DINCON 2017
CONFERÊNCIA BRASILEIRA DE DINÂMICA, CONTROLE E APLICAÇÕES
30 de outubro a 01 de novembro de 2017 – São José do Rio Preto/SP

Revisited Position-Velocity Integration Formulas for In-Flight Self-Alignment of GPS-Aided Inertial Navigation Systems

Rian Koja[1]
Instituto Nacional de Pesquisas Espaciais
Visiona Tecnologia Espacial
Waldemar de Castro Leite Filho[2]
Instituto Nacional de Pesquisas Espaciais
CLC - Castro Leite Consulting

Abstract. This paper reviews and adapts the framework for attitude initialization of an Inertial Navigation System with Position-Velocity Integration formulas, which are based on measurements from GPS receiver and inertial sensors. It is shown that some shortcomings of such methods are more critical for a simplified comparison algorithm, which in turn helps create a derived method, based on some logical conditions checks that allow on-line declaration of convergence for the computed attitude.

Keywords. Inertial Navigation Systems, In-Flight-Alignment, Attitude Estimation

1 Introduction

The alignment of an Inertial Navigation System (INS) consists in determining it's initial attitude from its sensors and possibly with external aid. A significant of research effort is dedicated to the development of alignment techniques, due both to the possible divergence when out of the linear region [4] and the improvement in convergence time that a good alignment provides. Self-alignment is the denomination given to a process that does not rely on external knowledge of the attitude.

A special case of self-alignment occurs for In-Flight Alignment, when the INS can neither be requested to remain stationary nor controlled to be leveled for the benefit of alignment. This can be useful when aiding systems had been unavailable for a period after which the propagated attitude is no longer reliable. Several works describe Kalman-filter like techniques for In-flight alignment, such as [8] or [1].

An interesting method is presented in [12] and related works that is not based on Kalman-Filtering techniques but rather on attitude optimization methods. This method

[1]rian.koja@gmail.com
[2]waldemar@castroleite.com

is briefly explained in Section 2, along with some of the related background. It is demonstrated that such method is not guaranteed to converge in Section 4. Also, the proposed optimization is shown to possibly provide little or no gain over a deterministic attitude determination method, which in turn makes the method's flaws more evident. Some preliminary research to overcome such shortcomings and developing on-line criteria to assert convergence is presented. This is achieved with analytical deductions and analyzing simulated data presented on Section 3.

2 Theoretical Background

Classical initialization schemes rely on solving an attitude estimation problem from the measurements available to the INS. The TRIAD algorithm, first described on [2] is a simple solution to the attitude (deterministic) determination problem, it is used both on the current work and for simple stationary alignment techniques. Given two (unit) vectors u^A and v^A expressed in A frame, and their counterparts u^B and v^B expressed in B frame, the rotation matrix between such frames C_A^B can be computed.

Stationary alignment can be proceeded by assuming the inertial system has no kinematic acceleration during initialization period, such that the measured specific force in body frame matches the local gravity $f^B = g^B$, then also the measured angular rate of the platform is the Navigation frame rate with respect to the inertial frame, which for a stationary object matches the Earth's rotation: $\omega_{BI}^B = \omega_{NI}^B = \omega_{EI}^B$.

Optimal attitude estimation traces back to the Wahba's problem and is a least square solution to the problem of determining attitude when several measured vectors u_k are available and the following cost functional J is to be minimized:

$$J = \frac{1}{2} \sum_k a_k \left| u_k^B - C_A^B u_k^A \right|^2 \tag{1}$$

Where a_k is a positive weighting coefficient. Several algorithms to find the solution of this problem exist, one of the most popular being the QUEST algorithm [9]. The Singular Value Decomposition (SVD) method will be used in this work for simplicity and robustness of implementation [6]. It consists of:

$$K = \sum_k a_k u_k^B (u_k^A) = U \Sigma V^T = U diag[\Sigma_{11}\, \Sigma_{22}\, \Sigma_{33}] V^T \tag{2a}$$

