Defence Research and Development Canada / Recherche et développement pour la défense Canada

MEMS/GPS Kalman Filter

Dale Arden

This work was completed in August 2004 and published in May 2007

Defence R&D Canada – Ottawa

CONTRACT REPORT

DRDC Ottawa CR 2007-093

May 2007

Canada

MEMS/GPS Kalman Filter

Dale Arden

Prepared By:

Dale Arden Consulting
RR3 Almonte ON K0A 1A0

DRDC Ottawa Contract W7714-010525
Contractor Document Number: DAC-0402

CSA: J. S. Bird, CNEW Section

This work was completed in August 2004 and published in May 2007

Defence R&D Canada – Ottawa

Contract Report

DRDC Ottawa CR 2007-093

May 2007

Table of Contents

This page intentionally left blank

1 INTRODUCTION

The contract under which this report has been prepared is entitled "*MEMS/GPS Integration Investigation for NAVWAR Applications*". Its purpose is to "research, design and build a working prototype of an integrated MEMS/GPS system". GPS has become a dominant navigation system for defence applications. However, GPS is highly susceptible to jamming (due to its low power signal) and signal loss (due its requirement for line-of-sight satellite visibility). A missile or aircraft approaching a target could very well experience increasing defensive GPS jamming noise just as its need for accuracy is increasing. A soldier fighting in a canyon – man-made or natural – will have the usefulness of his GPS navigator compromised or negated by the lack of visible satellites. NAVWAR aims to mitigate these vulnerabilities by improving GPS operation in high jamming, high multipath and low signal strength environments like these. Efforts will be directed in a number of different directions.

Integration of an Inertial Navigation System (INS) with GPS has been shown to mitigate NAVWAR problems: the INS is able to bridge GPS outages (and improve GPS signal tracking in tightly coupled systems); GPS is able to calibrate INS sensor errors (thus improving performance during GPS outages). However, traditional inertial systems are too expensive, too big and too complex to be practical on any but the costliest military platforms. A GPS-based navigation system that could be carried by a soldier would have to light and robust; a system fitted into a large number of land vehicles would have to be small and inexpensive; a system mounted in an artillery shell would have to survive 1000's of g's acceleration when fired and be capable of providing the required accuracy after decades of storage. MEMS inertial sensors are tiny, inexpensive and robust. However, at present, they are also very inaccurate. A MEMS/GPS system has the potential provide at least some of the benefits of INS/GPS integration while making robust navigation practically available to a much larger portion of the military in the field.

This report describes important parts of the Kalman filter that is used to optimally combine data from different navigation sensors in a way that will provide the best navigation solution in all situations. The implementation of this filter is based on earlier work: the Dual Inertial Integrated Navigation System (DIINS). Details of the DIINS Kalman filter are presented in reference [1]. Unless otherwise stated, the theory, algorithms, techniques and models applied to the MEMS/GPS Kalman filter are the same as those described in ibid.

Rather than repeating all these background details, this report was written as a kind of addendum to reference [1]. That is not to say that the differences are insignificant: the MEMS/GPS filter has completely different target applications; filter timing has changed from clock timed to data driven; there are new sensors, and new non-sensor measurements; there is a complementary IMU strapdown navigator running with the filter in a closed-loop fashion; and new IMU error states were added to try to better model the larger MEMS IMU errors. The main focus of the sensor integration has also been modified: in DIINS, the emphasis was on failure detection, isolation and reconfiguration; in the MEMS/GPS system, the emphasis is on maintaining navigation accuracy during periods of GPS loss or degradation.

The rest of this report describes the new or modified procedures.

2 FILTER TIMING

The predecessor to the MEMS/GPS Kalman filter was clock timed: functions were signalled to begin at regular preset clock time intervals. The MEMS/GPS Kalman filter is data driven: filter functions will be run as sensor data arrives. This chapter describes the methods to be used to control Kalman filter timing on the basis of sensor data arrivals.

2.1 Reading Sensor Data

Since sensor timing drives this new filter timing approach, it is instructive to fully understand how sensor data is read into the filter task.

The basis of the procedures described below is the assumption that the IMU data rate will be no less than that from any other sensor. This is a fairly safe assumption: at the present time, IMU data rates are one to two orders of magnitude higher than other common sensors (e.g. 100 Hz versus 1 Hz). The IMU strapdown navigator sends data to the filter at a selectable rate that can be as high the IMU data rate or can be reduced by integer fractions of the IMU rate. For example, currently used IMUs output data at rates between 100 Hz and 200 Hz, GPS and digital compass data is available at 1 Hz, and the strapdown navigator outputs data at 10 Hz.

All sensor data is sent to bounded (circular) buffers. At every processing cycle of the Kalman filter, the IMU data record is read from its buffer.

Since the other sensors' data is arriving at slower rates, they will not be read at every filter cycle. To determine whether or not a sensor data record must be extracted, its next expected time of data is computed and tested against the latest IMU time. To explain the algorithm used to do this, a brief overview of the process used to pre-process Kalman filter measurements is required (details can be found in the next section of this chapter).

When Kalman filter measurements are formed, the aiding sensor data is interpolated back to the latest IMU time. To allow this, aiding sensor data should be read from its buffer only when IMU data has been read up to but not past the next expected aiding sensor time. In other words, the aiding sensor should be read when its next expected time lies between the latest IMU time and next expected IMU time.

2.2 Filter Processing

A discrete Kalman filter like that used in the MEMS/GPS project has a number of steps that are generally occurring at different rates. It is assumed that the reader is familiar with Kalman filter processing. All terms and processes are described in detail in References [1] and [3].

The IMU data arrives at the filter from the strapdown navigator at a fixed rate that should be greater than or equal to other sensor rates.

1) At the arrival of every good IMU data record, the following steps are executed.

   A) Filter matrices are computed:

      i) A new dynamics matrix is computed using the new IMU data.

      ii) New diagonal elements for the system state block of the continuous process noise spectral density matrix are computed using IMU data and a set of coefficients. The Gauss-Markov diagonals are constant and computed once at filter initialisation.

      iii) The incremental, discretised process noise covariance matrix is computed using the spectral density matrix and the dynamics matrix.

      iv) The system state rows of the incremental transition matrix are computed. Assuming the dynamics matrix is constant over the IMU data interval, the transition matrix is computed as a Taylor series expansion in terms of the dynamics matrix and the IMU time interval. The Gauss-Markov rows are all constant: a zero sub-block in the system state columns, and a diagonal sub-block computed from correlation times and the IMU time interval in the Gauss-Markov columns.

   B) At filter initialisation and immediately after a measurement update, the full transition and discrete process noise matrices are re-initialised to their incremental counterparts.

   C) Between measurement updates, the error models are propagated over the time since the last measurement update by

      i) Pre-multiplying the previous transition matrix by the current incremental matrix,

      ii) And adding the current incremental process noise covariance matrix to the previous process noise covariance matrix.

2) The second filter "rate" is variable, dependent on the arrival of aiding sensor data or the signal that the time for non-sensor measurements has arrived. At each of these "update times," the following steps are executed.

   A) The full transition and process noise covariance matrices are used to propagate the state vector and its covariance matrix from the last update time to the current time.

   B) Propagated IMU states are sent to the strapdown navigator for closed loop error control. These states are saved for use in Step F).

   C) Each Kalman filter measurement corresponding to the new aiding sensor (or non-sensor) data is formed and tested. If the statistical test of the measurement's residual passes, the state vector and its covariance matrix are updated. If the test fails, no action is taken. All measurements are processed before any other steps are processed.

D) The updated state vector is used to correct IMU data to get the best navigation (position, velocity, attitude) estimates.

E) The corrected navigation data, measurement residual data, and state vector data is sent to a file for subsequent analysis.

F) The states sent to the strapdown navigator (saved in Step B) are subtracted from the updated state vector. This maintains consistency with strapdown navigator: the error control states were used to reduce IMU errors; the filter estimates of those errors (the IMU states) must be adjusted to account for this reduction.

The update interval is variable in the sense that the time of arrival of data from different sensors and the non-sensor measurement times are not coincident. However, if data and non-sensor measurement rates are constant, there will be a repeatable pattern in the measurement update times. For example, assume that GPS and compass data and non-sensor measurement time intervals are all one second. Further assume that GPS data arrives about time t_G, compass data arrives about time $t_C = t_G + 0.3$, and non-sensor measurement times arrive at $t_N = t_C + 0.1$. At each t_G, GPS measurement updates will be processed. Then 0.3 seconds later, a compass heading update will be processed. 0.1 seconds later, the non-sensor updates will be processed. Then, 0.6 seconds after that (when the next GPS record arrives), the process repeats.

Note that the all filter timing is driven by IMU time: all actions will be triggered at an IMU time of data.

3 FILTER MEASUREMENTS

The MEMS/GPS integrated navigation system is comprised of a MEMS IMU plus additional aiding sensors, GPS being the primary aid. A main goal of the MEMS/GPS project is the development of procedures that will allow successful navigation after the loss or degradation of GPS signals. To this end, non-GPS Kalman filter measurements have been developed. These are based on

1. Heading from a digital compass (the compass is also used for initial IMU alignment),

2. Non-sensor data like

    a. A fixed height,

    b. Zero velocities.

Future additions to this non-GPS suite of aiding information could include

1. A baro-altimeter to stabilise the IMU vertical channel in the absence of GPS (to replace or augment the fixed height measurements),

2. An vehicular odometer to provide direction-of-travel speed (e.g. a velocity measurement in the body frame),

3. Stride length algorithms to aid a dismounted soldier's personal navigator.

3.1 Overview

This chapter describes the MEMS/GPS current measurement models. GPS measurements are presented first, followed by all other measurements. Errors in all MEMS/GPS discussions are defined as *true minus approximate*. Therefore, measurements are all formed as *aid minus IMU*.

For each measurement type described below, the measurement vector and the associated rows of the (design) H-matrix will be derived. In brief, a Kalman filter updates its states by combining a weighted *residual* and its previous state estimates (see e.g. references [1] or [3] for more details). The H-matrix describes the linear transformation of the state vector into measurement space. Specifically, the full residual vector (for all measurements) is

$$\vec{\upsilon} = \vec{z} - H\,\widehat{\vec{x}}$$

The H-matrix can be derived as the partial differential equation

$$H = \frac{\partial \vec{z}}{\partial \vec{x}}$$

The standard practice (used in MEMS/GPS) processes measurements one at a time: this requires that all measurements be uncorrelated. Each row of the H-matrix, corresponding to one particular measurement type, is then

$$\vec{h}_i = \frac{\partial z_i}{\partial \vec{x}} \qquad (1)$$

The MEMS/GPS state vector is defined as

$$\vec{x} = \begin{bmatrix} \vec{x}_r^w \\ \vec{x}_v^w \\ \vec{x}_\psi^w \\ \vec{x}_A^b \\ \vec{x}_G^b \\ \vec{x}_{AID} \end{bmatrix} = \begin{bmatrix} \delta\vec{r}_r^w \\ \delta\vec{v}_v^w \\ \vec{\psi} \\ \vec{x}_A^b \\ \vec{x}_G^b \\ \vec{x}_{AID} \end{bmatrix} = \begin{bmatrix} IMU\ position\, states\ WA\ frame \\ IMU\ velocity\, states\ WA\ frame \\ IMU\ attitude\, states\ WA\ frame \\ IMU\ accel.bias\, states\ body\ frame \\ IMU\ accel.bias\, states\ body\ frame \\ Aiding\ sensor\ states \end{bmatrix} \qquad (2)$$

3.2 Temporal And Spatial Corrections

Kalman filter measurements in a system such as MEMS/GPS compare IMU quantities with similar reference values. In some cases, special constraints can be used to form measurements with no reference sensor. For example, when the system is stationary (i.e. with velocity of zero), IMU velocity can be compared with the known zero velocity to form Kalman filter measurements. However, most measurements are formed by comparing IMU quantities with reference values from an independent sensor, an example being IMU versus GPS velocity.

In the more common case of sensor-supplied reference data, corrections are often required before comparisons can be constructed. Firstly, data coming from the two different sensors to form a measurement are not usually synchronized in time. Over “short” differences, when appropriate rate data is available, data from one sensor (usually the reference) can be extrapolated to the time-of-validity of the second sensor (usually the IMU). Further, the two sensors are not, in general, co-located. Given the vector from one (usually the IMU) to the other (usually the reference) in the vehicle body frame (unless otherwise indicated, all coordinate frames used in this report are those defined in reference [1]) and the appropriate supplemental information, the data from one sensor can be transferred to the location of the other. The following sub-sections detail the procedures used in the MEMS/GPS Kalman filter.

3.2.1 Notation

The need to add time and relative body positions to a sensor quantity adds to the notational complexity. The full notation required for the following discussions will be developed here, starting from base notation.

Let’s start with a true velocity vector, $\vec{v}$. If the vector is presented in the a coordinate frame, it is written $\vec{v}^{\,a}$. If it is an estimated or measured value (any quantity containing errors), a hat is added - $\hat{\vec{v}}^{\,a}$. Now, if is has been measured by sensor X , that is added as a subscript - $\hat{\vec{v}}_X^{\,a}$. If the velocity of sensor X has been transferred to another location on the vehicle, say to the location of sensor Y , we will write $\hat{\vec{v}}_{XY}^{\,a}$. Finally, the time of validity is added as a suffix: $\hat{\vec{v}}_{XY}^{\,a}(t_i)$.

To summarise, the notation $\hat{\vec{v}}_{XY}^{\,a}(t_i)$ represents a velocity vector

In the a coordinate frame,

Measured by sensor X ,

Transferred to body location Y,

Valid for time t_i .

Note that location *Y* need not be different from location *X*. If they refer to the same point, and there is no chance of confusion, the second (location) subscript may be dropped $\hat{\bar{v}}_{XX}^{a}(t_i) \equiv \hat{\bar{v}}_{X}^{a}(t_i)$. The time of validity is not necessarily the sensor *X* time; it could any time shortly before or after sensor *X* time.

3.2.2 Temporal Measurement Extrapolation

Let's begin the discussion of adjustments in time by assuming that we wish to form a velocity measurement by comparing an IMU velocity with a reference velocity valid at a slightly different time. If the vehicle is moving (specifically, undergoing accelerations), one of the velocity vectors must be adjusted so it refers to the same point in time as the other. In general, we do not know how the velocity changed over the small time interval. However, we do have enough information to make an estimate: we have a series of previous and current velocities, and we may have acceleration outputs from the IMU (this is the case in MEMS/GPS). Extrapolation could be used to estimate a change in velocity forwards in time. The extrapolation could use a linear (or higher order) fit to the previous (one or more) velocity records (requiring that they be stored); or it could use past accelerations. Extrapolation is generally risky business; and the noisier the data, and the longer the extrapolation period, the greater the risk. A safer approach would be to interpolate backwards in time. Linear interpolations over short periods of time will suffice; more complex models may be required over longer periods of time. Keep in mind that short and long in terms of time periods in these discussions are relative: higher accelerations require shorter time intervals. In any event, it is safe to say that the shorter the interpolation or extrapolation time, the better the expected results will be.

