Performance analysis of MEMS based Pedestrian Navigation Systems

Damien Kubrak, Christophe Macabiau, Michel Monnerat

Performance Analysis of MEMS based Pedestrian Navigation Systems

Damien Kubrak, *Ecole Nationale de l'Aviation Civile / TéSA*, France
Christophe Macabiau, *Ecole Nationale de l'Aviation Civile*, France
Michel Monnerat, *Alcatel Space*, France

BIOGRAPHY

Damien Kubrak is a Ph.D. student in the field of personal positioning in the signal processing lab of the Ecole Nationale de l'Aviation Civile (ENAC) in Toulouse, France. He graduated in 2002 as an electronics engineer from the ENAC, and received the same year his Master degree in signal processing.

Christophe Macabiau graduated as an electronics engineer in 1992 from the Ecole Nationale de l'Aviation Civile (ENAC) in Toulouse, France. Since 1994, he has been working on the application of satellite navigation techniques to civil aviation. He received his Ph.D. in 1997 and has been in charge of the signal processing lab of the ENAC since 2000.

Michel Monnerat graduated from the ENSICA (Ecole Nationale Supérieure d'Ingénieur de Constructions Aéronautiques) engineering school. After being involved within Alcatel Space in many radar programs, in charge of the onboard processing of the ARGOS / SARSAT payload, he has been involved in the Galileo program since 1998, for the signal design and performance aspects. He is currently involved in Alcatel Space LBS programs.

ABSTRACT

First driven by the regulation on emergency call, several methods are studied to enable the location of one user whether he is outdoors or indoors. Moreover, location based services take more and more importance, and as a consequence, the ability of providing such a location becomes a great challenge. Assisted satellite navigation solutions are investigated, but they encounter big issues in indoor environment. The main reason is the weak power of the signals to acquire and process. Therefore, complementary techniques shall be developed to supply or enhance GPS based positioning solutions when GPS reception is disrupted. MEMS based techniques are promising ones.

In this paper, the performance of a MEMS based Pedestrian Navigation System (PNS) is studied. As the PNS is assumed to be used during GPS outages, only the navigation using exclusively MEMS sensors is addressed. The Pedestrian Navigation Module (PNM) relies on four components: a three axes accelerometer, a three axes gyro and a three axes magnetometer, as well as a pressure sensor. The four sensors are low-cost ones.

Two different MEMS based positioning methods are investigated, as well as two cases of use. In a first time, the traditional inertial navigation is discussed and its performance analysed. The distinction is made between the constrained navigation, meaning that the sensors are rigidly attached to the user body, and the non-constrained one, where the sensors are free of movement with respect to the user.

In a second time, the navigation based on sensor output signature is discussed. In this second positioning technique, the travelled distance and the heading estimation are done separately. The travelled distance is computed based on step length or velocity estimation, whereas the heading computation is based on gyro and magnetometer measurements. As for the inertial navigation, two cases of use are considered, whether the sensors are rigidly attached to the user body or not.

Results point out the fact that the unconstrained navigation leads to worse results than the constrained one, whatever the navigation method. The second method based on signal signature is the best one, and is well adapted for constrained navigation. It provides a good estimation of the travelled distance and also a reliable heading, even if low cost sensors are used. Moreover, the second method is more robust on sensor motion with respect to the user body. In the unconstrained navigation mode, the travelled distance as well as the heading can be estimated with accuracy depending on the type and number of PNM movements occurring during the walk. The availability is then all the more reduced as the number of PNM movements increase.

INTRODUCTION

The expanding demand of Location-Based Services

(LBS) is currently boosting personal positioning technologies. Among them, Assisted GPS (AGPS) and High Sensitive GPS (HSGPS) are very adapted to provide the location of the user in many use cases. However, even if the aforementioned techniques are very efficient, they do not cover all areas the LBS are likely to be provided. Especially in deep indoor environments, they suffer from a lack of availability.

To provide a continuous location solution, such GPS-based systems have to be augmented. In this paper, the augmentation using low-cost MEMS sensors for personal positioning is addressed. The focus is put on the performance of the sensor assembly as an alternative positioning system during GPS signals outages, in terms of accuracy and availability.

Two navigation methods are investigated, one based on traditional strap-down Inertial Navigation System (INS) algorithm, the other based on accelerometer signal signature algorithm. For both methods, the Pedestrian Navigation Module (PNM) used in the study is supposed to have its own dynamic. This dynamic would be those of the user or not. The two use cases are investigated, so that the MEMS-based PNM is tested as a typical handled positioning system.

The paper is organized as follows. In the first section, the sensor assembly is described, and accelerometers as well as gyrometers are roughly characterized. In the next two sections, navigation methods dedicated to the positioning of pedestrians are presented, as well as their specific algorithm. Section IV deals with typical issues encountered in pedestrian navigation that are taken into account in the aforementioned algorithms. Finally, section V presents the results of actual tests conducted with the PNM.

I – MEASUREMENT UNIT

The PNM is composed of four different types of sensors. They are accelerometers, gyrometers, magnetometers and a pressure sensor.

The pressure sensor is an Intersema MS5534 [1]. It is a stable and temperature compensated sensor capable of 0,1 mbar resolution (approximately 1 meter) with an accuracy within +/-0,5 mbar once calibrated. The sensor is used in a relative mode. Information about the height and vertical speed of the pedestrian is obtained using the following pressure-to-altitude relation :

$$z(P) = \frac{T_0}{\gamma}\left[\left(\frac{P}{P_0}\right)^{\frac{R_a \cdot \gamma}{g_0}} - 1\right]$$

where:

- z is the altitude of the sensor.
- P is the pressure at altitude z.
- P_0 is equal to 1013,25 hPa.
- T_0 is equal to 288,15 K.
- γ is equal to -6,5°C/km for z < 11km.
- R_a is equal to 287,1 J/kg/K for dry air.
- g_0 is equal to 9,80665 m/s².

