Yet another IMU simulator: validation and applications

M.E. Parés, J.J. Rosales and I. Colomina
Institute of Geomatics
Generalitat de Catalunya & Universitat Politècnica de Catalunya
Castelldefels
Spain

Keywords: Inertial Measurement Units, Simulations.

Abstract

The availability of inertial systems in the market, with different features, performances and prices, as well as advances in their integration within other navigation systems (mainly GPS) have brought the navigation and geodetic community to a wide spectrum of research and commercial possibilities. The question that, given an application and a budget, naturally arises is: which inertial system configuration would fit the requirements better?

Some of the Institute of Geomatics (IG) research lines require the development, implementation and validation of INS/GNSS algorithms and systems. This has lead the IG to develop an Inertial Measurement Unit (IMU) simulator for testing and validating navigation algorithms. For a given georeferenced trajectory (time, position, velocity and attitude), this software tool simulates the output of an IMU. One of its main characteristics is its ability to simulate different types of IMUs based on prior knowledge of the IMU error's properties. This might allow the researchers to simulate a wide range of scenarios, assess the performance of different inertial systems and to decide which configuration better meets their mission and budget requirements.

In this paper, a description of the algorithms and models implemented in the software is presented. Furthermore, the first results of this simulation tool are shown, paying special attention to its applications in several research projects. Finally, some further developments and applications in mission planning are discussed.

1 Introduction

From flight simulations to material endurance analysis, simulation has become instrumental in technology development. The improvement in computers' capability has allowed 21st century society to achieve milestones unbelievable just 60 years ago. It is not daring to say that, in a technical context, almost everything can be simulated with a computer and a good mathematical model.

As well as computers, Earth observation engineering has also gone through a revolution. Most of the new sensors for Earth observation, such as LiDAR or Radar, require the knowledge of attitude. Attitude that is usually obtained from INS systems. At the moment, a wide range of inertial systems are available on the market, from navigation ones (whose angular sensors drift less than 0.01 deg/h) to low-cost ones (whose angular sensors drift more than 1 deg/sec). Depending on the precision and accuracy required for an application, one system will be more suitable than another. For example, for images taken for forest surveying purposes, an automotive grade IMU might be enough, while for high precision agriculture purposes, a tactical grade unit is at least required.

The tool presented in this paper emulates the behaviour of standard IMUs with three linear accelerometers and three angular rate sensors in an orthogonal configuration. This tool has been mainly developed to validate new methods involving inertial technology.

In this paper, the motivations to implement and use an inertial simulator are presented first. Then, the theoretical basis for the simulations are described. Later, some of the main applications of the tool are presented, as well as some of the results obtained with it. Finally, the reader will find ongoing and future work and the main conclusions obtained from the use of the tool.

2 Motivation

Some of the IG research lines require the development, implementation and validation of externally aided INS algorithms, specially INS/GNSS. In this area, the IG is developing a system made up of signal generators, signal filters and signal analysers, as well as trajectory generators and trajectory analysers and, as its main component, an externally aided inertial navigator. In figure 1 the relationship between these components is shown.

The main purpose of the signal and trajectory generators is to provide data to test and validate new navigation algorithms under controlled circumstances. Signal and trajectory analysers are used to characterize the error of various signals. And the navigator is used to determine trajectories from different sensors.