Before continuing, let's look at the specifics of the MEMS/GPS system. IMU data is generally available at higher rates than other sensor data. In the MEMS/GPS strapdown navigator (reference [2]), any output rate up to the rate of the raw data can be selected. IMU data rates are typically 100-200 Hz. At present, the strapdown navigator output rate is set at 10 Hz. GPS output is receiver dependent: the Rockwell-Collins DAGR used in 2003 van testing could output NMEA-standard sentence sets at rates of 0.5 or 1 Hz. Other receivers have output rates of 10 Hz and even higher. Additional current and anticipated MEMS/GPS reference sensors can be assumed to have data rates on the order of 1 Hz.

In theory, it doesn't matter which sensor, the IMU, the reference or both, is time corrected. There are several options for setting measurement times. The following list describes the most likely possibilities and provides some observations that will guide us to the best choice:

1. A measurement rate higher than the lowest sensor rate would be counter-productive (in Kalman filtering theory, sensor data should be used to form measurements only once). This statement should be understood to apply to the following discussions.

2. Triggering measurements on the basis of clock time (every second on the second for example) leads to the possibility of having to adjust sensor times as much as the full sensor data time interval (if a measurement is triggered just before sensor data

arrives). And, of course, both IMU and reference data would, in general, have to be adjusted.

3. Triggering measurements on the basis of the time-of-data for one sensor eliminates the need to adjust the times of the selected sensor. However, the possibility of having to extrapolate the other sensor's times as much as its full data time interval remains.

4. IMU-supplied rate quantities that are not directly used for measurements may be required for temporal adjustments (e.g. accelerations for velocity measurements). They are required for the spatial (lever arm) adjustments described in the next section. Rather than adjusting all those other IMU quantities to the reference time, one adjustment of the reference quantity to IMU time is preferred.

Item 3 above leads to the conclusion that maximum possible extrapolation times will be minimised by adjusting the times of the sensor with the highest data rate (lowest time interval between data records). In most cases, this is the IMU.

On the other hand, item 4 states explicitly that reference sensor times should be adjusted to IMU times.

If the IMU data rate is less than the reference sensor's, items 3 and 4 both lead to the conclusion that reference times should be adjusted to IMU time.

Unfortunately, in the more common situation where IMU data arrives at a higher rate, there is a contradiction. Let's now try to find the best compromise solution to this problem.

Intuition (along with the preceding discussions) suggests that we should try to abide by the recommendation of item 4. Doing so should minimise software complexity, adjustment errors, and computational burden. Let's start with the position that reference times will be adjusted to IMU time. Item 3 then tells us that it may be necessary to extrapolate the other sensor's times as much as its full data time interval. Methods to minimise the adjustment errors need to be found.

In general, adjustment errors are reduced when adjustment times are reduced, and when interpolating rather than extrapolating. A hybrid solution is presented below that meets both error reduction criteria. Table 1 lists the proposed steps in measurement formation within an arbitrary measurement interval.

Table 1: Proposed Temporal Measurement Adjustment Process

1. IMU data records are received – no measurement activity;
2. A reference data record arrives, signalling the start of the measurement process;
3. Rather than extrapolating the IMU time to the reference time, the reference data is interpolated back to the previous IMU time;

4. The measurement is formed at the previous IMU time.

This is a hybrid solution in the sense that it is the arrival of reference data that triggers the start of the measurement process, but the measurement itself is formed at the last IMU time. This procedure

Uses the preferred interpolation method,

Limits the interpolation time to the (lesser) IMU data interval,

Applies to situations where the IMU rate is higher or lower than reference data rates.

Its weaknesses lie in the facts that

The reference data interpolation uses data points that are separated in time by the full reference data interval (even though the point of interest is as close as possible to one end point).

The Kalman filter update will be slightly stale by the time it is available for use (applicable, as it is, to a point in the near past).

When appropriate rate data is available (e.g. velocity data for position measurements), it could be used for adjustment. However, for practical reasons, this option is not recommended for MEMS/GPS:

At present (and for the foreseeable future), MEMS/GPS will use only position, velocity and heading measurements. The corresponding rate data (velocity, acceleration and heading rate) is available only from the IMU (with the exception of GPS position rate, i.e. velocity).

Using IMU rate data to time shift reference data is not recommended.

The added implementation complexities and inconsistency required switching between the application of rate methods (for GPS position measurements) and interpolation methods (for all MEMS/GPS measurements) eliminate any potential benefit of using rate methods.

GPS velocities computed from carrier phase information is more or less independent of code-derived positions, drawing into the question the validity of using such velocities for temporal adjustment of position data. GPS velocities derived from the numerical differentiation of position data offer little benefit relative to the (perhaps simpler) recommended interpolation procedure.

Given the constraints imposed, Table 1 presents an attractive solution to the problem of temporal data matching when forming position, velocity and heading measurements in the MEMS/GPS Kalman filter.

In mathematical terms, a vector of reference measurement data is linearly interpolated to IMU time as follows:

$$\hat{\bar{m}}_R(t_I) = \hat{\bar{m}}_R(t_{Rj}) - \frac{\hat{\bar{m}}_R(t_{Rj}) - \hat{\bar{m}}_R(t_{R(j-1)})}{t_{Rj} - t_{R(j-1)}}(t_{Rj} - t_I)$$

where

t_{Rj} is the time-of-data for the current reference data record,

$t_{R(j-1)}$ is the time-of-data for the previous reference data record,

$\hat{\bar{m}}_R(t_k)$ is the vector of reference measurement data applicable to time t_k.

The above equation is valid for extrapolation as well as interpolation.

3.2.3 Spatial Measurement Corrections

After sensor times have been rationalized, the problem of spatial rationalisation can be addressed. In this case, general procedures cannot be developed. Of the three anticipated measurement types, only position and velocity corrections will be developed here. The reasons for the lack spatial correction of heading measurements will be clear presently.

Recall that the problem involves the adjustment of measurement data to account for spatial displacements of the IMU and reference sensors. All displacements are measured in the (forward, starboard, down) body frame, relative to the navigation reference point. This reference point is the origin of the body frame. In many cases (MEMS/GPS included), the navigation point of reference is near the centre gravity of the vehicle. Often, the IMU is defined as reference point. But, it doesn’t really matter: the spatial corrections are computed in the same way regardless of the location of the reference point. Aside from the given definition of the body frame (origin and orientation), the only other constraint required to begin the derivations is the assumption that the vehicle is a rigid body. This assumption is required if the displacement vectors are to be assumed constant in the body frame.

3.2.3.1 Heading

This brings us the question of spatial corrections for heading measurements. If the rigid body assumption is valid, the heading measured by a sensor will have a constant offset from the heading of the vehicle (defined as the heading of the body frame X-axis) regardless of the location of the sensor on the vehicle. For example, if the sensor could be perfectly aligned to the vehicle heading, it would output the same heading regardless of its location on the vehicle – front or back, top or bottom. So, comparison of heading from two different sensors does not require any adjustments to account for differences in relative location.

3.2.3.2 Positions

Let's begin with position measurements. Clearly, positions collected from sensors at different locations on the vehicle need to be adjusted. Exceptions are made only when the errors introduced by neglecting the differences are negligible when compared to other position error sources: if two position sensors are separated by one centimetre and they measure position to an accuracy of 10 metres, there is nothing to be gained by applying spatial adjustments when forming measurements.

The vector from the origin of the body frame to the measurement centre of the sensor is conventionally called the *lever arm*. The three elements of the vector are measured in body frame coordinates. Positions in body frame coordinates are of no use – positions are conventionally provided in an earth-fixed frame: we generally need to know where we are on or near the Earth's surface. With a lever arm in body frame coordinates and positions in an earth-fixed frame, a method of rotating one into the other is required.

This is accomplished using the well-known direction cosine matrices (DCMs). The lever arm vector in the body frame can be transformed into the local geographic (north, east, down) frame as follows:

$$\delta\vec{r}_{\ell}^{\,g} = C_b^g \delta\vec{r}_{\ell}^{\,b}$$

where C_b^g is DCM from the body frame to local geographic frame. Again, all notation follows that of reference [1]. C_b^g is a function of the roll, pitch and heading angles (details of its formation are not given here). A sensor-derived position vector in the local geographic frame can be translated to the system reference point as follows:

$$\vec{r}_0^{\,g} = \vec{r}_S^{\,g} - \delta\vec{r}_{\ell}^{\,g}$$

One point of clarification: the local geographic is a locally level, topocentric frame whose origin is attached to the vehicle (coincident with the reference point for convenience). It is generally used to represent changes (e.g. velocity) or differences (e.g. the GPS position velocity equation in the next section).

In summary, the position lever arm correction requires (the one-time) measurement of the lever arm from the reference point to the measurement centre of the sensor as well as (on-going) measurement of vehicle roll, pitch and heading (or equivalent).

3.2.3.3 Velocities

Velocity lever arm corrections are slightly more complicated. They describe the velocity measured at the end of the lever arm that can be attributed to rotations of the lever arm relative to the Earth's surface. These rotations are the result of vehicle motions.

We'll begin by differentiating the position lever arm transformation equation, using the locally level, topocentric wander azimuth frame in place of the local geographic frame:

$$\delta\dot{\vec{r}}_{\ell}^{\,w} = \vec{v}_{\ell}^{\,w} = \dot{C}_b^w \delta\vec{r}_{\ell}^{\,b} + C_b^w \delta\dot{\vec{r}}_{\ell}^{\,b}$$

Recalling our previous rigid body assumption, we can set $\delta\dot{\vec{r}}_{\ell}^{\,b} \equiv \vec{0}$, so that

$$\begin{aligned} \vec{v}_{\ell}^{\,w} &= \dot{C}_b^w \delta\vec{r}_{\ell}^{\,b} \\ &= C_b^w \left(\vec{\omega}_{wb}^b \times \delta\vec{r}_{\ell}^{\,b}\right) \end{aligned}$$

where $\vec{\omega}_{wb}^b$ is the vector of rotations of the body frame with respect to the wander azimuth frame, expressed in the body frame; it is a by-product of IMU strapdown calculations. The DCM, $C_b^w = C_g^w C_b^g$, where C_g^w is simply a rotation about the local vertical through an angle known as the wander angle (see reference [1] for details). C_b^w may also be available from the strapdown navigator.

The velocity is corrected by removing the lever arm effects:

$$\vec{v}_0^{\,w} = \vec{v}_S^{\,w} - \vec{v}_{\ell}^{\,w}$$

The same equation can be used for the local geographic frame:

$$\vec{v}_{\ell}^{\,g} = C_b^g \left(\vec{\omega}_{gb}^b \times \delta\vec{r}_{\ell}^{\,b}\right)$$

The strapdown navigator does not provide the $\vec{\omega}_{gb}^b$ rotation vector, but $\vec{\omega}_{gb}^b$ can be computed:

$$\vec{\omega}_{gb}^b = \vec{\omega}_{gw}^b + \vec{\omega}_{wb}^b$$

where $\vec{\omega}_{gw}^b$ describes the rotation of the wander azimuth frame with respect to the local geographic frame, expressed in the body frame. In reference [1], $\vec{\omega}_{gw}^b$ is derived:

$$\vec{\omega}_{gw}^b = C_w^b \vec{\omega}_{gw}^w = \begin{bmatrix} 0 \\ 0 \\ \dot{\alpha} \end{bmatrix}$$

The wander angle rate is

$$\dot{\alpha} = \frac{-v_y^g \tan|\phi|}{R_E + h}$$

where $|\phi|$ is the absolute value of latitude. Is $\dot{\alpha}$ significant? At 89 degrees latitude (zero height), assuming a maximum velocity of 50 metres per second (180 km/hr),

$$\dot{\alpha} < -4.5 \times 10^{-4} \text{ rad/s}$$

Assuming a maximum lever arm of 10 metres, the lever arm velocity due to the wander angle rate is less than 5 millimetres per second. Even with these extreme limits, this is negligible. At latitudes above 89 degrees, or when velocities are much higher (aircraft speeds), the wander angle rate effects should be added.

3.3 GPS Measurements

GPS position and velocity measurement models are described below.

3.3.1 GPS Position Measurements

GPS positions expressed as latitude, longitude and height in the WGS84 geodetic system are used to form position measurements according to the following development.

3.3.1.1 The Measurements

Since the MEMS/GPS IMU system (navigation) states are all defined in the local level wander azimuth coordinate frame, the GPS position measurements are also formed in a local level frame, the local geographic:

$$\bar{z}_{r_{GPS}}^{g} = \begin{bmatrix} \left(\hat{\phi}_{GPS} - \hat{\phi}_{IMU}\right)\left(R_E + \hat{h}_{GPS}\right) \\ \left(\hat{\lambda}_{GPS} - \hat{\lambda}_{IMU}\right)\left(R_E + \hat{h}_{GPS}\right)\cos\hat{\phi}_{GPS} \\ \hat{h}_{IMU} - \hat{h}_{GPS} \end{bmatrix}$$

Notes:

A spherical earth model is used in the horizontal measurements.

Height is positive up while the local geographic z-axis is positive down – leading to the sign change in the z-measurement.

Strictly speaking, the right-hand side of the equation produces arcs while the left-hand components are chords. However, the difference is insignificant in this application.

In the radius estimates, GPS height is used to approximate true height. Similarly, GPS latitude is used to estimate true latitude. In the closed-loop formulation used in MEMS/GPS, GPS, IMU and filtered positions should all be close enough to one another that they are interchangeable when estimating these radii.

3.3.1.2 The Model

The corresponding rows of the H-matrix are derived using equation (1). To ease the differentiation, expand the first row of the measurement equation as follows

$$
\begin{aligned}
\left(\hat{\phi}_{GPS} - \hat{\phi}_{IMU}\right)\left(R_E + \hat{h}_{GPS}\right) &= \left[\left(\phi - \delta\phi_{GPS}\right) - \left(\phi - \delta\phi_{IMU}\right)\right]\left(R_E + \hat{h}_{GPS}\right) \\
&= \left(\delta\phi_{IMU} - \delta\phi_{GPS}\right)\left(R_E + \hat{h}_{GPS}\right) \\
&= \delta r^g_{IMU_Y} - \delta r^g_{GPS_X}
\end{aligned}
$$

Similarly,

$$\left(\delta\lambda_{IMU} - \delta\lambda_{GPS}\right)\left(R_E + \hat{h}_{GPS}\right)\cos\hat{\phi}_{GPS} = \delta r^g_{IMU_Y} - \delta r^g_{GPS_Y}$$

$$\delta h_{GPS} - \delta h_{IMU} = \delta r^g_{IMU_Z} - \delta r^g_{GPS_Z}$$

Now,

$$\bar{z}^g_{r_{GPS}} = \delta\bar{r}^g_{IMU} - \delta\bar{r}^g_{GPS} = C^g_w \delta\bar{r}^w_{IMU} - \delta\bar{r}^g_{GPS}$$

The 3 row by number-of-states column GPS position measurement matrix is derived through the partial differentiation of this equation:

$$H_{r_{GPS}} = \frac{\partial \bar{z}^g_{r_{GPS}}}{\partial \bar{x}} = \left[C^g_w \quad 0 \quad 0 \quad 0 \quad 0 \quad -I \ldots\right]$$

The 3 by 3 non-zero blocks fall in the IMU position error state and (optional) GPS error position error state columns, respectively. Columns corresponding to all other error states are zero.