$$C_A^B = U diag[1\, 1\, (detU)(detV)] V^T \tag{2b}$$

Where U and V are orthogonal and $\Sigma_{11} \geq \Sigma_{22} \geq \Sigma_{33}$. Having established such methods, they can be applied to INS alignment if a set of vectors is known in both Body frame (B) and Navigation Frame (N, or NED for North-East-Down). This allows computing the matrix C_B^N. Frames with unspecified time are to be understood as with respect to current time (i.e. $N = N(t)$), while 0 stands for the instant the alignment process is to start. While most filtering techniques try to improve the estimation of C_B^N as t progresses, it is noted that:

$$C_B^N = C_{N(0)}^{N(t)} C_{B(0)}^{N(0)} C_{B(t)}^{B(0)} = C_{E(t)}^{N(t)} C_{E(0)}^{E(t)} \left(C_{E(0)}^{N(0)}\right)^T C_{B(0)}^{N(0)} C_{B(t)}^{B(0)} \tag{3}$$

Where E stands for the Earth-Fixed Earth-Centered (ECEF) frame. $C_{E(t)}^{N(t)}$ can be computed directly from GPS position measurements, and its initial value $C_{E(0)}^{N(0)}$ can be stored. A simplified Earth rotation model is used to compute $C_{E(0)}^{E(t)}$. The right-hand side of the equation is used below for further developments. Meanwhile $C_{B(t)}^{B(0)}$ can be initialized with identity, and updated at high rate with measurements from the Inertial Measurement Unit (IMU), by using quaternions:

$$\hat{u} = \omega_{bi}^b / \left\|\omega_{bi}^b\right\| \tag{4a}$$

$$\Delta q = \left[\cos(|\omega_{bi}^b| dt/2), \sin(|\omega_{bi}^b| dt/2)\hat{u}^T\right]^T \tag{4b}$$

$$q_{B(0)}^{B(t)} = \Delta q \otimes q_{B(0)}^{B(t-dt)} \tag{4c}$$

Where $\otimes$ stands for quaternion multiplication. All frame transformations are performed with quaternions for the simulations described in Section 3, even the matrices computed with TRIAD and SVD methods are converted to quaternions with code adapted from [3], normalization and sign checks are also performed. The analytical developments are done with matrices for the sake of conciseness.

The technique proposed in [12] starts from the well known kinematic equation [5]:

$$\dot{v}^N = C_B^N f^B - (2\omega_{IE}^N + \omega_{EN}^N) \times v^N + g^N \tag{5}$$

Notably, assuming availability of GPS position and velocity measurements, the rotation rates and the local gravity acceleration can be computed with on-board models. Applying the expansion from equation 3:

$$\dot{v}^N = C_{N(0)}^{N(t)} C_{B(0)}^{N(0)} C_{B(t)}^{B(0)} f^B - (2\omega_{IE}^N + \omega_{EN}^N) \times v^N + g^N \tag{6}$$

Collecting alike terms:

$$C_{N(t)}^{N(0)} \left[\dot{v}^N - (2\omega_{IE}^N + \omega_{EN}^N) \times v^N + g^N\right] = C_{B(0)}^{N(0)} C_{B(t)}^{B(0)} f^B \tag{7}$$

The initial attitude matrix $C_{B(0)}^{N(0)}$ is the unknown to be estimated. The other right-hand side terms are computed from the IMU measurements. The left-hand side terms can be computed directly from the GPS measurements, except for $\dot{v}^n$, which can be eliminated by proceeding a time-integration:

$$\int_0^t C_{N(0)}^{N(t)} \dot{v}^N = C_{N(0)}^{N(t)} v^N - v(0) - \int_0^t C_{N(0)}^{N(t)} \left(\omega_{IN}^N\right)^\times v^N dt \tag{8}$$

