

Numerical Reconstructions from Reflected Global Navigation Satellite Data

JÉSSICA PERFEITO

jessica.r.luis.perfeito@tecnico.ulisboa.pt
Instituto Superior Técnico, Universidade de Lisboa, Portugal

June 2017

Abstract: Using GNSS-R (Global Navigation Satellite System Reflectometry) we aim to recover information on Earth's surface, where the satellite signal was reflected. The principal GNSS-R Bistatic Radar equation can be written in a convolution between two functions: one depends on the Earth's characteristics, and the other one only depends on features of the instrument measurement. We replace the convolution by a linear system that represents the equation, recovering the function that carries information about Earth's characteristics, and finally computing the bistatic radar cross section. Computational tests present different numerical methods for the inversion procedure, such as Tikhonov regularization and Truncated Singular Value Decomposition. **Keywords:** GNSS-R, numerical reconstruction, bistatic radar equation.

I. INTRODUCTION

According to *Giulio (2006)* [1], the aim of Global Navigation Satellite Systems (GNSS) is "to provide users with precise position/navigation information". This type of systems is constituted by a constellation of satellites spreading in a high altitude around the globe, to perform triangulation from every point on the Earth's surface, in the air and close space (in [1]). GNSS must be functional at every time, under all weather conditions and should be precise, in order to provide information for the user at real time.

This work will be based in one Global Navigation Satellite System: the GPS (Global Positioning System)¹. Basic information about structure of GPS signal needed in this work was consulted in [1, 2, 3], and provide a bridge between this topic and the next one, the Global Navigation Satellite Systems Reflectometry (GNSS-R). GNSS reflectometer composes a non monostatic radar in opposition with the usual GNSS. In other words, in case of GNSS, the emitted signal is measured in the same device and direction that was generated (the antenna is the same); and for GNSS-R, the signal is emitted by the transmitter but is measured in one (bistatic radar) or more (multistatic radar) receivers. Although this difference, GNSS-R techniques can be applied to remote sensing [1] as in GNSS techniques.

The main equation of GNSS-R is composed by the measurement of an instrument Y that results from the convolution between two functions: one is provided and build by the measurement instrument χ ; and the other depends on the scattering surface Σ . Giving that, and not knowing the properties and roughness of that

surface, is not easy to recover the information provided by Σ (which contains the information about scattering surface that we want to know). Apart from the deconvolution problem, another issue takes place when we need to match the instrument domain with the spatial domain.

II. GNSS-R FUNDAMENTAL EQUATION

First we will introduce some basic concepts that will be needed to understand the structure of the principal GNSS-R Bistatic Radar equation.

The specular point (SP) attains the minimum nominal distance from all the scattering points in the domain. In a mathematical sense, we choose the point that satisfies the minimum distance with respect to the receiver and the transmitter. Saying that, let $d(\vec{p}) = ||\vec{R}_x - \vec{p}|| + ||\vec{T}_x - \vec{p}||$ be the distance of a point in the Earth's surface \vec{p} with respect to the positions of the receiver \vec{R}_x and the transmitter \vec{T}_x . The position of the specular point satisfies $\vec{s}p = \arg \min_{\vec{p} \in E} \{d(\vec{p})\}$, where E is the Earth surface. This is a reference point to all others, since is the closest to the receiver and the transmitter.

The scattered points around the specular point are in the glistening zone (GZ) (here denominated by G), which is considered to be the area around the specular point that contains the most important information from the signal, since the one in the exterior border of this zone should have lower information about the signal in appreciation.

The delay $\tau(\vec{p})$ (in seconds) from a scattering point with a vector position \vec{p} , is given by $\tau(\vec{p}) = \frac{d(\vec{p})}{c}$, where c is the speed of light. It is normally expressed depending on the specular point delay $\tau(\vec{s}p)$, in a way that the domain of delays has its minimum value on the specular

¹There are more Global Navigation Satellite Systems that will not be mentioned on this report – to more details about the GNSS history see section 1.1 of [2].

point, where the shift delay is zero, and it is positive for the other scattering points. For simplicity, from now on, we use the shift delay with respect to the specular point as the delay, $\tau(\vec{p}) \equiv \tau(\vec{p}) - \tau(\vec{s}\vec{p})$.

The Doppler frequency depends on the velocities of the receiver and the transmitter, \vec{V}_r and \vec{V}_t respectively, and it's defined as $f_D = (\vec{V}_t \cdot \vec{n}_i - \vec{V}_r \cdot \vec{n}_s) / \lambda$ (in Hertz), where \vec{n}_s is the normal scattering vector, \vec{n}_i is the normal incident vector, and λ is the wavelength for the band considered.