The DCM C^g_w rotates the IMU position error states from the IMU (computer) wander azimuth frame into the GPS local geographic frame:

$$C_w^g = \begin{bmatrix} \cos\alpha_{IMU} & -\sin\alpha_{IMU} & 0 \\ -\sin\alpha_{IMU} & -\cos\alpha_{IMU} & 0 \\ 0 & 0 & -1 \end{bmatrix} \qquad (3)$$

where α_{IMU} is the wander angle at IMU longitude.

Note that this DCM is strictly correct only when IMU and GPS longitudes are the same (changes in wander angle being strictly a function of change in longitude). If there were significant differences (on the order of tens of kilometres) in longitude, it would be more correct to include the earth-centred rotation from the IMU to the GPS position. In MEMS/GPS, large IMU position errors will only occur when the IMU is unaided, and this problem will occur when GPS positions are re-acquired after a period of accumulating IMU longitude error. However, since this is really a second order affect, the IMU longitude will quickly converge to GPS longitude after reacquisition, even when ignoring the IMU to GPS rotation.

3.3.2 GPS Velocity Measurements

GPS velocity, expressed in the local geographic frame, is used to form velocity measurements according to the following development.

3.3.2.1 The Measurements

In principle, GPS velocity is computed in an earth-centred, earth-fixed coordinate frame, as a function of line-of-sight velocities to individual satellites. Then it is rotated into the local geographic frame at the GPS-computed position. Therefore, like positions, the GPS velocity measurements will be formed in the local geographic frame at the GPS-computed position - $\hat{\bar{v}}_{GPS}^g$.

The MEMS/GPS strapdown navigator provides IMU velocity in the wander azimuth frame.

A Kalman filter measurement formed using GPS and IMU velocities requires a common coordinate frame. The local geographic coordinate frame at the GPS-computed position has been selected as the measurement frame. This means that the IMU velocity must be rotated into this frame. Using the wander azimuth to local geographic DCM at the IMU-computed position, the GPS velocity measurement is:

$$\bar{z}_{v_{GPS}}^g = \hat{\bar{v}}_{GPS}^g - C_w^g \hat{\bar{v}}_{IMU}^w$$

where $$C_w^g = \begin{bmatrix} \cos\alpha_{IMU} & -\sin\alpha_{IMU} & 0 \\ -\sin\alpha_{IMU} & -\cos\alpha_{IMU} & 0 \\ 0 & 0 & -1 \end{bmatrix}$$

3.3.2.2 The Model

Identifying Kalman filter state dependencies of the measurement equation is somewhat more complex than it was for position measurements. In the -angle error formulation used in MEMS/GPS, IMU velocities are assumed provided in the "platform" frame (a nominally local level wander azimuth frame centred at the IMU-computed position) - $\hat{\vec{v}}^{p}_{IMU}$. All coordinate frames and their transformations are defined in reference [1].

The DCM used to form the measurements does not completely describe the transformation to the GPS local geographic frame:

> C^{g}_{w} describes a single rotation about the local level at the IMU position – it is more properly written $C^{g}_{c}(IMU)$, from the "computer" frame to the local geographic frame at the IMU-computed position.
>
> The DCM from the platform frame to the computer frame is a function of the - angles: $C^{c}_{p} \cong I + (\vec{\psi}\times)$, using small angle approximations.
>
> The local level at the IMU position is rotated with respect to local level at the GPS position because of earth curvature. The rotations are described (in standard notation) by the $\delta\vec{\theta}$ -angles. The $\delta\vec{\theta}$ -angles are dependent on IMU position errors (that are estimated using GPS positions).

The measurement equation can now be expanded to

$$
\begin{aligned}
\vec{z}^{g}_{v_{GPS}} &= \hat{\vec{v}}^{g}_{GPS} - C^{g(GPS)}_{w(GPS)} C^{w(GPS)}_{c} C^{c}_{p} \hat{\vec{v}}^{p}_{IMU} \\
&= \hat{\vec{v}}^{g}_{GPS} - C^{g(GPS)}_{w(GPS)} C^{t}_{c} C^{c}_{p} \hat{\vec{v}}^{p}_{IMU} \\
&= \hat{\vec{v}}^{g}_{GPS} - C^{g(GPS)}_{w(GPS)} \left(I + \delta\vec{\theta}\times\right)\left(I + \vec{\psi}\times\right) \hat{\vec{v}}^{p}_{IMU}
\end{aligned}
$$

Further expansion and elimination of second order error terms gives

$$
\vec{z}^{g}_{v_{GPS}} \cong \hat{\vec{v}}^{g}_{GPS} - C^{g(GPS)}_{w(GPS)} \left[I + \left(\delta\vec{\theta}\times\right) + \left(\vec{\psi}\times\right)\right] \hat{\vec{v}}^{p}_{IMU}
$$

This is the most complete form, required for open-loop INS/GPS filtering systems.

MEMS/GPS is a closed-loop system: whenever GPS data is available, the IMU strapdown navigator is continuously corrected with filter error estimates. This means that (whenever GPS data is available) IMU position, velocity, and attitude errors remain small. For this reason, this complete model is not required for MEMS/GPS GPS velocity measurements. The GPS velocity measurement model based on the complete model is derived in APPENDIX A. A hybrid model will be developed below.

The hybrid model will be developed by examining the relative sizes of the two error terms in the complete model, $\delta\vec{\theta}$ and $\vec{\psi}$. The former depends on IMU position errors. On a spherical earth, the $\delta\vec{\theta}$ -angles are an inverse function of the mean earth radius:

$$\delta\theta = \frac{\delta r}{R_E + h}$$

where δr is the position error. An error as large as 10 kilometres produces an angle of only about 5.5 arc minutes.

On the other hand, $\vec{\psi}$ -angle tilt errors are expected to be somewhat larger that 5 arc minutes even when all measurements are available.

The hybrid GPS velocity measurement model assumes position errors introduce negligible tilt relative to the $\vec{\psi}$ -angles:

$$\begin{aligned}\vec{z}^{\,g}_{v_{GPS}} &\cong \hat{\vec{v}}^{\,g}_{GPS} - C^{g(GPS)}_{w(GPS)}\left[I + (\vec{\psi}\times)\right]\hat{\vec{v}}^{\,p}_{IMU} \\ &\cong \hat{\vec{v}}^{\,g}_{GPS} - C^g_w \hat{\vec{v}}^{\,p}_{IMU} - C^g_w(\vec{\psi}\times)\hat{\vec{v}}^{\,p}_{IMU}\end{aligned}$$

Note that position errors produce wander angle changes at about the same rate as tilts. This allows us to use any wander azimuth to local geographic DCM.

At this point, the velocity errors will be introduced. Restricting ourselves to the first two terms,

$$\begin{aligned}\hat{\vec{v}}^{\,g}_{GPS} - C^g_w \hat{\vec{v}}^{\,p}_{IMU} &= \left(\vec{v}^{\,g}_{GPS} - \delta\vec{v}^{\,g}_{GPS}\right) - C^g_w\left(\vec{v}^{\,p}_{IMU} - \delta\vec{v}^{\,p}_{IMU}\right) \\ &= C^g_w \delta\vec{v}^{\,p}_{IMU} - \delta\vec{v}^{\,g}_{GPS}\end{aligned}$$

where C^g_w is given by equation (3), for convenience.

This expression is dependent only on IMU and GPS velocity errors. The partial differential equations needed to extract these dependencies are:

$$\frac{\partial\left(C^g_w \delta\vec{v}^{\,p}_{IMU} - \delta\vec{v}^{\,g}_{GPS}\right)}{\delta\vec{v}^{\,p}_{IMU}} = C^g_w$$

$$\frac{\partial\left(C^g_w \delta\vec{v}^{\,p}_{IMU} - \delta\vec{v}^{\,g}_{GPS}\right)}{\delta\vec{v}^{\,p}_{GPS}} = -I$$

Note: GPS velocity error states are often not included in a filter such as this.

The third term is bit more complicated. A very similar expression was derived in APPENDIX A. Those results will be modified for the present purposes:

$$C_w^g(\bar{\psi}\times)\hat{\bar{v}}_{IMU}^p = \begin{bmatrix} \cos\alpha_{IMU}\left(-\psi_z v_{IMU_Y}^p + \psi_y v_{IMU_Z}^p\right) - \sin\alpha_{IMU}\left(\psi_z v_{IMU_X}^p - \psi_x v_{IMU_Z}^p\right) \\ -\sin\alpha_{IMU}\left(-\psi_z v_{IMU_Y}^p + \psi_y v_{IMU_Z}^p\right) - \cos\alpha_{IMU}\left(\psi_z v_{IMU_X}^p - \psi_x v_{IMU_Z}^p\right) \\ \psi_y v_{IMU_X}^p - \psi_x v_{IMU_Y}^p \end{bmatrix}$$

Note that all velocity errors in this term will be second order and would be neglected as usual. The measured IMU velocity will be used here.

Once again neglecting second order effects, this expression is dependent only on the $\bar{\psi}$ -angles. The partial differential equation needed to extract these dependencies is:

$$\frac{\partial\left(-C_w^g(\bar{\psi}\times)\hat{\bar{v}}_{IMU}^p\right)}{\partial\bar{\psi}} = \begin{bmatrix} -v_{IMU_Z}^p \sin\alpha_{IMU} & -v_{IMU_Z}^p \cos\alpha_{IMU} & v_{IMU_X}^p \sin\alpha_{IMU} + v_{IMU_Y}^p \cos\alpha_{IMU} \\ -v_{IMU_Z}^p \cos\alpha_{IMU} & v_{IMU_Z}^p \sin\alpha_{IMU} & v_{IMU_X}^p \cos\alpha_{IMU} - v_{IMU_Y}^p \sin\alpha_{IMU} \\ v_{IMU_Y}^p & -v_{IMU_X}^p & 0 \end{bmatrix}$$

This can be simplified by writing the ψ_z column in terms of local geographic velocity:

$$\frac{\partial\left(-C_w^g(\bar{\psi}\times)\hat{\bar{v}}_{IMU}^p\right)}{\partial\bar{\psi}} = \begin{bmatrix} -v_{IMU_Z}^p \sin\alpha_{IMU} & -v_{IMU_Z}^p \cos\alpha_{IMU} & -v_{IMU_Y}^g \\ -v_{IMU_Z}^p \cos\alpha_{IMU} & v_{IMU_Z}^p \sin\alpha_{IMU} & v_{IMU_X}^g \\ v_{IMU_Y}^p & -v_{IMU_X}^p & 0 \end{bmatrix}$$

The resulting 3 by number-of-states GPS velocity measurement matrix is:

$$H_{v_{GPS}} = \frac{\partial\bar{z}_{v_{GPS}}^g}{\partial\bar{x}} = \begin{bmatrix} 0 & C_w^g & \begin{pmatrix} -v_{IMU_Z}^p \sin\alpha_{IMU} & -v_{IMU_Z}^p \cos\alpha_{IMU} & -v_{IMU_Y}^g \\ -v_{IMU_Z}^p \cos\alpha_{IMU} & v_{IMU_Z}^p \sin\alpha_{IMU} & v_{IMU_X}^g \\ v_{IMU_Y}^p & -v_{IMU_X}^p & 0 \end{pmatrix} & 0 & 0 & 0 & -I & \ldots \end{bmatrix}$$

In this case, the 3 by 3 non-zero blocks fall in the IMU velocity $\left(C_w^g\right)$, IMU attitude $\left(-\frac{\partial\left(C_w^g(\bar{\psi}\times)\hat{\bar{v}}_{IMU}^p\right)}{\partial\bar{\psi}}\right)$ and (optional) GPS velocity (-I) error state columns. Columns corresponding to all other error states are zero.

3.3.2.3 The Simplest Model

The GPS velocity model can be further simplified by assuming that the tilt errors also have a negligible effect on the measurement (i.e. set $\bar{\psi} \cong \bar{0}$ or $C_p^c \cong I$). Then the H-matrix is simply

$$H_{v_{GPS}} = \frac{\partial\bar{z}_{v_{GPS}}^g}{\partial\bar{x}} = \begin{bmatrix} 0 & C_w^g & 0 & 0 & 0 & 0 & -I & \ldots \end{bmatrix}$$

3.3.3 GPS Course-Over-Ground Measurements

In certain applications, the direction of the GPS velocity vector can be used to control system heading. These measurements can only be used for installations that ensure the direction of travel and heading are collinear. In general, this restricts GPS course-over-ground (COG) measurements to wheeled or tracked land vehicles.

If GPS COG is not available directly from the receiver, it is derived from the horizontal components of the GPS velocity vector:

$$\hat{\Psi}^{g}_{GPS} = \tan^{-1}\left(\frac{v^{g}_{y}}{v^{g}_{x}}\right)$$

3.3.3.1 The Measurement

The GPS COG measurement equation can be written as

$$\begin{aligned} z^{g}_{\Psi_{GPS}} &= \hat{\Psi}^{g}_{GPS} - \hat{\Psi}^{g}_{IMU} \\ &= \hat{\Psi}^{g}_{GPS} - \left(\hat{\Psi}^{w}_{IMU} - \alpha_{IMU}\right) \end{aligned}$$

where $\hat{\Psi}^{w}_{IMU}$ is the IMU's so-called "platform" heading.

3.3.3.2 The Model

To derive the H-matrix model, write the GPS COG measurement equation in terms of true values and errors:

$$\begin{aligned} z^{g}_{\Psi_{GPS}} &= \hat{\Psi}^{g}_{GPS} - \hat{\Psi}^{w}_{IMU} + \alpha_{IMU} \\ &= \left(\Psi^{g} - \delta\Psi^{g}_{GPS}\right) - \left(\Psi^{w} - \delta\Psi^{w}_{IMU}\right) + \left(\alpha - \delta\alpha_{IMU}\right) \\ &= \left(\Psi^{g} - \Psi^{w} + \alpha\right) + \delta\Psi^{w}_{IMU} - \delta\Psi^{g}_{GPS} - \delta\alpha_{IMU} \\ &= \delta\Psi^{w}_{IMU} - \delta\Psi^{g}_{GPS} - \delta\alpha_{IMU} \end{aligned}$$

Using equation (9.51) of reference [1],

$$\begin{aligned} \delta\Psi^{w}_{IMU} &= \delta r^{w}_{IMU_X} \frac{\tan\Theta \sin\Psi^{w}_{IMU}}{R_E + h} + \delta r^{w}_{IMU_X} \frac{\tan\Theta \cos\Psi^{w}_{IMU}}{R_E + h} \\ &\quad + \psi^{w}_{x} \tan\Theta \cos\Psi^{w}_{IMU} - \psi^{w}_{y} \tan\Theta \sin\Psi^{w}_{IMU} - \psi^{w}_{z} \end{aligned}$$

where is pitch.

