Implementation Of Robust Estimation Algorithms In The GALILEO Baseline Integrity Check

Philippe PAIMBLANC, *ENAC,*

Christophe MACABIAU, *ENAC,*

Bruno LOBERT, *Alcatel Space,*

Mathias VAN DEN BOSSCHE, *Alcatel Space*

Stephane LANNELONGUE, *Alcatel Space*

BIOGRAPHY

Philippe Paimblanc graduated as an electronics engineer from the ENAC (Ecole Nationale de l'Aviation Civile) in 2002 and received the same year his Master research degree in signal processing. He is now a Ph.D student at the satellite navigation lab of the ENAC. Currently he carries out research on the Galileo Integrity concept in collaboration with ALCATEL SPACE, in Toulouse, France.

Christophe Macabiau graduated as an electronics engineer in 1992 from the ENAC (Ecole Nationale de l'Aviation Civile) 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.

Bruno Lobert graduated from IDN (Lille, France) and IRR (Toulouse, France). He is currently responsible of the Galileo Performance team at Alcatel Space.

Mathias van den Bossche graduated with a Magistère in Physics at the École Normale Supérieure (Paris), and got PhD in Theoretical Physics at the Paris University. He turned to Space industry in 2001, and joined the Galileo team at Alcatel Space in 2003.

Stephane Lannelongue received in 1996 a M. Sc. Engineer degree in electronics and signal processing from the Ecole Nationale Supérieure d'Ingénieurs Electriciens de Grenoble (ENSIEG/France). From 1997 to 2000, he worked in the field of satellite navigation at the European Space Agency (ESA/ESTEC). He joined Alcatel Space Industries in 2000 as a system engineer on EGNOS and Galileo.

ABSTRACT

The European satellite navigation system GALILEO will provide radionavigation signals for a variety of applications. Safety Of Life users will get a safe navigation service through ranging signals carrying integrity information.

The Galileo Integrity Baseline algorithm includes the transmission of three parameters allowing users to monitor their integrity level. These parameters are the Signal-In-Space Accuracy (SISA: prediction of the minimum standard deviation of a Gaussian distribution overbounding the Signal-In-Space error in the fault-free case), the Signal-In-Space Monitoring Accuracy (SISMA: minimum standard deviation of a Gaussian distribution overbounding the difference between Signal-In-Space error and its estimation by ground control stations) and the Integrity Flag, which accounts for satellite status (it can be set to "OK", "DON'T USE" or "NOT MONITORED").

The work presented in this paper studies the possibility of computing SISMA using a statistically robust algorithm, so as to reject wrong measurements and decrease ground system False Alarm rate (fault-free satellites flagged "DON'T USE").

INTRODUCTION

This article presents the first results of the use of robust regression algorithms in the Integrity concept of GALILEO. Indeed, it consists in providing users with information concerning the system contribution to the final user position error, in order to allow them to autonomously check their integrity level. That information provided to the users is in fact a quantification of the quality of the SIS (Signal In Space). The SIS is the signal emitted by a satellite in the constellation as received by a fault-free receiver. In the present Integrity Check algorithm, propagation errors are not considered, only the contribution of the satellite payload will be included in the broadcasted information.

Indeed, the error induced by the ground and space segments on the determination of the user's position and clock bias is assumed to be essentially due to the difference between the satellite true position and clock bias and the values provided by the OD&TS (Orbit Determination and Time Synchronisation) through the broadcast ephemeris data. The projection of these position

and clock errors on the satellite user axis is called the SISE (Signal In Space Error): it is in fact the equivalent measurement error due to OD&TS errors. Thus, the integrity information to be provided by GALILEO to the users only concerns OD&TS errors: propagation errors or local errors such as multipath or jamming are not taken into account in the broadcast integrity parameters.

The integrity information broadcasted in the navigation message is composed of three parameters:

- Signal-In-Space accuracy (SISA): predicted smallest standard deviation of a Gaussian distribution overbounding the SISE distribution in fault-free mode for any user in the service area.
- Signal-In-Space Monitoring Accuracy (SISMA): smallest standard deviation of a Gaussian distribution overbounding the difference between SISE and $SISE_{est}$, its estimation by the ground segment.
- Integrity Flag (IF): depending on the results of the monitoring system, this flag can be set to one of the three following values: "USE", "DON'T USE" or "NOT MONITORED"

The value of the integrity flag for each satellite will be determined by the Integrity monitoring system. Composed of about 40 GALILEO Sensor Stations (GSS) in precisely known positions, it performs an estimation of the difference between the true satellite position and clock bias and the corresponding values provided by the OD&TS. This estimation is carried out by using the L1, E5b and E6 pseudorange measurements made by the GSS. The estimated vector and its covariance matrix are then projected over a user grid, so as to obtain respectively $SISE_{est,u}$ (an estimation of the SISE) and its standard deviation $\sigma_{check,u}$ for each potential user u. The WUL (Worst User Location) is determined by looking for the maximum of $\sigma_{check,u}$. The SISMA is the maximum value of $\sigma_{check,u}$. The corresponding $SISE_{est,u}$ for this worst user location is the chosen estimate of SISE and is called $SISE_{est}$. The Integrity Flag is raised when the $SISE_{est}$ has a magnitude which cannot be due to the normal ephemeris error nor to the noise on the GSS measurements.

The detailed way to determine SISMA and $SISE_{est}$ and the algorithm which computes the Integrity Flag based on $SISE_{est}$ and SISMA are described in the Integrity Check section.

The use, in the process of computing SISMA and $SISE_{est}$, of a least-squares algorithm implies that all input residuals are considered fault-free and Gaussian. In the case of an undetected malfunction (concerning either a satellite or a GSS), biased measurements may be included, thus leading to the overestimation of $SISE_{est}$ and thus possibly causing a fault-free satellite to be flagged "DON'T USE".

Therefore this paper studies the possibility of replacing the Least-Squares by a robust estimation algorithm in the Integrity check performed by Galileo ground segment in the baseline. The term "robust" refers here to the ability of such algorithms to estimate correct statistical parameters in the presence of some corrupted samples: while a single biased sample (called an outlier, a measurement not belonging to the same distribution as the other samples from the set) may lead a non-robust algorithm (e.g. Least-Squares) to provide an estimate far away from the underlying distribution of the sample space, a robust estimator (M-estimator, Least-Trimmed-Squares…) will resist such a small change and be able to provide an estimate close to the true distribution. Typically, one biased measurement in a set of otherwise Gaussian distributed ones will be either discarded (in the case of the LTS, for instance) or weighted down so as not to influence the estimation process (e.g. in the case of the M-estimator). The cost of this ability to detect and discard outliers is twofold: first, the loss of the optimality in the case of a truly Gaussian distributed sample set, and second, an additional computational cost (most robust algorithms are based on iterative residual computation ranking).

The performance analysis of the robust algorithm on the computation of the SISMA is performed in this paper with simulated data (with and without satellite or station failure), taking into account the steps of the pre-processing that impact most on the shape of the input signal (carrier-phase smoothing, ionospheric and tropospheric).

Thus the paper will be structured as follows: the first section describes the input signal simulator, the second section presents the Integrity Check algorithm. Section 3 gives the definition of overbounding used throughout the article. Then sections 4 to 8 introduce the notion of statistical robustness, the methods that were used, how they were optimized and the gains that could be expected from them. Section 9 presents the results obtained through simulation and finally conclusions are derived on the feasibility of introducing a robust algorithm in the Integrity Check.

