Robotic Ultrasound Manipulator: Calibration of Position and Orientation Measurement System

by

Brian Syverud

Submitted to the Department of Mechanical
Engineering in Partial Fulfillment
of the Requirements for the Degree of

Bachelor of Science

at the

Massachusetts Institute of Technology

June 2009

Signature of Author: ....
Department of Mechanical Engineering
May 11, 2009

Certified by: ....
Brian W. Anthony
Laboratory for Manufacturing and Productivity Research Scientist
Thesis Supervisor

Accepted by: ....
Professor J. Lienhard V
Collins Professor of Mechanical Engineering
Chairman, Undergraduate Thesis Committee

Robotic Ultrasound Manipulator: Calibration of Position and Orientation Measurement System

by

Brian Syverud

Submitted to the Department of Mechanical Engineering
on May 11, 2009 in Partial Fulfillment of the
Requirements for the Degree of Bachelor of Science in
Mechanical Engineering

Abstract

The ultimate goal of this project is to develop a fixture for holding and manipulating an ultrasound wand as adeptly as a trained technician. From the 2D slices recorded by an ultrasound transducer, a three-dimensional image can be reconstructed. To reconstruct the 3D image, position and orientation data are needed. In the interests of space and simplicity, a sensor integrating both a gyroscope and accelerometer was chosen for these measurements. Because of the inherent error induced by integrating acceleration to yield position, it was necessary to calibrate this instrument. A fixture was constructed with preset reference points for the purposes of comparison and error analysis. As expected, the measurements obtained directly from the gyroscope were sufficiently accurate to track orientation. The position values from integration of acceleration exhibited accumulation of error over time. From these data, it was clear that a secondary reference instrument is needed for accurate position measurements.

Thesis Supervisor: Brian W. Anthony
Title: Research Scientist

1. Introduction

Ultrasound imaging represents an expanding field of medical technology with a wealth of applications. After the one-dimensional scanner became widely available in the 1970s, the 2D system experienced similar success. At the moment, three-dimensional imaging is possible through a fixed 2D array of crystals or analysis of a fixed 3D volume. It is hoped that a more flexible approach can be developed that utilizes a freehand 2D ultrasound probe. As long as position and orientation of the individual two-dimensional slices are recorded, image processing can be used to reconstruct the 3D volume.

This thesis investigates the potential use of an inertial measurement unit (IMU) to record position and orientation simultaneously. In this case, a 3 degree of freedom IMU from GLI Interactive LLC was used. This sensor, a MotionNode, uses a gyroscope to measure angular velocity about three axes and an accelerometer to measure linear acceleration along three axes. In addition to the integrated gyroscope and accelerometer, it includes a magnetometer to limit drift in angular velocity measurement. Past analyses have examined the MT-9B from XSens Technologies[1]. The GLI device, however, provides similar functionality and compactness at a cheaper price and increased availability.

The use of freehand ultrasound imaging with the aid of an IMU for this 3D image reconstruction is preferable to other available methods because of its non-invasive and robust nature. Other imaging sources (infrared, acoustic, and magnetic) are all subject to environmental factors[2]. At the same time, tracking by means of an optical system or mechanical arm requires a large space commitment and is still physically intrusive. It seems clear that successfully combining tracking by IMU with a freehand 2D ultrasound probe presents an attractive alternative.

2. Background

The major challenge to utilization of a freehand ultrasound transducer in 3D imaging is accurate tracking of position and orientation. In this project, it is hoped that an IMU integrating a 3-axis gyroscope and accelerometer will prove sufficient. According to the MotionNode specifications [3], the device measures 3D orientation to within a half of a degree. One would expect this orientation data to experience drift over time due to the nature of this measurement. The gyroscope contained in this sensor actually measures angular velocity about the three axes pictured below. To reach the angular orientation values which the MotionNode ultimately outputs, it is necessary to integrate these data. Based on the nature of integration, any errors will accumulate during this calculation with the addition of each time step. In this manner, drift would be introduced to the measurements of the gyroscope, if not for the presence of the magnetometer. The magnetometer's sensing of the earth's magnetic field resets error accumulating from the orientation calculation, allowing the MotionNode to obtain such accuracy.