The accelerometer triad is composed of Analog Device ADXL202E accelerometers [2]. They are capable of 0,01 m/s² RMS noise for a bandwidth of 30 Hz. To characterize the turn-on bias (or static bias), 10 data sets were collected while one idling accelerometer was sensing the gravity vector upwards and downwards. Then, the mean value of the measurements was compared to zero to find the turn-on bias. The process was repeated for each of the three axes. Results are given in Figure 1 and summed up in Table 1.

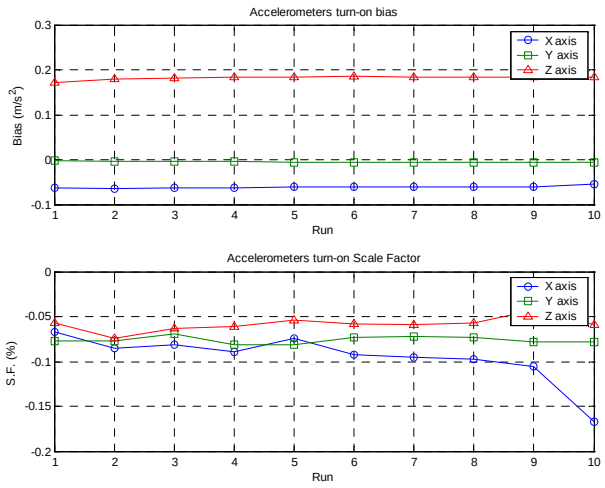

Figure 1 – Accelerometer turn-on bias and S.F.

	X axis		Y axis		Z axis	
	Bias (m/s²)	S.F. (%)	Bias (m/s²)	S.F. (%)	Bias (m/s²)	S.F. (%)
Mean	-0,061	-0,096	-0,005	-0,076	0,182	-0,058
Std	0,003	0,028	0,001	0,004	0,004	0,008

Table 1 – Accelerometers turn-on bias.

The gyros of the PNM are Analog Device ADXRS300 [3]. They are affected by a 0,7 deg/s RMS noise for a bandwidth of 50 Hz. The turn-on bias of the three sensors is characterized using the above 30 measurements. The turn on-bias is considered equal to the mean of gyros output while the PNM is idling. Results are plotted in Figure 2 and summed up in Table 2.

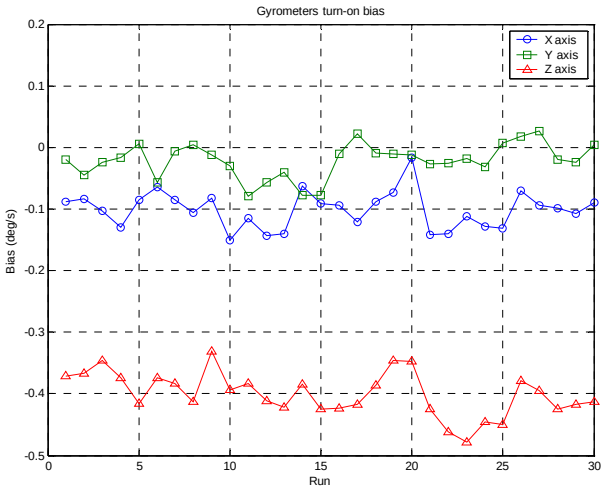

Figure 2 – Gyros turn-on bias.

	X axis (deg/s)	Y axis (deg/s)	Z axis (deg/s)
Mean	-0,1014	-0,0215	-0,4001
Std	0,0298	0,0280	0,0357

Table 2 – Gyros turn-on bias.

The magnetic sensors are Philips KMZ52 [3], [4]. They are very sensitive (16 mV/V / kA/m) but consequently measurements are easily affected by hard and soft magnetic perturbations.

This low-cost sensor assembly (about $150) is then used in the following for personal navigation. Two common navigation methods are implemented. They are discussed in the next two sections.

II – INERTIAL NAVIGATION

a) Principle

The first method studied for pedestrian navigation is based on the traditional INS algorithm. In the classical inertial navigation, accelerometer measurements are basically compensated for gravity and inertial forces. They are then integrated twice in the navigation frame to get the position of the unit.

Let $\vec{M}^{(I)}$ be the position vector of the PNM in the inertial frame ***(I)***. Its time derivative with respect to ***(I)*** and expressed in ***(I)*** is given by :

$$\left(\frac{d}{dt}\vec{M}^{(I)}\right)_I = \vec{v}_{m/e}^{(I)} + \vec{\Omega}_{e/I}^{(I)} \wedge \vec{M}^{(I)}$$

Equation 1

where :

- Superscript between brackets stands for the coordinate system in which vectors are expressed.
- m stands for mobile, or PNM.
- ***(e)*** stands for the ECEF reference frame.
- ***(I)*** stands for the Inertial reference frame.
- $\vec{\Omega}_{e/I}^{(I)}$ is the rotation vector of ***(e)*** relative to ***(I)***.
- $\vec{v}_{m/e}^{(I)}$ is the velocity of the mobile relative to ***(e)***.

The accelerometer triad senses the true acceleration of the PNM biased by the gravitational field. The fundamental principle of the dynamic applied to the PNM in the inertial frame ***(I)*** yields to :

$$\vec{f}^{(m)} = \vec{a}_{m/I}^{(m)} - \vec{G}^{(m)}(M)$$

Equation 2

where :

- ***(m)*** stands for the mobile reference frame.
- $\vec{f}^{(m)}$ is the specific force measured by the accelerometers.
- $\vec{a}_{m/I}^{(m)}$ is the true PNM acceleration relative to ***(I)***.
- $\vec{G}^{(m)}(M)$ is the gravitational field expressed in ***(m)***, at point M of the PNM.

Equation 2 combined with the time derivative of Equation 1 yields to the following relation :

$$\left(\frac{d}{dt}\vec{v}_{m/e}^{(I)}\right)_I = \vec{f}^{(I)} + \vec{g}^{(I)}(M) - \vec{\Omega}_{e/I}^{(I)} \wedge \vec{v}_{m/e}^{(I)}$$

Equation 3

where :

- $\vec{g}^{(I)}(M) = \vec{G}^{(I)}(M) - \vec{\Omega}_{e/I}^{(I)} \wedge \left(\vec{\Omega}_{e/I}^{(I)} \wedge \vec{M}^{(I)}\right)$.
  $\vec{g}$ is the local gravity vector.

