

# Development of a Payload Derived Position Acquisition System for Parachute Recovery Systems

Ryan D. Tiaden<sup>1</sup>

*U.S. Army Yuma Proving Ground, Yuma, AZ, 85365*

Oleg A. Yakimenko<sup>2</sup>

*Naval Postgraduate School, Monterey, CA, 93943*

**For parachute recovery systems (PRS) there is a requirement, for testing and operational use, to know the entire trajectory of the PRS. For testing, the trajectory is required to understand the opening characteristics and the flight performance of the PRS. For operational use, the trajectory information is utilized in real-time for the guidance and control of precision systems. Currently, there are certain limitations in how these trajectories are generated. The paper advocates the development of a Payload Derived Position Acquisition System (PDPAS) to overcome these problems. The PDPAS is an instrumentation set and software algorithm that is to be installed onto PRS in order to estimate PRS state vector parameters in real-time for testing and operational use. Ideally it needs to be done without continuous use of the Differential Global Positioning System receiver and should produce a six degree-of-freedom solution for the PRS's trajectory from aircraft exit to ground impact. The paper discusses the details of developed algorithms, results of computer simulations and processing of real drop data.**

## Nomenclature

CDS	=	Container Delivery System
DGPS	=	Differential Global Positioning System
DOF	=	Degrees of Freedom
GPS	=	Global Positioning System
HAHO	=	High Altitude High Opening
IC	=	Initial Condition
IMU	=	Inertial Measuring Unit
INS	=	Inertial Navigation System
KTM	=	Kineto Tracking Mount
PDPAS	=	Payload Derived Position Acquisition System
PGPRS	=	Precision Guided Parachute Recovery System
PRS	=	Parachute Recovery System

## I. Introduction

**T**HE Parachute Recovery Systems (PRS) trajectory information is a crucial element of any testing program. This information is used to characterize the PRS opening and flight performance, and to provide real-time situational awareness of PRS location. For operational utilization, PRS trajectory information is used in the guidance and control algorithms of precision guided systems.

Payload trajectories for testing applications are currently generated using a series of optical ground tracking stations, called Kineto-Tracking Mounts (KTM), which independently track the payloads from aircraft exit to ground impact, and/or utilizing Global Positioning System (GPS) receivers if available onboard the PRS. Both methods however have certain limitation. The primary limitations of optical ground stations include the following:

---

<sup>1</sup> Test Director / Aerospace Engineer, Air Delivery and Soldier Systems, U.S. Army Yuma Proving Ground, ATTN: TEDT-YPY-AVD, ryan.tiaden@us.army.mil, AIAA General Member.

<sup>2</sup> Research Associate Professor, Department of Mechanical and Astronautical Engineering, Code MAE/YK, oayakime@nps.edu, AIAA Associate Fellow.

KTMs track only one PRS at a time, require significant manpower resources and time to process data. Its usage is cost prohibitive to many testing programs, and is a limited material resource. In addition, most Precision Guided PRS (PGPRS) utilize GPS systems for real-time operational trajectories anyway. On the other hand, the GPS navigation system installed on top of the payload starts estimation parameters of the descent trajectory only after about 30 seconds after aircraft exit and is highly susceptible to GPS jamming, especially in the combat environment.

Therefore, the Payload Derived Position Acquisition System (PDPAS) is a solution developed to overcome the limitations of optical tracking and GPS usage for the generation of PRS trajectory information. The PDPAS is a system installed onto a PRS, which contains an instrumentation set and software algorithm that generates the trajectory information. When fully developed and implemented it should be initialized by aircraft data (through 1553 bus transmitter and a GPS signal re-broadcaster in the aircraft cargo bay) and produce an estimate of a descent trajectory from aircraft exit to ground impact without GPS using an Inertial Measuring Unit (IMU) only. The current design however does employ the GPS system to be able to tune algorithms and compare the produced Inertial Navigation System (INS) solution to.

Currently the entire process consists of three steps. The first step is the initialization of the INS with the initial conditions (IC) on Euler angles and position. The second portion of the algorithm generates the 6 Degree-of-Freedom (6DoF) solution from the IMU sensor data only. Finally, the estimates of PRS's position are updated using GPS data, when available, to correct the IMU sensors induced drifts in position and attitude.

The developed algorithm was first tested utilizing simulated sensor data, so that it could be refined against a known solution. Not only the simulated sensor data enabled refinement of the algorithm, but it also helped in understanding the limitations of the developed software. Subsequent to utilizing simulated sensor data, actual data from the Container Delivery System testing was utilized to test PDPAS software.

One of the advantages the PDPAS is that it could increase the speed in which test data could be collected. That reduces the overall cost of each trajectory test point. By reducing cost and testing time, additional test items will be able to collect this type of critical data to support their system's development. The PDPAS could also improve PGPRS performance. These systems with their control algorithms currently heavily dependent on GPS would definitely fail without it. Since PGPRS are typically used to provide cargo to personnel in forward combat areas, their reliability will be increased by reducing the susceptibility of a PGPRS to GPS jamming. Also, as PGPRS achieves a payload capability of 10,000 to 60,000 lb, the reliability of each drop becomes more important because of the increased cost of each payload. Finally, with PDPAS in place, the interactions between multiple PRS could be investigated.

This paper is organized as follows. Section II describes the types of PRS that would utilize PDPAS. Section III discusses the concept of PDPAS operation, followed by Section IV which reviews the mathematical model and processes used in the software algorithms. Section V shows the results of PDPAS data collected from a real PRS drop. The paper ends with conclusions and recommendations.

## II. Background

PRS trajectory data is primarily utilized in two functions. The first function is for testing of a new PRS. In testing, position trajectory data is utilized to characterize a system's performance at canopy opening, during flight, and upon landing. The attitude data is utilized to characterize the stability of a system and the effect of the wind on flight performance. The position and attitude data are critical in evaluating the suitability, performance, and safety of systems prior to being fielding, and can also be utilized in real-time to monitor the safety of range operations. The second area where PRS data are utilized is in the real-time control algorithms of PGPRS. The current technology utilized in the PGPRS is primarily GPS derived, which provides only position data to the control algorithm.

There are three primary types of PRS; personnel, unguided cargo, and precision guided cargo. Figures 1 through 3 are typical representations of these primary parachute types. For all three major types of PRS, trajectory data is required in the testing of these systems. Real-time data is utilized in High Altitude High Open (HAHO) PRS by providing canopy control information to the jumper. For PGPRS, the data is utilized to



**Figure 1. Airborne Systems Inc. – Megafly precision guided system**

control the system, which provides the precision performance of the system. Information regarding PRS types and their engineering is found in Ref. 1.

During PRS testing, the KTM is an optical manned ground station that tracks a payload at the U.S. Army Yuma Proving Ground. Each KTM's position is known and the slew and rise of each camera on the KTM is captured and recorded on the video of the drop. When three or more of these stations are used to track a payload during a drop, the data can be processed to accurately solve for the position of the payload through the geometry of the KTM locations. From this processing, attitude data for the payload can also be calculated. Detailed information of the algorithms used for processing the KTM data can be found in Ref. 2 through 4.

There are several limitations to the utilization of KTM derived data. The first limitation is that KTMs are critical range resources that are in high demand by multiple test programs. A second shortcoming is the cost to collect the KTM data. Each mount requires two personnel to operate and four mounts are utilized to generate the trajectory data. A third limitation is the extensive and time consuming data post-processing that is required to generate an accurate data solution. A final limitation (possibly the most restrictive), is that only one test item can be track at a time due to the number of existing KTMs.

Differential Global Positioning System (DGPS) systems are extensively utilized in test programs for data acquisition and are primarily utilized on PGPRS to provide position data to the control algorithms. One of the key advantages of DGPS systems is that they operate on their own once activated. This allows test programs to utilize as many systems as needed. However, there are several key limitations to data generated from DGPS.

