SANDIA REPORT
SAND2001-0172
Unlimited Release
Printed January 2001

Advanced Techniques for Real-Time Visualization of Data Intensive Missions

Mark R. Platzbecker, Gary W. Ashcraft, Todd E. Owen, Beverly R. Sturgis

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy under Contract DE-AC04-94AL85000.

Note: Public Relations & Communications Center 12600 publishes a printed version of this Guide annually. Between printings, we update the Guide on the Web. The current Web version can be accessed from Sandia's Home Page:

Services, Tech Library, Technical Library, SAND Reports, Direct links to special SAND Reports, SAND98-0730: Guide to Preparing SAND Reports.

Address: **<http://infoserve.library.sandia.gov/sand_doc/1998/980730.pdf>**

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

Available to DOE and DOE contractors from
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Prices available from (615) 576-8401, FTS 626-8401

Available to the public from
National Technical Information Service
U.S. Department of Commerce
5285 Port Royal Rd
Springfield, VA 22161

NTIS price codes
Printed copy: A11
Microfiche copy: A01

SAND2001-0172
Unlimited Release
Printed January 2001

Advanced Techniques for Real-Time Visualization of Data Intensive Missions

Mark R. Platzbecker
Telemetry Technology Development

Gary W. Ashcraft
High Integrity Software Systems Engineering

Todd E. Owen
Control Subsystems Department

Beverly R. Sturgis
Aerosciences & Comp Fluid Mechanics

Sandia National Laboratories
P.O. Box 5800
Albuquerque, NM 87185-0986

Abstract

Engineers at Sandia National Laboratories are combining entertainment industry software with traditional data collection techniques to create an interactive visualization tool. By replacing the usual flight simulator joystick with a telemetry data stream, experimental data is combined with existing three-dimensional (3D) engineering models. Users are immersed in their experiment, allowing interaction with and comprehension of complex data sets. Software tools are currently under development for post flight data visualization, and their usefulness and reusability have been demonstrated on numerous spaced-based programs within Sandia. However, data from remote sensors are subject to transmission errors that yield nonphysical behavior in real-time data visualization applications. We propose to investigate the applicability of real-time processing algorithms and estimation theories, such as Kalman filters, that have been successfully applied in other fields. Results will be integrated into existing postflight visualization tools for Proof-of-Concept validation and for potential integration of real-time applications.

Intentionally Left Blank

Table of Contents

This page intentionally left blank.

INTRODUCTION

Telemetry at Sandia National Laboratories focuses on gathering information from inaccessible flight test bodies such as satellites, space reentry vehicles, aircraft-delivered bombs, and earth penetrators. We help our customers acquire, process, protect, transmit, receive, and display scientific and engineering data using sensor-based information systems. Increased transmission rates from complex sensors are yielding ever larger data sets. Presenting results from these experiments as traditional numerical lists or two-dimensional plots is non-intuitive and time consuming.

TeleKnoSys™ (distant knowledge) changes the way that analysts, engineers and scientists explore and understand complex data sets. Telemetry data is fused with 3D solid models in an interactive pseudo-real environment that immerses users in their flight tests. Data viewed in its experimental context can be understood faster and more completely.

Combining commercial visualization and data acquisition software with mission specific analysis code allows us to generate rapid, cost effective data visualization applications.

CURRENT NEEDS

A real-time, field deployable, advanced visualization system is needed to enhance vital mission monitoring during missile launch and target deployment windows. Real-time visualization of deployed-instrumentation data would allow mission controllers, range safety and system engineers to quickly see, comprehend, and react to mission events in an interactive 3D environment that is compatible with very short decision timetables.

Software tools are currently under development for post flight data visualization, and their usefulness and reusability have been demonstrated on numerous spaced-based programs within Sandia. Unfortunately with currently available processes, technologies and software tools, data manipulation and visualization can only begin once data tapes are recovered after each specific mission has flown. A number of research and development issues must be addressed before a real-time capability can be fully defined and implemented. This LDRD focused on one of these issues, real-time display of information from conflicting, noisy data inputs.