Quantities of interest are those expressed in the navigation frame ***(n)*** (North, East, Down). Using the rotation matrix R_{I2n} from ***(I)*** to ***(n)***, Equation 3 can be re-written as :

$$\left(\frac{d}{dt}\vec{v}_{m/e}^{(n)}\right)_n = \vec{f}^{(n)} + \vec{g}^{(n)}(M) - \left(\vec{\Omega}_{n/e}^{(n)} + 2\vec{\Omega}_{e/I}^{(n)}\right) \wedge \vec{v}_{m/e}^{(n)}$$

Equation 4

where :

- $\vec{\Omega}_{n/e}^{(n)}$ is the rotation vector of ***(n)*** relative to ***(e)***.

Equation 4 is the differential equation of inertial navigation. Based on this equation, the usual INS mechanization is implemented as shown in Figure 3. The rotation matrix R_{m2n} that rotates measurements from the mobile frame ***(m)*** into the navigation frame ***(n)*** is computed using the rotation quaternion q. Position and velocity are computed every sampling period T_s.

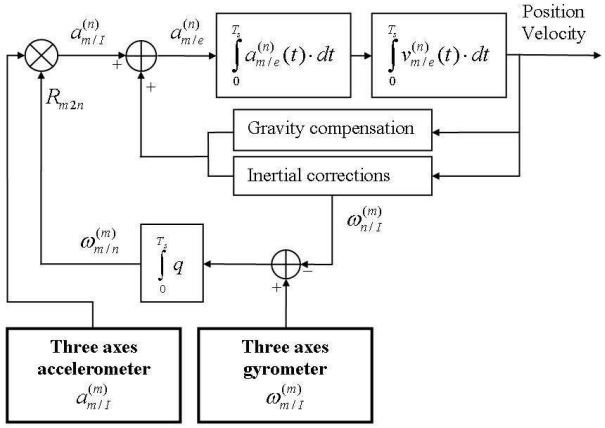

Figure 3 – INS mechanization.

The accuracy of such navigation algorithm is addressed in the next sub-section.

b) Accuracy

The accuracy of the navigation system presented in Figure 3 mainly depends on two issues. The most important one is the quality of the sensors used in the PNM. Because of manufacture process and sensor technology, the output of the sensor is usually different from the actual input, so that the general sensor measurement model is as follows :

$$a_{output}(t) = [1 + S(t)] \cdot a_{true}(t) + b(t) + n(t)$$

where :

- a_{true} is the actual quantity to measure.
- a_{output} is the value provided by the sensor.
- S is the scale factor affecting the true measurement.
- b is the bias affecting the true measurement.
- n is the sensor noise.

Both accelerometer and gyrometer measurements are affected. As a consequence, bias and scale factor introduce drift in the computation of the rotation matrix R_{m2n}, as well as in the integration of acceleration and velocity. They both contribute to a decrease of the overall accuracy.

The second factor responsible for position, velocity and attitude error comes from approximations made in the mechanization. They are twofold. Indeed, first order Taylor expansion models are used to compute some parameters, as for example quaternion. On the other hand, some quantities estimated at epoch k are used in the computation process at epoch k+1. Nevertheless, in the following this type of error is considered negligible with respect to the previous one.

In order to analyse the performance of the navigation algorithm, a propagation error method is used. Based on a dynamic error model of the INS as those presented in [6], the position, velocity and attitude error δp, δv and δρ are propagated using the state transition matrix F. The propagation model is those of Equation 5, where all sub-matrices are developed in [6] :

$$\begin{bmatrix} \delta\dot{p} \\ \delta\dot{v} \\ \delta\dot{\rho} \\ \delta\dot{\varepsilon}_a \\ \delta\dot{\varepsilon}_g \end{bmatrix} = \begin{bmatrix} Fpp & Fpv & F_{p\rho} & 0 & 0 \\ Fvp & Fvv & Fv\rho & F_{v\varepsilon_a} & 0 \\ F\rho p & F\rho v & F\rho\rho & 0 & F_{\rho\varepsilon_g} \\ 0 & 0 & 0 & F_{\varepsilon_a\varepsilon_a} & 0 \\ 0 & 0 & 0 & 0 & F_{\varepsilon_g\varepsilon_g} \end{bmatrix} \cdot \begin{bmatrix} \delta p \\ \delta v \\ \delta\rho \\ \delta\varepsilon_a \\ \delta\varepsilon_g \end{bmatrix} + \begin{bmatrix} n_p \\ n_v + n_{\varepsilon_a} \\ n_\rho + n_{\varepsilon_g} \\ n_{\varepsilon_a} \\ n_{\varepsilon_g} \end{bmatrix}$$

Equation 5 – INS error model

The state vector is augmented by accelerometers errors $\delta\varepsilon_a$ including bias and scale factor models, and gyrometers errors $\delta\varepsilon_g$ including bias model.

The performance of the sensor assembly is then analyzed using the propagation model of Equation 5. For the simulation, the PNM is assumed to be at rest, and biases and scale factor values are those computed in section I. The resulting predicted horizontal RMS position error is plotted in Figure 4 as the blue dashed curve. Given the low-cost sensors used in the assembly, the horizontal RMS error is predicted to be nearly 570 meters after 60 seconds.

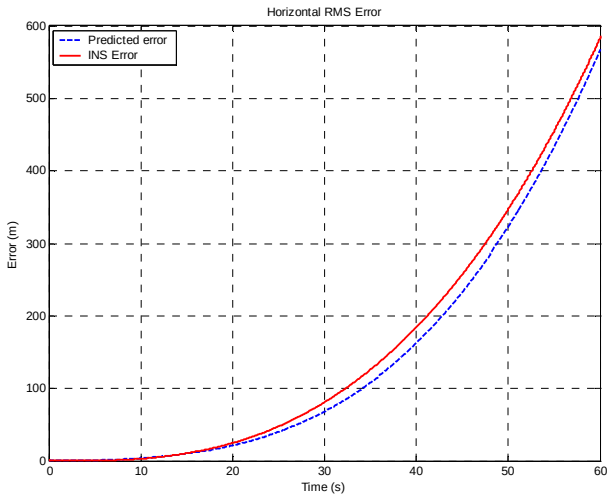

Figure 4 – Horizontal position error.