Equation (6) that will be the main focus of this work, represents a function dependent on several characteristics of the reflected signal from a surface, and it's known as the power distribution of the scattered signals from that surface. This function is directly related not only to the transmitter and receiver specifications, but also related to the surface shape and roughness, and characteristics from the measuring instrument. This power distribution is described by some functions and constants: τ and f_D are the delay and the Doppler frequency in handling; T_i is the coherent integration time, which is a constant; σ^0 is the scattering coefficient, or radar cross section, which is related to the characteristics of the surface; $R_0(\vec{p})$ is the distance from the transmitter to the considered scattering point \vec{p} ; $R(\vec{p})$ is the distance from the considered scattering point \vec{p} to the receiver; $D(\vec{p})$ is the antenna pattern; λ is the wavelength of the band of the scenario; $\Delta\tau = \tau - \tau_{xy}$ and $\Delta f_D = f_D - f_{D,xy}$ are the delay and Doppler frequency shifts of each point on the surface relative to the delay τ and Doppler f_D considered in the integration domain;² a is a function in \mathbb{R}^2 (for the variables τ and f_D).

Now we introduce Λ and S , the functions required to construct the power distribution Y . These two functions appear from an approximation of the ambiguity function χ (or Woodward ambiguity function (WAF)):

$$\chi^2(\tau, f_D) \simeq \Lambda^2(\tau) |S(f_D)|^2, \quad (1)$$

$$\Lambda(\tau) = \begin{cases} 1 - \frac{|\tau|}{\tau_c}, & \text{if } |\tau| \leq \tau_c, \\ 0, & \text{otherwise.} \end{cases} \quad S(f_D) = \text{Sinc}(f_D T_i), \quad (2)$$

where τ_c is the length (in seconds) of a pulse in the C/A code.

For simplicity of the expressions³, we introduce here a new function Σ dependent on the characteristics of the

²Note that τ and f_D are specific values for the delays and Dopplers in the domain considered, and τ_{xy} and $f_{D,xy}$ are the values for the delay and Doppler for a specific point $\vec{p} = (x, y)$ on the surface.

³For more details about the formula, see [4].

scenario

$$\Sigma(\tau, f_D) = \frac{T_i^2 \lambda^2}{(4\pi)^3} \frac{D(\vec{p}(\tau, f_D)) \sigma^0(\vec{p}(\tau, f_D))}{R_0^2(\vec{p}(\tau, f_D)) R^2(\vec{p}(\tau, f_D))} |J(\tau, f_D)|, \quad (3)$$

and where the change of variables from the spatial domain to the delay-Doppler domain are expressed by the jacobian J .

Given the functions, variables and constants, we are able to introduce the expression for $Y(\tau, f_D) = \langle \langle \Sigma, B_{\chi^2} \rangle \rangle$ in the following definition.

Definition 1. Let Σ and B_a be two functions with domain in the delay-Doppler, Σ defined as in (3), the operator $\langle \langle \Sigma, B_a \rangle \rangle$ is defined in (4) as an integral in G_1 and G_2 , the domains of the delays and the Doppler frequencies, respectively.

$$\langle \langle \Sigma, B_a \rangle \rangle := \int_{G_1} \int_{G_2} \Sigma(\tau_{xy}, f_{D,xy}) B_a(\tau_{xy}, f_{D,xy}) d\tau_{xy} df_{D,xy}, \quad (4)$$

with

$$B_a(\tau_{xy}, f_{D,xy}) := a(\tau - \tau_{xy}, f_D - f_{D,xy}). \quad (5)$$

The power distribution Y can be computed using the convolution, $Y(\tau, f_D) = \Sigma(\tau, f_D) * \chi^2(\tau, f_D)$ or using Definition 1 with $a = \chi^2$,

$$Y(\tau, f_D) = \int_{G_1} \int_{G_2} \Sigma(\tau_{xy}, f_{D,xy}) \chi^2(\Delta\tau, \Delta f_D) d\tau_{xy} df_{D,xy}. \quad (6)$$

III. DESCRIPTION OF THE METHODS

We will now introduce the methods used to obtain Σ , given the observations of Y and constructing χ^2 .

i. Noise and deconvolution

In this section we start by presenting the type of noise considered. From this we build a linear system which must represent the convolution product. The next step will be the process of the inversion, or the deconvolution, to obtain $\tilde{\Sigma}$ (the recovered variable obtain from the measurements with noise \tilde{Y}), and from that, we only need to compute $\tilde{\sigma}^0$. In the end of this section we will introduce an algorithm used to join various $\tilde{\sigma}^0$ regarding mutual areas in the surface but coming from measurements in different times.