The wander angle error is dependent solely on longitude (or east) error. In terms of the wander azimuth IMU position error states (again from reference [1]),

$$\delta\alpha_{IMU} = \frac{\left(\delta r^{w}_{IMU_X}\ \sin\alpha_{IMU} + \delta r^{w}_{IMU_Y}\ \cos\alpha_{IMU}\right)\tan\left|\phi_{GPS}\right|}{R_E + h_{GPS}}$$

Once again, the position errors will be assumed negligible in this closed loop model. In addition, since this measurement is restricted to wheeled or tracked land vehicles, it can be assumed that pitch will not approach 90 degrees, and the IMU heading error term can be assumed negligible. With these assumptions, the GPS COG measurement can be written

$$z^{g}_{\Psi_{GPS}} \cong -\psi^{w}_{z} - \delta\Psi^{g}_{GPS}$$

Now the row of the H-matrix corresponding to the GPS COG measurement can be written as

$$H_{\Psi_{CMPS}} = \frac{\partial z^{g}_{\Psi_{CMPS}}}{\partial\vec{x}} = \begin{bmatrix}0 & 0 & (0 & 0 & -1) & 0 & 0 & \ldots -1\end{bmatrix}$$

with minus ones in the IMU vertical attitude error and GPS heading error columns and zeroes everywhere else.

3.3.3.3 Measurement Variance

Since GPS COG is the direction of the GPS local geographic velocity vector, the accuracy of the COG will depend on the accuracy of the horizontal velocity as well as the vehicle speed. Thus, it is better to compute the COG measurement variance from GPS velocity variances, when they are available.

If GPS velocity accuracy estimates are available, they are used to estimate COG accuracy as follows. Differentiate the COG equation,

$$\hat{\Psi}^{g}_{GPS} = \tan^{-1}\left(\frac{v^{g}_{y}}{v^{g}_{x}}\right)$$

to get an expression for COG error in terms of velocity error:

$$\delta\hat{\Psi}^{g}_{GPS} = \delta\left[\tan^{-1}\left(\frac{v^{g}_{y}}{v^{g}_{x}}\right)\right]$$

In general,

$$\frac{d}{dx}\tan^{-1}u = \frac{1}{1+u^2}\frac{du}{dx}$$

Therefore,

$$
\begin{aligned}
d\hat{\Psi}^{g}_{GPS} &= \frac{1}{1+\left(\dfrac{v^g_y}{v^g_x}\right)^2} d\left(\frac{v^g_y}{v^g_x}\right) \\
&= \frac{\left(v^g_x\right)^2}{\left(v^g_x\right)^2+\left(v^g_y\right)^2} d\left(\frac{v^g_y}{v^g_x}\right)
\end{aligned}
$$

Now,

$$
d\left(\frac{v^g_y}{v^g_x}\right) = \frac{v^g_x \cdot d\left(v^g_y\right) - v^g_y \cdot d\left(v^g_x\right)}{\left(v^g_x\right)^2}
$$

So that

$$
d\hat{\Psi}^{g}_{GPS} = \frac{\left(v^g_x\right)^2}{\left(v^g_x\right)^2+\left(v^g_y\right)^2} \cdot \frac{v^g_x \cdot d\left(v^g_y\right) - v^g_y \cdot d\left(v^g_x\right)}{\left(v^g_x\right)^2}
$$

Finally, the error in GPS COG (in radians) in terms of GPS horizontal velocity errors is

$$
d\hat{\Psi}^{g}_{GPS} = \frac{v^g_x \cdot d\left(v^g_y\right) - v^g_y \cdot d\left(v^g_x\right)}{\left(v^g_x\right)^2+\left(v^g_y\right)^2}
$$

The variance of $d\hat{\Psi}^{g}_{GPS}$ can be estimated in terms of the velocity error variances by using the definition of variance:

$$
\sigma^2_x = \mathrm{E}\left[\left(x - \mathrm{E}[(x)]\right)^2\right]
$$

If it is assumed that $\mathrm{E}\left[d\hat{\Psi}^{g}_{GPS}\right] = 0$ (i.e. that COG is unbiased),

$$
\sigma^2_{d\Psi} = \mathrm{E}\left[\left(d\hat{\Psi}^{g}_{GPS}\right)^2\right]
$$

Squaring $d\hat{\Psi}^{g}_{GPS}$, we get

$$
\left(d\hat{\Psi}^{g}_{GPS}\right)^2 = \frac{\left(v^g_x\right)^2 \cdot d\left(v^g_y\right)^2 - 2 v^g_x v^g_y d\left(v^g_y\right) d\left(v^g_x\right) + \left(v^g_y\right)^2 \cdot d\left(v^g_x\right)^2}{\left(\left(v^g_x\right)^2+\left(v^g_y\right)^2\right)^2}
$$

Taking the expectation of the result,

$$\mathrm{E}\left[\left(d\hat{\Psi}^{g}_{GPS}\right)^2\right] = \sigma^2_{d\Psi} = \frac{\left(v^g_x\right)^2 \cdot \sigma^2_{v_Y} - 2v^g_x v^g_y \mathrm{E}\left[d\left(v^g_y\right)d\left(v^g_x\right)\right] + \left(v^g_y\right)^2 \cdot \sigma^2_{v_X}}{\left(\left(v^g_x\right)^2 + \left(v^g_y\right)^2\right)^2}$$

If velocity covariances are known, they can be used to compute $\mathrm{E}\left[d\left(v^g_y\right)d\left(v^g_x\right)\right]$. In the more usual cases where they are unknown, they will be assumed negligible, with the result

$$\mathrm{E}\left[d\left(v^g_y\right)d\left(v^g_x\right)\right] \cong 0$$

The COG variance can thus simplified to

$$\sigma^2_{d\Psi} \cong \frac{\left(v^g_x\right)^2 \cdot \sigma^2_{v_Y} + \left(v^g_y\right)^2 \cdot \sigma^2_{v_X}}{\left(\left(v^g_x\right)^2 + \left(v^g_y\right)^2\right)^2}$$

Note that when $\sigma^2_{v_X} = \sigma^2_{v_Y} = \sigma^2_v$, this expression can be further simplified to

$$\sigma^2_{d\Psi} \cong \frac{\left(\left(v^g_x\right)^2 + \left(v^g_y\right)^2\right)\cdot \sigma^2_v}{\left(\left(v^g_x\right)^2 + \left(v^g_y\right)^2\right)^2} = \frac{\sigma^2_v}{\left(v^g_x\right)^2 + \left(v^g_y\right)^2}$$

These results make intuitive sense: the COG variances are directly related velocity variances and inversely related to speed. The poorer the velocity accuracy, the poorer the COG accuracy; the higher the speed, the better the COG accuracy.

3.4 Compass Measurements

Since only a single channel digital compass is currently being used in the MEMS/GPS system, the simplest, scalar heading measurement model will be derived in this section. The compass heading model should be revised if a multi-channel compass is added in the future.

Note that this model is identical to the GPS COG measurement derived above.

3.4.1.1 The Measurement

The compass heading measurement equation can be written as

$$\begin{aligned} z^g_{\Psi_{CMPS}} &= \hat{\Psi}^g_{CMPS} - \hat{\Psi}^g_{IMU} \\ &= \hat{\Psi}^g_{CMPS} - \left(\hat{\Psi}^w_{IMU} - \alpha_{IMU}\right) \end{aligned}$$

3.4.1.2 The Model

To derive the H-matrix model, write this equation in terms of true values and errors:

$$\begin{aligned} z^{g}_{\Psi_{CMPS}} &= \hat{\Psi}^{g}_{CMPS} - \hat{\Psi}^{w}_{IMU} + \alpha_{IMU} \\ &= \delta\Psi^{w}_{IMU} - \delta\Psi^{g}_{CMPS} - \delta\alpha_{IMU} \end{aligned}$$

Assuming δc_{IMU} is negligible and pitch is small,

$$z^{g}_{\Psi_{CMPS}} \cong -\psi^{w}_{z} - \delta\Psi^{g}_{CMPS}$$

The digital compass currently being used is a KVH C100. The Technical Manual (reference [5]) limits valid heading to tilt angles of 16 degrees or less. The tangent of 16 degrees is less than 0.3, justifying the assumption of small pitch.

Now the row of the H-matrix corresponding to the compass heading measurement can be written as

$$H_{\Psi_{CMPS}} = \frac{\partial z^{g}_{\Psi_{CMPS}}}{\partial \vec{x}} = \begin{bmatrix} 0 & 0 & \begin{pmatrix} 0 & 0 & -1 \end{pmatrix} & 0 & 0 & \ldots -1 \end{bmatrix}$$

In this case, the second minus one is located in the compass heading error column.

In an application that is not limited in pitch (e.g. a soldier-mounted system), assuming the use of a multi-axis compass, the IMU tilt error terms cannot be neglected in the H-matrix. The following model should be used

$$H^{3D}_{\Psi_{CMPS}} = \frac{\partial z^{g}_{\Psi_{CMPS}}}{\partial \vec{x}} = \begin{bmatrix} 0 & 0 & \begin{pmatrix} \tan\Theta\cos\Psi^{w}_{IMU} & -\tan\Theta\sin\Psi^{w}_{IMU} & -1 \end{pmatrix} & 0 & 0 & \ldots -1 \end{bmatrix}$$

Note the singularity at the vertical where the tangent of pitch is infinite.

3.5 Non-Sensor Measurements

3.5.1 Non-Sensor Position Measurements

At present (and for the foreseeable future), the is only one non-sensor position measurement – a fixed height.

3.5.1.1 Fixed Height

An IMU is naturally instable in the vertical channel. When there are no other measurements able to stabilise the vertical channel, the system height can be held constant using a fixed height measurement. Note that a fixed height measurement can be used in conjunction with the vertical velocity measurement described below.

To use a fixed height measurement, it is advisable to track height and save it whenever it can be confidently assumed good. Then when fixed height measurements are required, the saved good height is used as the reference, fixed height for measurement formation.

3.5.1.1.1 The Measurement

The fixed height measurement is very simple:

$$z^{w}_{Z_{FIX}} = \hat{h}_{FIX} - \hat{h}_{IMU}$$

It is similar to the GPS height measurement, except that it is formed in the wander azimuth coordinate frame, resulting in a sign change in the measurement.

3.5.1.1.2 The Model

No coordinate conversion is necessary and the fixed height is assumed to be error-free. This leads to a measurement model is equally simple:

$$H_{r_{FIX}} = \frac{\partial z^{w}_{r_{FIX}}}{\partial \vec{x}} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & \ldots \end{bmatrix}$$

3.5.2 Non-Sensor Velocity Measurements

When the GPS signals are not available, it is important for the MEMS/GPS system to have backup measurements available to limit or slow the rate of growth of filtered errors. There is non-sensor information that can be used to form velocity measurements:

1. Zero velocity updates (often called ZUPTS) can be made whenever the system is known to be stationary. These can be used in a system mounted in a land vehicle or on a dismounted soldier.

2. An IMU is naturally instable in the vertical channel. When there are no other measurements able to stabilise the vertical channel, the system height can be held controlled by assuming zero vertical velocity and forming zero vertical velocity measurements (just like a regular vertical ZUPT except the vehicle may be moving).

3. If an IMU is aligned to the body of a land vehicle, the velocity in the body lateral axis (perpendicular to the direction of travel) can be assumed to be zero unless the vehicle's wheels (or tracks) are sliding sideways.

3.5.2.1 ZUPTs

3.5.2.1.1 The Measurement

The measurement model for a ZUPT is very simple. Since a zero velocity vector is zero in any coordinate frame, no transformations are necessary:

$$\vec{z}^{\,p}_{ZUPT} = \vec{0} - \hat{\vec{v}}^{\,p}_{IMU} = -\hat{\vec{v}}^{\,p}_{IMU}$$

3.5.2.1.2 The Model

Since $\hat{\vec{v}}^{\,p}_{IMU} = \vec{v}^{\,p}_{IMU} - \delta\vec{v}^{\,p}_{IMU}$, the ZUPT rows of the H-matrix are simply

$$H_{ZUPT} = \frac{\partial \vec{z}^{\,p}_{ZUPT}}{\partial \vec{x}} = \begin{bmatrix} 0 & I & 0 & 0 & 0 & 0 \end{bmatrix}$$

3.5.2.2 Land Vehicle Velocity Constraints

3.5.2.2.1 The Measurement

The lateral axis velocity constraint is somewhat more complicated than the ZUPT model. This is a scalar model that will be written in the vehicle body y-axis. Again, the reference is simply zero velocity. The IMU velocity must transformed from the wander azimuth platform frame into the Y-axis of the body frame:

$$z^{y^b}_{v_0} = 0 - C^b_w(2)\hat{\vec{v}}^{\,p}_{IMU} = -C^b_w(2)\hat{\vec{v}}^{\,p}_{IMU}$$

where the second row of the wander azimuth to body frame DCM,

$$C^b_w(2) = \begin{bmatrix} \sin\Phi_{IMU}\sin\Theta_{IMU}\cos\Psi^p_{IMU} & -\sin\Phi_{IMU}\sin\Theta_{IMU}\sin\Psi^p_{IMU} & \\ & & -\sin\Phi_{IMU}\cos\Theta_{IMU} \\ -\cos\Phi_{IMU}\sin\Psi^p_{IMU} & -\cos\Phi_{IMU}\cos\Psi^p_{IMU} & \end{bmatrix}$$

$\Phi_{IMU}, \Theta_{IMU}, \Psi^p_{IMU}$ are the IMU computed roll, pitch and platform heading (heading relative to the (platform) wander azimuth x-axis), respectively.

3.5.2.2.2 The Model

Once again, the IMU velocity must be rotated from the platform to the computer frame using the -angles. Following steps similar to those used for GPS velocity measurements:

$$
\begin{aligned}
z_{v_0}^{y^b} &= -C_w^b(2) C_p^c \hat{\vec{v}}_{IMU}^{p} \\
&= -C_w^b(2)\left(I + \vec{\psi}\times\right)\hat{\vec{v}}_{IMU}^{p} \\
&= -C_w^b(2)\hat{\vec{v}}_{IMU}^{p} - C_w^b(2)\left(\vec{\psi}\times\hat{\vec{v}}_{IMU}^{p}\right)
\end{aligned}
$$

Introducing velocity errors:

$$
\begin{aligned}
z_{v_0}^{y^b} &= -C_w^b(2)\left(\vec{v}_{IMU}^{p} - \delta\vec{v}_{IMU}^{p}\right) - C_w^b(2)\left(\vec{\psi}\times\hat{\vec{v}}_{IMU}^{p}\right) \\
&= -C_w^b(2)\vec{v}_{IMU}^{p} + C_w^b(2)\delta\vec{v}_{IMU}^{p} - C_w^b(2)\left(\vec{\psi}\times\hat{\vec{v}}_{IMU}^{p}\right)
\end{aligned}
$$

Expanding the last term gives

$$
\begin{aligned}
-C_w^b(2)\left(\vec{\psi}\times\hat{\vec{v}}_{IMU}^{p}\right) &= -C_w^b(2,1)\left(-\psi_z v_{IMU_Y}^{p} + \psi_y v_{IMU_Z}^{p}\right) \\
&\quad - C_w^b(2,2)\left(\psi_z v_{IMU_X}^{p} - \psi_x v_{IMU_Z}^{p}\right) \\
&\quad - C_w^b(2,3)\left(-\psi_y v_{IMU_X}^{p} + \psi_x v_{IMU_Y}^{p}\right)
\end{aligned}
$$

Finally, by taking the partial derivatives, we get the required row of the H-matrix:

$$
H_{v_0} = \frac{\partial z_{v_0}^{y^b}}{\partial \vec{x}} = \begin{bmatrix} 0 & C_w^b(2) & \begin{bmatrix} \{v_{IMU_Z}^{p} C_w^b(2,2) & \{-v_{IMU_Z}^{p} C_w^b(2,1) & \{v_{IMU_Y}^{p} C_w^b(2,1) \\ -v_{IMU_Y}^{p} C_w^b(2,3)\} & +v_{IMU_X}^{p} C_w^b(2,3)\} & -v_{IMU_X}^{p} C_w^b(2,2)\} \end{bmatrix} & 0 & 0 & 0 \end{bmatrix}
$$

3.6 Measurement Pre-Conditions

The Kalman filter measurements described above typically are run only when specific pre-conditions are met. This section describes the pre-conditions for each type of measurement.