To assess the simulation result, real data were collected from the idling PNM over 60 seconds and processed through the navigation algorithm depicted in Figure 3. The resulting horizontal RMS error is plotted as the red solid curve in Figure 4.

Both horizontal errors are close from each other. There is only a slight difference mainly due to stochastic models chosen for biases and scale factor, which probably do not match perfectly the reality. Nevertheless, the error propagation model matches quite well the INS static behaviour so that it can be used to find ways for improving the navigation solution.

As discussed above, errors in the navigation algorithm are basically those introduced by biases and scale factor. To analyze the effect of each factor, three independent error propagation simulations are conducted. They all involve only one error parameter among accelerometer bias, accelerometer scale factor and gyrometer bias to avoid interference between these factors. Values used in the error model are again these of section I.

Predicted horizontal RMS errors are plotted in Figure 5. From the three error factors, the bias affecting the gyrometer measurements is the one that degrades significantly the accuracy of the navigation algorithm. Opposite, the accelerometer scale factor has very little effect on the overall accuracy.

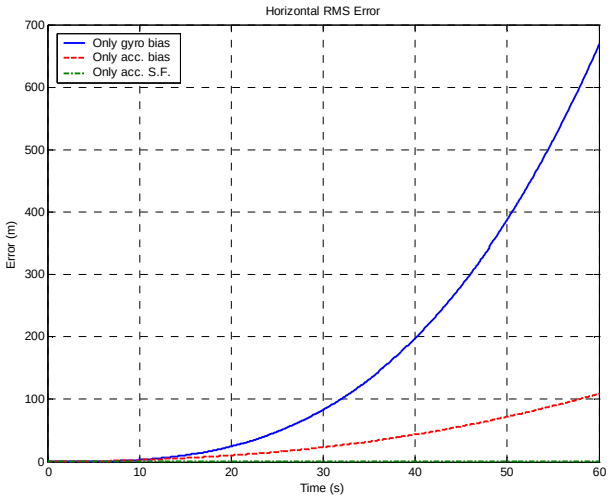

Figure 5 – Biases and Scale Factor impact.

Consequently, the improvement of the navigation algorithm relies first on the reduction of the effect of gyrometer bias and then accelerometer bias. Scale factor effect is from this sub-section neglected.

The next sub-section deals then with the improvement of the classical inertial navigation algorithm.

c) Enhanced Inertial Navigation (EIN) Algorithm

To compensate for accelerometer and gyrometer biases and enhance the inertial navigation algorithm, it is necessary to go beyond classical compensation methods based on stochastic error models. Indeed, these models are only approximations of the actual behaviour of biases affecting the measurements. Thus, there are still residual errors that degrade the overall performance.

One reliable way to improve bias estimation is to use external data. Given our sensor assembly, additive external measurements can indeed be used to provide redundant information for bias estimation. The gyrometer biases are likely to be estimated using accelerometers for inclination drift and magnetometers for heading drift. The combination of the three sensors provides a drift-free attitude, which can be fit in the inertial navigation algorithm to compensate for gyro biases effect.

Accelerometers biases are more difficult to estimate using our PNM assembly. Nevertheless, some assumptions can be made about the spectrum of the measurements to limit the residual error. Moreover, once rotated into the navigation frame ***(n)***, accelerometer vertical channel can also be checked with pressure sensor measurements to limit vertical velocity and vertical position drift.

With all these improvements, the classical inertial navigation algorithm is modified, such as the EIN System mechanization is as follows :

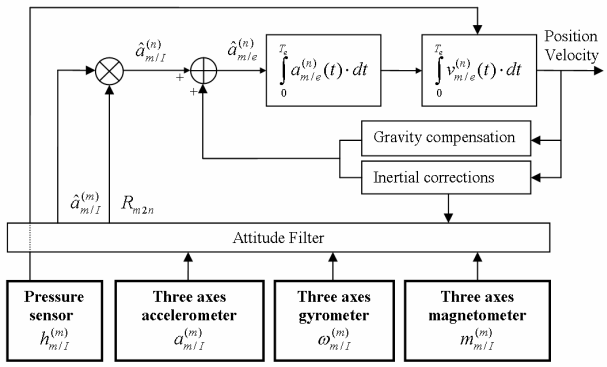

Figure 6 – EIN System Mechanization.

III – SIGNAL SIGNATURE-BASED NAVIGATION

a) Principles

The second pedestrian navigation method studied in this paper is based on the signature of the PNM accelerometer signal. To cope with the issue of accelerometer bias, a model linking the velocity or step length of the pedestrian and characteristics of the sensed acceleration is established as described in [7] and [8] :

$$s_p = f_1(individual\ parameters)$$
$$v_p = f_2(individual\ parameters)$$

where :

- s_p is the step length of the walking pedestrian.
- v_p is the velocity of the walking pedestrian.
- f_1 and f_2 are the model function. They can be linear or non linear.

The individual parameters used to model the actual pedestrian velocity are computed based on the acceleration signal. To get rid of the orientation of the PNM containing the accelerometers, the acceleration signal is considered to be the total acceleration magnitude defined as :

$$a_k = \sqrt{\left(a_{x,k}^{(m)}\right)^2 + \left(a_{y,k}^{(m)}\right)^2 + \left(a_{z,k}^{(m)}\right)^2}$$

Equation 6 – Global acceleration

where :

- a_k is the global acceleration magnitude sensed by the PNM at epoch k. This global acceleration is biased by the gravity vector.
- $a_{x,k}^{(m)}$ is the acceleration along the x axis of the PNM, expressed in the mobile frame ***(m)*** at epoch k.
- $a_{y,k}^{(m)}$ is the acceleration along the y axis of the PNM, expressed in the mobile frame ***(m)*** at epoch k.
- $a_{z,k}^{(m)}$ is the acceleration along the z axis of the PNM, expressed in the mobile frame ***(m)*** at epoch k.