Where $(.)^{\times}$ denotes the cross-product matrix. By integrating both sides of equation (7), one obtains, as in [12]:

$$\alpha_v = \int_0^t C_{B(t)}^{B(0)} f^B dt \tag{9a}$$

$$\beta_v = C_{N(0)}^{N(t)} v^N - v^N(0) + \int_0^t C_{N(0)}^{N(t)} \left[\left(\omega_{IE}^N\right)^{\times} v^N - g^N \right] dt \tag{9b}$$

$$\beta_v = C_{B(0)}^{N(0)} \alpha_v \tag{9c}$$

Hence, α_v and β_v can be used as entries for an optimal estimation method, with the K matrix from equation (2a) being updated at each time step t_k. This is referred to as the "Velocity Integration Formula" (VIF) in [12]. By integrating over time again, new vectors α_p and β_p are obtained:

$$\alpha_p = \int_0^t \alpha_v dt = \int_0^t \int_0^\tau C_{B(\tau)}^{B(0)} f^B d\tau dt \tag{10}$$

$$\beta_p = \int_0^t \beta_v dt = \int_0^t C_{N(0)}^{N(t)} v^N dt - t v^N(0) + \int_0^t \int_0^\tau C_{N(0)}^{N(t)} \left[\left(\omega_{IE}^N\right)^{\times} v^N - g^N \right] d\tau dt \tag{11}$$

Accumulating such vectors in the K matrix and estimating $C_{B(0)}^{N(0)}$ is what [12] refers to as "Position Integration Formula" (PIF).

Given that such quantities are distorted by noise, they should always span the whole Euclidean space, thus avoiding singularities on the SVD method, yet not granting accuracy of the solution.

If all four vectors α_v, β_v, α_p and β_p are known simultaneously, they can be used by TRIAD method to determine a one-shot attitude estimate, which is prone to divergence under certain circumstances, yet has shown better or slightly worse accuracy for several instances during simulations.

If only a specific subset of vectors is accumulated on the matrix K, satisfying properties that would avoid divergence of the TRIAD method, a reliable result may be achieved.

3 Simulations

Simulated data for a sounding rocket flight was used. The idea was to simulate the algorithm starting either on propelled stage or close to zero gravity. Hence for the result charts, the simulation may begin at different instants, before which no data is shown. The simulated flight is the same shown below, spanning 100s.

The data has noise compatible with a GPS receiver and inertial sensors described on Table 1, all at 1σ confidence level. It is noted that for both GPS velocity and position, the noise is modeled as being zero-mean gaussian with standard deviance given on the table.

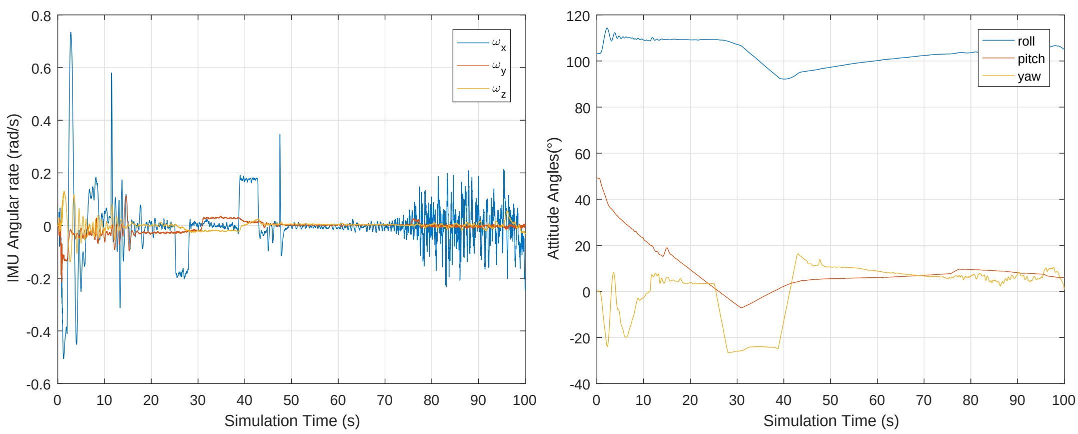
Notably there is a mismatch in the frequency of sensors and GPS. This is handled by running the simulated algorithm at the high rate sensors' frequency, but updating the vectors based on GPS measurements only at cycles when a new measurement arrives, causing a small delay due to the mismatch, whose effect is small compared to noise figures.

Table 1: Simulated Data Specifications

Specification	Value	Specification	Value
IMU Frequency	76Hz	Accelerometer Bias Repeatability	$< 550\mu G$
Angular Rate Noise	$1,2'/\sqrt{h}$	GPS Frequency	1Hz
Angular Rate Bias Repeatability	$2°/h$	GPS Position Accuracy	5m
Angular Rate Bias Instability	$1°/\sqrt{h}$	GPS Velocity Accuracy	$0,1m/s$
Accelerometer Noise	$7\mu G$		

The mission profile for the experiment consisted of a propelled flight, then a close to zero gravity parabolic movement and a lift-sustained glide.