The first limitation is that the systems lose GPS lock upon exit until about 30 seconds after aircraft exit. This loss of data is due to the opening shock of the canopy opening, causing saturating the clock oscillators, poor satellite coverage due to system inversion upon aircraft exit, and discontinuities in the solution due to the loss of the data. This loss of DGPS data has been reduced from >60 seconds due to the incorporation of a GPS rebroadcast kit inside of the aircraft, which prevents the system from having a "cold start" after aircraft exit. For PGPRS, the 30 second loss of data is not significant because the system will have control authority once DGPS data is re-acquired, but for testing programs, this 30 second window is critical for capturing dynamic information during the canopy opening sequence. So, for programs that require canopy opening performance data, KTM derived data is currently the only data source. Another limitation for PGPRS is that they will be utilized in hostile environments. One typical threat in a hostile environment is jamming of the GPS signal. When a current PGPRS loses its GPS data, it can not reach its desired target.



**Figure 2. G-12E parachute w/ Container Delivery System**



**Figure 3. MC-6 personnel parachute system**

### **III. PDPAS Concept**

The PDPAS concept is designed to overcome the limitations of the KTM derived solution for testing and to provide improved robustness to the design of the PGPRS. To overcome these limitations and provide a robust design, the trajectory data generated must only be from sensors on the payload during the drop, which will provide real-time data for the PGPRS and allow transmission of the trajectory data off of the payload. Having the sensors contained on the payload will allow any number of systems to be instrumented on a drop during a test.

The suggested solution to address these limitations for the PDPAS is a "strap down" Inertial Measuring Unit (IMU) and GPS system. An IMU is a sensor that provides at least three orthogonal accelerometers and three orthogonal rate gyros. Using the IMU data, with initial conditions for the position and Euler angles of the IMU, the trajectory data can be integrated. This IMU generated trajectory data provides the capability to fill in position trajectory data when the GPS solution is lost. Also, the IMU generated trajectory data will provide additional information about the attitude of the PRS throughout the drop, which is an increase in valuable data for testers and

PGPRS algorithms. The product of this sensor combination will be a 6DOF solution from aircraft exit to ground impact. Information regarding the integration of IMU data and GPS can be found in Ref. 5 and 6.

In addition to the IMU and GPS data on-board the payload, additional sensors on the aircraft are utilized to generate the initial conditions for the integration of the trajectory solution. The sensors on-board the aircraft are a 1553 Bus data recorder, GPS re-broadcast kit, and tilt sensors. The 1553 Bus provides a wealth of information regarding the position, velocity, attitude of the aircraft, and general aircraft conditions. The GPS re-broadcast kit provides the GPS satellite signal from the aircraft GPS antenna inside of the aircraft's cargo bay. The PDPAS utilizes the re-broadcast signal to acquire a GPS lock prior to aircraft exit, which eliminates the need for ephemeris and almanac data to be reloaded when the GPS signal is reacquired after aircraft exit. The tilt sensors provide more aircraft and payload attitude data, since the attitude data captured from the 1553 Bus is coarse.

Figure 4 shows that the IMU data alone will be used to fill in the position trajectory data until the GPS data can be incorporated. Figure 5 shows a graphical depiction of where the position and attitude data elements are generated from during the PRS drop.

Figure 6 depicts the data and processing flow for the PDPAS for a testing application. An advantage of this processing flow is that trajectory data processing does not necessarily need to be completed in real-time, since all of the data can be recorded. This provides the opportunity to utilize additional data resources to improve the accuracy of the trajectory data over a real-time solution. The increase in accuracy is important because in a test environment the control of error is critical to providing quality system analysis.

Figure 6 shows that the 1553 Bus data, IMU data, GPS data, Clinometer sensor data, and onboard measurements, and time of events captured from video are all inputs into the PDPAS processing algorithm. The output from this algorithm is the 6DOF solution that can be utilized for system analysis

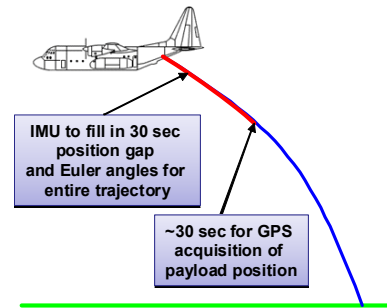


Figure 4. PDPAS position solution throughout PRS trajectory

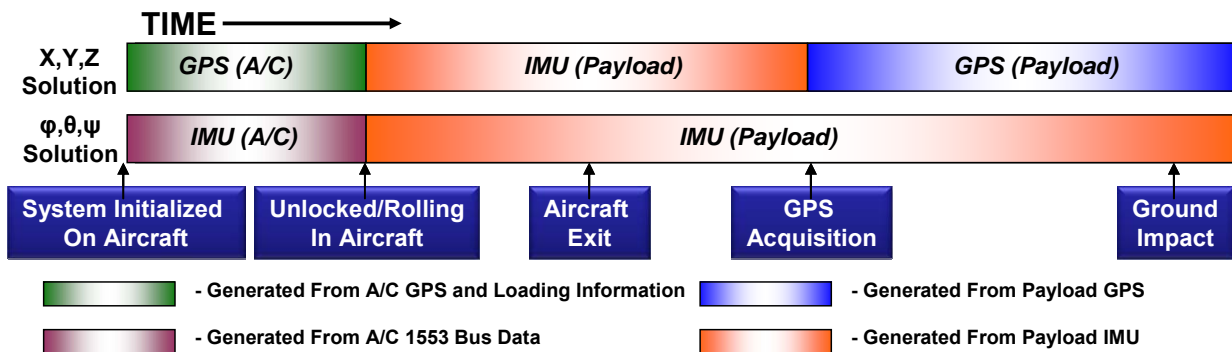
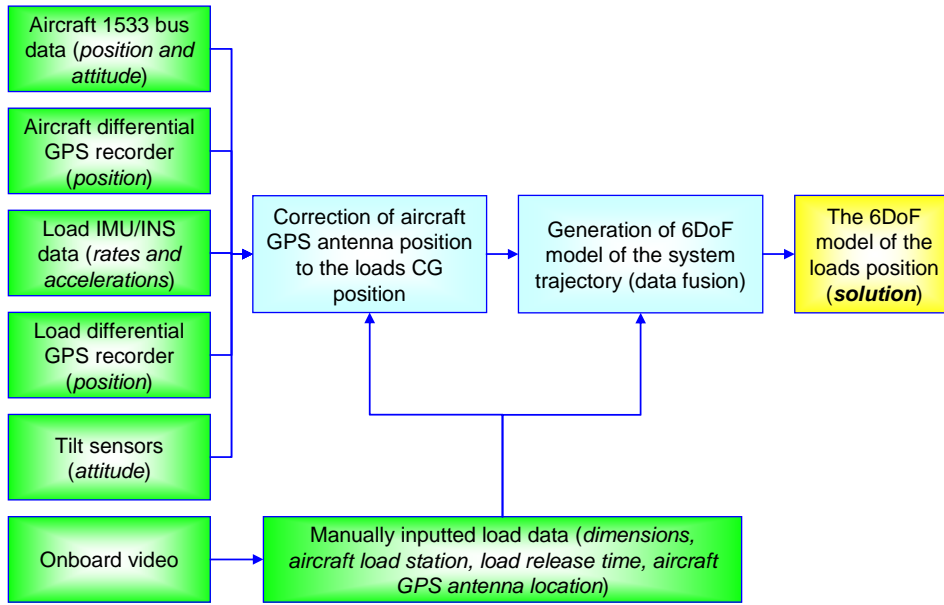