TECHNICAL ISSUES

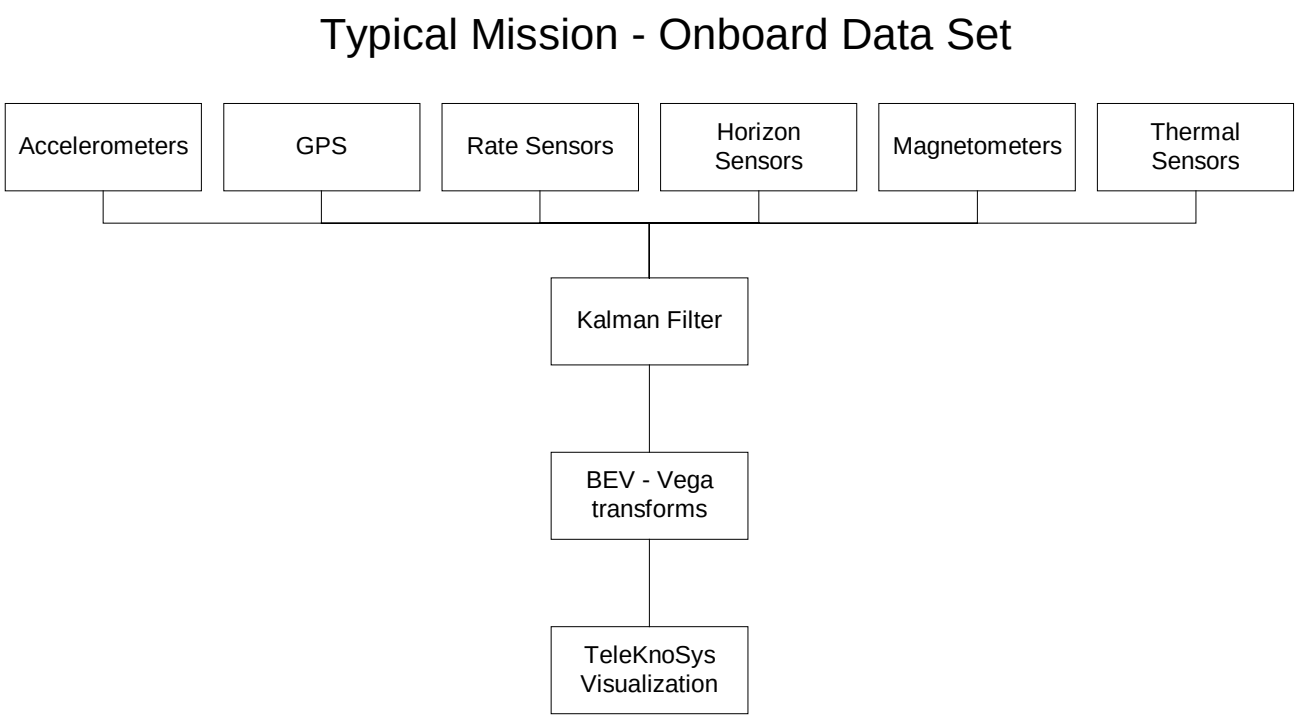
Data from remote sensors are subject to transmission errors that yield, if unfiltered, nonphysical behavior in real-time data visualization applications. Transmission errors arise from a number of different causal events and result in lost, inaccurate and conflicting data that can quickly lead to misinformation. For example, an uncorrected bit error may indicate an invalid spin rate for a body during deployment and free flight and could dramatically affect the visualized orientation of the body. Filtering of transmission errors is the key to a trusted visualization environment.

Real-time data interpretation can be computationally intensive. Some preliminary work has been completed to verify that sufficient computational resources are available to support a real-time application. These initial applications visualized the motions of exoatmospheric reentry vehicles and used simple rigid body dynamics equations to bridge transmission errors. These results demonstrated the data correction algorithm took only a few milliseconds of the tens of milliseconds available between data points. The applications made strong use of multiple processors, and demonstrated the feasibility of a more robust set of real-time data correction algorithms for visualization.

INVESTMENT RETURN

The above efforts are a firm foundation for moving forward with real-time visualization of data intensive scenarios where time critical decisions must be made. Knowledge from this effort will feed into ongoing data visualization efforts and will directly effect the efficiency and effectiveness of a number of ongoing tests. In addition, Sandia's customers have expressed an interest in the deployment of a real-time data visualization system and are considering a proposal for developing and fielding an early prototype upon the successful demonstration of real-time capabilities. If successful, this could lead to a new area of Sandia support and technology development for a large number of missile tests already planned for the future.

This proposal is considered the initial stepping stone to better understanding and development of tools to allow significantly better data to information to knowledge translation for a wide spectrum of military and civilian applications hampered today by intensive and conflicting data. Applications such as battlefield oversight and learning and critical infrastructure monitoring are expected to be major growth areas for the future.



FILTERING TECHNIQUES

1.1 Data Generation

The set of data used as the input to the Kalman filter and to the "old code" consists of the following sequence:

1) a 10 second segment of clean data,
2) a 30 second segment containing dropouts,
3) a 30 second segment containing spikes,
4) a 30 second segment of scale and bias errors (not activated at this point),
5) a 20 second segment containing dropouts and spikes.

These data were generated using a simulation of a reentry vehicle in exoatmospheric flight. A "perfect" data stream from the simulation was corrupted by including dropouts and spikes according to the schedule above. The uncorrupted data stream was also provided in order to show the actual orientation of the vehicle in space.

1.2 Threshold Filtering Technique

Two candidate real time telemetry data correction techniques were evaluated, Threshold filtering and Kalman filtering.