The relevant characteristics of the simulated profile are shown in the following figures.



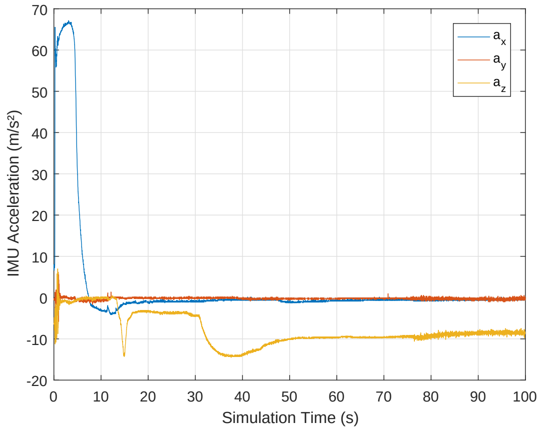
(a) Measured Angular Rate in Body Frame.

(b) Attitude Angles.

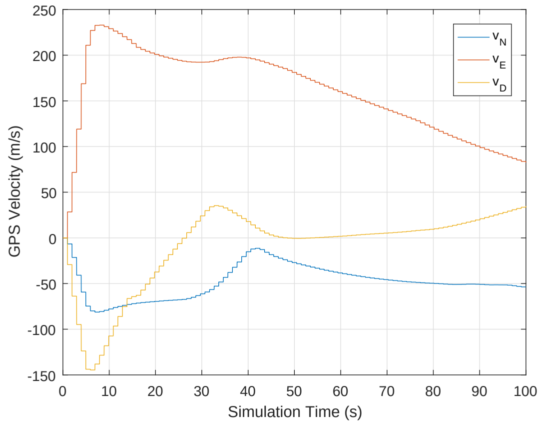
Figure 1: Attitude and angular rate.

4 Remarks on Methods

This section is devoted to point some conditions under which the proposed schemes may fail to converge or deliver poor performance, which also helps identifying on-line if the output of this method can be declared as converged. This is relevant since most Kalman filter navigation systems are expected to be initialized with a valid initial state. Some differences between the current implementation and that presented in [12] are to be noted, mostly because the aforementioned reference assumes a 50Hz GPS and 100Hz IMU, while this work assumes the specifications provided on Table 1.



(a) Measured Linear Acceleration in Body Frame.



(b) GPS Measured Velocity in NED Frame.

Figure 2: Acceleration and Velocity

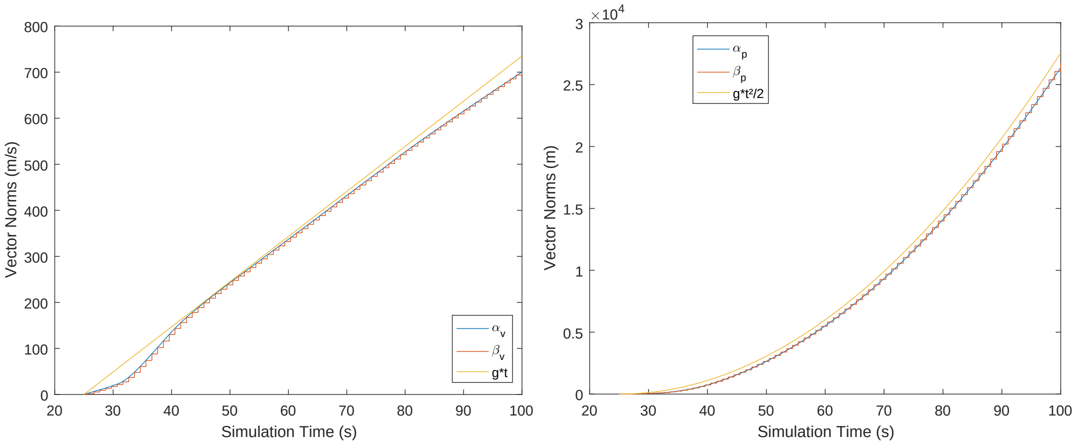
4.1 Norm threshold

A first shortcoming of such method occurs when the INS is free-falling, since under this condition, the measured specific force should be zero $f^b = 0$, though it will actually accumulate noise and sensor errors (e.g. bias). This causes α_v and consequently α_p to be invalid. Additionally, small magnitude for those vectors which are produced from noisy measurements, this implies smaller signal-to-noise ratio. It is then suggested to verify that the used vectors have their norm above a certain threshold in order to assume they're usable.