There are general pre-conditions applied to all measurements.

1) The MEMS/GPS Kalman filter software allows measurements to enabled or disabled. Only enabled measurements are processed.

2) Execution of each measurement is dependent on the availability of good IMU and aiding sensor data.

3) A particular measurement will be excluded from forming an update if its misclosure could not be properly formed.

Pre-conditions applied to specific measurements (or groups of measurements) are described below. Measurements are formed only when all pre-conditions are met: the general pre-conditions were listed above; measurement-specific pre-conditions are described below.

3.6.1 GPS Positions

Most GPS receivers will switch to 2-D(imensional) mode when they are not able to track enough pseudoranges to provide a full solution. A GPS receiver in 2D mode will output a fixed height equal to its last computed height as well as a flag informing the user of the change in operating mode. When fixed height data is being used in a Kalman filter, the measurement model should be adjusted to reflect the degraded height performance. Rather than adjusting the GPS height measurement model as a receiver switches to and from 2D mode, the MEMS/GPS filter uses a separate “fixed height” measurement. When 2D mode is signalled, the GPS height measurement is deactivated and the fixed height measurement is activated. Conversely, when GPS height estimation is resumed, the fixed height measurement is deactivated and the GPS height measurement is reactivated. To add some confidence in and control over the height fixing, the MEMS/GPS Kalman filter task keeps track of and uses its own fixed height value instead of relying on the receiver’s fixed height.

Therefore, to optimise filtered height performance during periods of 2D GPS, it is recommended that the fixed height measurement always be enabled. Note that fixed height measurements are activated whenever a sensor-based height measurement has not succeeded. This could be a single measurement residual rejection, or it could be a period of time with no GPS position fixes due to signal blockage or jamming. This is an important feature since the IMU strapdown navigator relies on the Kalman filter to control instability in the vertical channel. Even a short period without damping can result in an unstable height. Note further that this feature is suitable only for surface vehicles. An aircraft or submarine application requires an independent height (or depth) sensor to maintain acceptable height performance.

3.6.2 GPS Velocities

A specific pre-condition prevents GPS velocity measurements whenever the system speed is zero and zero velocity measurements have been enabled. In this situation, activation of zero velocity measurements coincides with the deactivation of GPS velocity measurements. When motion resumes, zero velocity measurements are deactivated and GPS velocity measurements are reactivated.

3.6.3 GPS Course-Over-Ground

The GPS COG measurement is based on the direction of the GPS local geographic velocity vector. If the speed is zero, the direction is undefined. Additionally, GPS velocity noise will cause COG to be very noisy at low speeds. Therefore, COG measurements are not processed when speed drops below a specified limit. In MEMS/GPS, the low speed test is implemented as a maximum COG measurement variance test: if the computed COG variance is greater than a specified maximum (or if GPS speed is zero, meaning the COG variance can not be computed), the COG measurement is disabled.

See Section 3.3.3.3 for a derivation of the dependence of COG variance as a function of velocity and velocity variances.

3.6.4 Compass Heading

The accuracy of compass heading measurements depends largely on the magnetic environment in which they required to operate: a large metallic vehicle with many electrical devices onboard is a very poor environment for a magnetic compass; a soldier-carried system would tend to be more magnetically benign. In addition, GPS COG data will be more accurate at vehicular speeds than at soldier speeds. Therefore, compass heading measurement pre-conditions will depend on the intended application.

3.6.4.1 On a Land Vehicle

On a land vehicle, GPS COG measurements are given precedence over compass heading measurements: the relatively high speeds produce accurate COG estimates, and the challenging magnetic environment produces compass data that is likely biased and noisy. In this application, compass heading is used only when the GPS COG measurement has not succeeded. This could be due to a single COG measurement residual rejection, or it could be a longer duration problem due loss of appropriate GPS data or low vehicle speed preventing the formation of a COG measurement (as described above).

Furthermore, a single-axis compass (suitable for a land vehicle) provides valid heading information only when its sensitive axis is “close” to horizontal. When a single-axis compass is used to form heading measurements, a maximum tilt pre-condition is required. In the MEMS/GPS Kalman filter, IMU roll and pitch are tested against a manufacturer-supplied maximum tilt value. The measurement proceeds only when roll and pitch are below the specified limit.

3.6.4.2 On a Soldier

Conversely, on a dismounted soldier, compass heading measurements can be expected to be superior to GPS COG measurements because of the relatively low speeds and benign magnetic environment. However, implementation of the compass heading pre-conditions need not differ from that used for a land vehicle application: compass heading is used only when a GPS COG measurement has not succeeded. While the implementation is the same, it may be advisable to modify the pre-condition model (i.e. the maximum GPS COG measurement variance). Or, GPS COG measurements could simply be disabled. In addition, it is expected that GPS COG measurements (if enabled) would succeed much less frequently than they do on a land vehicle.

The unrestricted movement of a dismounted soldier requires the use of a three-axis magnetometer. The ability of such a sensor to resolve heading at any orientation precludes the need for tilt-based measurement pre-conditions.

3.6.5 Fixed Height

The first fixed height measurement pre-condition is closely connected to those for GPS position measurements: the fixed height measurement is formed only if there has been no successful sensor-based (e.g. GPS) height measurement update. In principle, other sensors could be used to generate height measurements. In practice, only GPS is used. In addition, there must be a valid fixed height to use as the misclosure reference.

3.6.6 Zero Velocities

Horizontal and vertical velocity measurement pre-conditions differ. Firstly, horizontal zero velocity measurements are formed only when the system is stationary. Vertical measurements are formed when the system is stationary, or when the system is moving but the vertical channel is unconstrained, i.e. when there has been

No successful height update from a sensor (like GPS) or from a fixed height;

Nor a successful (GPS) vertical velocity measurement.

3.6.6.1 Zero Speed Detection

Zero speed is detected automatically by comparing sensor-supplied speeds with a fixed maximum.

The speed is calculated as the length of the velocity vector. The maximum allowable calculated speed is taken as the maximum of the ZUPT measurement noise standard deviations and a fixed minimum (e.g. 0.1 m/s).

Each sensor is checked for zero velocity, starting with GPS (if two receivers are active, both may be tested) and ending with the IMU. Zero speed is signalled as soon as any sensor passes the zero speed test. In other words, zero speed is signalled when the calculated speed from any active sensor is less than or equal to the limit.

4 NEW SENSOR STATES

The number of states in a Kalman filter mechanization is often the largest single factor in the determination of computational requirements: the covariance matrix of the state estimates must be inverted. For this reason, only those states that have the largest effects on the total error budget are traditionally included in the state vector. Other known but smaller errors may be neglected all together or are dumped into the Kalman filter process noise, where they are not explicitly estimated but are included in the filter's accuracy estimates (via the state covariance matrix).

MEMS inertial sensors currently have errors that are larger than most of their predecessors. In some cases, the size of these MEMS sensor errors requires that they be more rigorously modelled in a Kalman filter integration. Specifically, they may have to be added to the state vector where they can be explicitly modelled and estimated (assuming they are observable with the available measurements).

The software package used as the basis for the MEMS/GPS Kalman filter came from DIINS (as described in reference [1]). The only inertial errors that can be modelled in the state vector are accelerometer and gyro biases. The following errors are modelled in the process noise covariance:

1. Horizontal and vertical gravity modelling errors;
2. Accelerometer and gyro scale factor errors;
3. Accelerometer and gyro misalignments (relative to an ideal sensor coordinate frame);
4. Accelerometer and gyro random walk errors.

The gravity errors are not sensor errors and are less of a factor when sensor errors are larger: the gravity error modelling will not be changed. The remaining (sensor) errors are candidates for inclusion in the MEMS/GPS state vector.

This chapter describes the methodology needed to model these other inertial errors as states.

4.1 Expanded Gyro and Accelerometer Error Models

A Kalman filter for conventional inertial sensors typically models sensor errors as a dominant bias component and a process noise component that lumps the remaining (much smaller) errors into the velocity and attitude covariance estimates. These amalgamated “process noise” components were included effects arising from:

Anomalous gravity,

Sensor scale factors and misalignments,

Random drifts.

Reference [1] describes the details.

With MEMS sensors, noise levels, scale factor errors and misalignments are much larger than they are for conventional inertial sensors. This section begins with the error analysis used to include gyro and accelerometer scale factor and misalignment states in the MEMS/GPS Kalman filter system. It concludes with an expanded process noise methodology.

4.1.1 New Gyro Model

Inertial scale factor errors essentially describe the errors resulting from imperfect knowledge of the process required to transform the electrical signals produced by the sensors into data that can be processed in an inertial navigator. Scale factor errors are typically divided into a dominant linear portion and smaller deviations from the linear model. The linear part will be modelled explicitly; the non-linear part will included in the process noise covariance matrix.

Misalignment errors describe the rotations needed to align each sensor with an arbitrary, theoretical, Cartesian sensor frame.

To derive the expanded gyro error models, let the error vector, $\bar{\varepsilon}_G^s$, be a generic term containing all gyro errors as measured in the sensor frame.

A gyro error model dependent on (small) misalignments, scale factor errors and biases can be written (in an arbitrary coordinate frame) as

$$\bar{\omega} = \begin{bmatrix} 1 & \mu_{Gyx} & \mu_{Gzx} \\ \mu_{Gxy} & 1 & \mu_{Gzy} \\ \mu_{Gxz} & \mu_{Gyz} & 1 \end{bmatrix} \begin{bmatrix} S_{Gx}\hat{\omega}_{Gx} + \beta_{Gx} \\ S_{Gy}\hat{\omega}_{Gy} + \beta_{Gy} \\ S_{Gz}\hat{\omega}_{Gz} + \beta_{Gz} \end{bmatrix} \qquad (4)$$

where

$\bar{\omega}$ is the true (corrected) rotation rate vector.

$\hat{\bar{\omega}}$ is the measured rotation rate vector.

μ_{Gij} are the misalignment angles of the gyro's (nominally orthogonal) sensitive axes relative to a perfectly orthogonal true frame. Specifically, μ_{Gij} is the (small) misalignment of the gyro sensitive axis *i* relative to the true axis *i* measured in the true *i-j* plane. For example, μ_{Gyx} is the misalignment of the y-gyro in the x-y plane. μ_{Gij} can also be considered the fraction of rotation about the true x-axis that is sensed by the y-gyro.

S_{Gi} is the (full) scale factor applicable to gyro *i*.

β_{Gi} is the bias applicable to gyro *i*.

See reference [4] for additional details. All remaining gyro errors are modelled as process noise.

Note the order of the corrections:

1. The scale factor is applied to the measured rotation rate.

2. The bias error is removed.

3. The small misalignment rotations are applied.

In reference [1], it was assumed that that $\mu_{Gij} \cong 0$ and $S_{Gk} \cong 1$, so that $\vec{\varepsilon}_G \cong \vec{\beta}_G$.

To include misalignments and scale factors, begin by expanding the above equation.

$$\vec{\omega} = \begin{bmatrix} S_{Gx}\hat{\omega}_{Gx} + \beta_{Gx} + \mu_{Gyx}\left(S_{Gy}\hat{\omega}_{Gy} + \beta_{Gy}\right) + \mu_{Gzx}\left(S_{Gz}\hat{\omega}_{Gz} + \beta_{Gz}\right) \\ \mu_{Gxy}\left(S_{Gx}\hat{\omega}_{Gx} + \beta_{Gx}\right) + S_{Gy}\hat{\omega}_{Gy} + \beta_{Gy} + \mu_{Gzy}\left(S_{Gz}\hat{\omega}_{Gz} + \beta_{Gz}\right) \\ \mu_{Gxz}\left(S_{Gx}\hat{\omega}_{Gx} + \beta_{Gx}\right) + \mu_{Gyz}\left(S_{Gy}\hat{\omega}_{Gy} + \beta_{Gy}\right) + S_{Gz}\hat{\omega}_{Gz} + \beta_{Gz} \end{bmatrix}$$

This can be simplified by assuming that misalignments and scale factor errors are much smaller than the signal and the biases and ignoring second order terms. Note that the scale factor terms are the full scale factors: $S_i = 1 + c_i$, where c_i is the (small) scale factor error. Now,

$$\vec{\omega} = \begin{bmatrix} S_{Gx}\hat{\omega}_{Gx} + \beta_{Gx} + \mu_{Gyx}\left(\hat{\omega}_{Gy} + \beta_{Gy}\right) + \mu_{Gzx}\left(\hat{\omega}_{Gz} + \beta_{Gz}\right) \\ \mu_{Gxy}\left(\hat{\omega}_{Gx} + \beta_{Gx}\right) + S_{Gy}\hat{\omega}_{Gy} + \beta_{Gy} + \mu_{Gzy}\left(\hat{\omega}_{Gz} + \beta_{Gz}\right) \\ \mu_{Gxz}\left(\hat{\omega}_{Gx} + \beta_{Gx}\right) + \mu_{Gyz}\left(\hat{\omega}_{Gy} + \beta_{Gy}\right) + S_{Gz}\hat{\omega}_{Gz} + \beta_{Gz} \end{bmatrix}$$

The expanded expression for gyro errors can now be derived

$$\vec{\varepsilon}_G = \vec{\omega} - \hat{\vec{\omega}} = \begin{bmatrix} \beta_{Gx} + \sigma_{Gx}\hat{\omega}_{Gx} + \mu_{Gyx}\left(\hat{\omega}_{Gy} + \beta_{Gy}\right) + \mu_{Gzx}\left(\hat{\omega}_{Gz} + \beta_{Gz}\right) \\ \beta_{Gy} + \sigma_{Gy}\hat{\omega}_{Gy} + \mu_{Gxy}\left(\hat{\omega}_{Gx} + \beta_{Gx}\right) + \mu_{Gzy}\left(\hat{\omega}_{Gz} + \beta_{Gz}\right) \\ \beta_{Gz} + \sigma_{Gz}\hat{\omega}_{Gz} + \mu_{Gxz}\left(\hat{\omega}_{Gx} + \beta_{Gx}\right) + \mu_{Gyz}\left(\hat{\omega}_{Gy} + \beta_{Gy}\right) \end{bmatrix}$$

An exponentially correlated first-order Gauss-Markov model will be used in the filter to characterize the gyro errors. It is completely described by a variance and a correlation time. This is the same kind of model that is used in the software for almost all sensor errors.