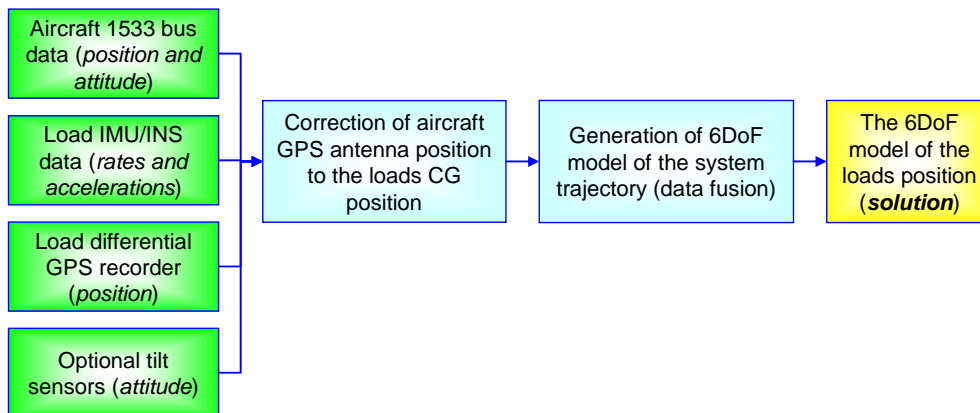
Figure 5. PDPAS position solution throughout PRS trajectory as a function of time



**Figure 6. PDPAS data and processing flow diagram (testing application)**

Figure 7 shows a data and processing flow diagram for the PDPAS for an operational application. A critical part of an operational application is that the data is provided in real-time, since the data is irrelevant after the drop is completed. An important aspect for operational applications is that the integrated solution of the trajectory does not diverge, since only a small error (~15 m) in the position of the payload is acceptable. A diverging solution can happen if the quality of the IMU data is low, the initial conditions are not correct, or when the GPS position data is lost. The diverging solution due to low IMU quality and the poor initial conditions is a result of the INS reference axis diverging from the true IMU axis, and without GPS the INS reference axis cannot be corrected to the true IMU axis. A ~15 m error in position is acceptable because this is the quality of the position data from GPS used in an operational environment. The choice of sensors and the configuration of the PDPAS hardware is a critical feature that is necessary for meeting the needs of an operational application.

Figure 7 shows that the 1553 Bus data, IMU data, GPS data, and tilt sensor data inputs into the PDPAS processing algorithm. The output from this algorithm is the 6DOF solution that can be utilized for the control algorithm of a PGPRS.



**Figure 7. PDPAS data and processing flow diagram (operational application)**

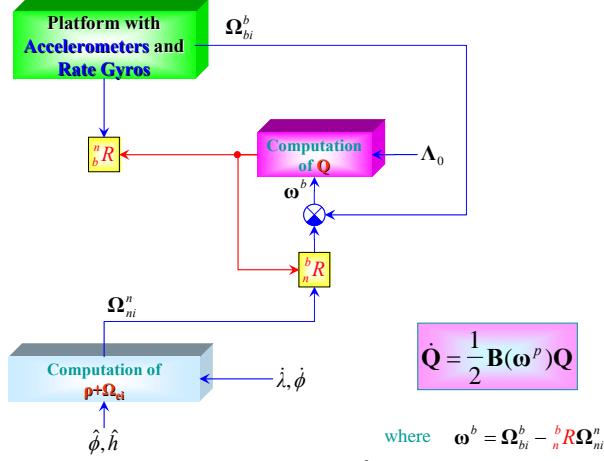


Figure 9. Computation of the rotation matrix<sup>6</sup>

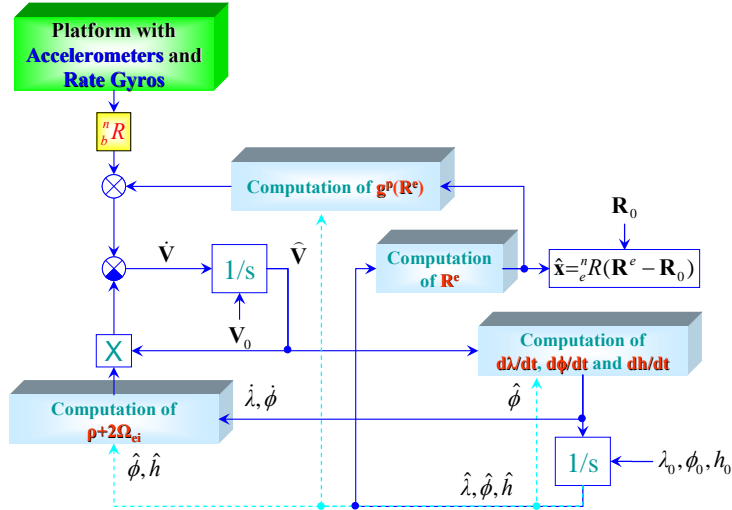


Figure 8. The INS mechanization<sup>6</sup>

## IV. Data Processing

### A. Inertial Navigation System Model Used

The Inertial Navigation System (INS) model used in the PDPAS to generate the position information is depicted in Figure 8. The inputs to the INS model are the accelerations, current Euler angles (generated from the rate gyros), the initial position, local gravity, and other constants. First, the acceleration data is rotated from body frame to the navigation coordinate frame. From the navigation coordinate frame, acceleration data due to gravity and the Coriolis Effect are subtracted to generate true accelerations as seen by the PRS. These true accelerations are then integrated to produce the velocity of the PRS and then integrated again to produce the position of the PRS. The velocity data and the position data is then fed into the calculations for the Coriolis Effect. The INS model for the generation of the attitude data from the rate gyros is depicted in Figure 9, which shows that the Euler angles of the system are the integration of the rate gyros.

The addition of GPS data into the INS model is depicted in Figure 10, which shows that the GPS data is utilized to correct for position errors and velocity of the INS by blending the data when the GPS data is available. The GPS update is important because the velocity and position data from the INS alone has an increasing error with time due to the numerical integration of the INS model. Also, Fig.10 shows that the aircraft bus data is used to initialize the numerical integration of the system.

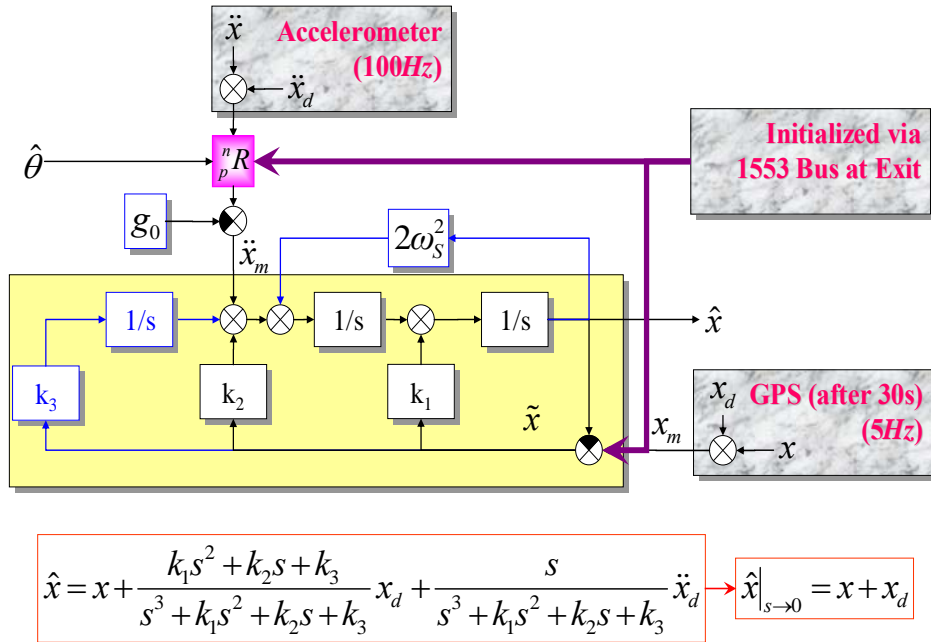


Figure 10. INS position model with GPS data blend

The mathematical modeling described is used in the Matlab® code and Simulink® modeling described to generate a solution from data sets. Matlab® and Simulink® were used due to its ability to model complex systems in a user friendly environment. If this process is utilized in an operational environment, the modeling will need to be completed utilizing a real-time operating environment and software. The INS block top level functions are shown in Figure 11.