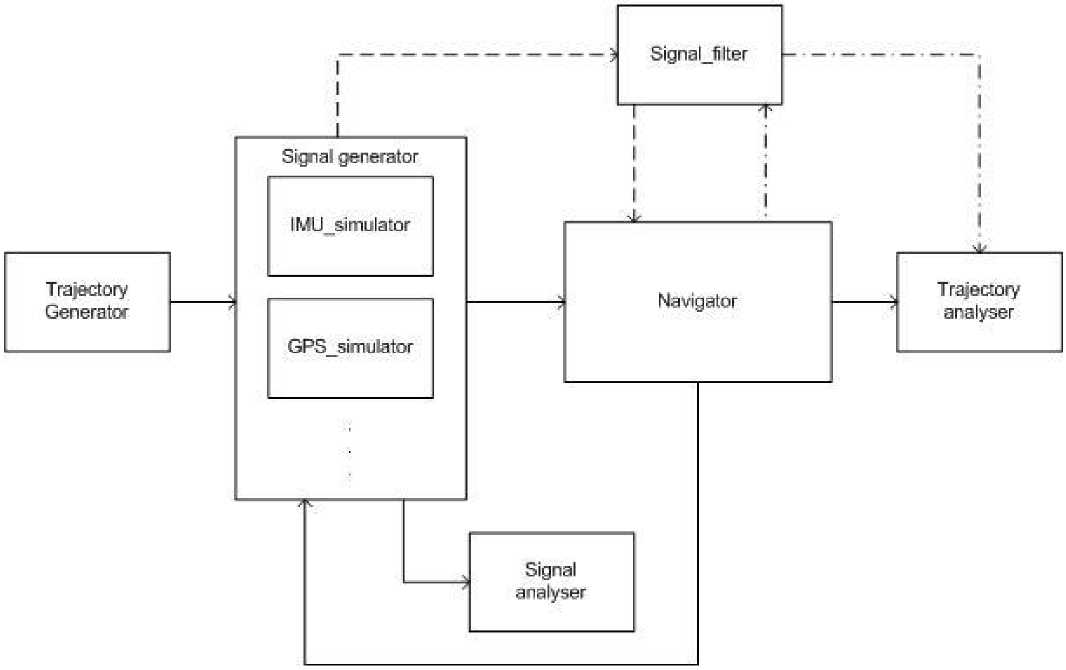

Fig. 1: IG's navigation system.

3 Simulator Design Principles

The fundamentals of the IMU simulator are the functional model that characterize the inertial movement (INS mechanization equations), the stochastic models that characterize IMU errors and the "geodetic" model that contextualizes the previous models.

As any real IMU, the simulator computes angular velocities and linear accelerations when fed with a certain trajectory. Given a set of times, positions, velocities and attitudes, the system provides the signal that an IMU measures as if it were in that situation.

The simulator computes first the errorless inertial measurements and then corrupts them with errors.

3.1 Software Architecture

The architecture of a software system is a description of the system which comprises components, the main properties of those components, and the relationships between them. Usually, in order to describe those relations, software modelling languages are used. The Unified Modelling Language (UML) has become the "de facto" standard. Among others there are the Object Model (where the main components and their relationship are presented) and the Activity Model (where the different actions done by the system during an execution are presented.)

3.1.1 Object Model

The simulator is designed under the Object-Oriented (O-O) paradigm and is coded in the `C++` language. The main reason for choosing the O-O paradigm instead of procedural programming is that it allows the system to be *modular* and *extensible*, two basic characteristics for a system whose main purpose is to validate methods.

As mentioned earlier, the errorless observations and the errors are computed separately. As it can be seen in figure 2 there is one primary control class (`kernel`) that steers the general data flow and two secondary control classes (`obs_kernel`) and (`error_kernel`) that drive the data flow during the generation of observations and perturbations. There is also a container class where all the characteristics of the IMU to be simulated are kept. Finally, there are observation computation classes (`obs_accel` and `obs_gyro`) where the correct signal is computed and perturbation computation classes (`error_b`, `error_sf`, `error_m`, `error_w` and `error_q`) where the noise is added to the signal.

3.1.2 Activity Model

In what concerns the data processing, the simulator reads the trajectory data sequentially and computes for each trajectory event or item the linear accelerations and the angular velocities first according to the models

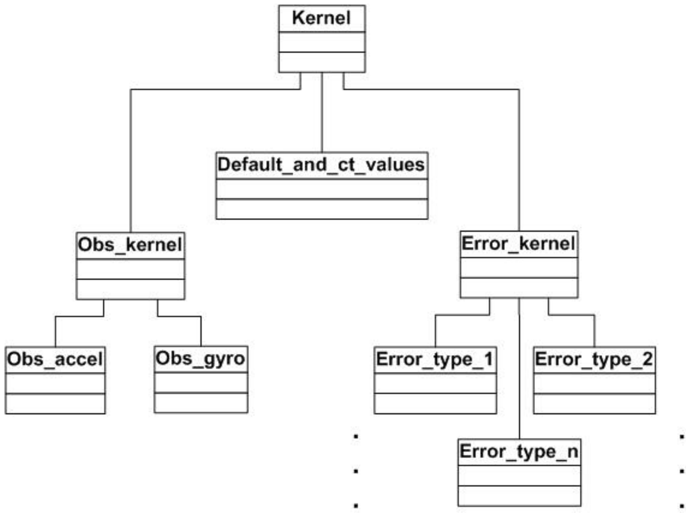

Fig. 2: Simulator object model.