MODELISATION AND GENERATION OF INPUT RESIDUALS

The pseudorange measurements made by the Galileo Sensor Stations used for the Integrity Check are L1, E5b and E6 code and carrier phase measurements. For the purpose of the integrity check by the Integrity Processing Facility, where the impact of satellite position and clock errors on user range measurements is checked, these measurements are first pre-processed in order to test anomalies and to remove residual errors that may affect these measurements.

The main steps of this pre-processing are the following:

- Detection of cycle slips on L1, E5b and E6 carrier phase measurements, and correction of these cycle slips if possible
- Correction of L1, E5b and E6 tropospheric errors
- Code-carrier smoothing of L1, E5b and E6
- Elaboration of composite range measurements L1-E5b and L1-E6, with weights inducing correction of ionospheric errors
- Computation of residual composite range measurements by differencing observed composite measurements and range measurement prediction made from the knowledge of GSS positions and GALILEO satellites ephemeris data
- Correction of GSS clock errors and residual tropospheric error

Thus, for any Galileo satellite i, the L1/E5b residuals for any GSS n can be modelled in the following way (the same model applies for the L1-E6 composite range measurements):

$$r^i_{L1,E5b} = \tilde{P}^i_{L1,E5b} - \hat{P}^i_{L1,E5b}$$

where:

- $\tilde{P}^i_{L1,E5b} = \dfrac{\gamma_{L1} P^i_{L1,sm} - \gamma_{E5b} P^i_{E5b,sm}}{\gamma_1}$ is the L1-E5b composite measurement made from smoothed L1 and E5b measurements $P^i_{L1,sm}$ and $P^i_{E5b,sm}$.
- $\hat{P}^i_{L1,E5b} = \sqrt{\left(x^n_r - \hat{x}^i_s\right)^2 + \left(y^n_r - \hat{y}^i_s\right)^2 + \left(z^n_r - \hat{z}^i_s\right)^2} + c\Delta\hat{t}^i_s - c\Delta\hat{t}_r + \hat{\tau}_{tropo}$ is the predicted measurement, where $\left(x_s, y_s, z_s, \Delta t_s\right)$ is the satellite true position and clock bias, and $\left(x^n_r, y^n_r, z^n_r, \Delta t^n_r\right)$ is the GSS true position and clock bias.

Therefore, these residuals are finally modelled as:

$$r^i_{L1,E5b} = SISE^i + \varepsilon^i_{cs,L1E5b} + \varepsilon^i_{tropo} + \varepsilon^i_{MP,L1E5b} + \varepsilon^i_{noise,L1E5b}$$

where

$SISE = h_x\delta_x + h_y\delta_y + h_z\delta_z + c\delta_t$ is the SIS Error in the direction of the GSS, with h_x, h_y, h_z the GSS-satellite direction cosines and $\delta_x, \delta_y, \delta_z, \delta_t$ is the satellite ephemeris error.

$\varepsilon_{cs,L1E5b}$ is the transient response of the smoothing filter to carrier cycle slips

ε_{tropo} is the residual tropospheric error

$\varepsilon_{MP,L1E5b}$ is the residual multipath error

$\varepsilon_{MP,L1E5b}$ is the residual

In order to test the performance of the Integrity check for several configurations of algorithms, a program was implemented that generates residuals with the same characteristics as residuals obtained by preprocessing measurements from GSS stations.

The goal here is to generate simulated residuals that would be representative of the final Galileo system, without oversimplifying the behaviour of the measurements. Each error component is therefore generated with reasonable characteristics. Two types of components need to be distinguished: the SISE originating from ephemeris errors, and the other components originating from measurement errors.

The SISE is generated quite simply by assuming reasonable ephemeris errors. It is generated from the model recalled above: $SISE = h_x\delta_x + h_y\delta_y + h_z\delta_z + c\delta_t$. The ephemeris errors $\delta_x, \delta_y, \delta_z, \delta_t$ are simulated as 1st order Markov processes such that the satellite radial error has a standard deviation of 1.34 m, the satellite tangential plane component has a standard deviation of 2.45 m, the satellite clock error has a standard deviation of 2 m.

Residual components originating from measurements errors are not stationary, and their characteristics do change over time. More precisely, two main effects must be taken into account: the dependency of the measurement errors over the satellite elevation, and the error reduction due to the smoothing filter. Therefore, the input parameter is a targeted standard deviation of the sum of these components over the elevation angle denoted $\sigma_{target}(\theta)$, and these components are generated sequentially, then they are affected by the smoothing filter.

As a summary, the technique used to generated these components is recalled below:

ε_{cs} is first simulated by inserting random cycle slips at random epochs following a Poisson distribution with mean time between occurrences of 10^{-8}. Then, these cycle slips go through the smoothing filter.

ε_{tropo} is generated as the product of a zenith residual bias multiplied by a mapping function depending on the elevation angle.

ε_{MP} is generated as the multipath error induced by two reflected rays in addition to the line of sight signal. As these multipath rays do not vary fast for a ground station, this error is solved as the solution ε_τ to the traditional set of equations originating from the PLL and DLL discriminators model:

$$\begin{cases} \tan(\varepsilon_\theta) = \dfrac{\sum_{i=1}^{2} \alpha_i K_c(\Delta\tau_i + \varepsilon_\tau)\sin(\Delta\theta_i)}{\sum_{i=0}^{2} \alpha_i K_c(\Delta\tau_i + \varepsilon_\tau)\cos(\Delta\theta_i)} \\ \left(I_E^2 + Q_E^2\right) - \left(I_L^2 + Q_L^2\right) = 0 \end{cases}$$

where

$$I_E = \sum_{i=0}^{2} \alpha_i K_c(\Delta\tau_i + \varepsilon_\tau + C_s/2)\cos(\varepsilon_\theta + \Delta\theta_i)$$

$$Q_E = \sum_{i=0}^{2} \alpha_i K_c(\Delta\tau_i + \varepsilon_\tau + C_s/2)\sin(\varepsilon_\theta + \Delta\theta_i)$$

$$I_L = \sum_{i=0}^{2} \alpha_i K_c(\Delta\tau_i + \varepsilon_\tau - C_s/2)\cos(\varepsilon_\theta + \Delta\theta_i)$$

$$Q_L = \sum_{i=0}^{2} \alpha_i K_c(\Delta\tau_i + \varepsilon_\tau - C_s/2)\sin(\varepsilon_\theta + \Delta\theta_i)$$

and

ε_τ, ε_θ are the code and phase tracking errors

α_i, $\Delta\tau_i$, $\Delta\theta_i$ are the relative amplitude, relative delay and relative phase of each reflected ray (assuming $\alpha_0 = 1$, $\Delta\tau_0 = 0$, $\Delta\theta_0 = 0$).

C_s is the DLL chip spacing

K_c is the PRN code correlation function

ε_{noise} is the residual error due to noise. At the smoothing filter input, this error is generated as a Gaussian white noise with standard deviation $\sigma_{target}(\theta) \times 0.5 \times \sqrt{2T_{smooth}}$ where T_{smooth} is the smoothing filter time constant.