## Strapdown Mechanization INS

Calculations are carried in {NED}

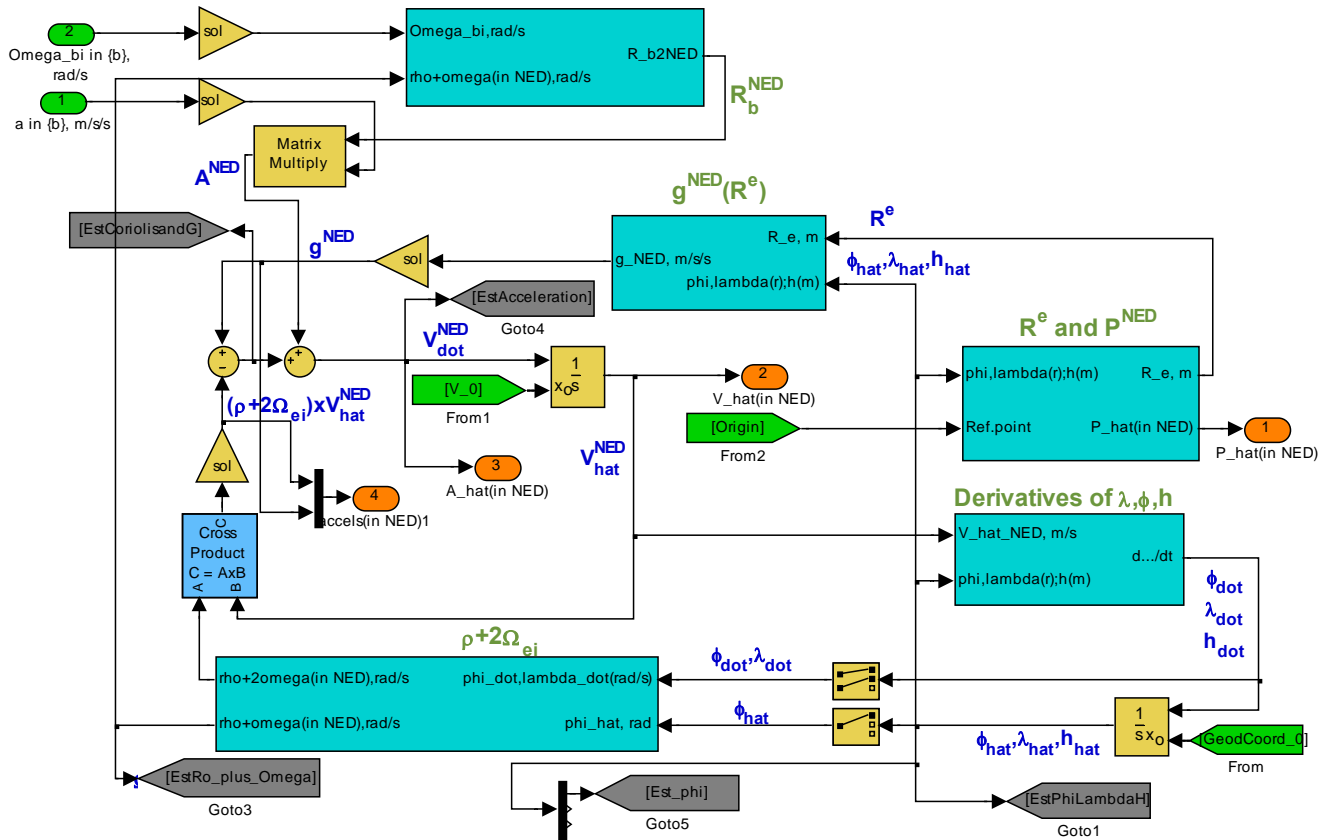


Figure 11. INS Simulink® model



## B. Matlab® Script

The Matlab® script developed for the PDPAS follows the data and processing flow diagram depicted in Figure 12. The first thing the PDPAS script requires is what subroutines the user wants to execute, which provides the user the opportunity to add truth source data for comparison purposes. The time of first motion, when a PRS first starts to roll out of the aircraft, is also an input by the user so the PDPAS script knows when to initialize the INS solution. Another user input is the location of the LTP origin to be used, which is typically the DZ IP for PRS.

The second step in the script is to load GPS data for the initialization of the INS and for blending with the INS solution. The aircraft 1553 Bus data is then loaded and the initial conditions of the PRS Euler angles are calculated. Loading KTM position and attitude data loaded next is optional, since it is only used in evaluating the INS solution and not in generating it.

The next step in the script is the calculation and correction of IMU biases. From the GPS data, the user selects a time span during which the aircraft was stationary on the ground after PDPAS activation. From this stationary time, the bias of the accelerometers and rate gyros is calculated, which is described in Section E. Following the bias correction, the data is truncated to the time interval of interest. This truncation is done to reduce the memory load of the computer so that the processing time of the INS solution can be reduced. Once the data is truncated, the time domain of all time dependent data is correlated to a time zero location at the first motion of the PRS on the aircraft. The correlation is done because the initial conditions of the INS solution are correlated to first motion and the Simulink® model starts its solution at time equal to zero. Next, the initial conditions for the position and Euler angles are setup, the Simulink® INS model is run. Once the INS model has run, the data is correlated, plotted, and stored for user evaluation.

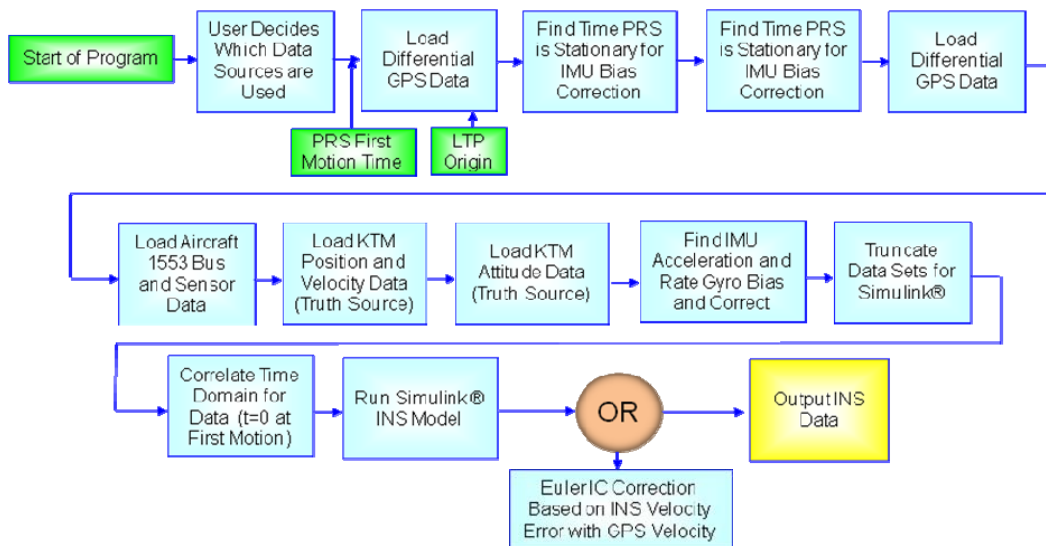


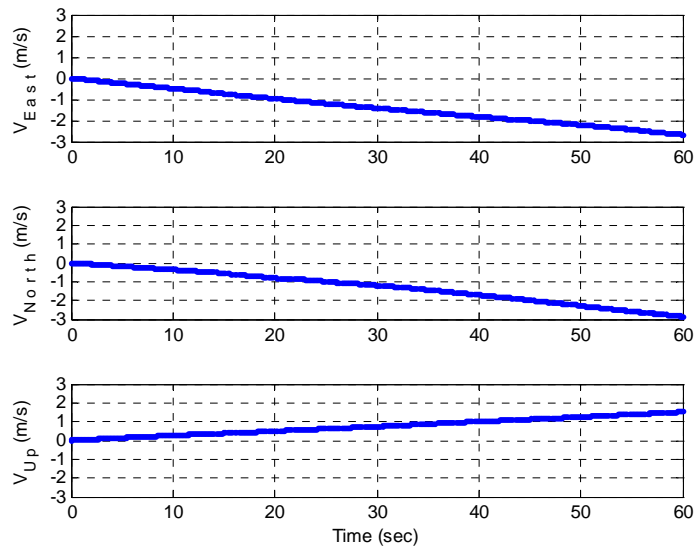
Figure 12. PDPAS data and processing flow diagram (testing application)