A threshold filtering technique already implemented for post-flight processing was evaluated. This technique utilizes three gyroscopic rate sensors and the equations of motion for a spinning rigid body to calculate the angular rates and orientation of the body. It is referred to as a threshold filtering technique because limits for valid rate sensor measurements are determined by examining the recorded telemetry data. When the telemetry rate sensor data is processed, if the values fall within the pre-determined threshold limits, they are accepted and the roll, pitch, and yaw rate estimates are simply set equal to the measurements.

If they fall outside the limits, they are rejected, and instead the body's previous good motion measurements are propagated forward to the time of the next set of measurements.
The differential angular rate equations, which assume a spinning rigid body with no cross-coupled moments of inertia, are:

where

$$\dot{roll\_rate} = \frac{(RIYY - RIZZ)}{RIXX} * pitch\_rate * yaw\_rate \quad (1)$$

$$\dot{pitch\_rate} = \frac{(RIZZ - RIXX)}{RIYY} * roll\_rate * yaw\_rate \quad (2)$$

$$\dot{yaw\_rate} = \frac{(RIXX - RIYY)}{RIZZ} * roll\_rate * pitch\_rate \quad (3)$$

RIXX is the moment of inertia about the X or roll axis,
RIYY is the moment of inertia about the Y or pitch axis, and
RIZZ is the moment of inertia about the Z or yaw axis.

Once the rates are defined, the differential equations of motion to calculate the roll, pitch, and yaw angles in an inertial reference frame (using a Body 3-2-1 rotation sequence[1]) are:

$$\dot{roll\_angle} = roll\_rate + [pitch\_rate * \sin(roll\_angle) + yaw\_rate * \cos(roll\_angle)] * \tan(pitch\_angle) \quad (4)$$

$$\dot{pitch\_angle} = pitch\_rate * \cos(roll\_angle) - yaw\_rate * \sin(roll\_angle) \quad (5)$$

$$\dot{yaw\_angle} = (pitch\_rate * \sin(roll\_angle) + yaw\_rate * \cos(roll\_angle)) / \cos(pitch\_angle) \quad (6)$$

These equations are integrated for each time step to solve for the attitude of the vehicle. An initial set of orientation angles is assumed based on the vehicle's nominal attitude at time t_o.

1.3 Kalman Filter Technique

1.3.1 Discrete Kalman Filter Equations

The second technique is a Kalman filtering of data. A Kalman filter is a set of mathematical equations that provides a recursive solution of the least squares method of estimation in which one tries to find a "best fit" to a set of data using all of the previously gathered data up to the present time. There are many excellent references to Kalman filter theory and derivation [Gelb, etc.], so only the filter equations are presented here.

The Kalman filter solves the general problem of trying to estimate the "state" $\underline{x}$ (where the "_" denotes a vector) of a discrete-time controlled process described by the equations:

$$\underline{x}_k = \phi_{k-1}\underline{x}_{k-1} + \underline{w}_{k-1} \tag{7}$$

and

$$\underline{z}_k = H_k\underline{x}_k + \underline{v}_k. \tag{8}$$

where $\underline{z}_k$ is a measurement vector. The matrix ϕ relates the state a time step (k-1) to the state at time step k and the matrix H relates the states at time step k to the measurements at time step k. The random variables $\underline{w}_k$ and $\underline{v}_k$ represent the process and measurement noise and are assumed to be independent of each other, white, and with normal probability distributions $p(\underline{w}) = N(0,Q)$ and $p(\underline{v}) = N(0,R)$. The expected value of $\underline{x}(0)$ is given as $\underline{\hat{x}}_0$ and the covariance is given a P_0.

Under these assumptions, the set of discrete Kalman filter equations can be written as:

$$\underline{\hat{x}}_k(-) = \phi_{k-1}\,\underline{\hat{x}}_{k-1}(+) \qquad \text{(state estimation extrapolation)} \tag{9}$$

$$P_k(-) = \phi_{k-1}P_{k-1}(+)\phi^T_{k-1} + Q_{k-1} \qquad \text{(error covariance extrapolation)} \tag{10}$$

$$K_k = P_k(-)H^T_k[H_k\,P_k(-)H^T_k + R_k]^{-1} \qquad \text{(kalman gain matrix calculation)} \tag{11}$$

$$\underline{\hat{x}}_k(+) = \underline{\hat{x}}_k(-) + K_k[z_k - H_k\,\underline{\hat{x}}_k(-)] \qquad \text{(state estimate measurement update)} \tag{12}$$

$$P_k(+) = [I - K_kH_k]\,P_k(-) \qquad \text{(error covariance update)} \tag{13}$$

where $\underline{\hat{x}}$ denotes the Kalman filter's estimate of the state $\underline{x}$, $(-)_k$ denotes the condition at time step k before the measurement update and $(+)_k$ is after a measurement update.

The five recursive Kalman filter equations above operate as follows. Equations 9 and 10 propagate the state and covariance estimates from the previous time step to the current time step, making use of the state transition matrix ϕ, and the variance of the uncertainty, in the state transition equations, Q.