INTEGRITY CHECK ALGORITM

At a given instant, each satellite *i* is seen by *N* GSS, which all perform pseudorange measurements based on satellite i signal. Let $(x_s, y_s, z_s, \Delta t_s)$ be the satellite true position (in the WGS-84 referential) and clock bias: these 4 values are the unknowns of the problem. Let (x_r^n, y_r^n, z_r^n) be the coordinates of the GSS of index *n*. The GSS coordinates are precisely known. The GSS clock biases will be determined through common view techniques, using a specific GSS as a reference station connected to the Galileo Time. Therefore, the relation between the residuals for GSS *n* and the coordinates of satellite *i* can be expressed as follows:

$$r^i = H \cdot \Delta X + E_i$$

where H is a (3x*N*) matrix, defined as follows

$$H = \begin{pmatrix} \dfrac{\hat{x}_s - x_r^1}{\rho^1} & \dfrac{\hat{y}_s - y_r^1}{\rho^1} & \dfrac{\hat{z}_s - z_r^1}{\rho^1} \\ \vdots & \vdots & \vdots \\ \dfrac{\hat{x}_s - x_r^N}{\rho^N} & \dfrac{\hat{y}_s - y_r^N}{\rho^N} & \dfrac{\hat{z}_s - z_r^N}{\rho^N} \end{pmatrix}$$ and ΔX is the satellite ephemeris error. Indeed, in the present case, a 3-parameter model is used rather than the usual 4-parameter one. This can be justified by projecting the measurement equation in the local referential of the GSS (north east down), in which the down component of the observation matrix is almost equal to 1. Thus, it is the sum of the down and clock parameters that are estimated. This causes no problem since ΔX is estimated only to be projected on the Worst User Location axis.

Let $\Delta\hat{X}$ be the estimation of ΔX and $\mathrm{cov}(\Delta\hat{X})$ its covariance matrix. To obtain an estimation of SISE, the current method consists in forming a user grid on the surface of the Earth. $\Delta\hat{X}$ and $\mathrm{cov}(\Delta\hat{X})$ are projected on each satellite-user axis. Let $X_0 = (x_0, y_0, z_0)$ be the OD&TS coordinates of satellite *i* and $X_g = (x_g, y_g, z_g)$ the coordinates of a user on the grid.

The relation between the pseudorange measurement performed by the ground user, user position and satellite position is the same as the relation between the pseudorange measurement performed by the GSS, GSS position and satellite position. Therefore, the projection vector h_u for the $(X_0 X_g)$ axis is obtained by replacing the coordinates of the GSS by the user's in a line vector of the observation matrix *H*:

$$h_u = \begin{pmatrix} \dfrac{x_0 - x_g}{\sqrt{(x_0 - x_g)^2 + (y_0 - y_g)^2 + (z_0 - z_g)^2}} \\ \dfrac{y_0 - y_g}{\sqrt{(x_0 - x_g)^2 + (y_0 - y_g)^2 + (z_0 - z_g)^2}} \\ \dfrac{z_0 - z_g}{\sqrt{(x_0 - x_g)^2 + (y_0 - y_g)^2 + (z_0 - z_g)^2}} \end{pmatrix}$$

Thus, for a given user *u*, the estimations of the SISE and its standard deviation are:

$$SISE_{est,u} = h_u^t \cdot \Delta\hat{X}$$

$$\sigma_{check,u} = h_u^t \cdot \mathrm{cov}(\Delta\hat{X}) \cdot h_u = h_u^t \cdot \left[H^t \cdot R^{-1} \cdot H\right] \cdot h_u$$

where $R = \mathrm{cov}(E_i)$

The maximum values of $SISE_{est,u}$ and $\sigma_{check,u}$ are called $SISE_{est}$ and SISMA:

$$SISE_{est} = \max_{u}\left(SISE_{est,u}\right)$$
$$SISMA = \max_{u}\left(\sigma_{chack,u}\right)$$

It appears that the standard deviation of $SISE_{est}$ depends on the observation time-span. Indeed, if $SISE_{est}$ is observed for a short time, SISE (projection of ΔX) will be considered a constant. $SISE_{est}$ will therefore follow a normal distribution centred on SISE, with a standard deviation equal to SISMA. If, on the other hand, $SISE_{est}$ is observed for a long time, the variations of SISE will have to be taken into account: $SISE_{est}$ will follow a centred normal distribution, with standard deviation equal to $\sqrt{SISA^2 + SISMA^2}$.

The integrity flag is raised when the observed $SISE_{est}$ departs significantly from this statistics. The decision threshold is tuned using the specified false alarm rate, so the integrity flag is raised when

$$\left|SISE_{est}\right| > k_{FA} \times \sqrt{SISA^2 + SISMA^2}$$

where k_{FA} =4.34 as per the design false alarm rate 1.5 x 10^{-5} per independent sample.

Thus, $SISE_{est}$ is computed from the residuals provided by the pre-processing algorithms, while SISMA is only computed from assumptions on the standard deviation of the measurement noise and geometrical data. If one of the input residuals does not respect the assumption on its standard deviation (because of a propagation problem or a reception problem in the vicinity of the GSS), then there is risk that $\left|SISE_{est}\right|$ will exceed the decision threshold despite the fact that the satellite is not malfunctioning. In this case, a robust algorithm, able to reject input data that do not respect the assumptions made on them, might help in not flagging unduly a satellite.

OVERBOUNDING

The SISMA is defined as the smallest standard deviation of a Gaussian distribution ***overbounding*** $SISE - SISE_{est}$, the difference between true and estimated pseudorange measurement error. The practical definition of overbounding is the following:

Let f be an experimental distribution and g a normal, centred distribution. g overbounds f means:

$$\int_{-\infty}^{-L} g(x)dx + \int_{L}^{+\infty} g(x)dx \geq \int_{-\infty}^{-L} f(x)dx + \int_{L}^{+\infty} f(x)dx, \forall L \geq 0$$

In general a given distribution *cannot be overbounded* e.g. Dirac distribution $\delta(x-b_{xp})$ corresponding to non-scattered measurements with bias b_{xp}

However one can show that a biased normal distribution $f \sim N(b_{xp},\sigma_{xp})$ $\sigma_{xp} < \infty$ with can be bounded by a centred normal distribution $g \sim N(0,\sigma_{ob})$ with:

$$\sigma_{ob} \geq \sigma_{xp} \exp\left(\frac{b_{xp}^2}{2\sigma_{xp}^2}\right)$$

This can be graphically assessed by plotting the tail weights of both the experimental and objective distributions. For a given distribution f, the tail weights are defined as:

$$P\left[\left|error\right| > L\right] = \int_{-\infty}^{-L} f(x)dx + \int_{L}^{+\infty} f(x)dx, \forall L \geq 0$$

The overbounding of f by g is achieved if the tail weights of g are superior to those of f for any L.

This definition is suited to distributions: in order to test whether a centred normal distribution of standard deviation $SISMA$ overbounds $SISE - SISE_{est}$, it is required to have a statistically significant number of $SISE_{est}$ measurements corresponding to the same value of $SISMA$. But in the present case, each value of $SISMA$ (for a particular satellite at a particular epoch) corresponds to only one value of $SISE_{est}$. Thus, in this paper, the test that will be used to compare the overbounding performances of the Least-Squares and robust methods consists in computing the standardized random variable $\left(SISE_{est} - SISE\right)/SISMA$ for all satellites at all epochs and dividing it by the corresponding SISMA, so as to obtain a normalized random variable. Data coming from all satellites are then considered as a single normalized random variable whose tail weights are compared to that of a standard Gaussian distribution.