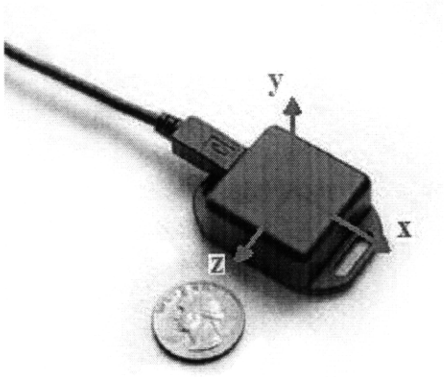


Figure 1: MotionNode Sensor with orthogonal axes indicated and quarter for scale.

Position data is obtained from the accelerometer in a similar manner. The sensor reports acceleration along each of the three axes above. For the purposes of this experiment, it is necessary to integrate separately. Unlike the gyroscope data, however, the integration is a two step process, from acceleration to velocity and from velocity to position. The second integration step compounds the error accumulation inherent to the process. In addition, the accelerometer has no secondary reference to reset this drift. As a result, this analysis focuses on quantifying the integrated position error. Experiments focus on determining the effects of various sources, including sensor noise, movement speed, and total travel distance.

After running initial tests on the MotionNode, it became clear that the constant presence of noise in the sensor data exerted a non-trivial influence. To characterize this effect, the signal-to-noise ratio (SNR) was determined, as governed by the following equation:

$$\mathrm{SNR} = \frac{P_{\mathrm{signal}}}{P_{\mathrm{noise}}} \tag{Eq. 1}$$

Since noise seemed to remain constant with or without motion, it was assumed that noise was independent of signal. As a result, it was possible to decouple sensor output according to the following relation:

$$P_{wave} = P_{signal} + P_{noise} \tag{Eq 2.}$$

Using the "var" function in Matlab, it was possible to extract two of the three quantities from Eq. 2. By examining the MotionNode under fixed conditions, the noise component was determined. For each trial involving translation, the wave component was determined. A subtraction operation could then be performed to find the signal value, and the SNR was ultimately measured.

3. Experimental Procedure

Analysis of the MotionNode was decoupled to focus on the following areas:

(1) Orientation data from the gyroscope
(2) Acceleration data from the accelerometer

Because orientation data is already provided by the MotionNode, analysis of the gyroscope was mainly performed to confirm the resolution listed in its specifications. As a result, the procedure followed was relatively simple. In this case, only rotation around the y-axis was analyzed. A device similar to a lazy susan was modified to fit the requirements of the experiment. Initially, the device merely consisted of two circular sections, one nested in the other. The outer section acted as a base, while the inner piece was free to rotate. After finding the center of rotation by careful measurement, the MotionNode was mounted accordingly so that the sensor's y-axis was centered on the origin. In addition, a line was stretched taut from rim to rim of the base piece, passing through the center of rotation, to act as a fixed reference point. Several markings were added to the inner, rotary section to indicate angular travel. The completed fixture, with mounted IMU is pictured below:

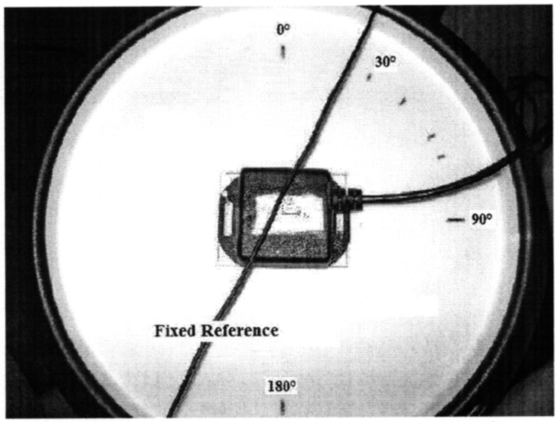


Figure 2: Device for testing orientation data.

After preparing the MotionNode for measurement, a series of trials was conducted. Before each test, pins were placed at the zero reference angle and the desired angle to constrain rotation to the relevant area. The various angles measured were 22.5°, 30°, 45°, 60°, 67.5°, 90°, 180° and 360°. Each trial was repeated three times to ensure consistency. The resulting data were analyzed in Matlab.