presented in the next subsection. After that, the signal is modified by adding errors such as biases or scale factors. This process is presented graphically in figure 3. In the latest version of the simulator, the errors added are scale factors, biases, misalignments, white noises and quantizations in this same order. But as mentioned before, the modularity of the system allows to change this order easily, if the user requests so.

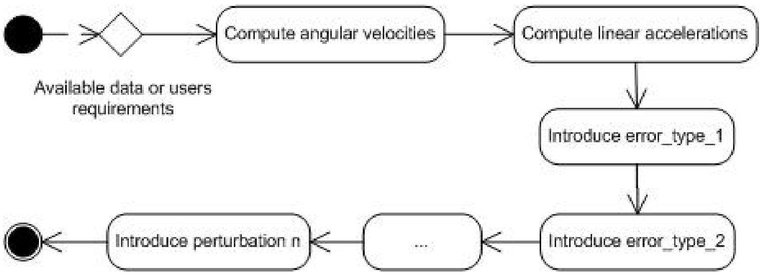

Fig. 3: Simulator activity model.

3.2 Errorless Signal: Mechanization Equations

In order to simulate an errorless signal, the mechanization equations are rewritten in such a way that angular velocities and linear accelerations are isolated. In the next subsections the models used are described.

3.2.1 Angular Velocities

Given a trajectory, the simulator computes angular velocities for an orthogonal triad of angular rate sensors. The model implemented is

$$\Omega_{ib}^{b} = C_{n}^{b}(\dot{C_{b}^{n}} + \Omega_{in}^{n}) \tag{1}$$

where Ω_{ib}^{b} is a skew-symmetric matrix relating the body frame (b) with the inertial frame (i) obtained from the angular local velocities measured by the rate sensors; Ω_{in}^{n} is a skew-symmetric matrix relating the inertial frame (i) with the navigation frame (n) and obtained from the position given by the reference trajectory; and C_{n}^{b} is a rotation matrix relating the body frame (b) with the navigation frame (n) and obtained from the attitude values.

For details on equation 1, the reader is referred to any textbook on inertial navigation such as [4] or [7].

Note that the matrix $\dot{C_{b}^{n}}$ can be either computed numerically from C_{n}^{b} or analytically after numerically differentiating its components. After analysing the results for both methods, even though both are equally valid, the first one was selected because of its low computational burden.

3.2.2 Linear Accelerations

Given a trajectory, the simulator computes linear accelerations for an orthogonal triad of accelerometers. The model implemented is

$$f^b = C_n^b(\dot{v^n} + (2\Omega_{ie}^n + \Omega_{en}^n)v^n - g^n(x^n)) \tag{2}$$

where f^b is the vector of linear accelerations in the body frame (b); v^n is the vector of velocities of the sensor in the navigation frame (n); C_n^b is a rotation matrix relating the body frame (b) with the navigation frame (n); Ω_{ie}^n is a skew-symmetric matrix relating the navigation frame (n), the inertial frame (i) and the terrestrial frame (e); Ω_{en}^n is a skew-symmetric matrix relating the navigation frame (n) and the terrestrial frame (e); and $g^n(x^n)$ is the gravity vector expressed in the navigation frame (n).

3.3 Corrupted signal

Once the signal has been computed, the perturbations are added. For information about errors affecting IMUs refer to [1] and [11]. Hereafter the models implemented for various error types are presented.

3.3.1 Scale Factor Errors

The scale factor of an inertial sensor is the relation between the output signal and the quantity which is measuring.