It must be added that the algorithms used for this paper (robust or not) do not include an additional overbounding procedure (such as inflating the SISMA by a given percentage), the overbounding performance being ensured by the fact that the predicted standard deviations of the measurement noise are actually superior bounds.

PRINCIPLE OF ROBUST ALGORITHMS

The aim of robust statistics is to construct estimation and regression algorithms able to provide reliable results when all the assumptions made on the observation data are not met in full. Indeed, it is generally assumed that all variables are normally distributed: it is the case for which the classical Least-Squares algorithm is optimal. When the underlying Gaussian model does not hold for every sample, for instance when a feared event causes one pseudorange measurement to deviate (such a measurement is called an ***outlier***), the results provided by the Least-Squares may actually be far away from the true data distribution. The whole point of robustness can be expressed in terms of continuity of the estimator: a small variation in the sample space (either a small change on the whole sample space, or great change on a small fraction of

the sample space) must bring only small variations to the estimated distribution.

In order to underline the differences between Least-Squares and robust algorithms, we will use the example of linear regression (which is the type of problem solved to obtain a position with satellite positioning systems):

$y_i = \beta_0 + \beta_1 x_{i1} + \ldots + \beta_p x_{ip} + e_i$ for $i = 1, \ldots, N$,

where y_i is the response variable (e.g. a pseudorange measurement), $x_{i1}, \ldots, x_{ip}$ are the regressors (e.g. elements of the user-satellite direction vector) and e_i is a zero-mean normal noise with σ standard deviation. The aim is to obtain $(\hat{\beta}_0, \ldots, \hat{\beta}_p)$, estimate of the set $(\beta_0, \ldots, \beta_p)$ of regression coefficients (which are user coordinates and clock bias in the case of satellite positioning). The regression residuals may be expressed as follows:

$r_i = y_i - (\hat{\beta}_0 + \hat{\beta}_1 x_{1i} + \ldots + \hat{\beta}_p x_{ip})$.

The estimate $(\hat{\beta}_0, \ldots, \hat{\beta}_p)$ computed by LS is the one which minimizes the sum of squared residuals:

$$\min_{(\hat{\beta}_0, \ldots, \hat{\beta}_p)} \sum_{i=1}^{N} r_i^2$$

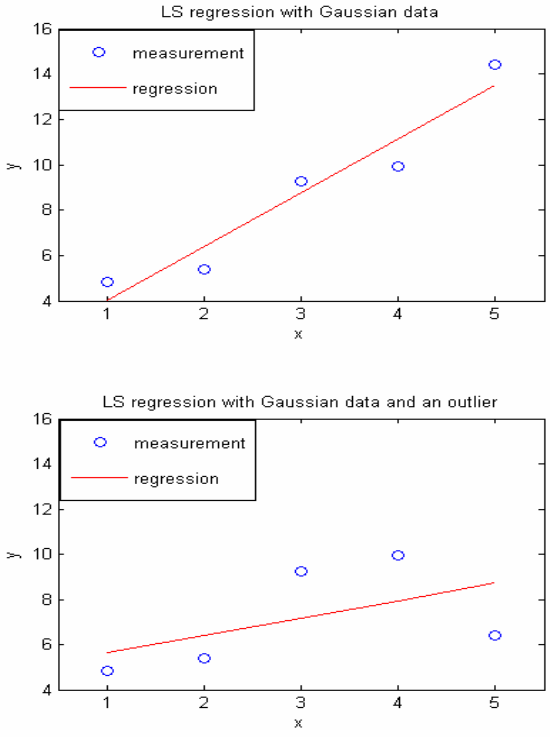


Figure 1: Example of the influence of a single outlier

The LS criterion brings optimal results when e_i is Gaussian. But there is no guarantee of reliability of the algorithm's results when e_i is not Gaussian (which is generally the case). Indeed, figure 1 illustrates the fact that one single outlying sample may cause a two-dimensional LS regression to break down.

This example may be used to introduce the notions of *breakdown value* and *breakdown bound*, as described by Huber in [1]: the breakdown value is the smallest fraction of contamination in the sample space that can cause the regression method to run arbitrarily far from the distribution of the majority of the samples. For instance, it can be seen from the preceding figure that the breakdown value of the LS algorithm is $1/N$. The breakdown bound is the limiting value (for $N \to \infty$) of the breakdown value. It is thus equal to 0 for the LS. Estimators with a breakdown bound strictly superior to 0 are called positive-breakdown methods (as will be seen in the following, robust methods do not systematically have positive breakdown bounds).

M-ESTIMATOR

In order to avoid the situation illustrated in the preceding figure, robust algorithms derive their estimates from a different minimization criterion. The type of criterion used defines the type of algorithm: in the case of M-estimation (or maximum likelihood estimation), the criterion is the following:

$$\min_{(\hat{\beta}_0, \ldots, \hat{\beta}_p)} \sum_{i=1}^{N} \rho\left(\frac{r_i}{\sigma}\right)$$

where ρ is any symmetric positive-definite function with a unique minimum at zero, r_i are the estimation residuals and σ their scale factor. The M-estimator of the $(\beta_0, \ldots, \beta_p)$ set is thus the solution of the p equations:

$$\sum_{i=1}^{N} \psi\left(\frac{r_i}{\sigma}\right) \frac{\partial r_i}{\partial \beta_j} = 0 \qquad \text{for } j = 1, \ldots, p,$$

where $\psi(x) = d\rho(x)/dx$ and is called the influence function. ψ describes the influence of one given sample on the computed estimate. In the LS case, where $\rho(x) = x^2$, the influence function is $\psi(x) = 2x$, which means that the influence of a sample grows linearly with the size of its error: an outlier will thus have more influence on the regression outcome than samples belonging to the regular underlying distribution, which explains the non-robustness illustrated in figure 1. Let us then define a weight function: $w(x) = \psi(x)/x$. The equation system to solve then becomes:

$$\sum_{i=1}^{N} w\left(\frac{r_i}{\sigma}\right) r_i \frac{\partial r_i}{\partial \beta_j} = 0 \quad \text{for } j = 1, \ldots, p,$$

which shows that M-estimation simply comes down to solving the following iterated reweighted Least-Squares problem:

$$\min \sum_{i=1}^{N} w\left(r_i^{(k-1)}\right) r_i^2$$

Thus, the efficiency of the M-estimator relies on the fact that its ρ function is chosen so that samples have a bounded influence.

Several ρ functions have been tested. The first one (a popular choice) is the Huber function:

$$\begin{cases} \rho(x) = \dfrac{x^2}{2} & if\ |x| \le k \\ \rho(x) = k \cdot |x| - \dfrac{k^2}{2} & if\ |x| \ge k \end{cases}$$

which is a parabola in the vicinity of 0 (the same as the LS) then grows linearly. k is a tuning variable which defines the value of the residual error beyond which the corresponding sample will be deemed an outlier. The influence function of Huber's M-estimator :

$$\begin{cases} \psi(x) = x & if\ |x| \le k \\ \psi(x) = k & if\ |x| \ge k \end{cases}$$

is thus bounded by k . The corresponding weight function is thus:

$$\begin{cases} w(x) = 1 & if\ |x| \le k \\ w(x) = \dfrac{k}{x} & if\ |x| \ge k \end{cases}$$

An outlying measurement, raising a large error residual, will therefore be weighted down. But its influence will never be completely set to zero. In order to Thus a different function was tried, namely the Tukey function, which is expressed as follows.