Equation 10 calculates that Kalman gain matrix, K, which will be used in the update equations that follow. The Kalman gain is a measure of the relative weighting that will be given to the new measurements, $\underline{z}$, and the current state estimates, $\hat{\underline{x}}$. If the ratio of the uncertainty associated with the measurements, R, is large relative to the uncertainty associated with the current state estimates, P, then the Kalman gain will be small and the new measurements will not have a large influence on the state estimates. Conversely, if the uncertainty of the measurements is small with respect to the uncertainty of the states, then the Kalman gain matrix will be larger and the new measurements will have a greater impact on the state estimates.

Equations 11 and 12 utilize the Kalman gain matrix, K, calculated in equation 10, to update the state and covariance estimates. The term $[z_k - H_k \hat{\underline{x}}_k(-)]$ in equation 11 is called the measurement residual. It is the difference be the new measurements, z_k and what one would expect the measurements to be based on the state estimates. The measurement residual is then multiplied by the Kalman gain and the product is added to the state estimates to yield an updated state estimate. Equation 12 updates the state estimate covariance based on the Kalman filter gain.

1.3.2 Extended Kalman Filter

Equations 9 through 13 apply if the system under consideration can be expressed as a set of linear equations. What about our case, however, in which the equations of motion that represent the system are highly non-linear, as illustrated by equations 1 through 6? One can linearize the system and measurement equations about the current best estimate in what is called an extended Kalman filter.

First, equations (7) and (8) are re-written to reflect the inclusion of non-linear process and measurement equations:

$$\underline{x}_k = f(\underline{x}_{k-1}, \underline{w}_{k-1}) \qquad (14)$$

and

$$\underline{z}_k = h(\underline{x}_k, \underline{v}_k) \qquad (15)$$

where f(*) is a non-linear function that relates the states at time (k-1) to the states at time (k) and h(*) is a non-linear function that relates the states at time (k) to the measurements at time (k).

Now, define the following matrices which linearize the above equations about the current best estimate of the state, $\hat{\underline{x}}$.

$$A_{[i,j]} = \frac{\partial f(i)}{\partial x(j)} \Big|_{x = \hat{\underline{x}}} \quad (16)$$

$$W_{[i,j]} = \frac{\partial f(i)}{\partial w(j)} \Big|_{x = \hat{\underline{x}}} \quad (17)$$

$$H_{[i,j]} = \frac{\partial h(i)}{\partial x(j)} \Big|_{x = \hat{\underline{x}}} \quad (18)$$

$$V_{[i,j]} = \frac{\partial f(i)}{\partial v(j)} \Big|_{x = \hat{\underline{x}}} \quad (19)$$

The extended Kalman filter equations are then:

$\hat{\underline{x}}_k(-) = f(\hat{\underline{x}}_{k-1}(+))$ (state estimation extrapolation) (20)

$P_k(-) = A_{k-1}P_{k-1}(+)A^T_{k-1} + W_{k-1}\,Q_{k-1}\,W^T_{k-1}$ (error covariance extrapolation) (21)

$K_k = P_k(-)H^T_k[H_k\,P_k(-)H^T_k + V_kR_kV^T_k]^{-1}$ (kalman gain matrix calculation) (22)

$\hat{\underline{x}}_k(+) = \hat{\underline{x}}_k(-) + K_k[z_k - h(\hat{\underline{x}}_k(-))]$ (state estimate measurement update) (23)

$P_k(+) = [I - K_kH_k]\,P_k(-)$ (error covariance update) (24)

1.3.3 Application of Kalman Filter to Telemetry Data

To apply the extended Kalman filter to the real time telemetry data problem, define

x_1 = roll rate ,
x_2 = pitch rate,
x_3 = yaw rate,
x_4 = roll angle,
x_5 = pitch angle and
x_6 = yaw angle.

Then, combining equations (1-6) with equation 20, we can write the state propagation equations and define the function f( ) as:

$$x_{1(k)} = x_{1(k-1)} + \Delta t^* \frac{(RIYY - RIZZ)}{RIXX} * x_{2(k-1)} * x_{3(k-1)} + w_{1(k)}\,, \qquad \text{(roll rate)} \qquad (25)$$

$$x_{2(k)} = x_{2(k-1)} + \Delta t^* \frac{(RIZZ - RIXX)}{RIYY} * x_{1(k-1)} * x_{3(k-1)} + w_{2(k)}\,, \qquad \text{(pitch rate)} \qquad (26)$$

$$x_{3(k)} = x_{3(k-1)} + \Delta t^* \frac{(RIXX - RIYY)}{RIZZ} * x_{1(k-1)} * x_{2(k-1)} + w_{3(k)}, \qquad \text{(yaw rate)} \qquad (27)$$