Based on the signal of Equation 6, parameters are computed to model the velocity or step length of the walking pedestrian. These parameters are chosen in such a way they reflect the behaviour of the walk. To fit the mathematical model with the actual velocity model, a regression algorithm is then applied on the computed parameters using external velocity measurements when they are available. The pedestrian velocity model can then be expressed as follows :

$$v_p = \alpha_1 \cdot PARAM_1^{\beta_1} + \alpha_2 \cdot PARAM_2^{\beta_2} + \cdots$$

where :

- α_i and β_i are the regression coefficients.
- $PARAM_i$ are the parameters computed based on the total acceleration a_k.

Once the velocity model is found, the curvilinear distance that is travelled is estimated by integration. In a second time, the azimuth of displacement is estimated. Then the trajectory can be reconstructed through a classical Dead Reckoning algorithm :

$$N_{k+1} = N_k + d_{[k,k+1]} \cdot \cos(\psi_k)$$
$$E_{k+1} = E_k + d_{[k,k+1]} \cdot \sin(\psi_k)$$

where :

- ψ_k is the azimuth of displacement at epoch k.
- N_k is the north position of the pedestrian at epoch k.
- E_k is the east position of the pedestrian at epoch k.
- $d_{[k,k+1]}$ is the curvilinear distance travelled between epoch k and k+1.

b) Signal Signature-Based Navigation Mechanization

As it has been presented in the previous sub-section, both distance and heading estimation are done separately. In this study, the travelled curvilinear distance computation is performed through the estimation of the pedestrian velocity. Depending on the navigation mode (constrained or unconstrained), the model is established using an association of parameters chosen among vertical velocity, frequency, mean, Root Mean Square and third-order moment. The vertical velocity is provided by the pressure sensor, whereas the others are computed from the global acceleration of Equation 6.

The heading information is usually obtained through the integration of gyrometer measurements. In this paper, the heading information is computed using the attitude filter described in section II.

The Signal Signature-Based Navigation (SSBN) System is then mechanized as depicted in Figure 7.

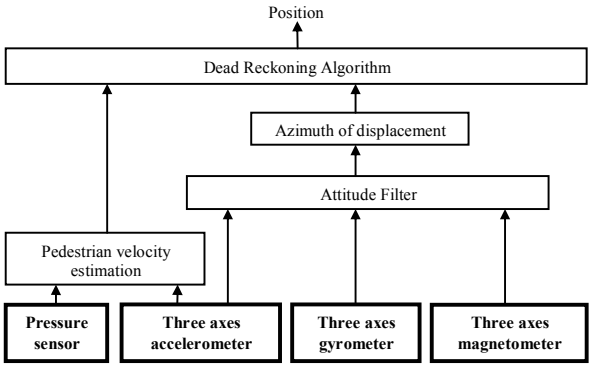

Figure 7 – SSBN System Mechanization.

IV – NAVIGATION ISSUES

Before testing the algorithms described in the two previous sections, two typical issues affecting both navigation methods and occurring while walking are investigated. They are Euler's angles singularity and displacement direction detection.

a) Euler Singularities

Euler's angles are used to define the attitude of the PNM in the navigation frame. The definition used in this paper is the one presented in Figure 8. φ is the roll angle, θ the pitch angle and ψ the yaw angle.

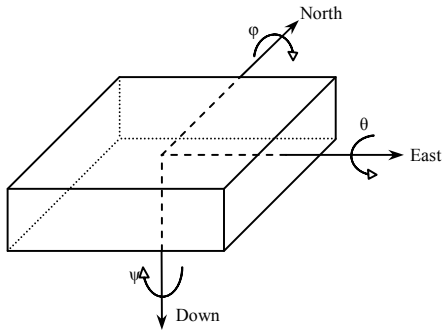

Figure 8 – Euler's angles definition.

Usually, these angles are computed from the rotation matrix R_{m2n} defined in section II and III. There are different ways of computing this rotation matrix, whether using direct cosine method or quaternion integration. The quaternion method is used in this paper.

The direct cosine expression of R_{m2n} is as follows :

$$R_{m2n} = \begin{bmatrix} c\theta \cdot c\psi & s\varphi \cdot s\theta \cdot c\psi - c\varphi \cdot s\psi & c\varphi \cdot s\theta \cdot c\psi + s\varphi \cdot s\psi \\ c\theta \cdot s\psi & s\varphi \cdot s\theta \cdot s\psi + c\varphi \cdot c\psi & c\varphi \cdot s\theta \cdot s\psi - s\varphi \cdot c\psi \\ -s\theta & s\varphi \cdot c\theta & c\varphi \cdot c\theta \end{bmatrix}$$

where :

- c stands for cosine.
- s stands for sine.

From these expressions, it is clear that the Euler's angles can be computed according to the three relations below :

$$\varphi = \arctan\left(\frac{R_{m2n}(3,2)}{R_{m2n}(3,3)}\right)$$
$$\theta = -\arcsin\left(R_{m2n}(3,1)\right)$$
$$\psi = \arctan\left(\frac{R_{m2n}(2,1)}{R_{m2n}(1,1)}\right)$$

Obviously, when the pitch angle reaches +/-90°, the roll and yaw angles are mathematically not defined; and thus can not be recovered from the rotation matrix R_{m2n}. Since the PNM will be put in a pocket in any direction, this use case may happen.

To avoid the singularity, the pitch angle is continuously checked. If it goes higher than a determined threshold, then the PNM is virtually rotated along the pitch angle, so that no singularity should appear. As an illustration, Figure 9 gives an example of the Euler's Angles Singularity Resolution (EASR) algorithm developed in this study. The bold part of each plot is the corrected angle. There are no singularities left after processing. Roll and yaw angles are perfectly observable, even for pitch angle values of +/-90°.