For scale factor effects, the model used is

$$\omega_o = (1 + Sc_\omega + Src_\omega + Sgm_\omega + Srw_\omega) * \omega_i, \tag{3}$$

$$f_o = (1 + Sc_f + Src_f + Sgm_f + Srw_f) * f_i \tag{4}$$

where ω_o and f_o are the output angular velocities and linear accelerations, ω_i and f_i are the errorless data, Sc is the constant component of the scale factor, Src is a random constant stochastic process, Sgm is a Gauss-Markov stochastic process and Srw is a random walk stochastic process. Note that for each component of each vector the values of stochastic errors could be different.

3.3.2 Bias Error

Bias refers to the offset in the measurement provided by an inertial sensor.

The model implemented to introduce them is different for rate sensors and accelerometers. For rate sensors

$$\omega_o = Bc_\omega + Brc_\omega + Bgm_\omega + Brw_\omega + Bgs_\omega * f_i + \omega_i \tag{5}$$

where ω_o is the output data, ω_i is the original data, Bc is the constant component of the bias, Brc is a random constant stochastic process, Bgm is a Gauss-Markov stochastic process, Brw is a random walk stochastic process and Bgs is a component depending on linear accelerations.

The model for accelerometers is

$$f_o = Bc_f + Brc_f + Bgm_f + Brw_f + f_i \tag{6}$$

where f_o is the output data, f_i is the original data, Bc is the constant component of the bias, Brc is a random constant stochastic process, Bgm is a Gauss-Markov stochastic process and Brw is a random walk stochastic process.

3.3.3 Misalignment Errors

Misalignments refer to the errors due to the non-orthogonality of the actual assembly mechanism.

To simulate misalignments the well known models have been implemented

$$\begin{pmatrix} \omega_{xo} \\ \omega_{yo} \\ \omega_{zo} \end{pmatrix} = \begin{pmatrix} 1 & (-y_z)_\omega & (z_y)_\omega \\ (x_z)_\omega & 1 & (-z_x)_\omega \\ (-x_y)_\omega & (y_x)_\omega & 1 \end{pmatrix} \begin{pmatrix} \omega_{xi} \\ \omega_{yi} \\ \omega_{zi} \end{pmatrix} \tag{7}$$

$$\begin{pmatrix} f_{xo} \\ f_{yo} \\ f_{zo} \end{pmatrix} = \begin{pmatrix} 1 & (-y_z)_f & (z_y)_f \\ (x_z)_f & 1 & (-z_x)_f \\ (-x_y)_f & (y_x)_f & 1 \end{pmatrix} \begin{pmatrix} f_{xi} \\ f_{yi} \\ f_{zi} \end{pmatrix} \tag{8}$$

where ω_o and f_o are the result data, ω_i and f_i are the original data, and x_y, x_z, y_x, y_z, z_x and z_y are the misalignments between axis.

3.3.4 White Noise

In order to add the non-systematic errors, white noise is added to the signal.

3.3.5 Quantization

Last the system quantize the signal to simulate the A/D conversors with the following model:

$$\omega_o = [\omega_i/quant] * quant \tag{9}$$

$$f_o = [f_i/quant] * quant \tag{10}$$

where ω_o and f_o are the outputs, ω_i and f_i is the data to correct, $quant$ is the number of quantization, and [.] means the integer part of a real number.

4 Validation

Sometimes verification and validation concepts are misused. We understand that the verification of a piece of software consists on assuring that the tool complies with the previously defined specifications, while the validation consists on checking that the tool accomplishes its intended requirements.

While, in our context, verification used to be a relatively easy process (it consists on checking some specifications), validation could become a puzzle, because of the difficulty to prove that the tested tool does what it is expected to do.

In this section the validation process for the IMU simulator is explained. Note that according to the implementation of the algorithms, the validation of the system was also divided between errorless and perturbed signal.

4.1 Errorless signal validation

In order to validate the errorless signal computed with this tool we begin by simulating static acquisitions across the planet. The output of one of those simulations can be seen in figure 4. After checking that the results are the expected ones (according to the Earth's characteristics) dynamic tests were performed: from straight lines and circular trajectories to more complex ones such as a typical aerial surveying trajectories. For the acceptance of these outputs the generated signal was processed with an INS/GPS navigation software, as the results were the reference trajectory, the errorless signal generated with the IMU simulator was validated.