### C. Bias Correction

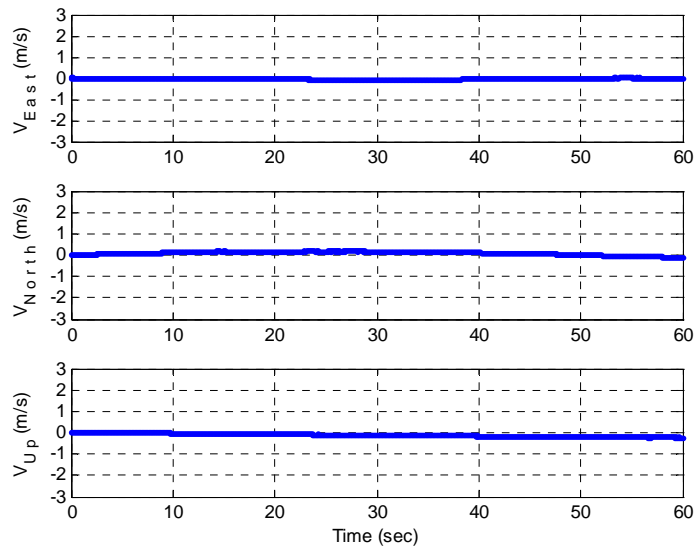
The correction of the system bias is an important part of generating an accurate INS solution. Small differences from the real accelerations and real rotation rates impact the INS model in two ways. The first way the bias impacts the solution is due to the fact that the position, velocity, and Euler angle are integrated from the recorded data, which amplifies small errors. The effect of small integration errors will “walk” the INS solution away from the correct solution. The second way that the bias impacts the INS solution is in the method in which the errors in the calculated Euler angles produce additional errors in the position and velocity terms. At each time step, the Euler angle is used to rotate the acceleration data from the body frame to the navigation frame. When the Euler angle that is used is not correct, the magnitudes of the accelerations are not correctly oriented to the coordinate frame, which causes the integration of the acceleration in the navigation frame to be in error. The effect of errors in the Euler angle is that the calculated data is in the incorrect direction, so the INS solution “wanders” around the true solution.

The bias correction sub-routine in the PDPAS Matlab® script utilizes the Simulink® INS model to calculate the errors. The biases are found during a time span where the IMU is motionless. This time span provides a known solution to the INS model, which is zero acceleration and zero rotation rate. Next, to find the biases for all three acceleration channels, the INS model is run with the rate gyro data set to zero and the initial Euler angles set to zero. This acceleration data run produces an INS solution in which each accelerometer is allowed to generate a velocity solution without interaction from the other accelerometers. When the INS solution is run over a significant time span, the acceleration bias for each of the channels can be calculated from the slope of the velocity data. The calculated acceleration biases are then subtracted from the IMU data. Figure 13 depicts the velocity generated from each accelerometer prior to the subtraction of the acceleration bias and Figure.14 depicts the velocity generated from each accelerometer after the biases are removed. One limitation to the bias correction process is that it only subtracts constant biases, where in fact the bias for each channel is a function of time.

The elimination of the biases for the rate gyro is very similar to the process used for the accelerometers. The key difference from the acceleration bias correction is that



**Figure 13. INS velocity solution with stationary IMU and no bias correction**



**Figure 14. INS velocity solution with stationary IMU and bias correction**

each rate gyro channel's bias is calculated independently because the rotation of one Euler angle will affect the result of another due to coupling. So for each rate gyro, the accelerations are set to zero and the other two rate gyro's data are set to zero for the INS model run. The limitation of linear bias corrections is also true for the rate gyros; the biases corrected for are only constants with time and no higher order corrections are made.

#### D. Euler Angle Alignment Error and Drift

One of the critical inputs to the INS solution for PDPAS is the initial condition (IC) of the Euler angles. From Euler angle IC, the Euler angles are integrated from the rate gyro data. An error in the IC will propagate throughout the INS solution and increase the error of the INS solution. However, these IC are difficult to accurately identify when the INS is initialized just prior to the drop. The IC is calculated from the aircraft attitude data via the 1553 Bus. The IC calculated from the 1553 Bus has a limited accuracy because the IMU is not hard mounted to the aircraft, but to the PRS. Each time that a PRS is rigged, the variation between the aircraft's Euler angles and the PRS will be different. Another issue is for personnel jumps, in which the jumper is not locked into the aircraft, the relationship between the jumper and the aircraft can not be know. In addition, the IMU used in this paper has a significant drift of the Euler angles as a function of time. To compensate for these errors in the IC and drifting of the Euler angle solution, an alternative method of calculating Euler angles was developed.

The method used to calculate Euler angle IC is based only on the IMU and GPS data. The basic premise is that there is only one set of Euler IC that will provide a correct INS solution, and errors in the Euler angle IC will produce errors in the INS solution. Based on the fact that there is only one correct set of Euler angles, the correct Euler angle IC will have the smallest amount of error. The error that is minimized is the difference between the GPS velocity solution and the INS velocity solution. The approach used in PDPAS is to use the MATLAB® 'fminunc' unconstrained optimization algorithm from the optimization toolbox to solve for the Euler angle IC that have the smallest error in all three velocity channels. Below are the steps in the algorithm used in PDPAS to solve for Euler angle IC. Figure 15 depicts the data flow of the Euler IC GPS correction algorithm.

Euler IC GPS Correction Algorithm Steps:

- 1) Generate Euler angle solution based on aircraft data.
- 2) Select region with sufficient GPS coverage.
- 3) Run 'fminunc' of Euler angle IC to find minimum error in the INS velocity.
- 4) Run INS solution forwards with new Euler angle IC.
- 5) Run INS solution backwards with new Euler angle IC if necessary.
- 6) Repeat process for other regions where Euler angle solution is drifting.