The Kalman filter dynamics matrix describes how the state vector changes in time. The effects due to the gyro errors are derived through partial differentiation. Following the development in reference [1], begin with the equations of motion in sections 4.3.2 and 4.3.3. Equation (4.42) is a first-order differential equation for the attitude errors (the -angle vector). It is repeated here:

$$\dot{\vec{\psi}} = -\left(\vec{\omega}_{IE}^{c} + \vec{\omega}_{Ec}^{c}\right) \times \vec{\psi} + C_s^p \vec{\varepsilon}_G^{s}$$

The c-frame is the wander azimuth *computer* frame,

The I-frame is the inertial frame,

The E-frame is the earth-centred, earth-fixed frame,

The s-frame is the sensor frame,

The p-frame is the wander azimuth *platform* frame.

The dynamics matrix is derived via the partial differentiation of equation (4.42) with respect to the Kalman filter states, as described in reference [1], Chapter 7. For the present purposes, we are concerned only with the gyro error term. There are no dependencies on non-gyro error states.

$$\frac{\partial \vec{\varepsilon}_G}{\partial \vec{\beta}_G} = \begin{bmatrix} 1 & \mu_{Gyx} & \mu_{Gzx} \\ \mu_{Gxy} & 1 & \mu_{Gzy} \\ \mu_{Gxz} & \mu_{Gyz} & 1 \end{bmatrix} \cong \begin{bmatrix} 1 & \hat{\mu}_{Gyx} & \hat{\mu}_{Gzx} \\ \hat{\mu}_{Gxy} & 1 & \hat{\mu}_{Gzy} \\ \hat{\mu}_{Gxz} & \hat{\mu}_{Gyz} & 1 \end{bmatrix} \quad (5)$$

$$\frac{\partial \vec{\varepsilon}_G}{\partial \vec{\sigma}_G} = \begin{bmatrix} \hat{\omega}_{Gx} & 0 & 0 \\ 0 & \hat{\omega}_{Gy} & 0 \\ 0 & 0 & \hat{\omega}_{Gz} \end{bmatrix} \quad (6)$$

$$\frac{\partial \vec{\varepsilon}_G}{\partial \vec{\mu}_G} = \begin{bmatrix} (\hat{\omega}_{Gy} + \beta_{Gy}) & (\hat{\omega}_{Gz} + \beta_{Gz}) & 0 & 0 & 0 & 0 \\ 0 & 0 & (\hat{\omega}_{Gx} + \beta_{Gx}) & (\hat{\omega}_{Gz} + \beta_{Gz}) & 0 & 0 \\ 0 & 0 & 0 & 0 & (\hat{\omega}_{Gx} + \beta_{Gx}) & (\hat{\omega}_{Gy} + \beta_{Gy}) \end{bmatrix}$$

$$\cong \begin{bmatrix} (\hat{\omega}_{Gy} + \hat{\beta}_{Gy}) & (\hat{\omega}_{Gz} + \hat{\beta}_{Gz}) & 0 & 0 & 0 & 0 \\ 0 & 0 & (\hat{\omega}_{Gx} + \hat{\beta}_{Gx}) & (\hat{\omega}_{Gz} + \hat{\beta}_{Gz}) & 0 & 0 \\ 0 & 0 & 0 & 0 & (\hat{\omega}_{Gx} + \hat{\beta}_{Gx}) & (\hat{\omega}_{Gy} + \hat{\beta}_{Gy}) \end{bmatrix}$$

(7)

where

$$\vec{\mu}_G^T = \begin{bmatrix} \mu_{Gyx} & \mu_{Gzx} & \mu_{Gxy} & \mu_{Gzy} & \mu_{Gxz} & \mu_{Gyz} \end{bmatrix}$$

Hats added to the error terms indicate filter-estimated values. Recall that all three of theses dynamics matrix sub-blocks have to be pre-multiplied by the sensor frame to platform frame direction cosine matrix.

The misalignments add as many states as the biases and scale factors combined. And, the misalignment states are not expected to add as much information to the solution as the other gyro errors. Therefore, they will likely remain in the indirect process noise covariance model and not be modelled in the state vector. In this simplified model, The misalignment sub-block will obviously be removed from the dynamics matrix.

The scale factor sub-block is unchanged.

The bias sub-block is simplified to a 3x3 identity matrix (prior to pre-multiplication with C_s^p).

Note that the INS velocity errors are also dependent on gyro errors (see reference [1], equation (4.59)). In this velocity differential equation, the $C_s^p \vec{\varepsilon}_G^s$ term is pre-multiplied by the wander azimuth velocity cross product, $\left(\vec{v}^p \times\right)$ - see the next section.

4.1.2 New Accelerometer Model

Note that the gyro correction model, equation (4), is quite generic: it can just as easily be applied to accelerometer bias, scale factor, and misalignment corrections, as follows.

$$\vec{a} = \begin{bmatrix} 1 & \mu_{Ayx} & \mu_{Azx} \\ \mu_{Axy} & 1 & \mu_{Azy} \\ \mu_{Axz} & \mu_{Ayz} & 1 \end{bmatrix} \begin{bmatrix} S_{Ax}\hat{a}_{Ax} + \beta_{Ax} \\ S_{Ay}\hat{a}_{Ay} + \beta_{Ay} \\ S_{Az}\hat{a}_{Az} + \beta_{Az} \end{bmatrix}$$

Equation (4.59) in reference [1] is the first-order differential equation for the velocity errors:

$$\begin{aligned} \dot{\tilde{\vec{v}}}^p = \quad & -\left(2\vec{\omega}_{IE}^c + \vec{\omega}_{Ec}^c\right) \times \hat{\vec{v}}^p + \vec{v}^p \times \left(\vec{\omega}_{IE}^c \times \vec{\psi}\right) + \\ & C_s^p \vec{\varepsilon}_A^s + \vec{v}^p \times \left(C_s^p \vec{\varepsilon}_G^s\right) + \tilde{\vec{g}}_{ct}^{cp} \end{aligned}$$

Only the accelerometer and gyro error terms are of interest here. Note that the accelerometer error term is completely analogous to the corresponding gyro equation: the accelerometer dynamics matrix sub-blocks can be derived using the corresponding gyro error sub-blocks. Simply replace

$\vec{\omega}$ by $\vec{a}$,

$\vec{\varepsilon}_G$ by $\vec{\varepsilon}_A$,

$\vec{\beta}_G$ by $\vec{\beta}_A$,

$\vec{c}_G$ by $\vec{c}_A$, and

$\vec{\mu}_G$ by $\vec{\mu}_A$

in equations (5), (6), and (7).

However, note the velocity error equation is also dependent on gyro errors. The velocity error rows of the dynamics matrix will therefore contain terms in the gyro error columns. Fortunately, the gyro error elements are simply those in the attitude error rows pre-multiplied by the velocity cross-product matrix.

Note that the acceleration should include gravity to properly estimate the scale factors.

4.1.3 New IMU Process Noise Models

In conventional inertial sensors, noise levels are very low; in MEMS sensors, they are high. To model the noise accurately, specific noise terms are needed in the process noise covariance matrix. Sensor noise may be reported in terms of the noise itself or in terms of the integral of the noise - random walk.

Reference [3] describes the relationship between noise and random walk. For the continuous process, the state variable differential equation for a random walk process (with its variance) is:

$$\begin{aligned} \dot{x} &= w \\ \dot{p} &= q \end{aligned} \qquad (8)$$

where

> *x* is the random walk variable,
>
> *w* is the noise,
>
> *p* is the random walk variance,
>
> *q* is the noise variance.

In the MEMS/GPS processing stream, IMU sensor data is first collected and run through the strapdown navigator (reference [2]) – effectively an integration process with gyro angular rates integrated once to give angular changes and accelerometer rates integrated once to give velocity changes and again to give change in position. Noise on the sensor inputs produces random walk error in the strapdown angular and velocity outputs. Integrating the second line in equation (8) gives the variance of the random walk process in terms of constant noise variance and time:

$$p = \int_0^\tau q\,dt = q\int_0^\tau dt = q\tau$$

The variance of the random walk process increases linearly with time at a rate equal to the noise variance.

It is important to keep track of the units in this analysis: random walk is generally reported in this context as so-called Angular Random Walk (ARW) and Velocity Random Walk (VRW) with units given in terms of standard deviations as degrees per square root hour and (metres per second) per square root hour. The corresponding noise standard deviation units are degrees per hour and (metres per second) per hour. Rates may also be expressed in terms of seconds (e.g. degrees per square root second and degrees per second) when they are large.

The strapdown data is passed on to the MEMS/GPS Kalman filter where the final navigation, error control and accuracy estimates are generated. To accurately propagate the IMU error models (via the filter propagation step), the velocity and attitude random walk errors must be accounted for. In the filter, this is done in the process noise covariance matrix.

Can a model be developed that will take **either** the noise variance or the random walk variance and give the appropriate velocity and attitude state process noise variances for a discrete Kalman filter? It should be noted that one model will be used for all accelerometer noise, and another for all gyro noise.

4.1.3.1 Units Conversion

Accelerometer and gyro random walk errors are each commonly quoted in two different units. The conversions between the units are presented here.

4.1.3.1.1 Accelerometer Random Walk Units

Accelerometer random walk errors are commonly quoted as

Acceleration per square root frequency (Rate or Acceleration Random Walk), or

Velocity per square root time (Velocity Random Walk).

A useful starting point is the following conversion:

$$1\frac{m/s^2}{\sqrt{Hz}} = 1\sqrt{\frac{m^2/s^4}{1/s}} = 1\sqrt{\frac{m^2/s^2}{s} \times \frac{3600\,s}{1\,hr}} = 60\frac{m/s}{\sqrt{hr}}$$

Therefore, a random walk error given in $\frac{m/s^2}{\sqrt{Hz}}$ is converted to $\frac{m/s}{\sqrt{hr}}$ by multiplying by 60.

Note that $\frac{m/s}{\sqrt{s}}$ is numerically equivalent to $\frac{m/s^2}{\sqrt{Hz}}$.

Conversions of the velocity and acceleration units is relatively simply. For example, if acceleration is given in micro ( ) g’s,

$$1\,\frac{m/s^2}{\sqrt{Hz}} = \frac{10^6\mu g}{1g}\cdot\frac{1g}{9.8\,m/s^2}\frac{m/s^2}{\sqrt{Hz}} = 102{,}040\,\frac{\mu g}{\sqrt{Hz}}$$

And

$$1\,\frac{m/s^2}{\sqrt{Hz}} = 60\,\frac{m/s}{\sqrt{hr}} = 102{,}040\,\frac{\mu g}{\sqrt{Hz}}$$

So that

$$1\frac{\mu g}{\sqrt{Hz}} = \frac{60}{102{,}040}\frac{m/s}{\sqrt{hr}} = \frac{1}{1700}\frac{m/s}{\sqrt{hr}}$$

A random walk error given in $\frac{\mu g}{\sqrt{Hz}}$ is converted to $\frac{m/s}{\sqrt{hr}}$ by dividing by 1700.

Other coordinate conversions can be derived in a similar fashion. And, of course, the inverse conversions are simply the numerical inverses of those given.

4.1.3.1.2 Gyro Random Walk Units

Gyro random walk errors are commonly quoted as

Angular rate per square root frequency (Rate Random Walk), or

Angle per square root time (Angular Random Walk).

Let’s begin by converting $\frac{\deg/s}{\sqrt{Hz}}$ to $\frac{\deg}{\sqrt{hr}}$. Note that this is completely analogous to the accelerometer conversion from $\frac{m/s^2}{\sqrt{Hz}}$ to $\frac{m/s}{\sqrt{hr}}$.

A random walk error given in $\frac{\deg/s}{\sqrt{Hz}}$ is converted to $\frac{\deg}{\sqrt{hr}}$ by dividing by 60. And again, $\frac{\deg}{\sqrt{s}}$ is numerically equivalent to $\frac{\deg/s}{\sqrt{Hz}}$.

Conversions to other angle or time units would proceed in the normal fashion.

5 SUMMARY

This report describes those parts of the MEMS/GPS Kalman filter that differ significantly from earlier work (the DIINS Kalman filter are presented of reference [1]). It describes

1. New data-driven filter timing procedures;
2. New and modified Kalman filter measurement algorithms;
3. New IMU error states that were added to try to better model the larger MEMS IMU errors.

REFERENCES

[1] *Kalman Filter Design of the Dual Inertial Integrated Navigation System*, Dale Arden Consulting, Report #DAC-02-97, Ottawa, 1997.

[2] *The MEMS/GPS Strapdown Navigator*, Dale Arden, Dale Arden Consulting, (under the current contract) 2003.

[3] *Applied Optimal Estimation*, Arthur Gelb, ed., M.I.T. Press, Cambridge, Massachusetts, 1974.

[4] *Self-Alignment and Navigation Algorithms for DREO Navigation Laboratory Heading Reference Unit*, Carole R.M. Bolduc, M.Eng. Thesis, Carleton University, Ottawa, 1995.

[5] *C100 Compass Engine, Technical Manual*, Part No. 54-0044 Revision G2, KVH Industries, Inc., Middletown, RI.

APPENDIX A ALTERNATIVE GPS VELOCITY MEASUREMENT MODELS

A.1 Open-Loop Model

In section 3.3.2 , the GPS velocity measurement equation was derived. It was given as

$$\vec{z}^{\,g}_{v_{GPS}} \cong \hat{\vec{v}}^{\,g}_{GPS} - C^g_t \left[I + \left(\delta\vec{\theta}\times\right) + \left(\vec{\psi}\times\right)\right]\hat{\vec{v}}^{\,w}_{IMU}$$

The measurement model resulting from this equation will be derived below.