When examining the acceleration data, the stage of an EZ-Trak CNC Mill was used as a ground reference for position. The stage limited analysis to the x- and z-directions but also presented two significant advantages. As an instrument of precision machining, the mill stage was able to record translation to the ten-thousandth of an inch. Furthermore, the feed rate was easily set and adjusted, allowing for analysis of acceleration drift at varying velocities. Once the MotionNode was firmly fixtured to the stage, a G code [Appendix A] was input to the CNC mill to run a series of translation trials. Before running any trials, acceleration data was taken while the

MotionNode and stage remained motionless. Analysis of this stationary situation allowed for characterization of the DC noise contribution.

Initially, the program moved the stage and sensor only in one direction. Translations of three and twelve inches were examined in the x-direction at feed rates of sixty and thirty inches per minute. In the y-direction, displacements of three and nine inches were analyzed at the same feed rates. (It is important to note that the y-direction references the mill's coordinate system. The y-direction on the mill corresponds to the z-direction in the MotionNode coordinate system.) To further quantify the acceleration drift, the sensor was also moved in a rectangular circuit. The first circuit traced once the edges of a twelve inch by nine inch rectangle and returned to the origin. The second circuit traced thrice the same rectangle and also returned to the origin. Again, all resulting data was analyzed in Matlab.

4. Results and Discussion

After following the procedure described above, Matlab was used to import and analyze the data. The angular velocity output from the MotionNode was integrated to yield overall orientation. The following graphic portrays the angular velocity data for a rotation of 22.5°:

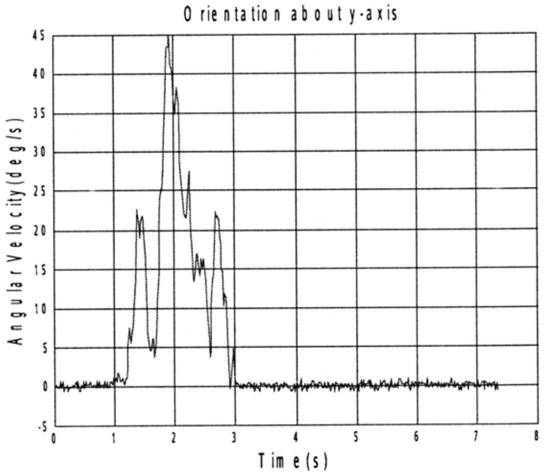


Figure 3: MotionNode output from gyroscope with rotation about y-axis.

For this specific trial, the integrated sensor reading was 22.37°. The values for each of the three repeated trials were averaged. In this manner, the average angular drift at each change in orientation was measured. The final values were tabulated and compared to the MotionNode specifications. As expected, the drift in orientation was relatively small. Average values ranged between -0.85° and 1.87° with a root mean square error of 0.048°. These deviations fell within the resolution of ±0.5° to ±2.0° listed by the sensor manufacturers. The overall spread of average drift values for each rotation is portrayed below:

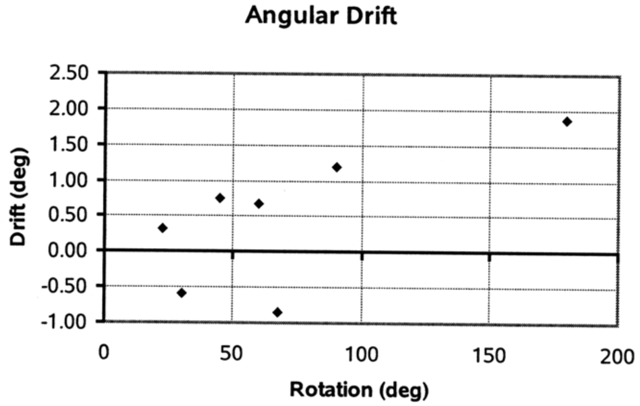


Figure 4: Summary of drift in orientation measurements.

In contrast to the orientation data, the measurement of integrated position data was a more complicated process. Initially, the MotionNode was examined while motionless to determine the contribution of sensor noise. As figure 5 shows, the noise behavior was sinusoidal in nature. Both the variance and mean of this profile were calculated for future use.