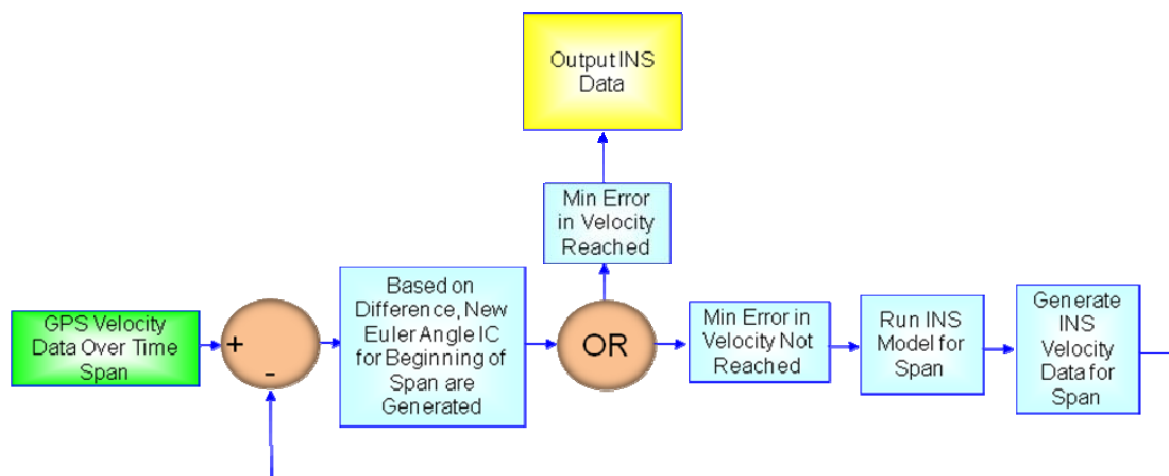
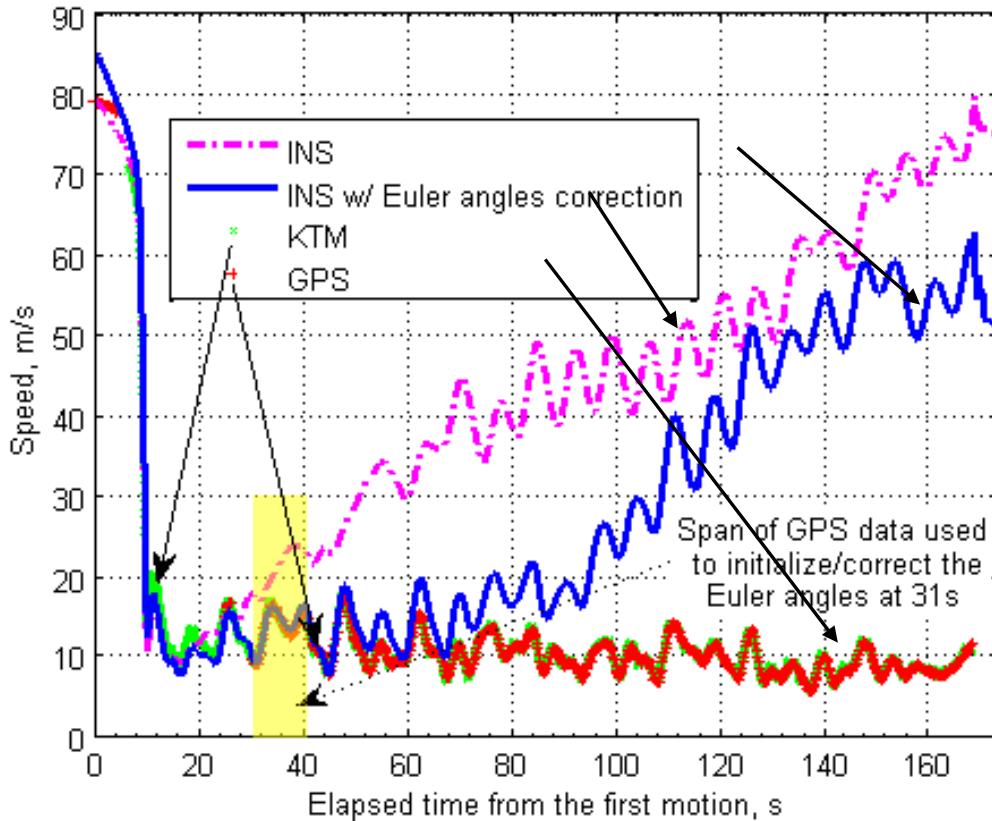


Figure 15. Euler IC GPS correction algorithm

The limitation of the Euler angle correction using the GPS algorithm approach is that the Euler IC can only be calculated when there is GPS coverage, but the system does not need to be at rest. Each Euler IC correction pass can take several minutes to calculate, which is a limitation of this algorithm. This extended processing time is acceptable for post processing, but a more rapid solver and a creative implementation would need to be utilized for operational

applications. However, an advantage of this approach is that the IC can be updated at any time when there is GPS coverage, so the drift of the Euler angles can be corrected during the solution. Figure 16 is a graph of the total velocity of a real PRS, which shows the effect of using the GPS data to correct the Euler angle IC. The Euler angle was corrected at the 31-second mark over a period of 10 seconds. The INS solution was calculated forward and backwards from the 31 second mark. As seen from Figure 16, the solution to the INS is accurate for just beyond the period that was used to correct the Euler angle IC (after which the drift of the IMU data creates errors in the INS solution). Subsequent runs of the Euler angle correction algorithm can be used to correct this drift.



**Figure 16. Effect of using a GPS data span to correct Euler angles at one small span of time on PRS data**

## V. PDPAS Trajectory Solution Using Real PRS Data

### A. INS Solution

The data presented in Figure 17 through 20 incorporates all of the error correction algorithms presented in this paper; bias correction, smoothing, Euler angle update every five seconds after GPS acquisition, velocity update after GPS acquisition, and position update after GPS acquisition. Additionally, each graph presents GPS and KTM truth source data to demonstrate the quality of the PDPAS INS solution. The PRS LTP velocity data is presented in Figure 17, which shows the expected track of horizontal velocity being reduced significantly. Figure 17 also shows the vertical velocity increasing after aircraft exit until the canopy can deploy and reduces the vertical velocity to steady state. Figure 18 shows the same PRS LTP velocity data, but zoomed to the region prior to GPS acquisition (the region with the highest error).

Figure 19 depicts the PRS position data from the same data sources as the previously mentioned velocity data. This data follows as expected; the PRS follows the aircraft track until aircraft exit, where the canopy dissipates the energy from the horizontal motion and the system is affected by the wind. At GPS acquisition there is a jump in the position data, which is due to the update of the position with GPS data.

Figure 20 is a plot of the Euler angles of the PRS, which is not plotted with truth source data because not an accurate source was available this test. There are two lines for each Euler angle. The blue line is the PDPAS INS solution with Euler Angle IC calculated from aircraft data, which is not updated throughout the drop. The red line is the Euler Angle of the system with Euler angle updates every five seconds after GPS acquisition. Euler angle updates with GPS, shown in red, is more accurate because it minimizes the error in the velocity terms, as there is only one set of Euler angle values that can achieve this.