In this work, we will study the impact of multiplicative noise, an usual noise present in real observations/measurements from signal processing. A particular

type of multiplicative noise is speckle noise, which follows $\tilde{f} = (1 + n)J$, where J is the image, \tilde{f} the respective image with speckle noise, and n is the random noise distributed in a uniform way. The uniformly distributed random noise has mean value equal to zero and the variance can be chosen.

The introducing of this type of noise can be done with the auxiliary function `imnoise` from the MATLAB directory, using the option "speckle".

Keeping in mind the equation (6) and the fact that we want to reconstruct σ_0 , variable present in the expression for Σ (see formula (3)), is necessary overcome the convolution.

In the convolution procedure, we will need information about the scattering signal present in the power distribution Y given in the Delay-Doppler (DD) domain. Indeed, to recover $\tilde{\Sigma}$, the ambiguity function will be considered too.

The standard ambiguity function χ^2 is known for every scenario, once the domain of $Y(\tau, f_D)$ is known, and the domains are the same for Y and χ^2 . To build $\chi^2(\tau, f_D)$ that generated $Y(\tau, f_D)$, we know the domain (the delays and Dopplers considered) of the measurements, and the functions (present in χ^2) that we need to compute in the refereed domain.

Recalling that the power distribution is obtained by $\tilde{Y}(\tau, f_D) = \tilde{\Sigma}(\tau, f_D) * \chi^2(\tau, f_D)$, the goal is to recover the values of $\tilde{\Sigma}$ which generated the observations of \tilde{Y} , given the observations of $\tilde{Y}(\tau, f_D)$ and building χ^2 .

This becomes a linear system $Ax = b$. Having the observables b , and building the matrix A it is possible to invert A and obtain the vector x , instead of using the convolution product.

We will present here the one dimensional case of the problem (only having the delays for the instrument domain), for details of the problem in two dimensions see [4]. Since we are dealing only with the delays, \tilde{Y} and $\tilde{\Sigma}$ are vectors, and we just need to build a matrix AF that verifies $\tilde{Y}(\tau, \cdot) = AF\tilde{\Sigma}(\tau, \cdot)$. Keeping in mind the definition for the convolution (recall that $\tilde{Y} = \tilde{\Sigma} * \chi^2$), we get

$AF =$

$$\begin{bmatrix} \chi^2(\tau_0 - \tau_0, \cdot) & \chi^2(\tau_0 - \tau_1, \cdot) & \dots & \chi^2(\tau_0 - \tau_{m-1}, \cdot) \\ \vdots & \ddots & \ddots & \vdots \\ \chi^2(\tau_{m-1} - \tau_0, \cdot) & \chi^2(\tau_{m-1} - \tau_1, \cdot) & \dots & \chi^2(\tau_{m-1} - \tau_{m-1}, \cdot) \end{bmatrix}. \quad (7)$$

ii. Numerical methods

A regularization technique is introduced here, since we are dealing with an inverse problem with (or without) noise. The main goal to introduce regularization is to prevent a ill-posed problem or to overcome the inversion procedure.

Here we will adopt a simple type of regularization. The problem is solve the equation $\tilde{Y}(\tau, \cdot) = AF\tilde{\Sigma}(\tau, \cdot)$, but there is no guaranty that AF is even a square matrix. To overcome that problem, it is possible to build a new system, where we invert a square matrix using

$$AF^\top \tilde{Y}(\tau, \cdot) = AF^\top AF \tilde{\Sigma}(\tau, \cdot). \quad (8)$$

In this new system we obtain

$$\tilde{\Sigma}(\tau, \cdot) = (AF^\top AF)^{-1} AF^\top \tilde{Y}(\tau, \cdot), \quad (9)$$

but we use Tikhonov regularization, that has the vantage to deal with the condition number of the matrix in the inverting process. If the matrix $AF^\top AF$ has a low condition number, the system will be seen as a well conditioned problem, and so will be easier to invert.