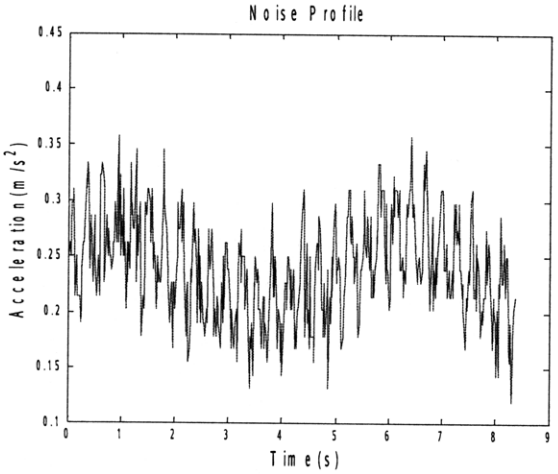


Figure 5: MotionNode noise characterization as measured by accelerometer while sensor remained motionless.

Once the typical variance of the background noise was quantified, it was possible to analyze the wave signal to determine the SNR. As figure 6 indicates, acceleration data measured from each trial was clipped to include only the period during which the sensor was in motion. In this region, the wave energy consisted of both the noise and signal contributions. During the periods before and after this section, the MotionNode experienced no acceleration besides gravity, and noise was the only component of the wave energy. In the relevant period, marked as "Sensor + Noise" below, the "var" function in Matlab was used to calculate the overall wave energy. This process was repeated for each trial, and equation 2 was applied to extract the signal energy.

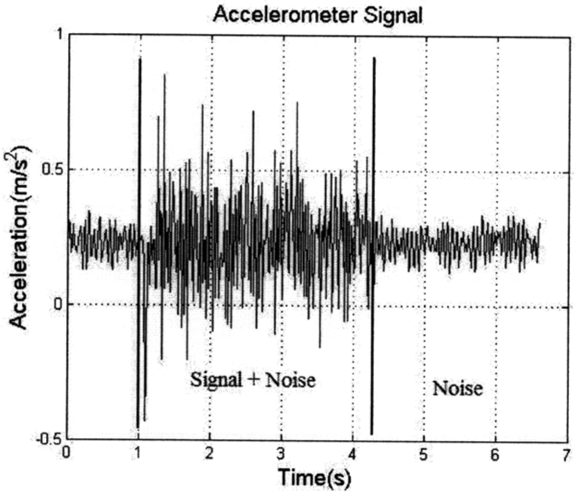


Figure 6: MotionNode output from accelerometer with translation in x-direction.

Trial	SNR (dB)
3in Translation in x, High Feed	18.6
3in Translation in x, Low Feed	14.8
12in Translation in x, High Feed	15.9
12in Translation in x, Low Feed	14.4
3in Translation in z, High Feed	11.4
3in Translation in z, Low Feed	7.1
9in Translation in z, High Feed	10.4
9in Translation in z, Low Feed	2.6

Table 1

The signal-to-noise ratio for each trial is listed in the table above. It is worth noting that even the most favorable ratios are not particularly high, and in every case, a significant portion of the signal is lost in the background noise. It also appears that the accelerometer is more sensitive in the x-axis than the z-axis. In addition, the ratio also declines as travel distance increases, suggesting that the signal decreases as time period increases.

With this knowledge in mind, the acceleration data from each trial was analyzed. The accelerations were integrated twice to yield final position measurements. The graphs in Appendix C display the resulting position values. The behavior of the position data matches initial expectations. The position values experience an approximately linear increase, corresponding to a constant velocity. Because the feed rate of the mill was preset, the translation velocity must have been constant. Upon closer examination, the position data were not nearly as accurate as they appeared. As the tabulated values below indicate, error accumulates exponentially with time. In both the x- and z-directions, the high feed translation of only 3 inches exhibited less than half an inch of drift. Cutting the feed rate in half, from 60 inches per minute to 30, doubled the number of time steps involved and increased the drift to 4.34" and 9.16" in x and z, respectively. When the translation distance was increased, along with the number of time steps, error accumulation was overwhelming.