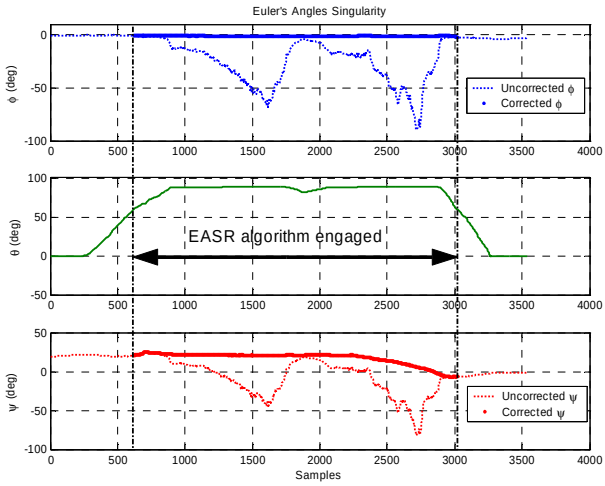

Figure 9 – Euler's angles singularity resolution.

b) Displacement Direction Detection

For both navigation methods, there are two typical cases of use in which the azimuth of displacement has to be estimated. It depends on the motion of the PNM relative to the user, whether it follows the movement of the pedestrian or not.

In the first case called constrained navigation, the PNM is fixed anywhere onto the user, or simply put in a pocket of the user. The heading computed by the PNM is thus the displacement azimuth, biased by an additive constant because of the non alignment of the PNM heading axis and the body direction of walk.

In the second case called unconstrained navigation, the true azimuth of displacement is not the heading provided by the PNM, since the PNM has got its own movement relative to the pedestrian. Figure 10 illustrates the issue. The orientation of the heading axis of the PNM is likely to change, as well as the direction of walk, but both are not correlated. Thus, tracking the heading of the PNM is

no more relevant.

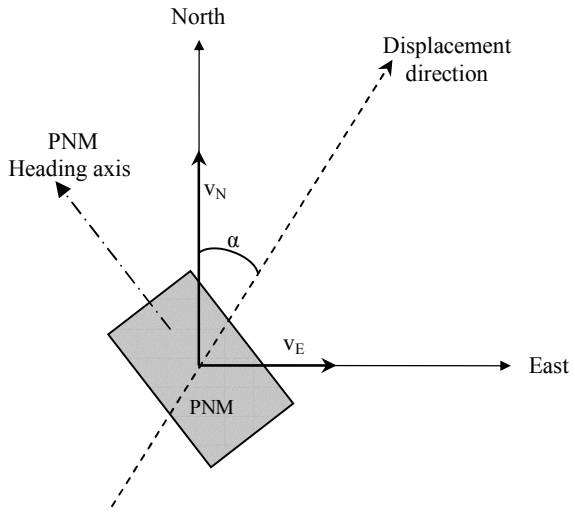

Figure 10 – Unconstrained navigation issue.

The solution investigated in this paper is based on velocity estimation in the navigation frame ***(n)***. Whatever the technique used for navigation, the velocity accessible from the PNM measurements is the velocity of the PNM relative to the ground. It can be decomposed as follows :

$$v^{(n)}_{PNM/gr} = v^{(n)}_{PNM/u} + v^{(n)}_{u/gr}$$

where :

- u stands for user.
- gr stands for ground.

Assuming the velocity of the PNM relative to the user $v^{(n)}_{PNM/u}$ negligible compared to those of the user relative to the ground $v^{(n)}_{u/gr}$, east and north velocities of both navigation algorithms can then be used for the computation of the true displacement azimuth. As shown in Figure 10, it is given by :

$$\alpha = \arctan\left(\frac{v_E}{v_N}\right)$$

Equation 7 – True Displacement Azimuth.

V – TESTS RESULTS

In this section, the two methods presented in Figure 6 and in Figure 7 are tested as pedestrian navigation systems. In the tests conducted, GPS data are only used to recover the initial heading information. These data also give the different reference trajectories.

Experiments are conducted in three phases. First, the static behaviour of the algorithms is tested. Second, the performance of the constrained navigation algorithm is analysed. Finally, the unconstrained navigation is addressed.

a) Static algorithm performance

In this test, the PNM is laying at rest in a pocket of the user for 60 seconds, in a random attitude. Three algorithms are compared : the Classical and Enhanced Inertial Navigation algorithms (respectively CIN and EIN), and the Signal Signature-Based Navigation (SSBN) algorithm. Results are plotted in Figure 11.

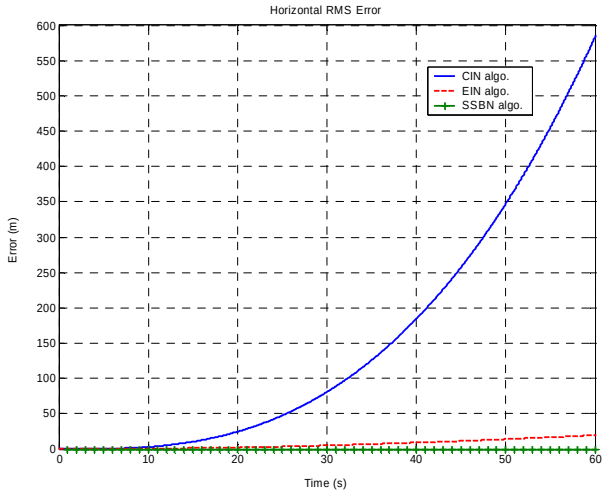

Figure 11 – Static Error.

As it could be expected, The SSBN algorithm outperforms the two other algorithms. Indeed, the error is 0 meter at the end of the run, since it is based on motion detection. Although it still remains some residual error, the EIN algorithm gives good results compared to the CIN one. After 60 seconds of navigation, the error is indeed reduced by 95% (from 570 meters for the CIN algorithm to 25 meters for the EIN algorithm). The improvement provided by the attitude filter is thus very effective.

b) Constrained Navigation

Dynamic tests are conducted to analyse the behaviour of the different navigation algorithms. In this sub-section, only constrained pedestrian navigation is addressed. This means that the PNM is put on a pocket of the user with a random attitude, and is no more moved until the end of the run.