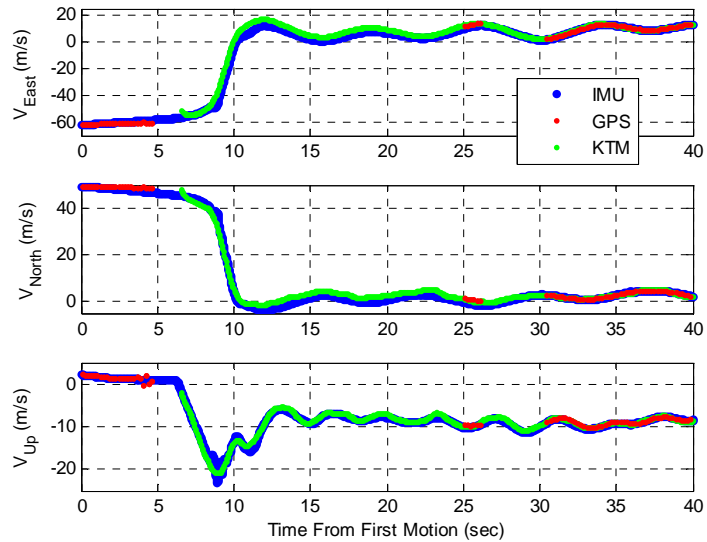


Figure 17. PDPAS velocity solution of a PRS

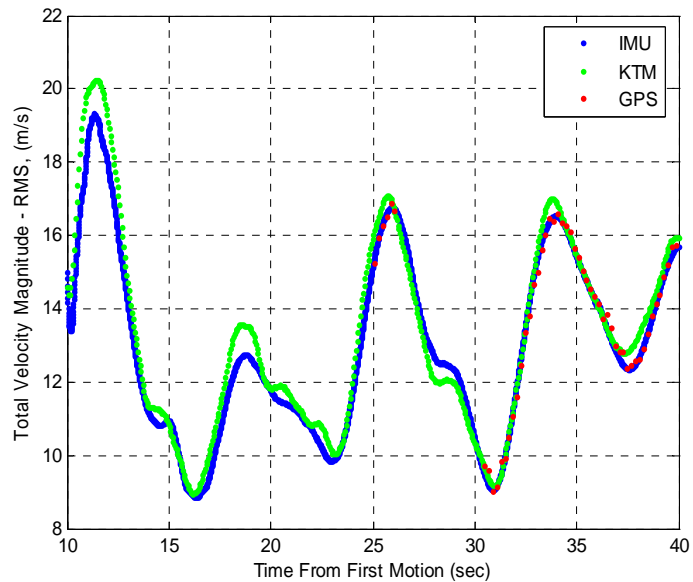
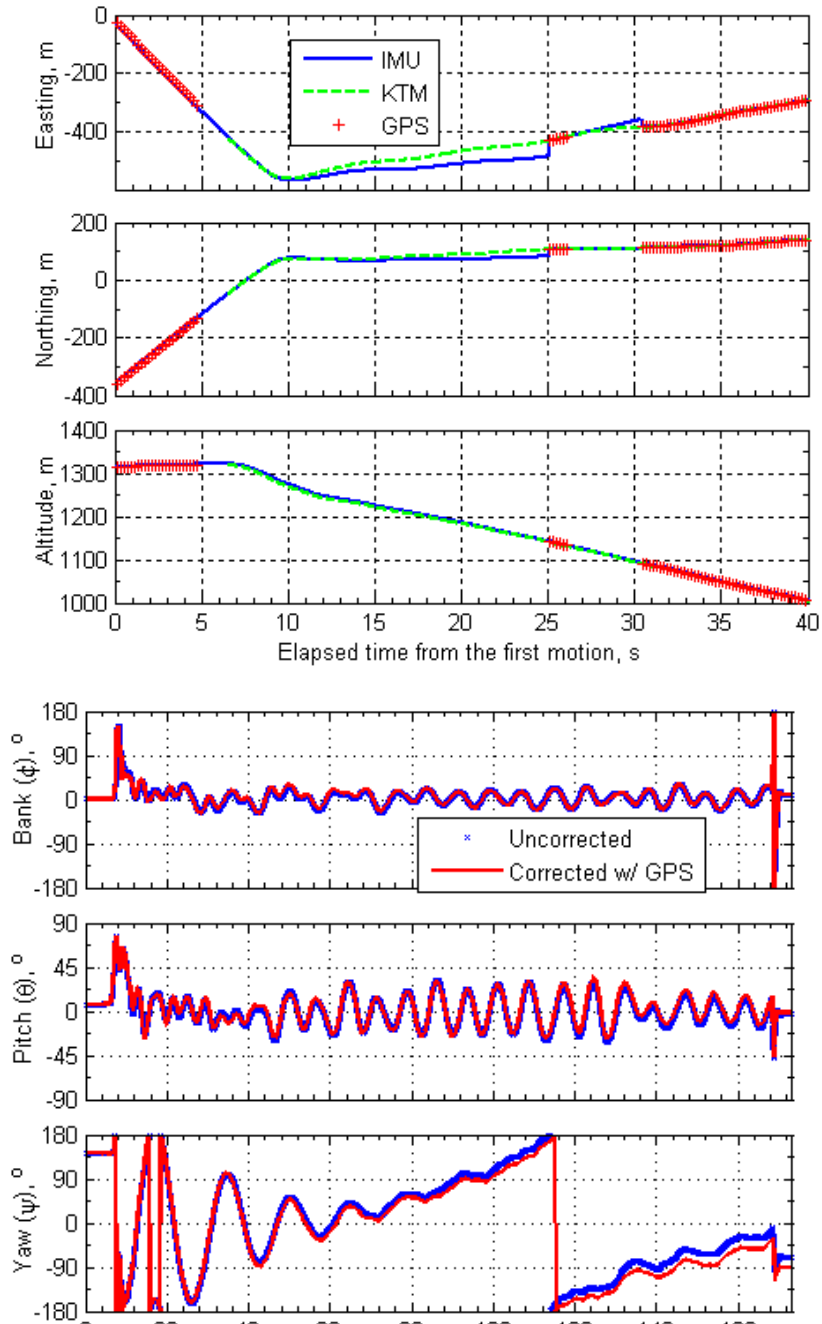


Figure 18. PDPAS velocity solution of a PRS (zoomed in)



## B. Error Analysis

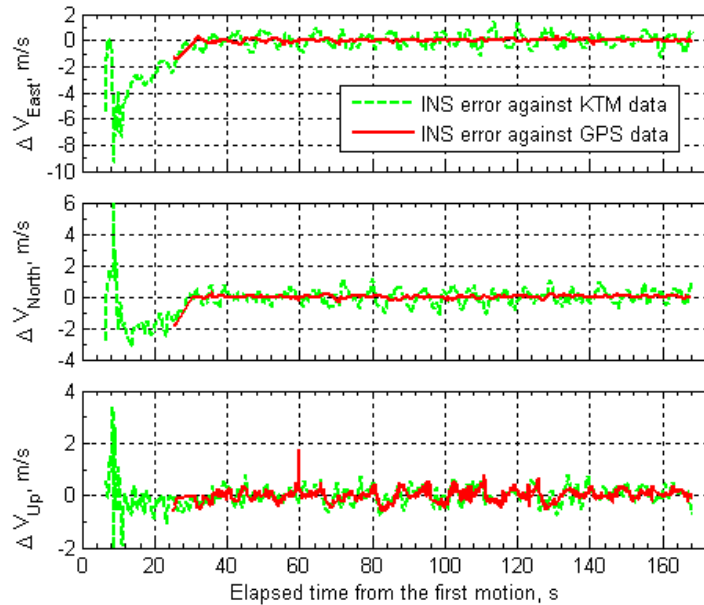
Figure 21 is a plot of the velocity error of the PDPAS INS solution from the truth source differential GPS and KTM data. Figure 22 is a velocity plot from the same data sources as the velocity data. The KTM data collected starts at aircraft exit, where GPS acquisition takes approximately 30 seconds for acquisition. This data does incorporate the blending of GPS data when available, which produces an accurate solution as expected.

The time prior to GPS acquisition is the key area for the PDPAS INS. This region is where the INS solution is only using IMU accelerometer and rate gyro data to generate the solution. During this region, the error in the horizontal channel reaches almost 20 meters and around 6 meters of error in the vertical channel. One known reason for the mentioned errors is an error in the IC of the IMU. IC error comes from two sources; the first is the Euler angle data from the aircraft is at a slow data rate and not of the accuracy to control the IMU solution. The slow data rate from the aircraft could be overcome by utilizing a sensor package on-board the aircraft. However, the GPS Euler angle correction algorithm could be utilized to generate the IC of the PDPAS INS. This alternative generation

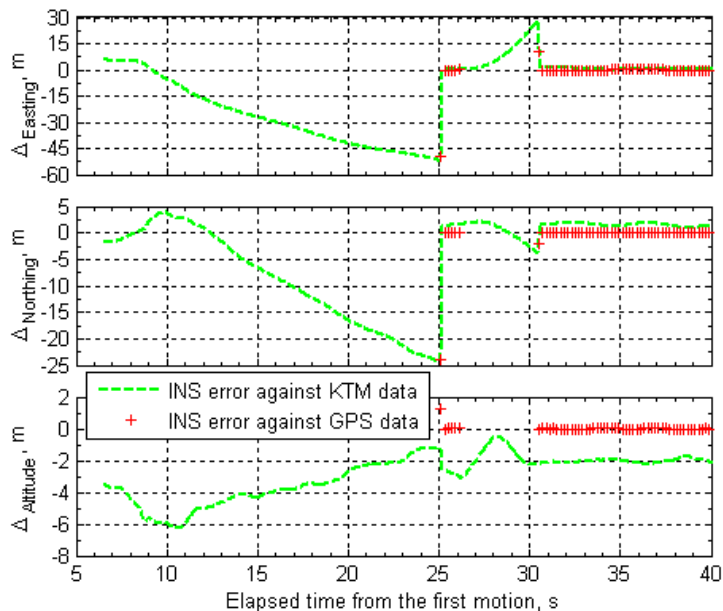
of IC is attractive because it is totally independent of aircraft data, so only the IMU and GPS receiver on-board the load would be necessary for the generation of the INS solution.