$$x_{4(k)} = x_{4(k-1)} + \Delta t^*[x_{1(k-1)} + (x_{2(k-1)}*\sin(x_{4(k-1)}) + x_{3(k-1)}*\cos(x_{4(k-1)}))*\tan(x_{5(k-1)})] + w_{4(k)}$$
$$\text{(roll angle)} \qquad (28)$$

$$x_{5(k)} = x_{5(k-1)} + \Delta t^*[x_{2(k-1)}*\cos(x_{4(k-1)}) - x_{3(k-1)}*\sin(x_{4(k-1)})] + w_{5(k)} \qquad \text{(pitch angle)} \qquad (29)$$

$$x_{6(k)} = x_{6(k-1)} + \Delta t^*[x_{2(k-1)}*\sin(x_{4(k-1)}) + x_{3(k-1)}*\cos(x_{4(k-1)})] / \cos(x_{5(k-1)}) + w_{6(k)}$$
$$\text{(yaw angle)} \qquad (30)$$

The matrix A(6x6) (eqn. 16) is the partial derivative of f( ), as defined by the six equations above, with respect to each of the states evaluated at the current best estimate of the state. The matrix W (eqn. 17) is simply the identity matrix.

1.3.4 Kalman Filter Measurements

The Kalman filter measurements, $\underline{z}$, are of the form $\underline{z}_k = h(\underline{x}_k, \underline{v}_k)$. For this telemetry application, six measurements were chosen to be processed. Three are the angular rate measurements given by the three gyroscopes that measure the rate about the x, y, and z axis of the body. The second group of three measurements are magnetometer measurements which measure the pull of the earth's magnetic field along the three body axes. Therefore, the measurement function h( ) for $\underline{z}$ as a function of the states $\underline{x}$ can be written as:

$z_{1(k)} = x_{1(k)} + v_{1(k)}$ (x or roll gyro measurement) (31)
$z_{2(k)} = x_{2(k)} + v_{2(k)}$ (y or pitch gyro measurement) (32)
$z_{3(k)} = x_{3(k)} + v_{3(k)}$ (z or yaw gyro measurement) (33)
$z_{4(k)} = -\sin(x_{5(k)}) + v_{4(k)}$ (x magnetometer measurement) (34)
$z_{5(k)} = \cos(x_{5(k)})*\sin(x_{4(k)}) + v_{5(k)}$ (y magnetometer measurement) (35)
$z_{6(k)} = \cos(x_{5(k)})*\cos(x_{4(k)}) + v_{6(k)}$ (z magnetometer measurement). (36)

The matrix H(6x6) (eqn. 18) is the partial derivative of h( ), as defined by the six equations above, with respect to each of the states evaluated at the current best estimate of the states. The matrix V (eqn. 19) is the identity matrix.

RESULTS

To evaluate the operation of the extended Kalman filter defined above, a set of telemetry data was simulated and noise and dropouts were added to mimic effects seen in actual real time telemetry data. Simulated data was used to avoid any possible classification problems with using real data and to assist in the development and debugging of the software since the "true" vehicle rotational rates and orientations would be known and could be compared to those estimated by the Kalman filter and the threshold filtering techniques.

Both the Kalman filter and threshold filtering equations were coded in "C" and the functions were integrated with the TeleKnoSys visualization tool so the comparisons between the truth, Kalman, and threshold outputs could be visualized. In addition, the outputs of each filter were written to files for further detailed analysis.

The simulated telemetry data consisted of 120 seconds of 263 Hz data which included a time tag, gyro measurements, and magnetometer measurements. The data stream included periods of dropouts and additive noise. A sample of the simulated input data is given in figure 1.