Figure 12 shows the trajectories of the first constrained navigation test. This test lasts 1min and 10s, and the true DGPS trajectory is represented as the blue solid curve.

The offset between the displacement direction and the PNM heading axis was found manually.

The SSBN solution is the more accurate one, with a final horizontal RMS error of about 5m. No major drift whether in attitude or in position is noticeable on this short duration run.

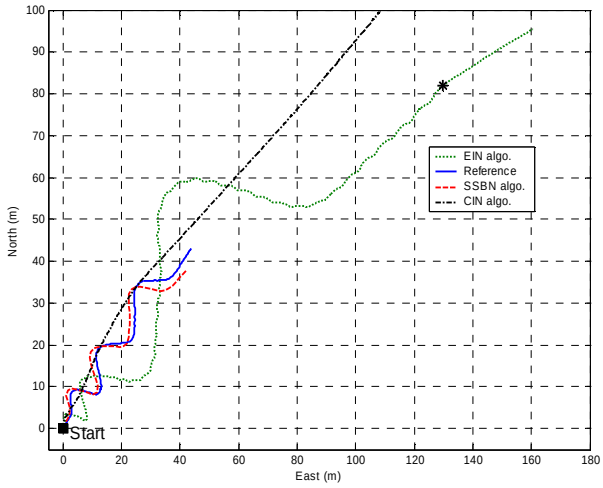

Figure 12 – Navigation Solutions.

The green dotted plot stands for the EIN position solution. Its final horizontal error is about 90 meters. Although the pedestrian is waiting for 3 seconds and no drift is remarkable at the beginning, this is not the case at the end of the run. Indeed the black asterisk is located at the actual end of the walk, so that the position drift while the pedestrian is stopped for another 3 seconds is clearly observable. On the other hand, the attitude of the pedestrian matches quite well those of the reference, with no apparent heading drift. Consequently, it can be stated that the attitude provided by the attitude filter is reliable. The filter is efficient in removing gyro drifts. However, it fails to estimate accurately the acceleration biases affecting the measurements, leading to a drifting position solution. Nevertheless, it clearly outperforms the CIN algorithm, whose position solution is represented by the black dash-dotted plot. The improvement is of about 90% since the corresponding final horizontal error is 950 meters (The whole trajectory is not plotted here for visual convenience).

As seen above, the attitude filter is drift-free but does not remove all the bias contained in the accelerometer measurements. On the other hand, the SSBN algorithm is more robust on such bias. Therefore, to assess the complementary of the attitude filter and the SSBN algorithm, another test is conducted over a longer period. In this run, the pedestrian walks 1 km in 13 min with the PNM put in a pocket with a random attitude. The PNM is still not moved during the run. The resulting trajectory using 100% of GPS data for velocity model calibration is plotted in Figure 13. The navigation solution is accurate within 20 meters over the whole test period, with a final horizontal RMS error of 16 meters.

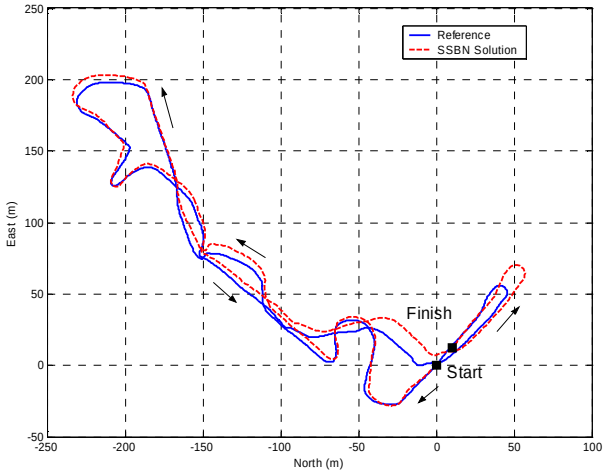

Figure 13 – SSBN Solution – 100% GPS data used.

The same trajectory is plotted using only 10% of GPS data for velocity model calibration. Those 10% of data are collected between the beginning of the walk and the green asterisk, as shown in Figure 14 (approximately 1 minute). The accuracy of the SSBN solution stays within 25 meters over the whole path. This corresponds to 12 minutes of navigation without GPS data. The final horizontal RMS error is 18 meters.

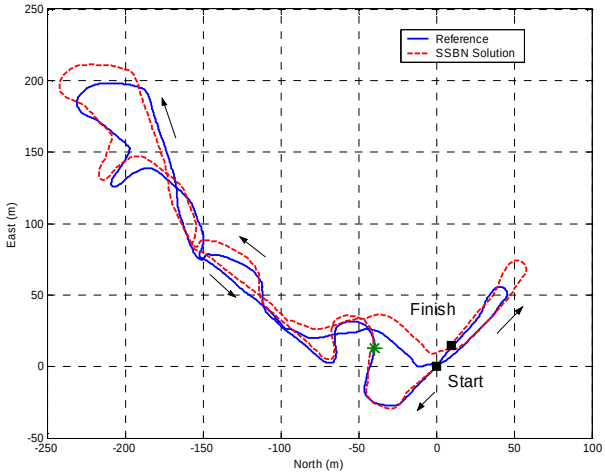

Figure 14 – SSBN Solution – 10% GPS data used.

Even with a little part of data for calibration, the velocity model provides good estimates of the curvilinear travelled distance. Accuracy stays within acceptable limits that are suitable for LBS applications during GPS outages. Moreover, trajectories from Figure 12 and Figure 13 are not affected by Euler's angles singularities, since both reconstructed and reference path match. A position solution can then be provided regardless of any

orientation of the PNM.

c) Unconstrained Navigation

However, the position solution may be affected by the movement of the PNM, if the latter is contained in a handheld device such as a cell phone or a PDA. To assess the capability of the SSBN algorithm to provide a reliable position solution while the PNM is moved during the walk, another dynamic test is performed.