The system (8), instead of using $AF^\top AF$ we will taken $AF^\top AF + \alpha I$, where I is the identity matrix with the same dimensions of $AF^\top AF$, and $\alpha > 0$ is a regularization parameter which we can shape having in account the problem in hand. Therefore, we get

$$AF^\top \tilde{Y}(\tau, f_D) = (AF^\top AF + \alpha I) \tilde{\Sigma}(\tau, f_D), \quad (10)$$

$$\tilde{\Sigma}_\alpha(\tau, f_D) = (AF^\top AF + \alpha I)^{-1} AF^\top \tilde{Y}(\tau, f_D). \quad (11)$$

With this regularization parameter α , the solution $\tilde{\Sigma}_\alpha$ can be closer to the real solution or not. In fact, if $\alpha = 0$ we have the original problem: values for α close to zero will imply small changes in the original problem. In the other hand, bigger values for α , will introduce a bigger variation in the original system and can perturb the solution $\tilde{\Sigma}$, far from the original one Σ (see [5] for details on the Tikhonov regularization).

The Truncated Singular Value Decomposition can also be used as a regularization technique for the inversion (see e.g. [6]).

The singular values of the matrix AF are ordered $\mu_1^2 \geq \mu_2^2 \geq \dots \geq \mu_n^2 \geq 0$ which correspond to the eigenvalues of the matrix $AF^* AF$. Consider the system (μ_k, v_k, w_k) given by (see [7])

$$\begin{cases} AFv_k = \mu_k w_k \\ AF^* w_k = \mu_k v_k \end{cases}, \quad (12)$$

where v_k is the eigenvector associated to μ_k^2 , and $w_k = \frac{AFv_k}{\mu_k}$.

Then we can solve the linear system $\tilde{Y} = AF\tilde{\Sigma}$ using

$$\tilde{\Sigma} = \sum_{k=1}^m \langle \tilde{Y}, w_k \rangle v_k \frac{1}{\mu_k}. \quad (13)$$

The use of truncation in this method allows us to reject eigenvalues that are close to zero. If $m = n$ we are using all the singular values, but the value for m can be

set as another integer value smaller than (or equal to) n and bigger than 1. If m is a small number, then the results could not be as good as expected since in that case we are losing the information about to a significant number of singular values.

Having an estimation to $\tilde{\Sigma}(\tau, f_D)$ (provided by the inversion using the mentioned methods) and recalling the expression (3), to recover the scattering coefficient distribution σ^0 we just need to compute multiplication and divisions. For that is necessary a transformation matrix T which permits pass from delay coordinates into the spatial domain (x, y) , since the variable σ^0 is expressed in the spatial domain.

iii. Uniqueness and Stability

The matrix AF is obtained using the convolution matrix function `convmtx` available in MATLAB, for the central columns of χ^2 , which corresponds to $\Lambda^2(\tau)$ since $\chi^2(\tau, 0) = \Lambda^2(\tau)$ (see e.g. [4]).

Studying the uniqueness of the system, we must have non zero singular values for AF , in other words, the matrix resulting of the product $AF^T AF$ must have non zero eigenvalues, and so the matrix $AF^T AF$ is an invertible matrix.

In Figure 1 can be found the matrix AF , and the distribution of the respective singular values.

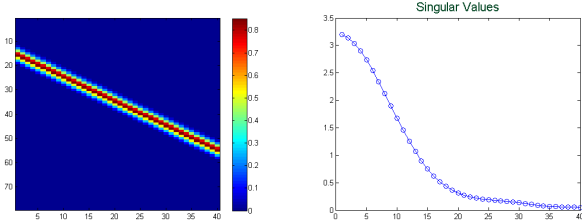


Figure 1: Matrix AF (left) and respective singular values of AF (right), relatives to 1D case.

For uniqueness we must have the singular values higher than zero. In this case, the singular values can be seen in right image of Figure 1, where the maximum value is 3.2007 and the minimum is 0.0503. In these 40 singular values, 27 (67.5%) of them are smaller than 1 (47.5% are between 1 and 0.1 and 20% are smaller then 0.1). Due to that the matrix inversion does not present stability problems.

iv. Measurements along the time

It is usual in this type of problems that we have more than one observation for \tilde{Y} . These observations are called snapshots that represent information from particular instants in time. If we can merge information from all the snapshots available, is expected that the

final recovering will be more precise than using only one observation.

From now on, we will accept that we have $t_0, t_1, t_2, \dots, t_{n_t-1}$ instants in time, and n_t observations of \tilde{Y} , and therefore n_t observations of $\tilde{\sigma}^0$. For the time, we introduce the index t in all variables included in (6) and for simplicity of notation we will skip (in this subsection) the dependence of the spatial of the delay-Doppler coordinates.