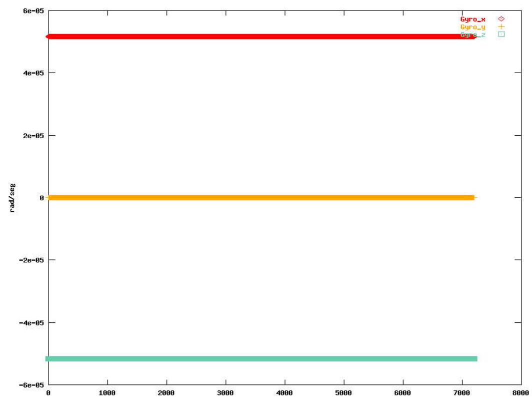

Fig. 4: Errorless simulated angular velocities for an IMU fixed at 41 deg lat, 0 deg lon, 0 m heigh.

4.2 Perturbed signal validation

As mentioned previously, the IMU simulator perturbs the errorless signal with a combination of errors obtained from different stochastic processes (constant, random constant, Gauss_Markov, random walks, etc. See figure 5.) Once it was checked that the implemented stochastic processes work properly, the capability of the tool for reproducing any IMU with this combination had to be proven.

In order to check those, we used various signal analysis techniques. The idea was to prove that the spectral characteristics of the simulated data are similar to the real ones.

The tools chosen to analyse the signals were an empirical statistical analysis of data, a Power Spectral Density (PSD) analysis of data and an Allan Variance (AV) analysis of data. For more information on PSDs and AVs refer to [5] and [6].

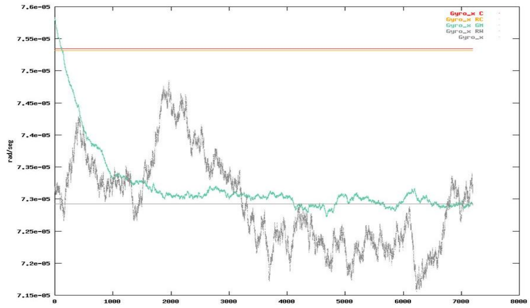

Fig. 5: Stochastic errors computed with the simulator.

The reference data used for those comparative analysis correspond to a static two hours acquisition at the Institute's lab. The data originally comes from a navigation grade IMU (IMAR FJI), a tactical one (Litton LN200), an automotive one (IMAR SSKS) and a Low-Cost one (Motion Pack II). In figure 6 a subset of the data collected during those acquisitions for the vertical accelerometers of FJI, SSKS and MPK-II is presented.

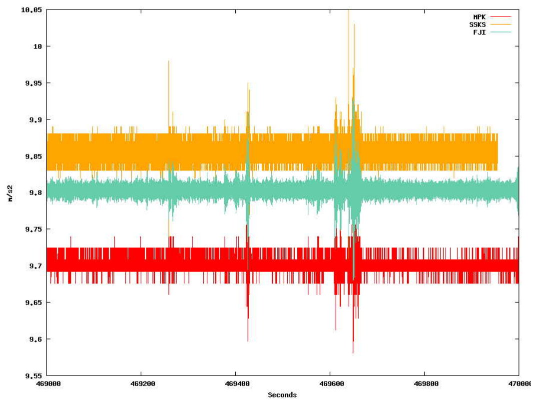

Fig. 6: Static real acquisition for different IMUs.

Once the necessary tools were developed and the reference data were obtained, various simulations were done characterizing each of the previous IMUs, where the simulated signals correspond to a static acquisitions in the Institute's lab.

Hereafter, a summary of the results of signal analysis for real and simulated sensors were presented.

- Statistic analysis

  For the simulated IMUs the mean and the standard deviation of these series are more or less of the same order, so we could accept those values and keep on with the analysis.

- PSD analysis

  For the simulated navigation grade and the simulated low-cost grade IMUs the PSD of the real sensors shown various peaks at hight frequencies that seem to correspond to vibrations of the building at the moment of the acquisition. Except for those peaks, that do not appear in the PSD of the simulated data, both PSDs are quite similar, so the signals were not rejected and the next step of the analysis was done.

  The PSD of the LN-200 signal shows a quite irregular behaviour for different frequencies that the simulated data do not reproduce, this is because of the special errors that characterize this unit. The LN-200 simulated data were not validated and need to be analysed deeper.