High Feed		Low Feed	
Global Translation(in)	Sensor Translation(in)	Global Translation(in)	Sensor Translation(in)
3 (x-dir)	3.48	3 (x-dir)	7.34
12 (x-dir)	58.62	12 (x-dir)	130.1
3 (z-dir)	3.38	3 (z-dir)	12.16
9 (z-dir)	25.42	9 (z-dir)	104

Table 2: Comparison of integrated sensor measurements to those of the CNC mill. All data is normalized to the sensor coordinate frame as pictured in Figure 1.

5. Conclusions and Recommendations

Analysis of MotionNode data, specifically gyroscope and accelerometer readings in orthogonal directions, was successfully conducted by integration in Matlab. Examination of results confirmed the reliability of the integrated gyroscope and magnetometer as a means of measuring orientation. Accumulated drift in every trial was less than 2°, well within the sensor's advertised resolution. Position data obtained by double integrating acceleration, however, exhibited significant drift. After continually sensing for more than 10 seconds, the accumulated error was so great that integrated position was no longer on the same order as the global value. In order to use this IMU for 3D image reconstruction, such drift is unacceptable. In the same manner that the gyroscope is aided by the magnetometer, the accelerometer also needs a secondary reference to correct drift and provide accurate position readings.

Acknowledgments

I would like to thank Brian Anthony and Shih-Yu Sun for their input on this project. Their expertise was an invaluable asset. I would also like to thank Matthew Gilbertson and Dean Ljubicic for their assistance.

References

1. A.A. Abdul Rahni, I. Yahya, and S.M. Mustaza. "2D Translation from a 6-DOF MEMS IMU's Orientation for Freehand 3D Ultrasound Scanning." IFMBE Proceedings, vol. 21. 2008.
2. A. Gee, R. Prager, G. Treece, and L. Berman. "Engineering a Freehand 3D Ultrasound System." Pattern Recognition Letters, vol. 24, pp. 757-777. February 2003.
3. MotionNode Specification, GLI Interactive LLC at http://www.motionnode.com/motionnode_spec_r00.pdf

Appendix A

CNC Mill G Code Test Program

```
N2 G00 G90 G54 X0 Y0 Z0        %Rapid move to origin
N3 M00                         %Program stop (Begin recording data)
N4 G01 X12.0 F60.0             %Trial 1: move 12in along x-axis at 60in/min
N5 M00                         %Program stop (Stop recording data)
N6 G00 X0                      %Return to origin
N7 M00
N8 G01 X12.0 F30.0             %Trial 2: move 12in along x-axis at 30in/min
N9 M00
N10 G00 X0
N11 M00
N12 G01 X3.0 F60.0             %Trial 3: move 3in along x-axis at 60in/min
N13 M00
N14 G00 X0
N15 M00
N16 G01 X3.0 F30.0             %Trial 4: move 3in along x-axis at 30in/min
N17 M00
N18 G00 X0
N19 M00
N20 G01 Y9.0 F60.0             %Trial 5: move 9in along y-axis at 60in/min
N21 M00
N22 G00 Y0
N23 M00
N24 G01 Y9.0 F30.0             %Trial 6: move 9in along y-axis at 30in/min
N25 M00
N26 G00 Y0
N27 M00
N28 G01 Y3.0 F60.0             %Trial 7: move 3in along y-axis at 60in/min
N29 M00
N30 G00 Y0
N31 M00
N32 G01 Y3.0 F30.0             %Trial 8: move 3in along y-axis at 30in/min
N33 M00
N34 G00 Y0
```

```
N35 M00
N36 G01 X12.0 F60.0          %Trial 9: trace rectangle in xy-directions, finishing at origin
N37 Y9.0
N38 X0
N39 Y0
N40 M00
N41 G01 X12.0                %Trial 10: trace rectangle twice in xy-directions, finishing at origin
N42 Y9.0
N43 X0
N44 Y0
N45 X12.0
N46 Y9.0
N47 X0
N48 Y0
N49 M00
N50 G01 X12.0                %Trial 11: trace rectangle thrice in xy-directions, finishing at origin
N51 Y9.0
N52 X0
N53 Y0
N54 X12.0
N55 Y9.0
N56 X0
N57 Y0
N58 X12.0
N59 Y9.0
N60 X0
N61 Y0
N62 M00
N63 M02
```