The objective is identification of all state dependencies. Begin by expanding the IMU term, temporarily making use of the -angles (see reference [1]):

$$C_t^g\left[I+\left(\vec{\phi}\times\right)\right]\hat{\vec{v}}_{IMU}^w=\begin{bmatrix}\cos\alpha_t & -\sin\alpha_t & 0\\ -\sin\alpha_t & -\cos\alpha_t & 0\\ 0 & 0 & -1\end{bmatrix}\begin{bmatrix}1 & -\phi_z & \phi_y\\ \phi_z & 1 & -\phi_x\\ -\phi_y & \phi_x & 1\end{bmatrix}\begin{bmatrix}v_{IMU_X}^w\\ v_{IMU_Y}^w\\ v_{IMU_Z}^w\end{bmatrix}$$

$$=\begin{bmatrix}\cos\alpha_t & -\sin\alpha_t & 0\\ -\sin\alpha_t & -\cos\alpha_t & 0\\ 0 & 0 & -1\end{bmatrix}\begin{bmatrix}v_{IMU_X}^w-\phi_z v_{IMU_Y}^w+\phi_y v_{IMU_Z}^w\\ \phi_z v_{IMU_X}^w+v_{IMU_Y}^w-\phi_x v_{IMU_Z}^w\\ -\phi_y v_{IMU_X}^w+\phi_x v_{IMU_Y}^w+v_{IMU_Z}^w\end{bmatrix}$$

$$=\begin{bmatrix}\cos\alpha_t\left(v_{IMU_X}^w-\phi_z v_{IMU_Y}^w+\phi_y v_{IMU_Z}^w\right)-\sin\alpha_t\left(\phi_z v_{IMU_X}^w+v_{IMU_Y}^w-\phi_x v_{IMU_Z}^w\right)\\ -\sin\alpha_t\left(v_{IMU_X}^w-\phi_z v_{IMU_Y}^w+\phi_y v_{IMU_Z}^w\right)-\cos\alpha_t\left(\phi_z v_{IMU_X}^w+v_{IMU_Y}^w-\phi_x v_{IMU_Z}^w\right)\\ \phi_y v_{IMU_X}^w-\phi_x v_{IMU_Y}^w-v_{IMU_Z}^w\end{bmatrix}$$

To be completely rigorous, the true wander angle should written

$$\alpha_t=\alpha_{IMU}+\delta\alpha_{IMU}$$

From reference [1],

$$\delta\alpha_{IMU}=\frac{\left(\delta r_{IMU_X}^w\sin\alpha_{IMU}+\delta r_{IMU_Y}^w\cos\alpha_{IMU}\right)\tan\left|\phi_{GPS}\right|}{R_E+h_{GPS}}$$

where $\left|\phi_{GPS}\right|$ is the absolute value of the GPS latitude.

Using small angle approximations on $\delta\alpha_{IMU}$, we get

$$\cos\alpha_t\cong\cos\alpha_{IMU}-\frac{\left(\delta r_{IMU_X}^w\sin^2\alpha_{IMU}+\delta r_{IMU_Y}^w\sin\alpha_{IMU}\cos\alpha_{IMU}\right)\tan\left|\phi_{GPS}\right|}{R_E+h_{GPS}}$$
$$\sin\alpha_t\cong\sin\alpha_{IMU}+\frac{\left(\delta r_{IMU_X}^w\sin\alpha_{IMU}\cos\alpha_{IMU}+\delta r_{IMU_Y}^w\cos^2\alpha_{IMU}\right)\tan\left|\phi_{GPS}\right|}{R_E+h_{GPS}}$$

Expand the α_t terms, neglecting second order terms:

$$C_t^g\left[I+\left(\vec{\phi}\times\right)\right]\hat{\vec{v}}_{IMU}^w=$$
$$\begin{bmatrix}\cos\alpha_t v_{IMU_X}^w-\sin\alpha_t v_{IMU_Y}^w+\cos\alpha_{IMU}\left(-\phi_z v_{IMU_Y}^w+\phi_y v_{IMU_Z}^w\right)-\sin\alpha_{IMU}\left(\phi_z v_{IMU_X}^w-\phi_x v_{IMU_Z}^w\right)\\ -\cos\alpha_t v_{IMU_Y}^w-\sin\alpha_t v_{IMU_X}^w-\sin\alpha_{IMU}\left(-\phi_z v_{IMU_Y}^w+\phi_y v_{IMU_Z}^w\right)-\cos\alpha_{IMU}\left(\phi_z v_{IMU_X}^w-\phi_x v_{IMU_Z}^w\right)\\ \phi_y v_{IMU_X}^w-\phi_x v_{IMU_Y}^w-v_{IMU_Z}^w\end{bmatrix}$$

Collect terms according to the   -angles:

$$C_t^g\left[I+\left(\vec{\phi}\times\right)\right]\hat{\vec{v}}_{IMU}^{w} =$$

$$\begin{bmatrix} \cos\alpha_t v_{IMU_X}^w - \sin\alpha_t v_{IMU_Y}^w + \sin\alpha_{IMU} v_{IMU_Z}^w \phi_x + \cos\alpha_{IMU} v_{IMU_Z}^w \phi_y - \left(\cos\alpha_{IMU} v_{IMU_Y}^w + \sin\alpha_{IMU} v_{IMU_X}^w\right)\phi_z \\ -\cos\alpha_t v_{IMU_Y}^w - \sin\alpha_t v_{IMU_X}^w + \cos\alpha_{IMU} v_{IMU_Z}^w \phi_x - \sin\alpha_{IMU} v_{IMU_Z}^w \phi_y + \left(\sin\alpha_{IMU} v_{IMU_Y}^w - \cos\alpha_{IMU} v_{IMU_X}^w\right)\phi_z \\ \phi_y v_{IMU_X}^w - \phi_x v_{IMU_Y}^w - v_{IMU_Z}^w \end{bmatrix}$$

To complete the derivation, the α_t and $\vec{\phi}$ terms must be expanded (they are both dependent on error states). From reference [1] once again,

$$\vec{\phi} = \delta\vec{\theta} + \vec{\psi} = \begin{bmatrix} \dfrac{\delta r_{IMU_Y}^w}{R_E + h_{GPS}} + \psi_x \\ -\dfrac{\delta r_{IMU_X}^w}{R_E + h_{GPS}} + \psi_y \\ \psi_z \end{bmatrix}$$

A.1.1 First Row of the H-Matrix

But, to keep the equations to a manageable size, the partial differentiation will be started next, one term at a time, neglecting second order effects, starting with the X GPS velocity measurement row and X IMU position error column:

$$\begin{aligned} H_{GPS_v}(\partial z_{v_X}^g, \delta r_{IMU_X}) &= \frac{\partial z_{v_X}^g}{\partial\left(\delta r_{IMU_X}\right)} \\ &= \frac{\partial\left(-\cos\alpha_t v_{IMU_X}^w\right)}{\partial\left(\delta r_{IMU_X}\right)} + \frac{\partial\left(\sin\alpha_t v_{IMU_Y}^w\right)}{\partial\left(\delta r_{IMU_X}\right)} + \frac{\partial\left(-\cos\alpha_{IMU} v_{IMU_Z}^w \phi_y\right)}{\partial\left(\delta r_{IMU_X}\right)} \\ &= \frac{\sin^2\alpha_{IMU}\tan\left|\phi_{GPS}\right| v_{IMU_X}^w + \sin\alpha_{IMU}\cos\alpha_{IMU}\tan\left|\phi_{GPS}\right| v_{IMU_Y}^w + \cos\alpha_{IMU} v_{IMU_Z}^w}{R_E + h_{GPS}} \\ &= \frac{-v_{IMU_Y}^g \sin\alpha_{IMU}\tan\left|\phi_{GPS}\right| - v_{IMU_Z}^g \cos\alpha_{IMU}}{R_E + h_{GPS}} \end{aligned}$$

Similarly,

$$H_{GPS_v}(\partial z_{v_X}^g, \delta r_{IMU_Y}) = \frac{\partial z_{v_X}^g}{\partial\left(\delta r_{IMU_Y}\right)}$$

$$= \frac{\partial\left(-\cos\alpha_t v_{IMU_X}^w\right)}{\partial\left(\delta r_{IMU_Y}\right)} + \frac{\partial\left(\sin\alpha_t v_{IMU_Y}^w\right)}{\partial\left(\delta r_{IMU_Y}\right)} + \frac{\partial\left(-\sin\alpha_{IMU} v_{IMU_Z}^w \phi_x\right)}{\partial\left(\delta r_{IMU_Y}\right)}$$

$$= \frac{\sin\alpha_{IMU}\cos\alpha_{IMU}\tan|\phi_{GPS}|v_{IMU_X}^w + \cos^2\alpha_{IMU}\tan|\phi_{GPS}|v_{IMU_Y}^w - \sin\alpha_{IMU} v_{IMU_Z}^w}{R_E + h_{GPS}}$$

$$= \frac{-v_{IMU_Y}^g\cos\alpha_{IMU}\tan|\phi_{GPS}| + v_{IMU_Z}^g\sin\alpha_{IMU}}{R_E + h_{GPS}}$$

$$H_{GPS_v}(\partial z_{v_X}^g, \delta r_{IMU_Z}) = \frac{\partial z_{v_X}^g}{\partial\left(\delta r_{IMU_Z}\right)} = 0$$

The IMU velocity error elements (where $\hat{\vec{v}}_{IMU}^p = \vec{v}^p - \delta\vec{v}_{IMU}^p$) are

$$H_{GPS_v}(\partial z_{v_X}^g, \delta v_{IMU_X}) = \frac{\partial z_{v_X}^g}{\partial\left(\delta v_{IMU_X}\right)} = \cos\alpha_{IMU}$$

$$H_{GPS_v}(\partial z_{v_X}^g, \delta v_{IMU_Y}) = \frac{\partial z_{v_X}^g}{\partial\left(\delta v_{IMU_Y}\right)} = -\sin\alpha_{IMU}$$

$$H_{GPS_v}(\partial z_{v_X}^g, \delta v_{IMU_Z}) = \frac{\partial z_{v_X}^g}{\partial\left(\delta v_{IMU_Z}\right)} = 0$$

Continuing with IMU attitude errors,

$$H_{GPS_v}(\partial z_{v_X}^g, \psi_x) = \frac{\partial z_{v_X}^g}{\partial(\psi_x)} = \frac{\partial\left(-\sin\alpha_{IMU} v_{IMU_Z}^w \phi_x\right)}{\partial(\psi_x)} = -\sin\alpha_{IMU} v_{IMU_Z}^w = v_{IMU_Z}^g \sin\alpha_{IMU}$$

$$H_{GPS_v}(\partial z_{v_X}^g, \psi_y) = \frac{\partial z_{v_X}^g}{\partial(\psi_y)} = \frac{\partial\left(-\cos\alpha_{IMU} v_{IMU_Z}^w \phi_y\right)}{\partial(\psi_y)} = -\cos\alpha_{IMU} v_{IMU_Z}^w = v_{IMU_Z}^g \cos\alpha_{IMU}$$

$$H_{GPS_v}(\partial z_{v_X}^g, \psi_z) = \frac{\partial z_{v_X}^g}{\partial(\psi_z)} = \frac{\partial\left(\left(\sin\alpha_{IMU} v_{IMU_X}^w + \cos\alpha_{IMU} v_{IMU_Y}^w\right)\phi_x\right)}{\partial(\psi_z)} = \sin\alpha_{IMU} v_{IMU_X}^w + \cos\alpha_{IMU} v_{IMU_Y}^w = -v_{IMU_Y}^g$$

A.1.2 Second Row of the H-Matrix

The Y GPS velocity measurement matrix row is similar to the X row:

$$H_{GPS_v}(\partial z_{v_Y}^g, \delta r_{IMU_X}) = \frac{\partial z_{v_Y}^g}{\partial(\delta r_{IMU_X})}$$

$$= \frac{\partial\left(\sin\alpha_t v_{IMU_X}^w\right)}{\partial(\delta r_{IMU_X})} + \frac{\partial\left(\cos\alpha_t v_{IMU_Y}^w\right)}{\partial(\delta r_{IMU_X})} + \frac{\partial\left(\sin\alpha_{IMU} v_{IMU_Z}^w \phi_y\right)}{\partial(\delta r_{IMU_X})}$$

$$= \frac{\sin\alpha_{IMU}\ \cos\alpha_{IMU}\ \tan|\phi_{GPS}|v_{IMU_X}^w - \sin^2\alpha_{IMU}\ \tan|\phi_{GPS}|v_{IMU_Y}^w - \sin\alpha_{IMU} v_{IMU_Z}^w}{R_E + h_{GPS}}$$

$$= \frac{v_{IMU_X}^g\ \sin\alpha_{IMU}\ \tan|\phi_{GPS}| + v_{IMU_Z}^g\ \sin\alpha_{IMU}}{R_E + h_{GPS}}$$

$$H_{GPS_v}(\partial z_{v_Y}^g, \delta r_{IMU_Y}) = \frac{\partial z_{v_Y}^g}{\partial(\delta r_{IMU_Y})}$$

$$= \frac{\partial\left(\sin\alpha_t v_{IMU_X}^w\right)}{\partial(\delta r_{IMU_Y})} + \frac{\partial\left(\cos\alpha_t v_{IMU_Y}^w\right)}{\partial(\delta r_{IMU_Y})} + \frac{\partial\left(-\cos\alpha_{IMU} v_{IMU_Z}^w \phi_x\right)}{\partial(\delta r_{IMU_Y})}$$

$$= \frac{\cos^2\alpha_{IMU}\ \tan|\phi_{GPS}|v_{IMU_X}^w - \sin\alpha_{IMU}\ \cos\alpha_{IMU}\ \tan|\phi_{GPS}|v_{IMU_Y}^w - \cos\alpha_{IMU} v_{IMU_Z}^w}{R_E + h_{GPS}}$$

$$= \frac{v_{IMU_X}^g\ \cos\alpha_{IMU}\ \tan|\phi_{GPS}| + v_{IMU_Z}^g\ \cos\alpha_{IMU}}{R_E + h_{GPS}}$$

$$H_{GPS_v}(\partial z_{v_Y}^g, \delta r_{IMU_Z}) = \frac{\partial z_{v_Y}^g}{\partial(\delta r_{IMU_Z})} = 0$$

$$H_{GPS_v}(\partial z_{v_Y}^g, \delta v_{IMU_X}) = \frac{\partial z_{v_Y}^g}{\partial(\delta v_{IMU_X})} = -\sin\alpha_{IMU}$$

$$H_{GPS_v}(\partial z_{v_Y}^g, \delta v_{IMU_Y}) = \frac{\partial z_{v_Y}^g}{\partial(\delta v_{IMU_Y})} = -\cos\alpha_{IMU}$$

$$H_{GPS_v}(\partial z_{v_Y}^g, \delta v_{IMU_Z}) = \frac{\partial z_{v_Y}^g}{\partial(\delta v_{IMU_Z})} = 0$$

$$H_{GPS_v}(\partial z_{v_Y}^g, \psi_x) = \frac{\partial z_{v_Y}^g}{\partial(\psi_x)} = \frac{\partial\left(-\cos\alpha_{IMU} v_{IMU_Z}^w \phi_x\right)}{\partial(\psi_x)} = -v_{IMU_Z}^w\ \cos\alpha_{IMU} = v_{IMU_Z}^g\ \cos\alpha_{IMU}$$

$$H_{GPS_v}(\partial z^g_{v_Y}, \psi_y) = \frac{\partial z^g_{v_Y}}{\partial(\psi_y)} = \frac{\partial\left(\sin\alpha_{IMU} v^w_{IMU_Z}\phi_y\right)}{\partial(\psi_y)} = v^w_{IMU_Z}\sin\alpha_{IMU} = -v^g_{IMU_Z}\sin\alpha_{IMU}$$

$$H_{GPS_v}(\partial z^g_{v_Y}, \psi_z) = \frac{\partial z^g_{v_Y}}{\partial(\psi_z)} = \frac{\partial\left(\cos\alpha_{IMU} v^w_{IMU_X} - \sin\alpha_{IMU} v^w_{IMU_Y}\right)\phi_z}{\partial(\psi_z)} = \left(\cos\alpha_{IMU} v^w_{IMU_X} - \sin\alpha_{IMU} v^w_{IMU_Y}\right) = v^g_{IMU_X}$$