- AV analysis

  The Allan variance of the angular rate sensors for the navigation and Low-Cost IMUs shown a behaviour quite similar, so the angular rate sensors were validated. As far as accelerometers is concerned, the hight frequencies of the Allan Variance shown discrepancis that seem to be due to the modelling of the Gauss-Markov errors. In that case a fine tuning of the Gauss-Markov stochastic errors should be done. The simulation of the IMAR FJI and MPK-II was not validated for this reason, but the simulation of typical navigation and Low-Cost IMU was, because of the behaviour of other known IMUs.

According to these results we consider that the simulator is delivering data quite close to the real one.

5 Applications

An IMU simulator can be used for several applications. We illustrate them by describing some of the projects where the simulator has already been used.

- **Calibration of IMUs.**

  One research line at the IG is to develop methods for robust IMU calibration. Among others, this method includes signal analysis techniques. In order to distinguish the motion signal from the noise while analysing different IMU acquisitions, a comparative analysis with a reference IMU is advised. In this sense, specially for static acquisitions, an IMU simulated signal could be used as a reference one.

  This research is funded by the GENIA project, whose aim is the implementation of a physical prototype of the "inertial GNSS antenna" concept with low-cost inertial systems.

- **Validation of new mathematical models involving inertial technology.**

  When a new method involving inertial technology is developed, the number of elements that affects the results is so large that, in case of unfavourable results, it might be impossible to determine if the problem is the method itself or the elements involved in it. In order to decrease the "unknowns" related to inertial measurements, an appropriate way to start the testing is to use a well-known signal, which means, a signal where the dynamics and errors are known. The IG is using the simulator to do a first validation in the following projects with acceptable results:

  - **Evaluation of different Receiver Architectures for Inertial Aiding and Coasting.** The IG together with DEIMOS Engenharia has developed new hybridization strategies of INS and GNSS systems. These new strategies imply the utilization of the INS navigation solution to determine the satellite's signal carrier Doppler shift for a better tuning of the GNSS receiver PLL / DLL blocks and the use of Galileo and low-medium cost MEMS. In this project, the IMU simulator was used to validate the hybridization software. By means of simulated data first, and real data later, the different integration algorithms were validated and verified.

    This research project was funded by the IADIRA project, a joint proposal of DEIMOS Engenharia and the IG as a 6th Frame Program managed by the Galileo Joint Undertaking. For more information, see [10].

  - **Investigations on different numerical integration methods.** Together with the Department of Applied Mathematics of the University of Barcelona (MAIA-UB), the Institute is working on a new mathematical method to numerically integrate the dynamic model. With this research we expect to implement a method capable of reducing the impact of vibrations and noise on the determination of trajectories with INS systems. For more information on this topic, please see [8]. Here, the simulator is useful to analyse the behaviour of the method. Trajectories computed with different grade simulated IMU data and various integrators (Euler, Predictor-Corrector [9]) are being used to validate it.

    The result of this research will be applied on the uVISION project, where a precise trajectory from an INS/GNSS system on an unmanned airborne helicopter should be obtained. [3])

  - **Validation of new sensor Calibration/Orientation models.** Most of the models to calibrate/orient sensors installed on a moving platform assume that the standard deviation of heading is constant along straight lines trajectories. As it is shown in figure 7 this assumption is not close to reality. The reader can see that, for smoothed solutions (those used in network adjustment for calibration/orientation of sensors), the standard deviation is not constant.

    Another research line of the IG in the network adjustment group of the Institute is to derive new methods to improve the classical photogrammetric control models. Among those methods the IG is developing a new model in which the modelling of the attitude errors take into account the variability of the heading error. In this project the simulator will be useful for testing this new models because the estimated errors could be checked by simulated inertial data.

  - **To validate the use of a specific IMU for specific purposes.** The validation of new methods related to inertial technology, such as airborne gravimetry and pedestrian navigation, could become really expensive. In order to know which is the suitable IMU grade for each purpose there are basically two options, to try all available IMUs (which could be extremely expensive), or to run simulations until the required IMU grade is found. The usefulness of the simulator for this purposes is clearly proved.