In this test, a pedestrian follows an athletic track of 250 meters for 4 minutes. At the beginning of the test, the PNM is in the pocket of the pedestrian, with a random attitude. While walking, he pulls randomly the PNM out and moves it as if he would look at a cell phone. Then he replaces it inside his pocket. The movements of the PNM are done according to the assumption made in section IV b). The PNM is moved 13 times during the run. The resulting trajectories are given in Figure 15.

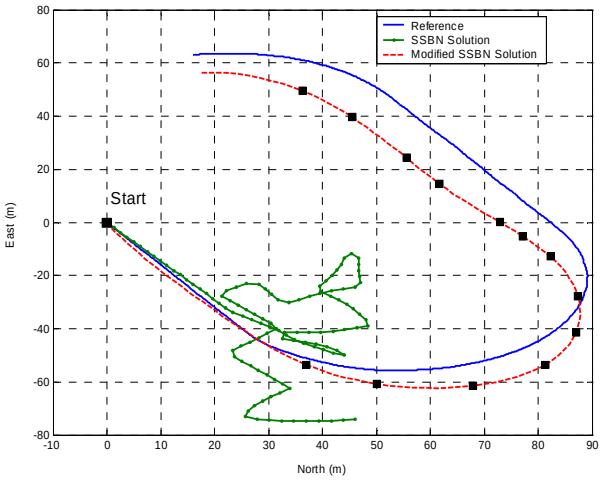

Figure 15 – Unconstrained navigation solutions.

The black squares are for epochs at which the PNM is moved (except the first one). The uncorrected trajectory computed with the SSBN algorithm is clearly not relevant of the actual path followed by the pedestrian.

However, the corrected trajectory computed with the Modified SSBN (MSSBN) algorithm matches quite well the true trajectory. Both travelled distance and displacement direction are coherent with the actual ones. Indeed, there is no noticeable drift in the curvilinear travelled distance. The velocity model once calibrated is thus robust to such PNM movements during the walk.

For visual convenience, the trajectory computed with the Modified EIN algorithm, (i.e. the EIN algorithm modified to estimate both North and East velocities) is not plotted here for comparison. The acceleration biases yield to a drifting position that goes quickly out of the figure.

Moreover, as shown is Figure 16, the MSSBN heading shape is approximately those of the GPS, which is taken as reference. Consequently, this tends to prove the efficiency of the displacement direction detection algorithm discussed above.

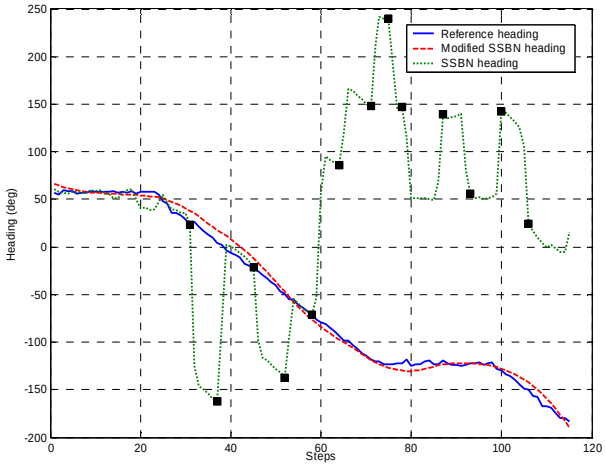

Figure 16 – Displacement direction detection result.

CONCLUSION

The aim of this paper was to analyse the performance of two MEMS-based navigation algorithms used in an alternative positioning system during GPS outages. The algorithms rely on four common low-cost sensors. They have been tested under two different use cases: first, the constrained navigation, and second, the unconstrained navigation. Since in both cases the PNM is placed in a pocket of the pedestrian with a random attitude, an Euler's angles control algorithm has been successfully implemented to avoid singularities that would degrade the heading information.

Errors sources have been identified and classified with respect to their impact on the overall accuracy. It has been shown that the gyro biases are responsible for most of the errors, more than accelerometer biases. Consequently, a filter has been designed to cope with these issues and provide a drift-free attitude. Its integration with the CIN algorithm has led to the implementation of an EIN algorithm. The performance of the EIN algorithm is more than 90% better than the CIN algorithm using the same sensors. However, even if the navigation solution drift is well reduced because of the attitude filter, this is not enough to be used as an alternative positioning system for pedestrian. Indeed, accelerometer biases are badly estimated and consequently are responsible for the residual errors. The position is drifting so that this algorithm can only be used for short term positioning for both constrained and unconstrained navigation.

To cope with this issue, a SSBN algorithm has been implemented. Coupled with the attitude filter, this algorithm provides reliable position solutions over time, with a horizontal RMS error within 25 meters for 12 minutes of constrained navigation without GPS assistance for velocity model calibration.

The unconstrained navigation mode has also been studied and tested with the MSSBN algorithm. In this mode, the PNM has got its own motion relative to those of the pedestrian. Assuming a low PNM dynamic with respect to pedestrian motion, it has been shown that the true displacement direction could be estimated with an acceptable accuracy (25 meters for 4 minutes of navigation). Even if the horizontal RMS error is higher than in the constrained navigation mode, the modified algorithm makes the system robust to medium motion of the PNM.

Consequently, the SSBN and MSSBN algorithms are the more suitable for pedestrian navigation during GPS outages.

ACKNOWLEDGMENTS

The author would like to thank Ph.D. student Benjamin Chibout and Dr. Christophe Macabiau for their help during the collect of the data used in this paper.

REFERENCES

[1] – Intersema MS5534 data sheet.

[2] – Analog Devices ADXL202E data sheet.

[3] – Analog Devices ADXRS300 data sheet.

[4] – Philips KMZ52 data sheet.

[5] – Philips Application Note AN00022.

[6] – Farrell J. A., Barth M. *The Global Positioning System & Inertial Navigation*.

[7] – Ladetto Q. (2003). *Ph.D. thesis report*.

[8] – Gabaglio V. (2003). *Ph.D. thesis report*.