Assuming that the problem of inversion was overcome, we can join the information for $\tilde{\sigma}^0$. $\tilde{\sigma}_t^0$ is a variable in spatial domain, and for each recovered observation at time t we have information for slightly different zones due to the movement of the Earth. For realistic and good results, small steps for t should be used.

We now introduce a simple algorithm to merge the n_t recovered $\tilde{\sigma}_t^0$. For that use a recursive average, which is an average calculated in each point in time $t = 0, 1, \dots, n_t - 1$ and where the average at t have in account the previous observations. The average at time t will be expressed by $\overline{\sigma}_t^0$ and computed by the formula

$$\overline{\sigma}_t^0 = \frac{t-1}{t} \overline{\sigma}_{t-1}^0 + \frac{1}{t} \tilde{\sigma}_t^0. \quad (14)$$

The last formula is the general formula, however in a fixed snapshot we can have pixels that have information and pixels that don't have information (NaN). This means that the pixels, where the values for $\tilde{\sigma}$ are NaN , were not "caught" by the instrument of measurement, in other words, that pixels are out of the delay-Doppler domain of the instrument. Given that, we can't use the general formula (14) for each matrix $\tilde{\sigma}_t^0$. Instead of that, we will consider a recursive average for each pixel.

The Swath Map is introduced to have an extended version of the spatial domain for each time t . This new map can provide us a simple view of all information along the time. To merge the snapshots we have the start index along columns I_t^4 for each time t . So the correspondence is $\tilde{\sigma}^0(i, j)_t = \tilde{\sigma}_{SWATH}^0(i, j + I_{n_t})$, and we just apply the formula for the recursive average (14) for each pixel in the swath map.

The advantage in having this conversion to the swath map remains in the fact that all recovered information can be summarized in one image.

v. Comparing the results

Now we will introduced the Morozov criteria, usual used to choose parameter which better minimizes the difference between the measurements generated using that parameter and the original measurements (see [8]).

⁴Here we only have start index along columns due to the information and the characteristics of the considered scenarios. In other cases, can be needed two start index, one for the lines and other for the columns.

This criteria will be useful to choose the value of α in the Tikhonov regularization and the number of singular values used in the truncated singular value decomposition.

In the following, we introduce this criteria for Tikhonov regularization, that allows us to choose wisely the parameter α used to recover.

Morozov Criteria: Let Y_t be the original measurements, and $Y_t(\alpha)$ be the measurements generated by the recovered images \hat{X} using Tikhonov regularization with regularization parameter α , for $t = 1, 2, \dots, N_t$. We must choose the parameter α which minimizes the difference $\|Y_t - Y_t(\alpha)\|$, in other words, we choose the value α which produces the measurements $Y_t(\alpha)$ closer to the original measurements Y_t . Generalizing, we can apply the Morozov criteria not only for Tikhonov regularization, we can also use this criteria to choose a parameter in a specific method, or even to choose a method in the list of methods that we used.

A Building a new grid

To improve the results and get a better comparison between every methods we will introduced an approach that consists in reducing the grid.

We compute the new matrix where one pixel will be the mean value of a "rectangle" in the original image. The "rectangle" built in the new matrix will be fixed in each new grid. For the new grid, each new pixel corresponds to establish a number of columns and lines from the old grid. For the value of the pixel in the new grid, we compute the mean value of the respective pixels values containing in the old image (or old grid).

In the cases where the number of new lines and columns is not a divisor of the original number of lines and columns, respectively, we will build the grid until $\lfloor \frac{\text{original lines}}{l} \rfloor \lfloor \frac{\text{original columns}}{c} \rfloor$, and for the rest of the lines and columns the procedure is the same, but the information in this last line and column will contain information about less than $l \times c$ old pixels. Recall that if l and c are factors of the number of original lines and columns, respectively, in one new pixel we will have the information about $c \times l$ old pixels.

Besides the Morozov criteria, we can compute the error between the original image and the recovered one. For that we introduce the definition for the error in the new grid, where U and U_{method} are the original and the recovered image in the grid (with dimensions $N_{\text{Grid}} \times M_{\text{Grid}}$), respectively.