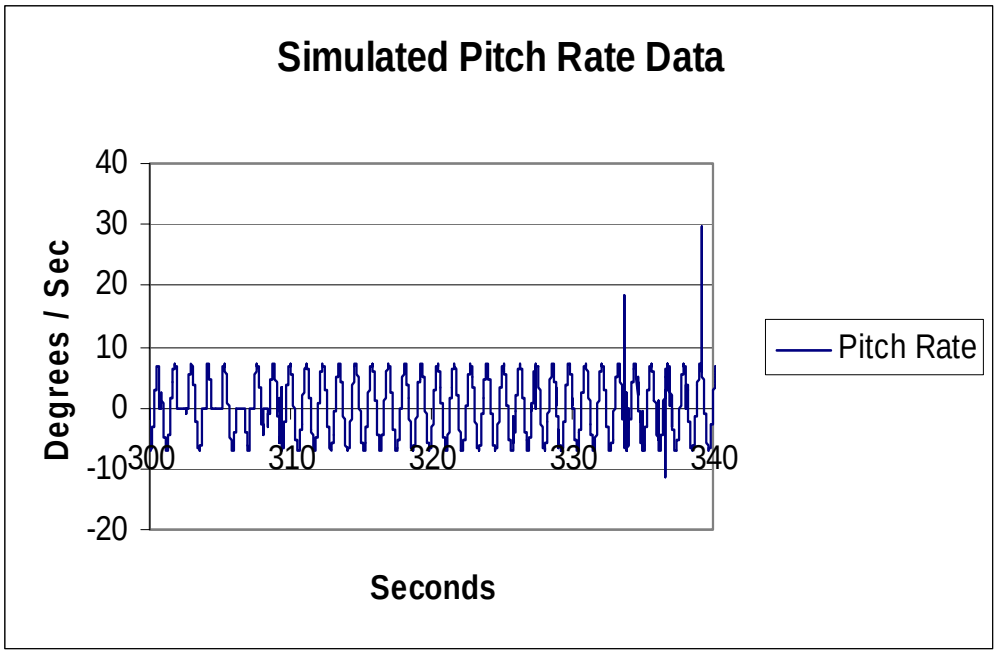


Figure 1. Simulated Telemetry Data

The threshold technique and the equations of motions work well to fill in the dropouts and remove the large noise spikes as shown in the output of the threshold technique for the same time period as shown in figure 1.

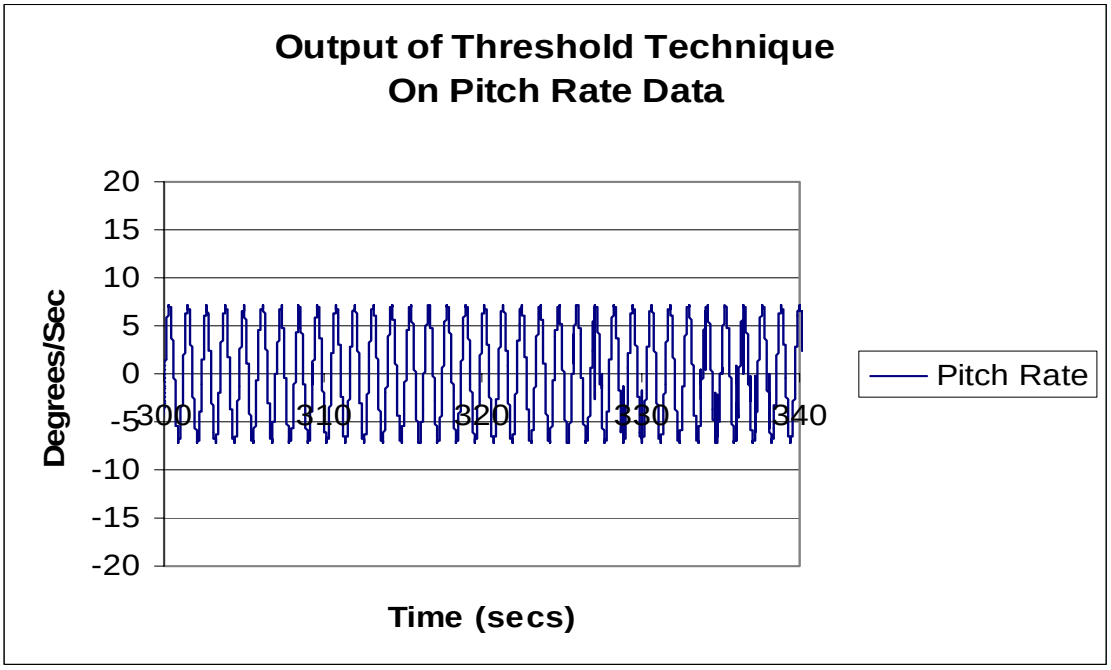


Figure 2. Output of Threshold Technique

However, if one zooms in on the data between 330 and 340 seconds, additive noise that does not exceed the preset thresholds is seen to still remain.