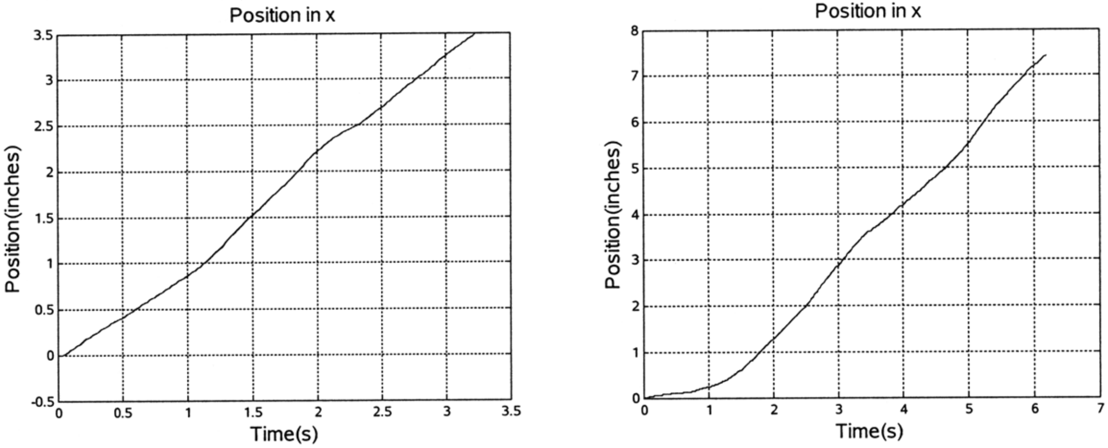
Appendix B

Sample Matlab code for integrating acceleration to position

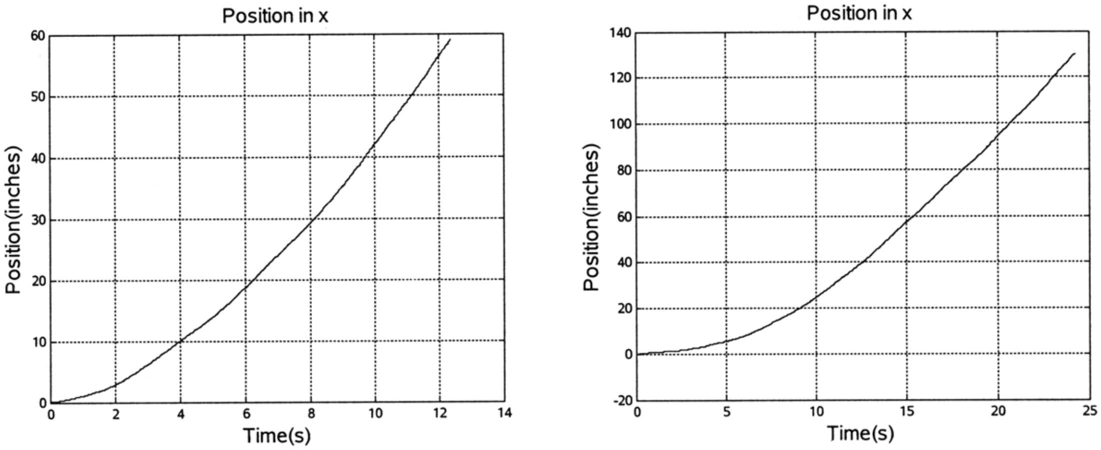
```
cd ‘C:\Documents and Settings\LLP\My Documents\MotionNode\take\x3highfeed’
%Choose trial to be analyzed
data = plot_sensor('sensor/motionnode');        %Import data from MotionNode
ax = data(:,1);                                 %Select acceleration in x-direction column
g = 9.80665;
ax = ax * g;                                    %Scale acceleration from g’s to m/s^2
figure, plot(ax)
ax = ax(67:312);                                %Clip data to include only sensor motion
ax = ax - DC;
for i = length(ax)                              %Integrate acceleration to velocity
   vx(i) = sum(ax(1:i));
end
for i = length(vx)                              %Integrate velocity to postion
   px(i) = sum(-vx(1:i));
end
px = px * (1/60)^2;                             %Scale postion to account for sample rate
px = px * 39.37;                                %Scale position to inches
t = (0:length(px)-1)*(1/60);                    %Insert time variable
figure, plot(t,px)
grid on
title('position in x', 'FontSize', 14)
xlabel('time(s)', 'FontSize', 14)
ylabel('position(inches)', 'FontSize', 14)
```