Definition 2. Let us define the L^2 norm of a grid as

$$\|U\|_{\text{GRID}} := \sqrt{\frac{1}{N_{\text{Grid}} \times M_{\text{Grid}}} \sum_{i=1}^{N_{\text{Grid}}} \sum_{j=1}^{M_{\text{Grid}}} (U(i, j))^2}. \quad (15)$$

Therefore, the error is given by $\|E\|_{\text{GRID}} = \|U - U_{\text{method}}\|_{\text{GRID}}$, and $\|R\|_{\text{GRID}} = \frac{\|E\|_{\text{GRID}}}{\|U\|_{\text{GRID}}}$ for the relative error. As U and U_{method} are matrices, (15) can be written with $U(i, j) = U_{ij}$ and $U_{\text{method}ij}$.

B Adjustment by mean value

Before computing the relative error between the grids selected by Morozov criteria, we decided to adjust the recovered values. This idea came from the visual analysis of the original and the recovered images, where the shapes are identical but the values are not the same. For this process we need to know an a priori value from the original image, the mean value. In this case we have access to the original image, and therefore we can compute the mean value of the matrix, but in other cases we can use a reference value (for example use values obtained in other simulations) or even an expected mean value.

In the adjustment, we first shift the non positive values to positives, and then compute a new image where the mean value will be the same as in the original image. The algorithm is presented below.

```

compute mean_original = mean(mean(original));
if min(min(method)) < 0 then
    method ← method - min(min(method));
compute mean_method = mean(mean(method));
factor = mean_original \ mean_method, and the new
matrix new_method = method × factor;

```

In the algorithm, we use the words "original" and "method" to represent the original and recovered matrices which represent the respective images, and the new matrix will be called "new_method".

IV. RESULTS ANALYSIS

All data and scenarios were provided by *Deimos Engenharia*, which developed a tool to simulate GNSS-Reflectometry called *SARGO Simulation Tools (DEIMOS)*, where all the scenarios were generated.

All the scenarios are identical, apart from the the variable σ^0 . In the following we will present the main difference in each scenario.

The three scenarios are from a UAV (unmanned aerial vehicle) (the receiver) at 2 km of altitude above the ocean. The scenarios were generated under the following parameters: 68 lines (n_x) and columns (n_y) in the spatial domain, respectively; 212 snapshots (n_t); 68 lines and 251 columns of the swath map ($n_{x_{\text{swath}}}$ and $n_{y_{\text{swath}}}$); the integration time (T_i) is equal to 0.001; the wavelength λ is 0.2548; and the instrument domain has 40 delays n_d .

Before presenting the final results, to choose the number of singular values used and the values for the parameter α in the Tikhonov regularization, we used Morozov criteria and the selected parameters for the methods in each scenario are present in Table 1.

Table 1: Chosen parameters for the numerical methods in each scenario, with and without noise.

without noise	with noise		without noise	with noise
1.51	1.51	Nominal	3	3
0.74	0.74	Wind	33	16
1	1	Target	37	40
Tikhonov			TSVD	

Besides from Tikhonov regularization and TSVD, we will present the original image and the recovered one using MATLAB inversion for a better comparison. The measurements taken for the inversion were generated by a 2D problem, where the Doppler frequency is considered, although we present all procedure in one dimension. This is a usual way to treat this type of scenarios, since the shapes are recovered as well and the computational cost is reduced than the two dimensional case. The figures are then presented converting to the 2D setting.

i. Nominal scenario

Nominal scenario has data from a symbolic scenario.

Looking at the obtained results using the methods mentioned above, we have in the following the final recoverege images for σ_{SWATH}^0 in Figure 2.

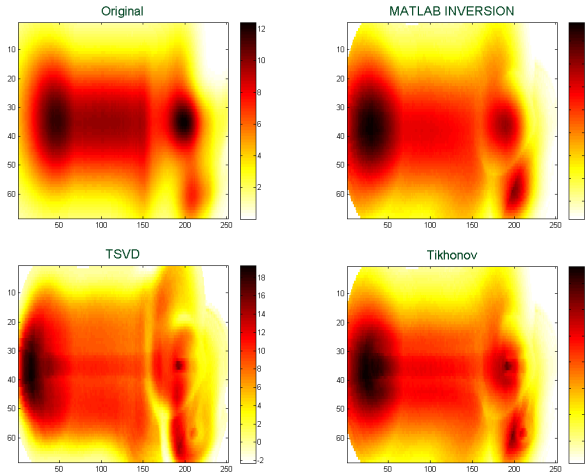


Figure 2: σ_{SWATH}^0 original, and recovering using MATLAB inversion, TSVD and Tikhonov, in nominal scenario without introducing noise.