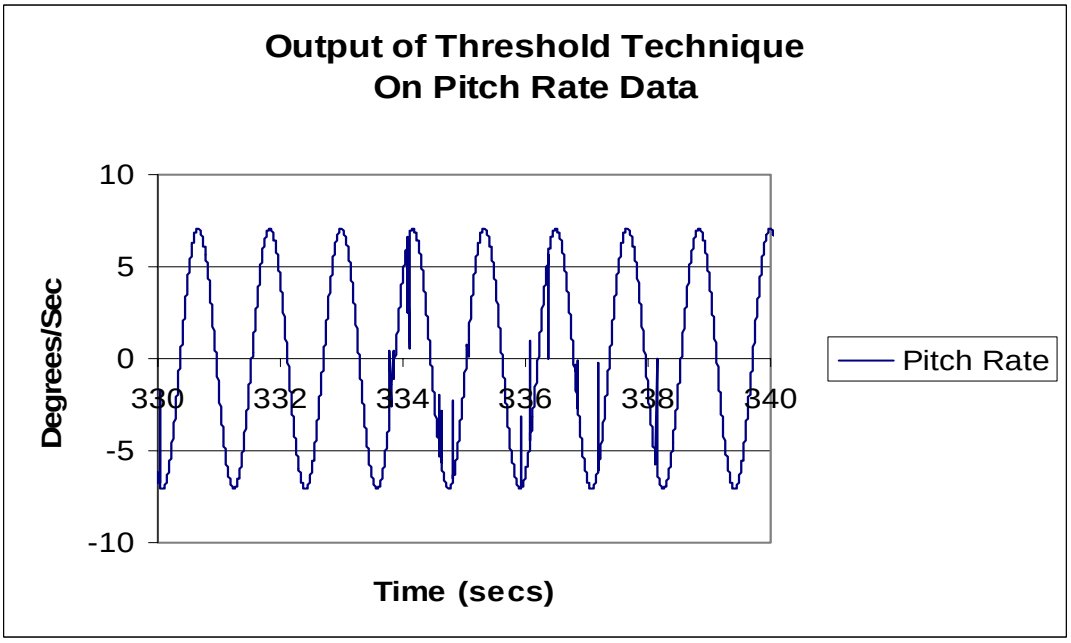


Figure 3. Threshold Technique Leaves Noisy Data

The extended Kalman filter, however, does a better job of blending the noisy gyro measurements with the current estimates of the states to remove much of the noise.

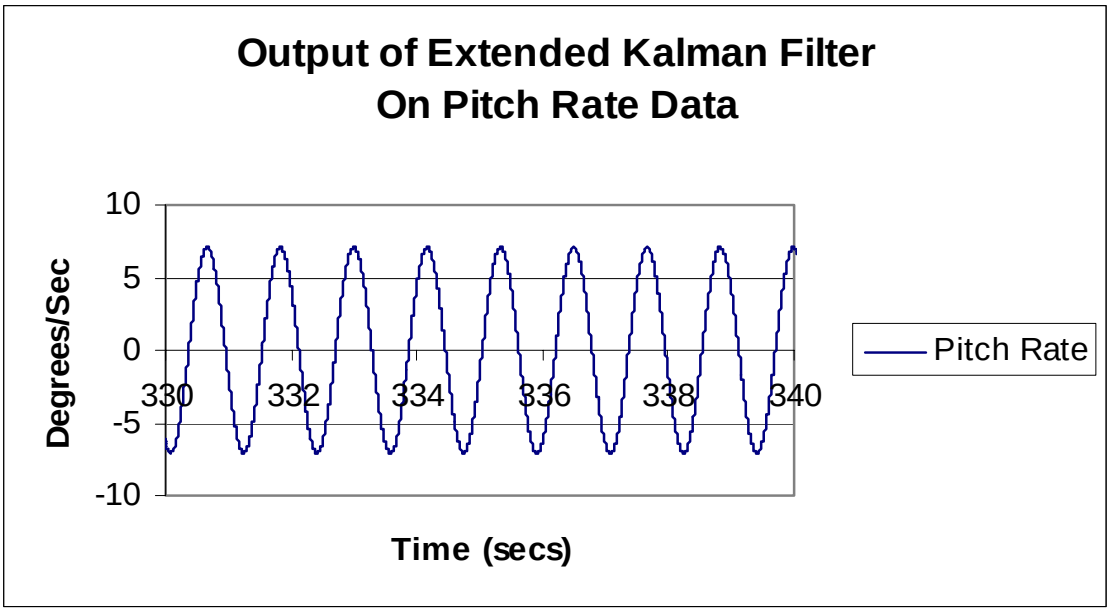


Figure 4. Extended Kalman Filter Removes Noise

The effect of not filtering out the noise on the gyro rate data as it is integrated to arrive at vehicle orientation is shown in the following figure. Figure 5 shows the estimate of the vehicle's pitch angle using the threshold technique. The noise spikes that remain in the rate data are integrated to yield errors in the pitch angle. The "truth" is that the vehicle is coning about a fixed orientation in inertial space whereas the output using the threshold technique shows noticeable errors.