The second source of IC error is the actual position of the payload on-board the aircraft. The payload position is not accurately known because the position used is the aircraft GPS antenna location, which is well above the PRS on the aircraft and has a horizontal separation based on the PRS location in the cargo bay of the aircraft. The GPS error could be overcome by knowing the PRS load station in the aircraft relative to the GPS antenna location and translating the GPS antenna location to the PRS location. Utilizing load station data is a good approach for an operational system where data is needed in real-time, but there is an alternative approach for data that is not needed in real-time. This alternative approach is to calculate the Euler angle during the flight from the GPS Euler angle correction algorithm. The position data from this alternative approach would be from the PRS GPS, which would not require any corrections. The data prior to the generated IC would be generated by solving the PDPAS INS solution in reverse, which is partially done in the solution presented.

The primary location of error in the PDPAS INS solution is the divergence of the integrated rotation rate to produce the Euler angle of the PRS. Any small errors of even less than one degree have a significant impact on the quality of the solution. The PDPAS INS tries to compensate for the drift errors and does for most of the linear bias, but the bias for the IMU used is very inconsistent and non-linear over very small times of five to ten seconds. Figure 23 shows the difference between the INS solution utilizing the GPS Euler angle correction algorithm every five seconds after GPS and the INS solution without Euler angle corrections. The level of error in the solution shows that the rate gyro error is significant and the position and velocity solutions will diverge within five seconds after the INS is initialized. A higher quality IMU would produce a better solution for the Euler angle with much smaller position and velocity errors. The IMU used in future versions of the PDPAS should incorporate an IMU that has smaller bias and non-linear error in its rate gyro data.

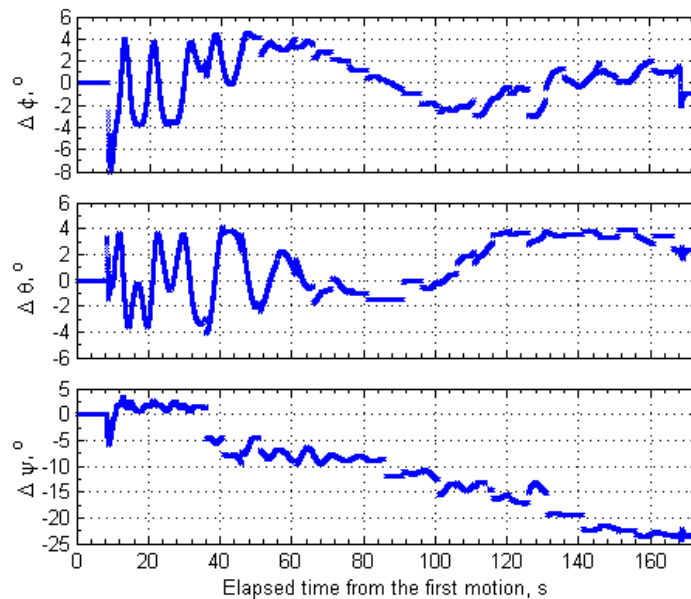


**Figure 21. PDPAS velocity error for a PRS**



**Figure 22. PDPAS position error for a PRS**





**Figure 23. Euler angle calculation difference using the Euler IC GPS correction algorithm for a PRS**

## VI. Conclusion

The paper addressed the development of the Payload Derived Position Acquisition System to obtain the descending PRS trajectory information. Incorporation of the PDPAS would significantly improve an operational system's performance and data collection required to support PRS developmental testing. This research proved that the primary source of errors in the suggested PDPAS scheme is the IMU sensors. Specifically, the IMU utilized in current configuration of the PDPAS when run without GPS updates failed to provide data at the required level of accuracy. Hence, further developments should be focused on the selection of an IMU that meets the need for accuracy of the solution for the duration of time when the system will not have GPS updates. Nevertheless, even this not very accurate IMU allowed to build and test the prototype of the future PDPAS. The concept was proved and all necessary algorithms were developed. These algorithms included modeling a strapdown mechanization of INS using a quaternion. Several efforts were dedicated to account for imperfection of IMU programmatically. Another major development included initialization of PDPAS angular position based on comparison of velocity data provided by GPS and INS. The Euler angle IC provided by this algorithm proved to be very accurate due to the minimization of the error processes utilized. If sufficient computing power is available onboard (to run a minimization procedure) the PDPAS algorithm currently tested off-line could be implemented for in real-time as well. In this latter case, the PDPAS could be developed to incorporate only the IMU, GPS receiver, and processor, which would reduce the PDPAS reliance on aircraft data sources. Without constraints from external sources, the PDPAS could be utilized on other systems where a 6DoF solution is needed during times where the GPS solution is lost.

## References

- <sup>1</sup>T. W. Knacke, *Parachute Recovery Systems Design*, Santa Barbra, CA: Para Publishing, First Edition, 1992.
- <sup>2</sup>R. Berlind, "Autonomous Video Scoring of Air Delivery Payloads," M.S. Thesis, Naval Postgraduate School, Monterey, CA, March 2006.
- <sup>3</sup>O. Yakimenko, V. Dobrokhodov, and I. Kaminer, "Autonomous Video Scoring and Dynamic Attitude Measurement," in *18<sup>th</sup> AIAA Aerodynamic Decelerator Systems Technology Conference and Seminar*, 2005.
- <sup>4</sup>O. Yakimenko, R. Berlind, and C. Albright, "Status on Video Data Reduction and Air Delivery Payload Pose Estimation," in *19<sup>th</sup> AIAA Aerodynamic Decelerator Systems Technology Conference and Seminar*, 2007.
- <sup>5</sup>R. M. Rogers, *Applied Mathematics in Integrated Navigation Systems*, Reston, VA: American Institute of Aeronautics and Astronautics, 2000.
- <sup>6</sup>O. Yakimenko, "Marine Navigation," Class Notes for ME4821, Naval Postgraduate School, Monterey, CA, 2006.
- <sup>7</sup>O. Yakimenko, "Introduction to Engineering System Dynamics," Class Notes for ME2801, Naval Postgraduate School, Monterey, CA 2006.



<sup>8</sup> O. Yakimenko, "Introduction to Digital Computations," Class Notes for AE2440, Naval Postgraduate School, Monterey, CA 2006

<sup>9</sup>"Department of Defense World Geodetic System 1984, Its Definition and Relationship with Local Geodetic Systems." National Mapping and Imagery Agency, Technical Report TR8350.2, July 1997.

<sup>10</sup> R. Tiaden, "Test Plan: Payload Derived Position Acquisition System," U.S. Army Yuma Proving Ground, Yuma, AZ, March 2007.

<sup>11</sup> R. Tiaden, "Payload Derived Position Acquisition System for Parachute Recovery Systems," M.S. Thesis, Naval Postgraduate School, Monterey, CA, December 2007.