSMV solutions experience occasional random walk behavior. With the IMU solution, for all cases 3–6, the vertical cargo control is noticeably smoother and smaller. The addition of the band-pass filter on the measured accelerometer data improves the results even further.

A final disclaimer. The presented performance comparison is done using a pure sinusoidal motion with the ship filter frequency set nearly equal to the actual ship motion. While adding the additional band-pass filter consistently improved the results here, the actual ship motion is typically a combination of the resonant ship motion, as well as much longer periods sinusoidal motion. This will cause the ship motion amplitudes to grow and shrink. The additional filter will cause the performance to

be more sensitive to miss-matched ship motion and filter frequencies. The self-tuning ship motion filter helps in making sure that the filter is tuned correctly. However, more studies are required where realistic ship motion is used to see if this additional filter is beneficial. Either way, the IMU based solution (with LN200 levels of corruptions) appears to provide very satisfactory performances. Recall that the modeled sensor corruption levels reflected very conservative LN200 specification values.

### 3.3 Influence of Sensor Bias

This section discusses how a constant accelerometer bias will influence, and provide estimates of the large the steady-state cargo position offsets will be. Let  $a(t)$  be the erroneous measured acceleration information, while  $A(s)$  is the corresponding Laplace transform of this signal. If the erroneous acceleration is a constant bias  $b$ , then we find

$$A(s) = \frac{b}{s} \quad (3.1)$$

To obtain the erroneous velocity estimate  $V(s)$ , we first integrate  $A(s)$  without filtering:

$$V(s) = \frac{b}{s} \cdot \frac{1}{s} = \frac{b}{s^2} \quad (3.2)$$

To determine the steady-state behavior as  $t \rightarrow \infty$ , we make use of the theorem<sup>4</sup>

$$\lim_{t \rightarrow \infty} v(t) = \lim_{s \rightarrow 0} sV(s) \quad (3.3)$$

Thus, for the un-aided acceleration integration we find

$$\lim_{s \rightarrow 0} sV(s) = \lim_{s \rightarrow 0} \frac{b}{s} \rightarrow \infty \quad (3.4)$$

This illustrates how an accelerometer bias will cause a secular error growth in the velocity estimate.

Next, let us investigate what happens if the band-pass filter is added to the integration step. The resulting velocity  $V(s)$  is

$$V(s) = \frac{b}{s} \cdot \frac{1}{s} \cdot \frac{BW s}{s^2 + BW s + \omega_c^2} \quad (3.5)$$

where  $\omega_c$  is the filter center frequency and  $BW$  is the filter bandwidth parameter. The steady-state response is then

$$\lim_{s \rightarrow 0} sV(s) = \lim_{s \rightarrow 0} \frac{bBW}{s^2 + BW s + \omega_c^2} = bBW \quad (3.6)$$

The filter stabilizes the otherwise unstable integration. However, note that the velocity error does not asymptotically converge to zero. Rather, a residual velocity bias error will result.

Next, we examine the velocity error behavior if the acceleration measurement receives an additional band-pass filtering. In this case we find

$$V(s) = \frac{b}{s} \frac{1}{s} \left( \frac{BW s}{s^2 + BW s + \omega_c^2} \right)^2 \quad (3.7)$$

The corresponding steady-state response is

$$\lim_{s \rightarrow 0} sV(s) = \lim_{s \rightarrow 0} \frac{bBW^2 s}{s^2 + BW s + \omega_c^2} = 0 \quad (3.8)$$

Adding this extra bandpass filter will cause the accelerometer bias to be completely rejected. The steady-state velocity errors will go to zero.

Repeating this process, estimates for the integrated/filtered position error motion are obtained. Without the extra filter, the accelerometer will cause a steady-state offset in position of

$$\lim_{s \rightarrow 0} sX(s) = bBW^2 \quad (3.9)$$

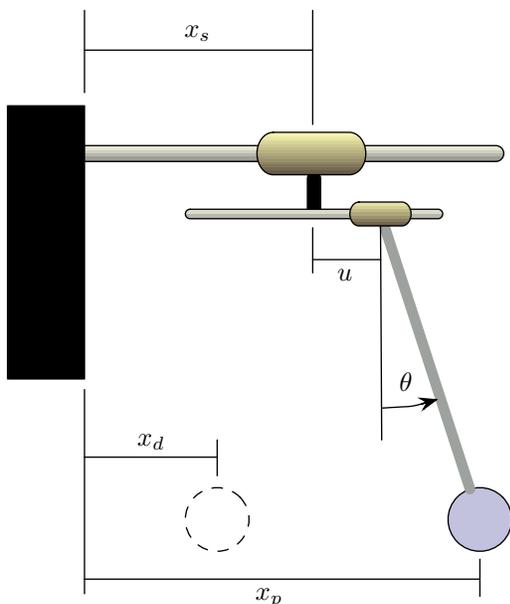
With the additional filter added, the steady-state position error will be