ρ function:

$$\begin{cases} \rho(x) = \dfrac{k^2}{6}\left(1 - \left[1 - \left(\dfrac{x}{k}\right)^2\right]^3\right) & if\ |x| \le k \\ \rho(x) = \dfrac{k^2}{6} & if\ |x| \ge k \end{cases}$$

Influence function:

$$\begin{cases} \psi(x) = x\left(1 - \left[1 - \left(\dfrac{x}{k}\right)^2\right]^2\right) & if\ |x| \le k \\ \psi(x) = 0 & if\ |x| \ge k \end{cases}$$

Weight function:

$$\begin{cases} w(x) = \left(1 - \left[1 - \left(\dfrac{x}{k}\right)^2\right]^2\right) & if\ |x| \le k \\ w(x) = 0 & if\ |x| \ge k \end{cases}$$

The Tukey function actually gave poor results as is shown and explained in the results section. This lead to the implementation of a custom-made triple weight function, designed to combine the qualities of both Huber and Tukey functions. It is expressed as follows:

ρ function:

$$\begin{cases} \rho(x) = \dfrac{x^2}{2} & if\ |x| \le k_1 \\ \rho(x) = \dfrac{k_2^6}{6\left(k_2^2 - k_1^2\right)^2} \cdot \left(1 - \left[1 - \left(\dfrac{x}{k_2}\right)^2\right]^3\right) & if\ k_1 < |x| \le k_2 \\ \rho(x) = \dfrac{k_2^6}{6\left(k_2^2 - k_1^2\right)^2} & if\ |x| > k_2 \end{cases}$$

Influence function:

$$\begin{cases} \psi(x) = x & if\ |x| \le k_1 \\ \psi(x) = \dfrac{k_2^2}{\left(k_2^2 - k_1^2\right)^2} \cdot x \cdot \left(1 - \left(\dfrac{x}{k_2}\right)^2\right)^2 & if\ k_1 < |x| \le k_2 \\ \psi(x) = 0 & if\ |x| > k_2 \end{cases}$$

Weight function:

$$\begin{cases} w(x) = 1 & if\ |x| \le k_1 \\ w(x) = \dfrac{k_2^2}{\left(k_2^2 - k_1^2\right)^2} \cdot \left(1 - \left(\dfrac{x}{k_2}\right)^2\right)^2 & if\ k_1 < |x| \le k_2 \\ w(x) = 0 & if\ |x| > k_2 \end{cases}$$

Figure 2 compares ρ, the influence function and the weight function for the LS, the Huber, Tukey and custom functions. It is clear from this figure that the Tukey and custom functions are not convex: in order to ensure convergence of the solution, the M-estimators based on these functions are initialized with a vector resulting from several iterations of Huber's M-estimator.

It must be noted that M-estimators are not positive breakdown estimators. However, their combination of robustness and small additional computational cost has led us to use them as a reference.

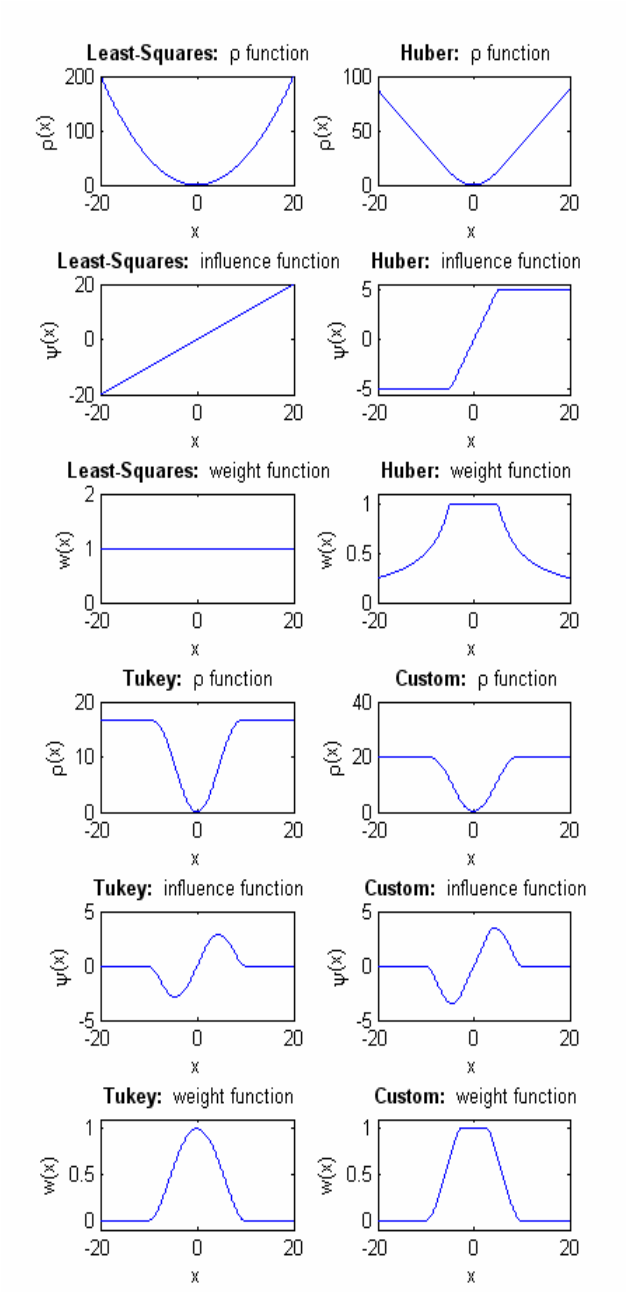


Figure 2: Comparison of ρ, influence and weight functions

LEAST-TRIMMED-SQUARES AND MINIMUM COVARIANCE DETERMINANT REGRESSION

Two positive-breakdown methods were also tested, both consisting in selecting a subset of samples. Indeed, the Least Trimmed Squares (LTS) criterion is the following:

$$\min_{(\hat{\beta}_0,\ldots,\hat{\beta}_p)} \sum_{i=1}^{h} \left(r^2\right)_{i:N}$$

where $\left(r^2\right)_{1:N} \le \left(r^2\right)_{2:N} \le \ldots \le \left(r^2\right)_{N:N}$ are the ordered squared residuals. Practically, the method consists in forming all sample subsets of cardinal h (out of N available samples) and then computing the Least-Squares solution on the subset. The solution subset is the one for which the sum of squared error residuals is minimum.

Directly outputting the solution computed with the subset would mean systematically considering one of the measurements to be biased. Such an assumption is unrealistic and statistically inefficient. Therefore, a reweighing of the samples is performed. It consists in computing a robust (according to the LTS criterion) estimate of the residuals:

$$\hat{\sigma}_{r,h} = \sqrt{\sum_{i=1}^{h} \frac{y_{red,i}^2}{h-1} + \frac{1}{N-1} \cdot \hat{x}_{LTS}^t \cdot H^t \cdot H \cdot \hat{x}_{LTS}}$$

where:

$y_{red,i}$ is the i^{th} element of y_{red}, the vector composed of the h measurements selected by the LTS,

$\hat{x}_{LTS}$ is the state vector estimated from the subset,

H is the complete design matrix.