A.1.3 Third Row of the H-Matrix

$$H_{GPS_v}(\partial z^g_{v_Z}, \delta r_{IMU_X}) = \frac{\partial z^g_{v_Z}}{\partial(\delta r_{IMU_X})} = \frac{\partial\left(-\phi_y v^w_{IMU_X}\right)}{\partial(\delta r_{IMU_X})} = \frac{v^w_{IMU_X}}{R_E + h_{GPS}}$$

$$H_{GPS_v}(\partial z^g_{v_Z}, \delta r_{IMU_Y}) = \frac{\partial z^g_{v_Z}}{\partial(\delta r_{IMU_Y})} = \frac{\partial\left(\phi_x v^w_{IMU_Y}\right)}{\partial(\delta r_{IMU_Y})} = \frac{v^w_{IMU_Y}}{R_E + h_{GPS}}$$

$$H_{GPS_v}(\partial z^g_{v_Z}, \delta r_{IMU_Z}) = \frac{\partial z^g_{v_Z}}{\partial(\delta r_{IMU_Y})} = 0$$

$$H_{GPS_v}(\partial z^g_{v_Z}, \delta v_{IMU_X}) = \frac{\partial\left(v^w_{IMU_Z}\right)}{\partial(\delta v_{IMU_X})} = 0$$

$$H_{GPS_v}(\partial z^g_{v_Z}, \delta v_{IMU_Y}) = \frac{\partial\left(v^w_{IMU_Z}\right)}{\partial(\delta v_{IMU_Y})} = 0$$

$$H_{GPS_v}(\partial z^g_{v_Z}, \delta v_{IMU_Z}) = \frac{\partial\left(v^w_{IMU_Z}\right)}{\partial(\delta v_{IMU_Z})} = -1$$

$$H_{GPS_v}(\partial z^g_{v_Z}, \psi_x) = \frac{\partial z^g_{v_Z}}{\partial(\psi_x)} = \frac{\partial\left(\phi_x v^w_{IMU_Y}\right)}{\partial(\psi_x)} = v^w_{IMU_Y}$$

$$H_{GPS_v}(\partial z^g_{v_Z}, \psi_y) = \frac{\partial z^g_{v_Z}}{\partial(\psi_y)} = \frac{\partial\left(-\phi_y v^w_{IMU_X}\right)}{\partial(\psi_y)} = -v^w_{IMU_X}$$

$$H_{GPS_v}(\partial z^g_{v_Z}, \psi_z) = \frac{\partial z^g_{v_Z}}{\partial(\psi_z)} = 0$$

A.1.4 The Full H-Matrix

The full GPS velocity measurement H-matrix is presented below. Note that GPS velocity states are not represented in this matrix. If GPS velocity is used, expand GPS velocity to

$$\hat{\vec{v}}_{GPS}^{g} = \vec{v}^{g} - \delta\vec{v}_{GPS}^{g}$$

Then, add a 3 by 1 minus identity matrix (-*I*) in the GPS velocity state columns of the H-matrix.

One note regarding the H-matrix below: the velocities in the first two rows are in the local geographic frame; the velocities in the third row are in the wander azimuth frame.

$$H_{GPS_V} = \frac{\partial \bar{z}^g_{v_{GPS}}}{\partial(\bar{x})} =$$

$$\begin{bmatrix} \frac{-v^g_{IMU_Y}\sin\alpha_{IMU}\tan\left|\phi_{GPS}\right| - v^g_{IMU_Z}\cos\alpha_{IMU}}{R_E + h_{GPS}} & \frac{-v^g_{IMU_Y}\cos\alpha_{IMU}\tan\left|\phi_{GPS}\right| + v^g_{IMU_Z}\sin\alpha_{IMU}}{R_E + h_{GPS}} & 0 & & v^g_{IMU_Z}\sin\alpha_{IMU} & v^g_{IMU_Z}\cos\alpha_{IMU} & -v^g_{IMU_Y} & \ldots & 0 \\ \frac{v^g_{IMU_X}\sin\alpha_{IMU}\tan\left|\phi_{GPS}\right| + v^g_{IMU_Z}\sin\alpha_{IMU}}{R_E + h_{GPS}} & \frac{v^g_{IMU_X}\cos\alpha_{IMU}\tan\left|\phi_{GPS}\right| + v^g_{IMU_Z}\cos\alpha_{IMU}}{R_E + h_{GPS}} & 0 & C^w_g & v^g_{IMU_Z}\cos\alpha_{IMU} & -v^g_{IMU_Z}\sin\alpha_{IMU} & v^g_{IMU_X} & \ldots & 0 \\ \frac{v^w_{IMU_X}}{R_E + h_{GPS}} & \frac{v^w_{IMU_Y}}{R_E + h_{GPS}} & 0 & & v^w_{IMU_Y} & -v^w_{IMU_X} & 0 & \ldots & 0 \end{bmatrix}$$

A.2 An Alternative GPS Velocity Formulation

In section 3.3.2 , a GPS velocity was constructed as follows:

$$\vec{z}^{\,g}_{v_{GPS}} = \hat{\vec{v}}^{\,g}_{GPS} - C^{g}_{w}\hat{\vec{v}}^{\,p}_{IMU}$$

Recall that the C^{g}_{w} DCM is the transformation from the IMU computer frame to the IMU local geographic frame: it is used as an approximate transformation from the IMU platform frame to the GPS local geographic frame. To develop the measurement model, it was expanded to

$$\begin{aligned}
\vec{z}^{\,g}_{v_{GPS}} &= \hat{\vec{v}}^{\,g}_{GPS} - C^{g(GPS)}_{w(GPS)}C^{w(GPS)}_{c}C^{c}_{p}\hat{\vec{v}}^{\,p}_{IMU} \\
&= \hat{\vec{v}}^{\,g}_{GPS} - C^{g(GPS)}_{w(GPS)}C^{t}_{c}C^{c}_{p}\hat{\vec{v}}^{\,p}_{IMU} \\
&= \hat{\vec{v}}^{\,g}_{GPS} - C^{g(GPS)}_{w(GPS)}\left(I + \delta\vec{\theta}\times\right)\left(I + \vec{\psi}\times\right)\hat{\vec{v}}^{\,p}_{IMU}
\end{aligned}$$

Here, an alternative formulation will be presented.

Instead of transforming the computer frame velocity to the GPS local geographic frame in this way (as a function of IMU position error), an exact transformation can be used. The new measurement is

$$\vec{z}^{\,g}_{v_{GPS}} = \hat{\vec{v}}^{\,g}_{GPS} - C^{g(GPS)}_{E}C^{E}_{c}\hat{\vec{v}}^{\,p}_{IMU}$$

where the earth-centred, earth-fixed coordinate frame is used as an intermediary. The first new DCM, C^{E}_{c} , is an exact function of IMU-computed latitude, longitude and height; the second, $C^{g(GPS)}_{E}$, is an exact function of GPS-computed latitude, longitude and height. All IMU position error dependence has been removed – at the expense of the extra computations required to form the new DCMs.

The H-matrix for the new model is derived from

$$\begin{aligned}
\vec{z}^{\,g}_{v_{GPS}} &= \hat{\vec{v}}^{\,g}_{GPS} - C^{g(GPS)}_{E}C^{E}_{c}C^{c}_{p}\hat{\vec{v}}^{\,p}_{IMU} \\
&= \hat{\vec{v}}^{\,g}_{GPS} - C^{g(GPS)}_{E}C^{E}_{c}\left(I + \vec{\psi}\times\right)\hat{\vec{v}}^{\,p}_{IMU} \\
&= \hat{\vec{v}}^{\,g}_{GPS} - C^{g(GPS)}_{E}C^{E}_{c}\hat{\vec{v}}^{\,p}_{IMU} - C^{g(GPS)}_{E}C^{E}_{c}\left(\vec{\psi}\times\hat{\vec{v}}^{\,p}_{IMU}\right)
\end{aligned}$$

In terms of -angles, neglecting second order terms,

$$\begin{aligned}
-C^{g(GPS)}_{E}C^{E}_{c}\left(\vec{\psi}\times\hat{\vec{v}}^{\,p}_{IMU}\right) &= C^{g(GPS)}_{E}C^{E}_{c}\left(\hat{\vec{v}}^{\,p}_{IMU}\times\vec{\psi}\right) \\
&= \left(C^{g(GPS)}_{E}C^{E}_{c}\hat{\vec{v}}^{\,p}_{IMU}\right)\times\vec{\psi} \\
&\cong \hat{\vec{v}}^{\,g}_{IMU}\times\vec{\psi}
\end{aligned}$$

Without expanding the terms, the new H-matrix is

$$H_{GPS_V} = \frac{\partial \bar{z}^{g}_{v_{GPS}}}{\partial(\vec{x})} = \begin{bmatrix} 0 & C_E^{g(GPS)} C_c^E & \dfrac{\partial\left(C_E^{g(GPS)} C_c^E \left(\hat{\vec{v}}^{\,p}_{IMU} \times \vec{\psi}\right)\right)}{\partial \vec{\psi}} & \ldots & 0 \end{bmatrix}$$

Using the above approximation,

$$H_{GPS_V} = \frac{\partial \bar{z}^{g}_{v_{GPS}}}{\partial(\vec{x})} = \begin{bmatrix} 0 & C_E^{g(GPS)} C_c^E & \begin{bmatrix} 0 & -v_z^g & v_y^g \\ v_z^g & 0 & -v_x^g \\ -v_y^g & v_x^g & 0 \end{bmatrix} & \ldots & 0 \end{bmatrix}$$

All non-zero terms are in the IMU velocity and attitude error state columns.

DOCUMENT CONTROL DATA

(Security classification of title, body of abstract and indexing annotation must be entered when the overall document is classified)

1. ORIGINATOR (the name and address of the organization preparing the document. Organizations for whom the document was prepared, e.g. Establishment sponsoring a contractor's report, or tasking agency, are entered in section 8.)

   Dale Arden Consulting
   R.R.. #3
   Almonte, Ontario

2. SECURITY CLASSIFICATION (overall security classification of the document, including special warning terms if applicable)

   UNCLASSIFIED

3. TITLE (the complete document title as indicated on the title page. Its classification should be indicated by the appropriate abbreviation (S,C or U) in parentheses after the title.)

   MEMS/GPS KALMAN FILTER (U)

4. AUTHORS (Last name, first name, middle initial)

   Arden, Dale A.G.

5. DATE OF PUBLICATION (month and year of publication of document)

   May 2007

6a. NO. OF PAGES (total containing information. Include Annexes, Appendices, etc.)

   49

6b. NO. OF REFS (total cited in document)

   5

7. DESCRIPTIVE NOTES (the category of the document, e.g. technical report, technical note or memorandum. If appropriate, enter the type of report, e.g. interim, progress, summary, annual or final. Give the inclusive dates when a specific reporting period is covered.)

   Contract Report

8. SPONSORING ACTIVITY (the name of the department project office or laboratory sponsoring the research and development. Include the address.)

   Defence R&D Canada – Ottawa
   3701 Carling Avenue
   Ottawa, Ontario, K1A 0Z4

9a. PROJECT OR GRANT NO. (if appropriate, the applicable research and development project or grant number under which the document was written. Please specify whether project or grant)

9b. CONTRACT NO. (if appropriate, the applicable number under which the document was written)

   W7714-010525/001/SV

10a. ORIGINATOR'S DOCUMENT NUMBER (the official document number by which the document is identified by the originating activity. This number must be unique to this document.)

   DAC-0402

10b. OTHER DOCUMENT NOS. (Any other numbers which may be assigned this document either by the originator or by the sponsor)

   DRDC Ottawa CR 2007-093

11. DOCUMENT AVAILABILITY (any limitations on further dissemination of the document, other than those imposed by security classification)

   (X) Unlimited distribution
   (   ) Distribution limited to defence departments and defence contractors; further distribution only as approved
   (   ) Distribution limited to defence departments and Canadian defence contractors; further distribution only as approved
   (   ) Distribution limited to government departments and agencies; further distribution only as approved
   (   ) Distribution limited to defence departments; further distribution only as approved
   (   ) Other (please specify):

12. DOCUMENT ANNOUNCEMENT (any limitation to the bibliographic announcement of this document. This will normally correspond to the Document Availability (11). However, where further distribution (beyond the audience specified in 11) is possible, a wider announcement audience may be selected.)

13. ABSTRACT (a brief and factual summary of the document. It may also appear elsewhere in the body of the document itself. It is highly desirable that the abstract of classified documents be unclassified. Each paragraph of the abstract shall begin with an indication of the security classification of the information in the paragraph (unless the document itself is unclassified) represented as (S), (C), or (U). It is not necessary to include here abstracts in both official languages unless the text is bilingual).

This report describes important parts of the Kalman filter that is used in MEMS/GPS to optimally combine data from different navigation sensors in a way that will provide the best navigation solution in all situations. The implementation of this filter is based on earlier work: the Dual Inertial Integrated Navigation System (DIINS). Unless otherwise stated, the theory, algorithms, techniques and models applied to the MEMS/GPS Kalman filter are the same as those described in DIINS documentation. This report was written as an addendum to the DIINS documents. That is not to say that the differences are insignificant: the MEMS/GPS filter has completely different target applications; filter timing has changed from clock timed to data driven; there are new sensors, and new non-sensor measurements; there is a complementary IMU strapdown navigator running with the filter in a closed-loop fashion; and new IMU error states were added to try to better model the larger MEMS IMU errors. The main focus of the sensor integration has also been modified: in DIINS, the emphasis was on failure detection, isolation and reconfiguration; in the MEMS/GPS system, the emphasis is on maintaining navigation accuracy during periods of GPS loss or degradation.

14. KEYWORDS, DESCRIPTORS or IDENTIFIERS (technically meaningful terms or short phrases that characterize a document and could be helpful in cataloguing the document. They should be selected so that no security classification is required. Identifiers such as equipment model designation, trade name, military project code name, geographic location may also be included. If possible keywords should be selected from a published thesaurus. e.g. Thesaurus of Engineering and Scientific Terms (TEST) and that thesaurus-identified. If it is not possible to select indexing terms which are Unclassified, the classification of each should be indicated as with the title.)

Integrated navigation; Inertial Measurement Units (IMUs); Micro-Electro-Mechanical Systems (MEMS); Accelerometer; Gyroscope; GPS, Kalman filter

Defence R&D Canada

Canada's leader in Defence
and National Security
Science and Technology

R & D pour la défense Canada

Chef de file au Canada en matière
de science et de technologie pour
la défense et la sécurité nationale

DEFENCE R&D DÉFENSE

www.drdc-rddc.gc.ca