$$\lim_{s \rightarrow 0} sX(s) = 0 \quad (3.10)$$

### 3.4 2D-Model Sensitivity Study

A two-dimensional cart-pendulum model was developed to study the sensitivity of the various control strategies to the ship motion estimation filter settings and IMU sensor errors. The

**CraneSim** simulation program offers advanced crane, ship and payload motion modeling capabilities. This is very useful when performing detailed proof-of-concept simulation and detailed control performance analysis. However, the fidelity of the simulation also makes it more difficult to perform simpler sensitivity studies. For example, **CraneSim** includes bit-level modeling of the ship motion sensor, and well as crane control and swing sensor communication modeling. Further, several crane drive system models are available high various degrees of fidelity.



**Figure 3.2:** Illustration of the Less-Complex Cart-Pendulum Dynamical Model.

Figure 3.2 illustrate the simplified dynamical system considered. Here the relevant dynamical components are still modeled, and it is much simpler to just what *relative* performance gains or losses will result with different control strategies. Instead of having the crane boom tip moving through 3D space to cause payload swing, a cart moving on a 1D track is modeled. This simulates the essentially 1D nature of the boom tip motion when the ship is undergoing mostly rolling motions (dominant ship motion mode). A second rail is attached to the primary cart. The hinge point of the spherical pendulum is attached to a 2<sup>nd</sup> cart. To compensate for the ship motion, as the primary cart (simulated

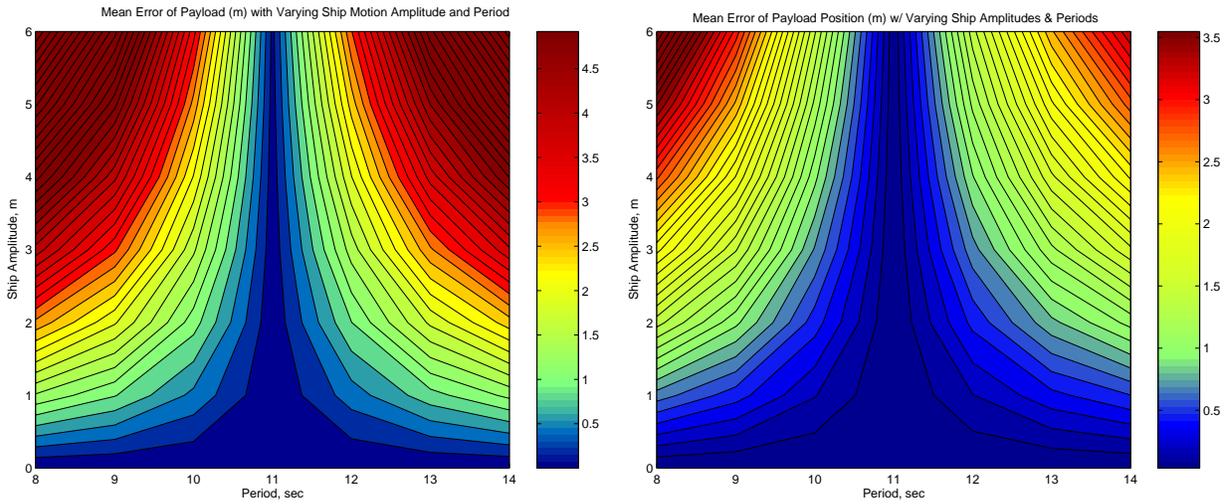
ship motion) moves through inertial space, the secondary cart (simulated crane motion) must compensate to stabilize the payload at a desired position. The control  $u$  must compensate for the perceived ship motion  $x_s$  and cargo pendulation  $\theta$  to stabilize the payload at the desired inertial location  $x_d$ . A simplified 1D version of the position- and velocity-based PCS strategies is used to control this simplified systems. The perceived ship motion is corrupted using the identical types and levels of IMU corruptions as discussed earlier. The estimated ship (primary cart) motion is estimated using the previously presented filtering algorithms. However, no detailed communication of crane servo drive system are modeled here.