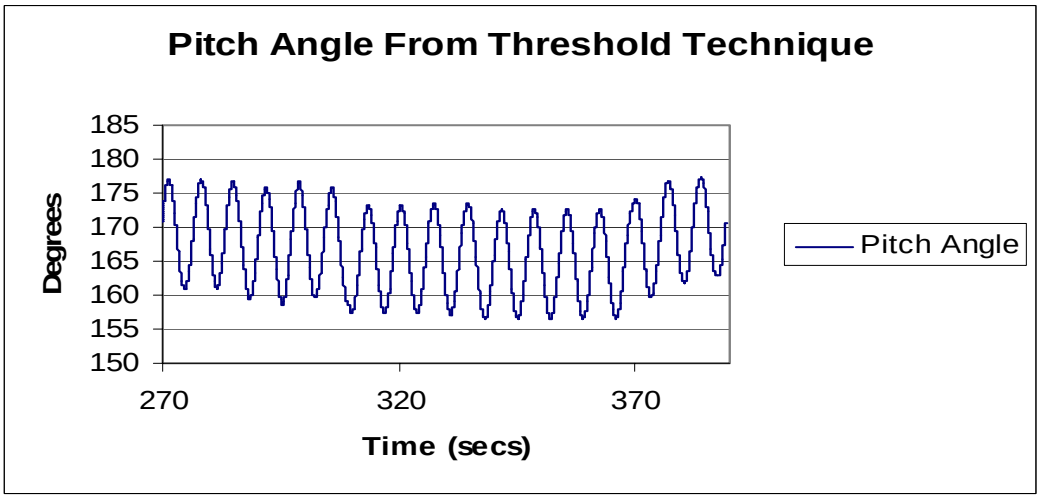


Figure 5. Pitch Angle As Determined By Threshold Technique

Since the extended Kalman filter does a better job at removing the noise spikes, the resulting orientation angles are smoother as well. However, the extended Kalman filter implementation also makes use of an additional set of measurements, the simulated magnetometer data. These measurements are in a geodetic or earth-fixed frame and the results must be transformed to an inertial frame for comparison with the results shown above. A set of transformation equations was written to do that. The resulting pitch angle estimate, in inertial coordinates, from the extended Kalman filter is shown in Figure 6.

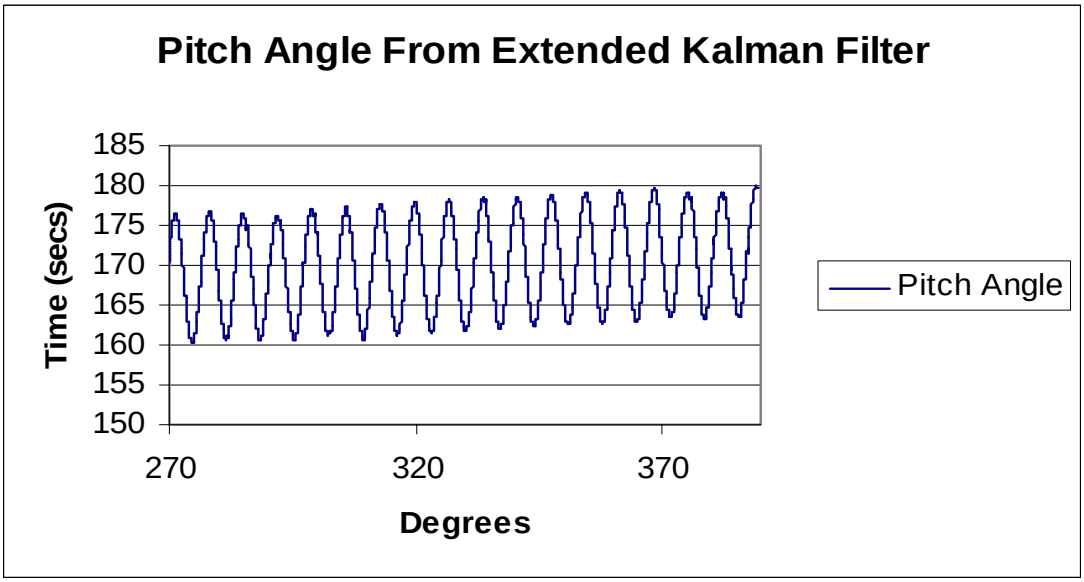


Figure 6. Pitch Angle From Extended Kalman Filter

As can be seen the errors seen in the threshold technique output have been removed, but there is a slow drift of the nominal angle away from the true value. The same effect is seen in the yaw angle result. It has not been determined if the drift is due to an error in the transformation from geodetic to inertial coordinates or an error in the equations incorporating the magnetometer measurements into the extended Kalman filter.

CONCLUSION

In summary, this LDRD illustrated that the addition of a Kalman filter and magnetometer measurements to the current approach for processing telemetry data results in an improved, less noisy estimate of the vehicle's angular rates and orientation. Work still needs to be done to investigate the reason for the slow drift in the estimates of pitch and yaw angles. We would also like to explore the incorporation of additional data sources such as sun sensors, star trackers, and GPS into the Kalman filter algorithm.

Distribution:

2	MS 0188	Donna L. Chavez, 1030
2	MS 0501	Todd E. Owen, 2338
2	MS 0825	Beverly Sturgis, 9115
2	MS 0986	Mark Platzbecker, 2664
2	MS 0986	Gary Ashcraft, 2662
1	MS 0986	Robert J. Longoria, 2663
1	MS 0986	Vincent P. Salazar, 2660
1	MS 0986	Lorraine S. Baca, 2661
1	MS 0986	Michael E. Partridge, 2665
1	MS 0986	Larry J. Dalton, 2662
1	MS 0986	Stephen C. Roehrig, 2664
1	MS 0986	Jeremy Giron, 2664
1	MS 0986	Dave R. Sandison, 2666
1	MS 9018	Central Technical Files, 8945-1
2	MS 0899	Technical Library, 9616
1	MS 0612	Review & Approval Desk, 9612 for DOE/OSTI