All the results present a similar shape to the original one. However in all cases, we have a slightly shift to

the left side in comparison to the original image. Then, the locations are not precisely obtained but the shape is similar to the original one, with a slightly shift to the left.

The scales of the values are different from the original data, where we have a scale from 0 to 12, and all the scales in the methods are different, bigger than the original one. In particular, Tikhonov regularization looks the best choice for the recovery procedure, since this method has the scale $[0, 14]$ closer to the scale of the original image $[0, 12]$.

TSVD using only 3 singular values (see Table 1) seems a bad option, as the scale takes non expected negative values. Even the shape is not the same as in the original image, and it is also different from the other methods.

Introducing noise of 1% we obtained the results in Figure 3 and we are facing other type of results.

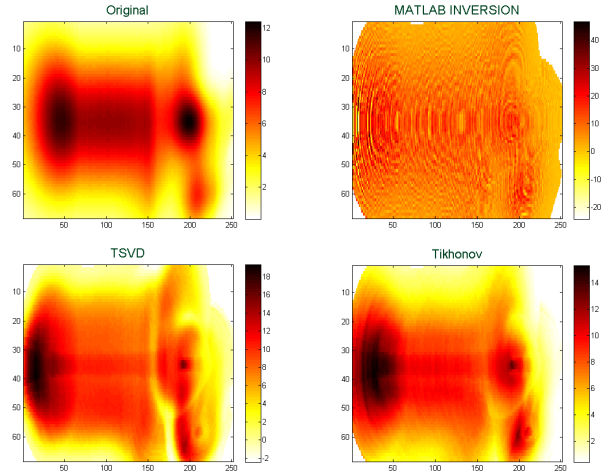


Figure 3: σ_{SWATH}^0 original, and recovering using MATLAB inversion, TSVD and Tikhonov, in nominal scenario introducing 1% of noise.

All the results, except the Tikhonov regularization, have a scale of values with minimum smaller than zero.

The first method, MATLAB inversion, seems to be sensitive to the introduction of noise. This method loses the "defined" shape obtained in the previous case (without noise) and the scale of the values are much bigger than the original one, containing negative values.

Truncated Singular Value Decomposition with only 3 singular values (chosen by Morozov criteria) seems to be not a good option as the shape continues to be more distant from the original image than Tikhonov regularization. However this method has a more "defined" shape, unlike the MATLAB inversion.

As in the previous case, Tikhonov regularization seems to be the best option to confront the introduction of noise. This regularization option presents the same scale of values as in the previous case, without noise. The shape obtained with this method is not ex-

actly the same as in the case without noise, but remains true to the original image: the shape is the same but with some little shadows when passing from the red parts to the yellow parts.

The shift to the left in the recovered images (comparing to the original image) also appears in this case.

ii. Wind scenario

This scenario is similar to the previous one (Nominal scenario), but this case presents values from wind intensity. The difference remains in intensity of the σ^0 . The shapes are identical, only the values for the intensities are smaller than the nominal scenario.

Apparently, in this scenario the results (see Figure 4) appear to be better than the scenario introduced before. Some reasons for that are explained in the comments bellow.

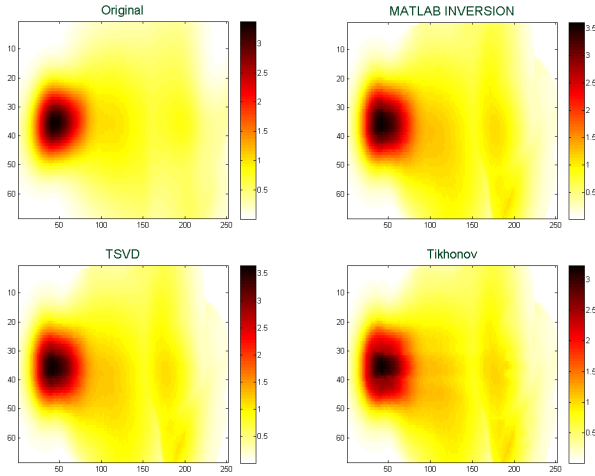


Figure 4: σ_{SWATH}^0 original, and recovering using MATLAB inversion, TSVD and Tikhonov, in wind scenario without introducing noise.

Here we do not have the shift to the left in the images from the methods, when comparing with original image.

The obtained shapes are similar to the original image, only Tikhonov seems to have a slight "non defined" form in the right side of the black/red circles.