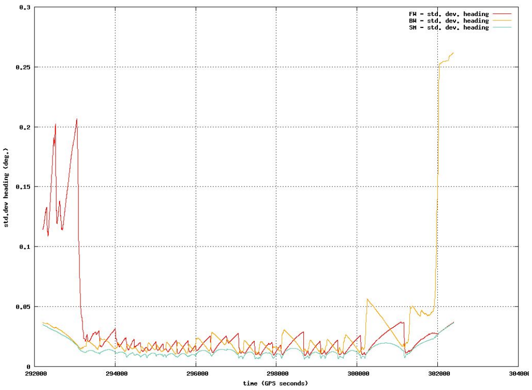

Fig. 7: Standard deviation of the heading along a flight. Values obtained for forward (red), backward (yellow) and smoothed (green) solution.

- **Teaching and training purposes.**

  Last but not least, an inertial simulator is an excellent tool for teaching and training new users of INS technology. It makes it possible to understand which is the expected signal for an IMU and also to visualise the different stochastic errors that can perturb the signal of one of those systems.

  The simulator has been used as a support tool in the International Executive M. Sc. in Airborne Photogrammetry and Remote Sensing of the IG [4].

6 Further developments

In order to improve the system, some new utilities have been considered and will be applied shortly. From those, the most remarkable are:

- To include the error models that characterise popular IMUs, which means the user will be provided with pre-defined templates that characterise the most popular IMUs (LTN-101, LN-200, DMARS-I, SSKS, Motion Pack-II, etc.)
- To add new types of errors such as temperature dependence, non-centralised measurements or vibration effects [1].
- To include simulations for skew redundant IMUs (SRIMUs) [2].

7 Conclusions

In this paper the simulator has been presented, the motivation to implement it, the theoretical basis and some of its applications. It has been demonstrated that an IMU simulator is a useful tool for research and development of INS and INS related applications.

Acknowledgements

The research reported on this paper has been partially funded by the Spanish Ministry of Education and Science, through the GENIA project of the Spanish National Space Research Programme (reference: ESP2005-07599).

The authors sincerely thank Assumpció Térmens, Mariano Wis and the IADIRA team at DEIMOS Engenharia for their advice and their feedback on the system.

References

[1] Bhatti, U.I., Ochieng, W.Y., *"Failure Modes and Models for Integrated GPS/INS Systems"*, The Journal of Navigation, p 327-348, 2007.

[2] Colomina, I. et al., *"On the Use of Redundant Inertial Data for Geodetic Applications"*, Proceeding of the ION GNSS - 2004, ION, Long Beach, California, 2004.

[3] Colomina, I. et al., *"The uVISION project for helicopter-UAV photogrammetry and remote-sensing"*, Proceedings of the 7th Geomatic week, Barcelona, 2007.

[4] Colomina, I., Parés, M.E., *"Sensor Orientation (2): precise trajectory & attitude determination with INS"*, Lecture Notes International Executive M. Sc. in Airborne Photogrammetry and Remote Sensing, 2007.

[5] Hou, H., *"Modeling Inertial Sensors Errors Using Allan Variance"*, PhD Thesis, University of Calgary, 2004.

[6] IEEE, *"IEEE Recommended Practice for Inertial Sensor Test Equipment, Instrumentation, Data Acquisition and Analysis"*, IEEE std 1554 - 2005, 2005.

[7] Jekeli, C., *"Inertial Navigation Systems with geodetic applications"*, Walter de Gruyter, 2001.

[8] Molina, P, *"Different approaches to the numerical solution of the Inertial Navigation mechanization equations: a comparative analysis"*, Practicum Report, Universitat de Barcelona, 2007.

[9] Rosales, J.J., Colomina, I., *"A flexible approach for the numerical solution of the INS mechanization equations "*, Proceedings of the 6th Geomatic week, Barcelona, 2005.

[10] Silva, P.F. et al, *"Evaluating Receiver Architectures for Inertial Aiding and Coasting"*, Proceeding of the ION GNSS - 2007, ION, Fort Worth, Texas, 2007.

[11] Titterton, D.H., Weston, J.L., *"Strapdown inertial navigation technology"*, Peter Peregrinus Ltd., on behalf of the Institution of Electronical Engineers, London, 1997.