Let r be the $(N \times 1)$ standardized residual vector, computed as:

$$r = \frac{1}{\hat{\sigma}_{r,h}} \left(y - H \cdot \hat{x}_{LTS}\right)$$

Each element of r is then squared and compared to a threshold computed as :

$$T = \left(\chi_1^2\right)^{-1} \left(P_{threshold}\right)$$

where

$\left(\chi_1^2\right)^{-1}$ is the inverse of the chi-square function with one free variable

$P_{threshold}$ is a probability (typically comprised between 0.9 and 1) used to adapt the algorithm to user's needs.

The solution is obtained by applying the Least-Squares to the measurements corresponding to the elements of r^2 that are smaller than T.

The Minimum Covariance Determinant is a high-breakdown estimation algorithm: it determines the mean and covariance matrix of a collection of vectors by computing the sample mean and sample covariance of the subset of h vectors out of N for which the determinant of the covariance matrix is minimum. Its application to linear regression consists in applying the algorithm to the measurement vector y in order to select a subset of h elements, on which the least-squares is applied to obtain an estimate of the state vector and its covariance matrix.

Thus in practice, the standard deviation of each measurement subset of length h (set to $N-1$ or $N-2$) is computed. The subset with smallest standard deviation is used to compute a first estimate. Then, the same additional reweighing as for the LTS is performed to achieve better statistical efficiency.

The breakdown bound of LTS and MCD is then a function of h: for instance, if $h = 0.75 \times N$, the breakdown bound is 25%; for $h = N/2$, it reaches 50%. But such values of h are adapted for statistical problems where samples are very abundant (in such cases, it is not feasible to compute all subsets: Huber and Rousseeuw thus proposed in [Huber-Rousseeuw] a method to converge towards the optimal subset without having to compute all possible solutions). The value of h was thus set to the expected maximum proportion of outlying measurements in sample sets: $N-1$ or $N-2$.

OPTIMISATION OF ROBUST METHODS

All the robust methods presented above have a parameter enabling users to tune them to suit the characteristics of the experimental distribution (k or k_1 and k_2 for the M-estimators, $P_{threshold}$ for the LTS and MCD). The values that are assigned to these parameters set the percentage of the experimental data distribution that will kept by the robust procedure. In order to find the values best suiting the needs of the Integrity flag algorithm, a quick optimisation procedure was set up. The key aspects of the robust procedures are their ability to reject outlying pseudorange measurements and the fact that the $SISMA$ they provide is at best equal to (but often larger than) the Least-Squares $SISMA$. If the tuning parameter of a given method is small, the algorithm will be very selective: outlying samples will be detected more easily, but regular samples belonging to the tails of the experimental data distribution will also be rejected more often. The resulting $SISMA$ will be higher (since computed with fewer samples). Thus a series of short simulations (performed on three hours of data each) was conducted: for each method and each tuning parameter value, two simulations were performed, one with fault-free measurements, the other with a ten-meter bias on one of the measurements for each of the satellites so as to obtain, for each method, the value of the maximum $SISMA$ with fault-free measurements and the value of the detection rate with one outlying measurement, as a function of the value of the parameter. The aim was to find, if possible, a trade-off between the ability to reject outliers and the ability to keep as many regular samples as possible. Figure 3 displays simulation results for the Huber M-estimator. It appears that the maximum $SISMA$ significantly drops between $k = 0.5$ and $k = 1.25$, then stays nearly constant. On the other hand, the detection rate increases slowly for $k \leq 1.5$, then rapidly grows to reach the same value as the Least-Squares. Thus k was set to **1.25.**

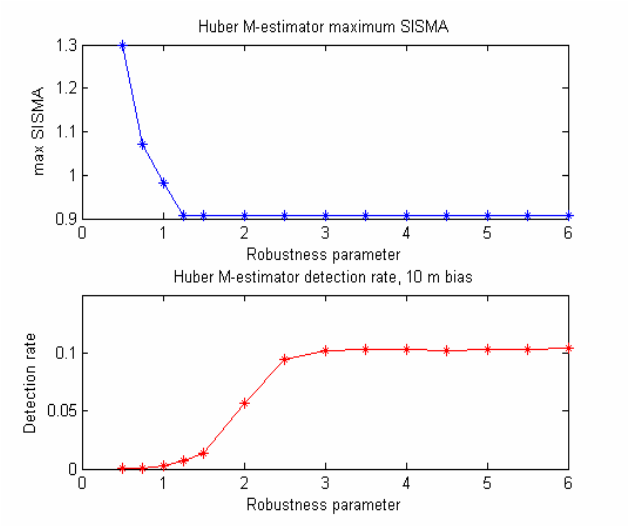


Figure 3: Results of Huber M-estimator optimisation

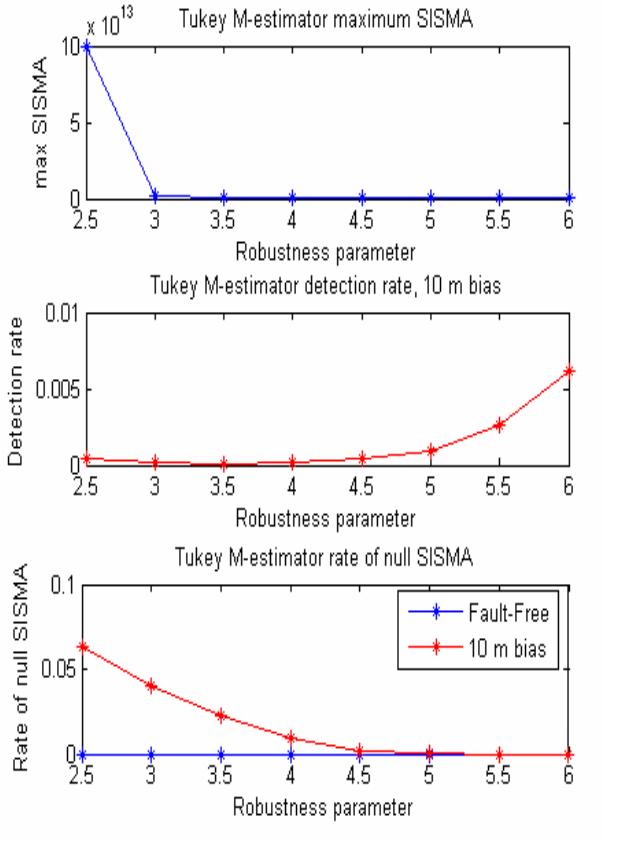


Figure 4: Results of Tukey M-estimator optimisation

Figure 4 represents the same type of data for the Tukey M-estimator, with the addition of the so-called "rate of null $SISMA$". Indeed, when the k parameter is too small, the weight function is over selective and rejects all or all but one measurement, leading to $SISMA = 0$. Additionally, if few measurements are kept, the reweighting of the covariance matrix by the inverse of the residual scale can lead to huge values of SISMA (see

figure 4). Thus, the k parameter was set to 6. However, this potential instability of the algorithm was what lead to the implementation of the Custom weight function.

Figure 5 represents the optimization curves for the Custom M-estimator. These are 3-dimensional, since there are two parameters to choose. Their analysis lead to choose k1 = 2 and k2 = 7.