Appendix C

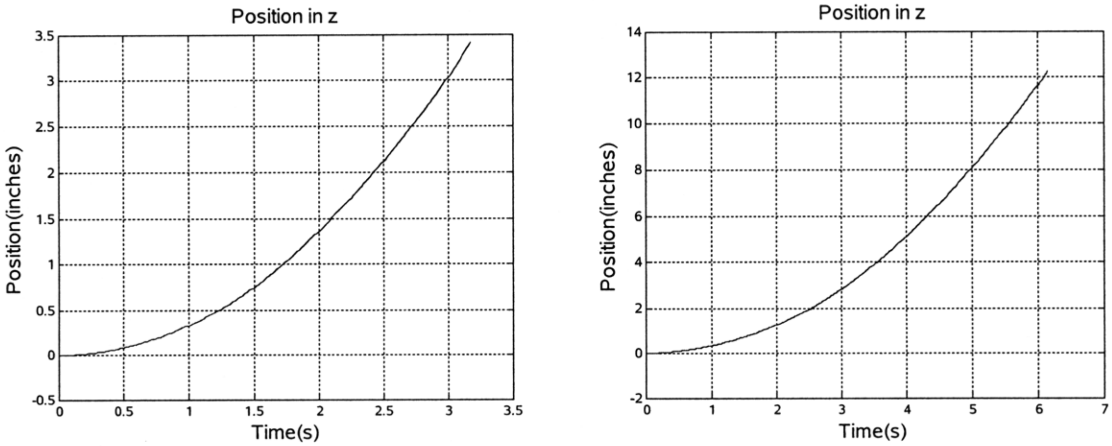
Position data as Integrated from Matlab



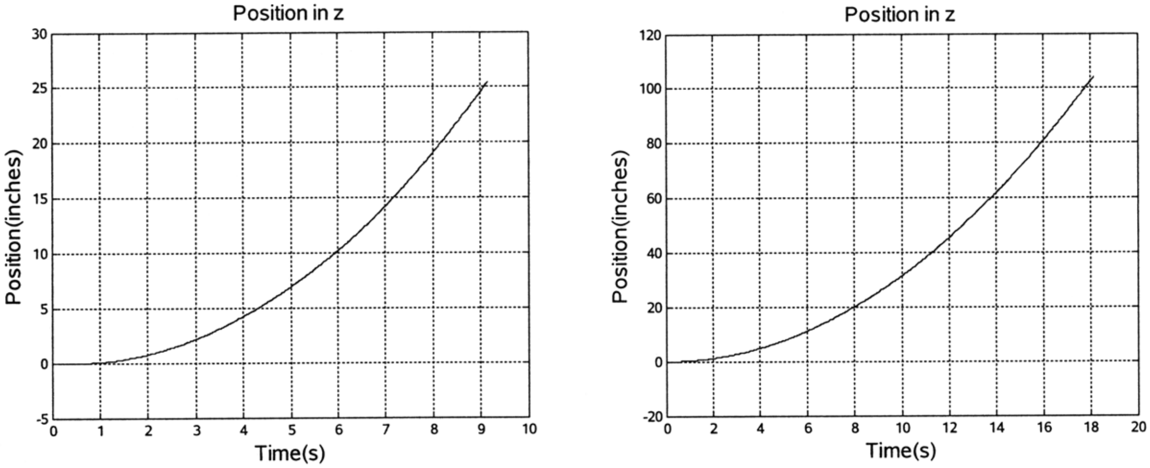
3" global translation in x at high feed (left) and low feed (right)



12" global translation in x at high feed (left) and low feed (right)



3” global translation in z at high feed (left) and low feed (right)



9” global translation in z at high feed (left) and low feed (right)