Despite the previous claim, most mission envelopes involve small accelerations compared with the gravity magnitude of $9,8\text{m/s}^2$. Hence the norm of these vectors tends to increase at a rate around $g_0\,\text{m/s}$ for α_v depending as well on attitude changes. Likewise a norm around $tg_0\,/2\,\text{m}$ is expected for α_p. This also causes the last computed sets of vectors to be more relevant than the first ones, since the lack of normalization on said vectors causes the norm to take the role of the weights a_k in equation (1). These trends, along with the computed norms for each vector in the simulated data are shown in Figure 3(a) and Figure 3(b).

4.2 Angle Between Vectors

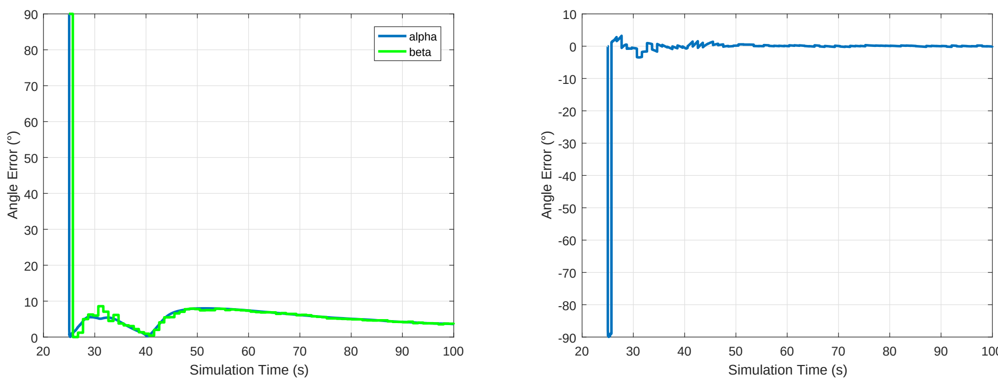
Since the same transformation represented by $C_{B(0)}^{N(0)}$, transforms α_v and α_p into β_v and β_p respectively, the angle between these pairs of vectors should be unchanged. However, due to noise and integration errors, these angles will have a difference that may be large, specially on the beginning of the simulation when the norms of such values tend to be small. When using the PIF or VIF methods separately, this property cannot be checked, one can only hope that the accumulated set of vectors spans the whole three-dimensional space, which is mostly granted to happen due to noise. Furthermore, the angle between such vectors may be zero, a situation that causes the TRIAD method to diverge. Even at



(a) Comparison of norms for v vectors (b) Comparison of norms for p vectors

Figure 3: Comparison of norms

small angles, TRIAD method's accuracy significantly decreases. Such zero angle occurs around 40s as seen on Figure 4(a). Also the difference between angles, shown in Figure 4(b) can be monitored, and a set of vectors can be declared invalid if the error (in absolute value) is above a certain threshold. Unless the attitude matrix to be estimated is the identity, it is also necessary that such angles varies over time during the simulation, to provide enough information for the computation.



(a) Computed Angles. (b) Angle Discrepancy.

Figure 4: Angles between Errors.

4.3 Norm Discrepancy

As far as online estimations are concerned, it is not possible to verify that the computed norms are accurate. However, it is possible to compute the norms for the estimates and know that their discrepancy comes from errors in their calculations and noise from inputs.

This error however, requires normalization, which again requires a reference, but for this purpose the average of the norms suffices. In equation form, irrespective of the subscripts v and p:

$$\Delta = 2\frac{\|\alpha\| - \|\beta\|}{\|\alpha\| + \|\beta\|} \tag{12}$$

Simulated results show that indeed a relatively large error appears between the vectors, shown in Figure 5. If the error was related only to the norm itself (and not to the direction), the TRIAD method (that normalizes vectors) would output the same result while the SVD method (that weights vectors by the norm when they're not unitary, but assumes norms to be the same) would be slightly affected. Norm discrepancy however is an indication that errors are still relevant within the computed vectors.