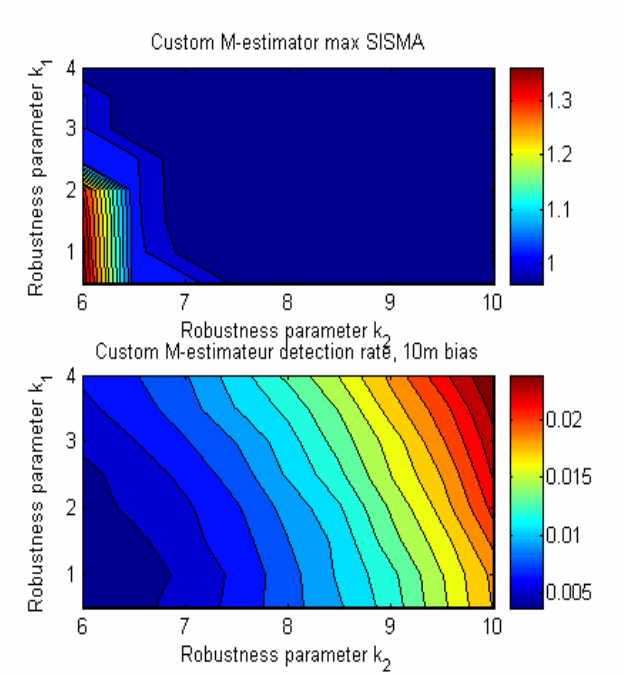


Figure 5: Results of custom M-estimator optimisation

As for the LTS and MCD, the parameter to optimize is the $P_{threshold}$, with which a chi square threshold is computed, so as to perform a final selection of the input data. The analysis of the data represented in figures 6 and 7 lead to the choice of $P_{threshold} = 0.97$ for both methods.

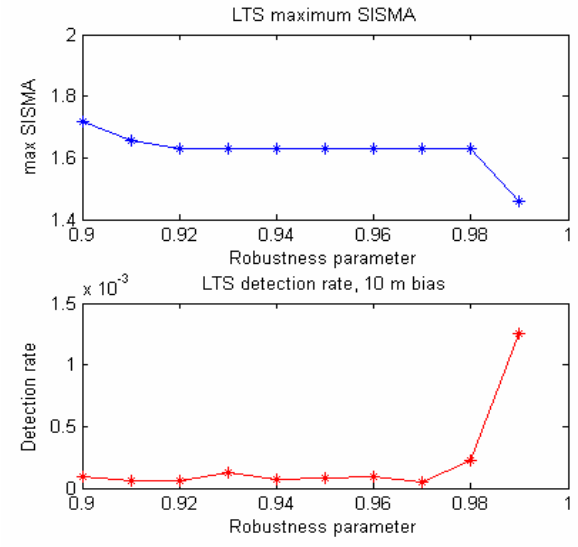


Figure 6: Results of LTS optimisation

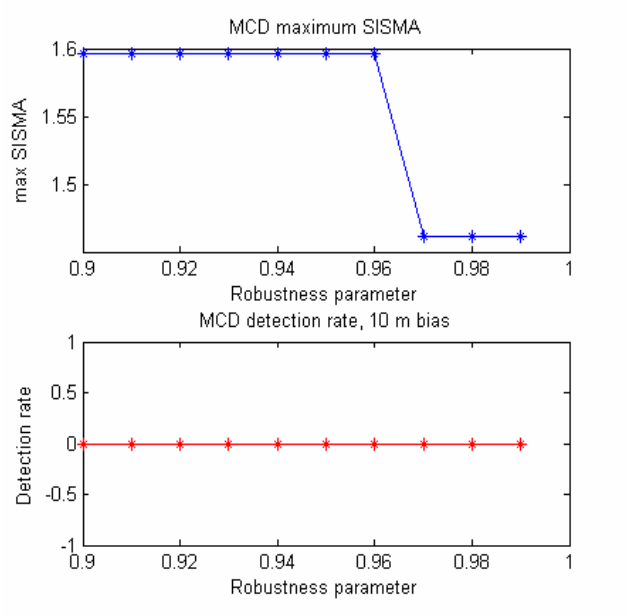


Figure 7: Results of MCD optimisation

EXPECTED GAINS FROM ROBUST METHODS AND TESTING PROCEDURE

Now that the different algorithms are ready to be used, we must explain what can be expected from them and how they can be tested. Indeed, the reason to use robust methods in the Galileo ground segment is obviously to remove outlying measurements when computing $SISE_{est}$ and $SISMA$, as the $SISE_{est}$ computed from outlier-free input would be less likely to cause false alarms. The Integrity Flag Algorithm would thus be prevented from sending a "DON'T USE" flag because of propagation or reception problems.
Such an isolated propagation and/or reception problem would appear as one outlier in the set of pre-processing residuals. Therefore the test procedure for this feature consists in adding a bias to one of the inputs, for all satellites and all epochs, so as to cause a potential false alarm at every test. The detection rate of each robust method is then compared to the LS detection rate: the smaller the rate, the more robust the estimator is.

Then, three aspects must be monitored. The first is that the derivation of high-level availability requirements implies that all SISMA values must be under a certain value, depending on the station configuration. Indeed, as SISMA reflects satellite position and clock estimation errors, it is expected to be as small as possible in order to satisfy system availability requirements. In the present case, the ground network comprising 40 stations, the maximum allowed value of SISMA is 0.7m. As the LS is optimal when measurements are truly Gaussian (fault-free case), robust estimators are expected to degrade performances in this respect. This degradation is assessed by performing a simulation in the fault-free case and

computing, for each method, the cumulative density function of SISMA. The two quality criteria that were chosen to compare the different methods are the maximum value of the SISMA, and the proportion of SISMA values inferior to 0.7m, as shown for the LS in figure 8.

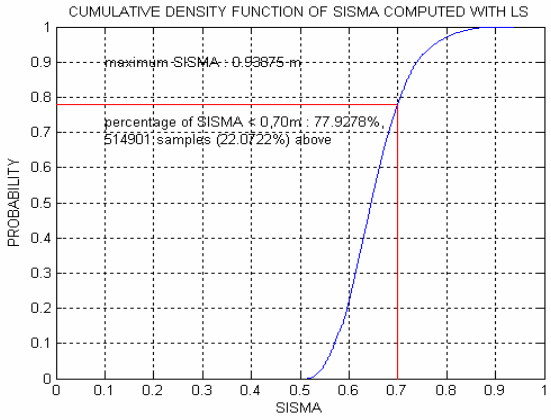


Figure 8: CDF of SISMA computed with LS

The second aspect is the overbounding capability: in the fault-free case, the true distribution of $SISE_{est}$-SISE must be overbounded by a centred Gaussian distribution of standard deviation equal to SISMA. The performances in this respect are assessed by computing the difference between tail weights of the Gaussian distribution $N(0, SISMA)$ and the tail weights of the experimental distribution of $SISE_{est}$-SISE (in the present case, all data being simulated, the true SISE is known). This difference should ideally be always positive. The quality criterion for this simulation is the aspect of the curve, showing the minimum value of the difference and the proportion of negative values.

The third aspect is true failure detection: the ability to reject outliers may degrade failure detection capability. In order to assess this degradation, a third type of simulation was performed, where, at every epoch, a failure of one satellite was simulated by adding a bias to all the measurements performed by GSS on the faulty satellite. The quality criterion for this experiment is the detection rate on the faulty satellite, which should be equal to 1.