The ship estimation filter frequency is set to a constant value. The **CraneSim** implementation shows how the ship motion period could be determined through the sensed motion. The period of a nearly sinusoidal motion can easily be estimated to within less than 0.5 seconds. Performance studies are conducted to explore how accurate the ship motion filter settings must be. The nominal simulation parameters are listed in Table 3.5. Note that some of these values are varied during the following sweeps. This table only provides the nominal values.

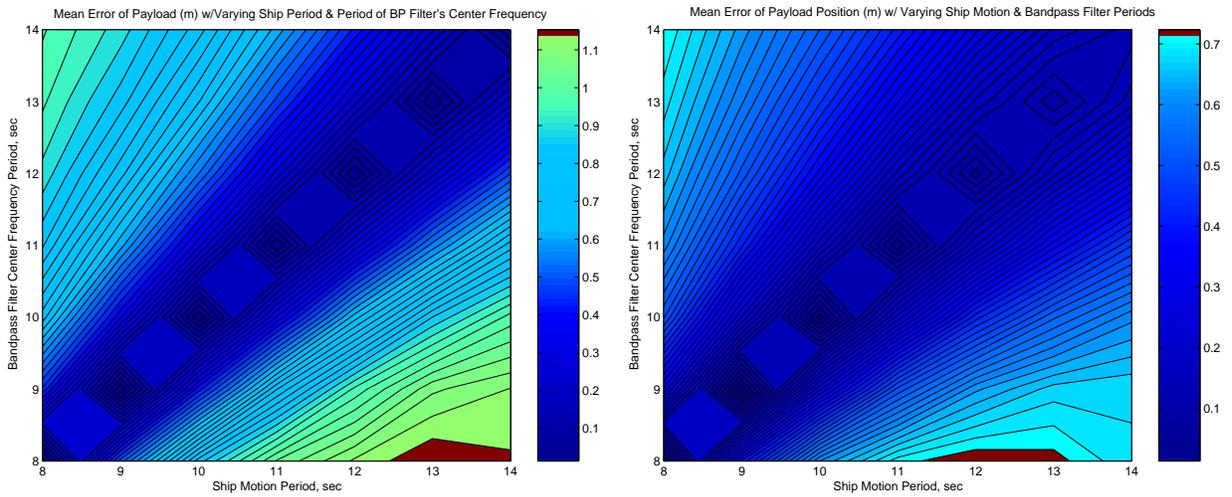
**Table 3.5:** Nominal 2D Simulation Parameters

Parameter	Value
Hoist Length $L$	35 m
Control Time Step	1/40 sec
Ship Motion Period	11 sec
Ship Motion Amplitude	1 m
Filter Center Frequency	11 sec
Filter Bandwidth $BW$	0.1 Hz
Filter Damping Coefficient $\xi$	0.707
Accelerometer Bias	$2 \cdot 10^{-3}$ g

The first study of interest is a sweep of the ship motion amplitude (of  $x_s$ ) versus the ship motion period. Once case uses the existing PCS control strategy, but models the ship motion as being sensed by an IMU. The second case has the same ship motion sensor models, but uses the new velocity-based PCS strategy. All



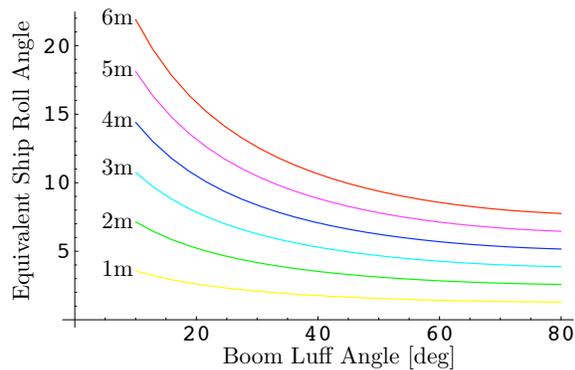
(a) Position-Based PCS – Boom tip motion vs Ship Period (b) Velocity-Based PCS – Boom tip motion vs Ship Period



(c) Position-Based PCS – Filter center frequency vs Ship Period (d) Velocity-Based PCS – Filter center frequency vs Ship Period

**Figure 3.3:** 2D Cart-Pendulum Performance Tests for the Position- and Velocity-Based PCS Concepts. Ship Motion is Measured using an IMU Sensor Only in Both Cases.

studies show the final steady-state inertial payload motion. Note that this rejects any constant steady-state errors. It is assumed that a human operator will be in the loop and should be able to compensate for static payload offsets.