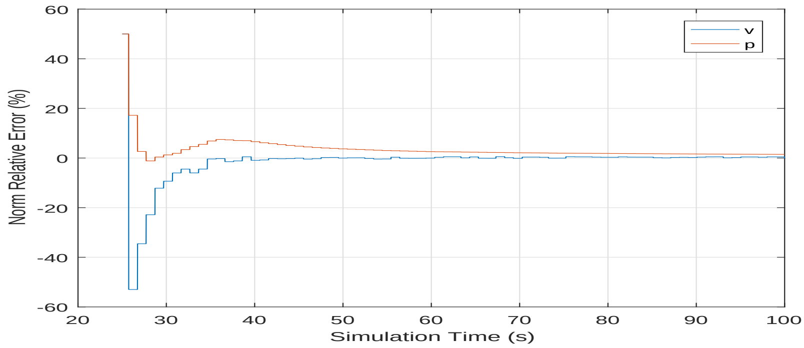


Figure 5: Discrepancy Between Norms.

4.4 Selectivity

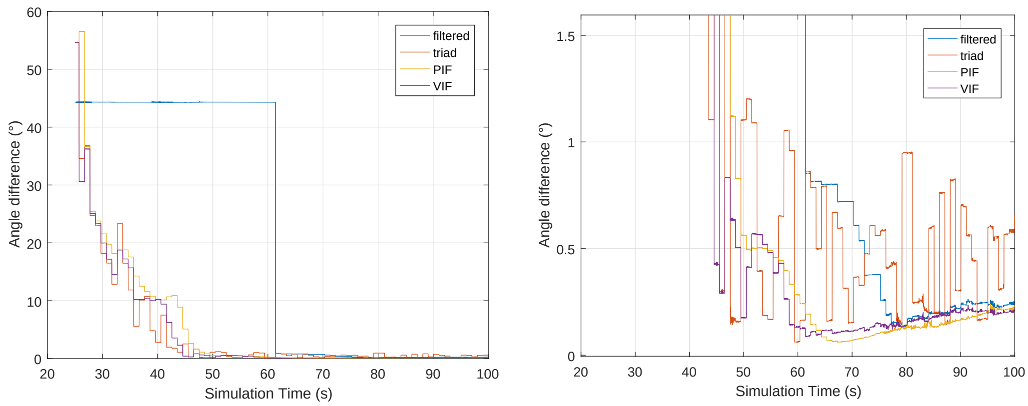
While the TRIAD based method is mentioned as a base check, it has less memory than other computed vectors and is more susceptible to the aforementioned flaws. This is solved by other methods by the accumulation of vectors (either only v or p ones) on the K matrix. By performing the aforementioned checks, the sets of vectors used can be selected, or filtered, before being added to the matrix K, this allows creating a modified method that takes long before providing any output, which in turn can be considered close to convergence, unlike the other methods for which no simple indication (other than estimated time span) to assert online if convergence has been achieved or not. This is shown on Section 5.

5 Performance Results

For each estimated attitude (or quaternion) $\hat{C}_{B(t)}^{N(t)}$, the discrepancy between the real matrix $C_{B(t)}^{N(t)}$ from the simulated data and the estimation can be represented by the rotation angle between them. In a simplified manner:

$$\theta_{err} = \arccos\left[0.5\left(tr\left(C_{B(t)}^{N(t)}\left(\hat{C}_{B(t)}^{N(t)}\right)^{T}\right) - 1\right)\right] \quad (13)$$

Ideally, θ_{err} should be zero if a perfect estimation was possible. Since inertial systems require external aiding in order not to diverge, due to the calculation of $C_{B(t)}^{B(0)}$, there is bound to be some discrepancy. In particular, biases of the angular rate sensor (and also for the accelerometer) are not considered nor computed on this method, tough present on simulation. This metric allows comparing results to see which method gave smaller errors and after how much time. This summary is shown on Figure 6(a) and Figure 6(b).