As mentioned above, the false detection and true detection tests require a bias. Thus all the corresponding simulations were performed for biases ranging from 10cm to 10m.

SISMA COMPUTATION RESULTS

The reader must first be warned against considering that the following results are representative of the true performances of the future GALILEO system. Indeed, the present study aimed at appreciating the impact of using robust statistics rather than classical Least-Squares and is only representative of the relative efficiency of the proposed methods as compared to LS. All the following tests have been performed on 24 hours of simulated pseudorange residual data.

Figure 9 displays the results obtained in detection of fault-free and faulty satellites. In the top figure, all detections cause a fault-free satellite to be flagged "DON'T USE" and it is expected from robust algorithms that they may decrease the false alarm rate. This result is achieved since detection rates are always smaller for robust algorithms than for LS. In particular, the LTS and MCD algorithms perform well: the maximum detection rate of the LTS (obtained for a bias of 10m) is inferior to 1.10^{-4}, while MCD actually succeeds in never raising the Integrity Flag for all but one bias value (for a bias of 4m, the detection rate is $1.569.10^{-4}$).

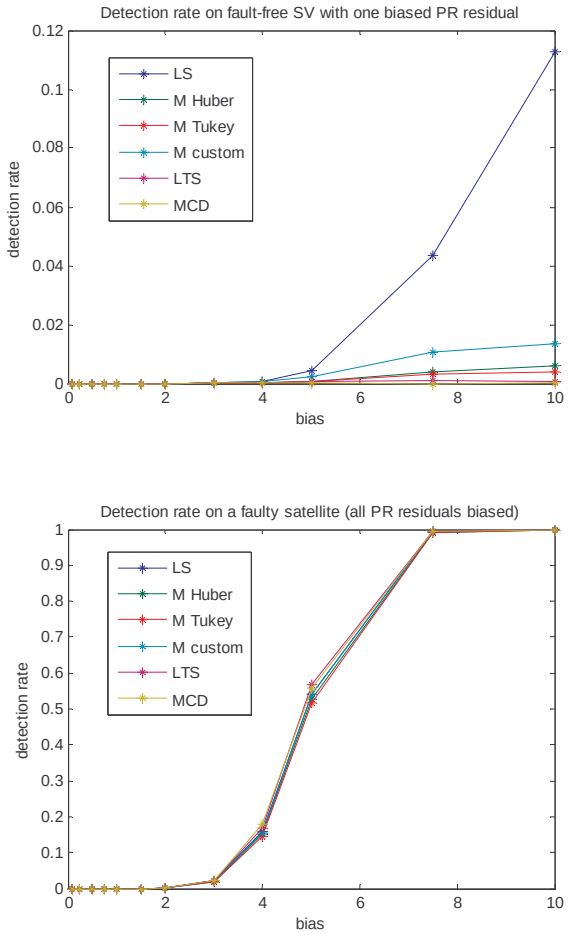


Figure 9: False alarm and true fault detection rates

Then, the bottom figure shows the fault detection rate as a function of the bias. M-estimators tend to slightly decrease detection rate (as compared to LS), thus increasing missed detection rate. On the opposite, LTS and MCD perform once again particularly well, since they actually provide higher detection rates than the LS.

On the other hand, this efficiency in false alarm and fault detection cause LTS and MCD to provide mediocre

results in terms of SISMA distribution. Indeed, table 1 shows maximum values of SISMA and percentage of SISMA inferior to 0.70m for all methods: while M-estimators (particularly with the custom function) only slightly degrade SISMA distributions, system availability might be seriously impacted by the use of LTS or MCD.

	Max SISMA (m)	Percentage of SISMA < 0.7m
LS	0.9388	77.9278
M Huber	0.9760	74.9155
M Tukey	1.2638	70.4186
M custom	0.9931	77.5716
LTS	1.7262	64.5722
MCD	1.9752	63.9085

Table 1: SISMA distribution for all estimation methods

Finally, the last feature that has been studied is the overbounding capability, as shown in figure 10. It appears that the performance of M-estimators is comparable to that of LS, while that of MCD is particularly good (minimum value of the difference much higher than for other methods), and that of LTS is particularly bad (minimum value of the difference much inferior to that of other methods).

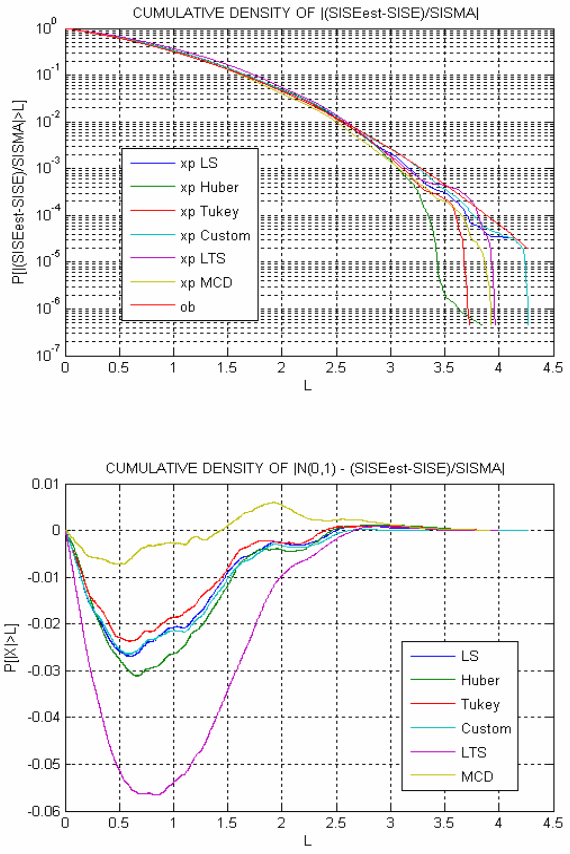


Figure 10: Overbounding capability

CONCLUSION

Thus the results obtained through our simulations can be summarized by the following table, which compares the performance of integrity check achieved through LS with the other procedures:

	M-estimators	LTS	MCD
False alarm rate	Good	Excellent	Excellent
SISMA distribution	Slightly Degraded	Degraded	Degraded
True fault detection rate	Lowered	Increased	Increased
Overbounding capability	Comparable to LS	Degraded	Increased

Therefore, although the feasibility is achieved, there is no candidate algorithm improving all the quality criteria chosen to compare all methods, since algorithms that perform best in detection have serious drawbacks on other respects (SISMA distribution for MCD and LTS, overbounding for LTS).

However, the performance of MCD (or LTS) is very interesting, and it is important to determine the budget of the degradation in SISMA distribution (caused for example by the MCD), especially when a robust algorithm is also implemented on the user side.

REFERENCES

[1] PJ Huber, “Robust Statistics”, *Wiley*, 1981

[2] PJ Rousseeuw, AM Leroy, “Robust Regression and Outlier Detection”, *Wiley*, 1987

[3] G Pison, *et al.*, “Small Sample Corrections for LTS and MCD”, *Metrika*, **55**, 111-123.

[4] V Oehler, *et al.*, “The Galileo Integrity Concept”, *Proceedings of ION GNSS 2004*. Long Beach, CA, Sept. 21-24, 2004.