**Figure 3.4:** Comparison of Cart Motion Amplitude Versus Equivalent Ship Roll Motion for Various Boom Luff Angles.

Figures 3.3(a) and 3.3(b) show the performance comparisons where the simulated ship motion amplitude and ship roll period are swept. As the the ship period varies, the ship motion estimator filter will not be tuned correctly and an increase in payload steady-state errors is expected. Further, as the boom tip motion is increased, these payload motion errors should increase as well. Both figures clearly illustrate this behavior. Figure 3.4 provides a convenient comparison of cart motion to equivalent ship roll motion. The boom is assumed to be about 35 meters long, and about 10 meters off the water level. For severe ship roll motion up to 5 degrees, we find that the equivalent cart motion amplitude must be within 1–3 meters, depending on the current boom luff angle. The sweeps consider cart motions up to 6 meters. This represents very extreme motion that should not typically be encountered by the PCS. Note that the velocity-based control strategy does provide slightly better results than the position-based strategy. This reflects the findings of the full 3D *CraneSim* simulations presented earlier. Further, note that the steady-state payload errors don’t start to significantly increase until the ship filter frequency is off by

about 0.5–1.0 seconds. This illustrates that the proposed self-tuning ship filter strategy should function very well, where the sinusoidal ship period estimation errors are far less than 0.5 seconds. This results in a more robust PCS strategy that can self-tune its filter as the natural ship period varies from day to day due to different loading conditions.

Figures 3.3(c) and 3.3(d) show the steady-state payload motion errors if the bandpass filter and ship motion period are swept between 8 and 14 seconds. As expected, the performance drops off as the filter and ship periods become more and more out of alignment. Overall the velocity-based solution provides slightly better results in these tests. Further, a 1 second corridor is apparent where the period misalignment has a minor effect on the crane performance.

## Chapter 4

# Conclusion

This report outlines how the GPS/IMU and IMU based ship motion sensors were modeled numerically. The PCS performance of the existing implementation, as well as the proposed IMU position-based, as well as the IMU velocity-based PCS strategies are evaluated using a high fidelity `CraneSim` simulation. The proposed ship motion filtering method is very effective in removing the typical IMU sensor biases. The simulations are performed using the Litton LN200 values, a common and popular IMU. Further, the IMU sensor scaling errors of the LN200 type IMU were modeled as well. The performance penalty on the simulated PCS was minimal. The `CraneSim` simulation also included a nominal drive system model of the current Flickertail State TG3637 class crane (unmodified). The new control methods (using IMU to measure ship) motion are compared to the existing PCS strategy where the ship motion is measured using a GPS/IMU sensor (POS-MV 320). Typical GPS induced drift behaviors are included. The new strategies appears to consistently perform at least as well, and often better than the current strategy. These results are promising in that it appears it is possible to replace the more expensive ship motion sensor with a more cost-effective solution.

The new position-based strategy on average performed slightly worse than velocity-based PCS strategy. In particular, when compared to the drive system induced payload error motion, these performance differences were small, but noticeable. While the velocity-based strategy could provide better steady-state performance,

it does have two drawbacks. First, the current payload deck tracking strategy would have to be redone to be made compatible with this velocity based strategy. As the payload is picked up, there will be some small amount of drift in the vertical direction (due to not being able to reject the gravity term perfectly from the IMU sensor information). The filters will quickly reject this drift as shown in the numerical simulation. From then on, the operator will easily be able to control the cargo position. However, the impact on this drift on the actual pick up process would still need to be studied. This study shows that adding the additional bandpass filter to the measured IMU data could help reduce this pick-up drift substantially.

In conclusion, both IMU based PCS strategies look very promising and should provide cost-effective cargo control strategies.

# Bibliography

- [1] Robinett, R. D., Groom, K. N., Feddema, J. T., and Parker, G. G., “Control system and method for payload control in mobile platform cranes,” December 17 2002, Patent Number 4,596,765.
- [2] Schaub, H., “Rate-Based Pendulation Control System Study,” Technical report, Virginia Tech, 2005.
- [3] Systems, N. G. E., “LN-200 Fiber Optic Inertial Measurement Unit,” Pamphlet, 2000.
- [4] Van De Vegte, J., *Feedback Control Systems*, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 2nd ed., 1990.