(a) Performance Error. (b) Performance Error (zoomed).

Figure 6: Angles between Errors.

While all methods provide performance better than 1° within 25 seconds, the instability of the TRIAD method can be noted observing that contrary to other methods it still oscillates more than half degree after 50 seconds. Nevertheless, it often provides better or as good estimates at the beginning of the simulation.

The filtered method only decreases in error at around 61 seconds because this is the first instance when all aforementioned criteria are met. At this instant, it receives the same input of that is given to TRIAD method, hence it outputs the same result until another valid set of vectors is available. Its result improves in small steps until it starts decreasing in quality due to the natural accumulation of errors.

Additionally, if the GPS noise is removed, some interesting effect can be seen on Figure 7, where around 40s the error for TRIAD method while the VIF and PIF methods stop improving. This is caused by the angles between vectors, shown in Figure 4(b), approaching zero, causing the TRIAD method to approach a mathematical exception and making the new entries add no new information for the SVD method. The same occurs with the presence of noise but is more visible when noise is removed. Even in this case, the filtered method takes longer to converge, but provides reliable results when does so.

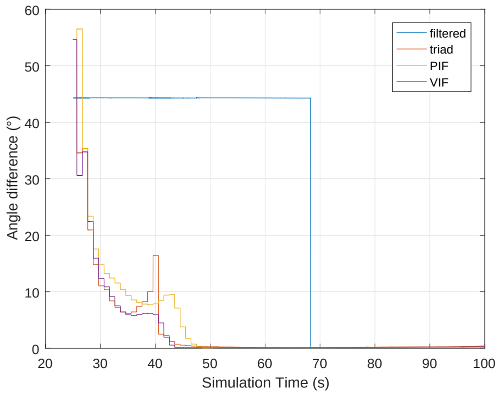


Figure 7: Performance Error (GPS noise removed).

6 Conclusion and Further Work

An analysis of the methods proposed on [12] was carried out, and very good results were obtained for the current simulated experiments. The proposed selectivity scheme provided indeed an added value in terms of reliability, but for the current work it depends on designer chosen threshold values, while in theory such tests should be made based on error models. Methods to improve overall convergence time, specially for the filtered model are to be investigated along with means to estimate inertial sensor biases.

Acknowledgments

The authors express their acknowledgment for the support received from the National Institute for Space Research - INPE, at São José dos Campos, Brazil.

References

[1] J. Ali and M. Ushaq, A consistent and robust Kalman filter design for in-motion alignment of inertial navigation system, Measurement, 42(4), 577-582 (2009)

[2] H. D. Black, A passive system for determining the attitude of a satellite, AIAA Journal, Vol. 2, No. 7 (1964)

[3] V. Carrara, An open source satellite attitude and orbit simulator toolbox for Matlab. In Proceedings of the 17th International Symposium on Dynamic Problems of Mechanics (2015)

[4] J. L. Crassidis, CRASSIDIS, John L. Sigma-point Kalman filtering for integrated GPS and inertial navigation. IEEE Transactions on Aerospace and Electronic Systems, v. 42, n. 2, p. 750-756, (2006)

[5] J. Farrell, Aided navigation: GPS with high rate sensors,McGraw-Hill, Inc. (2008)

[6] F. L. Markley, and D. Mortari, How to estimate attitude from vector observations, Astrodynamics Specialist; 16-19 Aug. 1999; Girdwood, AK; United States (1999)

[7] F. L. Markley and J.L. Crassidis. Fundamentals of spacecraft attitude determination and control. Vol. 33. New York: Springer, (2014)

[8] P. J. Pei, X. Liu and L. Zhu, In-flight alignment using filter for strapdown INS on aircraft. The Scientific World Journal (2014)

[9] M.D. Shuster and S.D.Oh, Three-axis attitude determination from vector observations. Journal of Guidance, Control, and Dynamics.(2012)

[11] D. Titterton and J. Weston, Strapdown inertial navigation technology, IET, (2004)

[12] Y. Wu and X. Pan, Velocity/position integration formula part I: Application to in-flight coarse alignment. IEEE Transactions on Aerospace and Electronic Systems, 49(2), 1006-1023 (2013)