The scale of values is not exact but it is closer to the original values. In the original image we have values from 0 to above the 3.5, which in the most of the cases (MATLAB inversion and TSVD) is the maximum value. None of the methods recovered non positive values, what happened in the nominal scenario. In terms of the scale of values, Tikhonov regularization seems to be the best approach, even not being a perfect match with the original scale.

Generally, the results obtained with all methods are good.

Introducing 1% of noise in this scenario, the results are different than the scenario without noise. In Figure 5 it can be found σ_{SWATH}^0 obtained with the recovering methods. In the comments bellow we express the main conclusions.

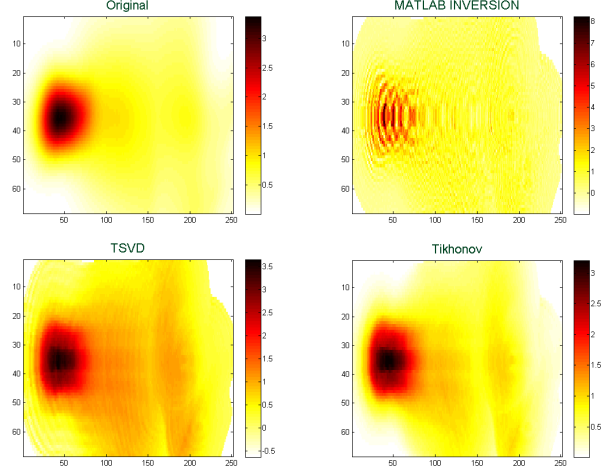


Figure 5: σ_{SWATH}^0 original, and recovering using MATLAB inversion, TSVD and Tikhonov, in wind scenario introducing 1% of noise.

There are two types of recovering. MATLAB inversion seems to be more sensitive to the noise, since we have rough images instead of smooth forms, as it occurs in TSVD or Tikhonov regularization, where the shapes are more well defined.

In terms of the scales, the first method has non positive values and values above 3.5 (which is approximately the maximum value of the original image).

TSVD with 16 singular values (in a total of 40, consult see 1) presents a good result, however the effect of the noise is seen in the shadows of vertical lines, all over the image. The scale of values are almost the same as in the original image but there are not the exact ones.

Tikhonov regularization is the better choice in this scenario with noise. The effect of noise is not detected if we only look at this image and the original one. Comparing this image with the one recovered without noise, the difference is smaller and almost undetected. The values in this image are the most similar ones to the original values.

iii. Target scenario

This scenario is a special one. This case is exactly the previous one with the values from wind intensity, but with the difference in one pixel in σ_{SWATH}^0 which has an intensity much higher than the other pixels. Here the main goal is detect the original location of this different pixel. This pixel appears from snapshot 128 to 211. In that 84 snapshots only appear the information

about that pixel, however, all the white part of it has information about the others pixels, but as the intensity of the special pixel is much higher than the rest of the pixels, the information about the others pixels is hidden in the image (the respective matrix will have all the information).

In Figure 6 the reader can find the results obtained for σ_{SWATH}^0 in this scenario without introducing noise.

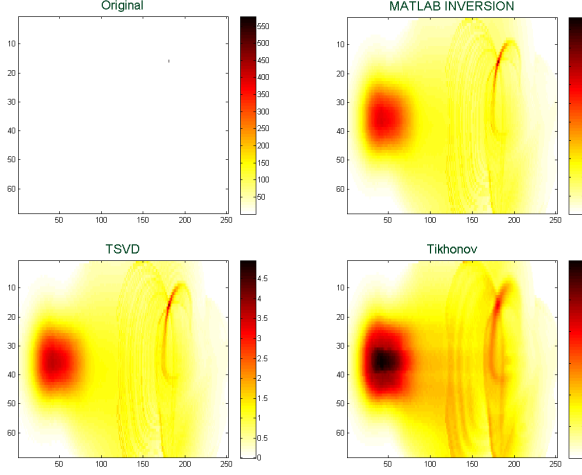


Figure 6: σ_{SWATH}^0 original, and recovering using MATLAB inversion, TSVD and Tikhonov, in target scenario without introducing noise.

All the recovered images are not near the original image. All the methods detected a singular and different pixel at column 181 and line 16. Also the information about the first 100 columns are detected (recall that the information in the rest of the pixels is the same as in the previous wind scenario).

In this case, looking at the scales we see that it is influenced by the pixel with the highest intensity. This intensity is recovered with a lower value, since the information about that pixel is not "clean", in each recovered snapshot of σ^0 (all the pixels with the same delay as the higher information pixel will share similar information), and only joining the snapshots we can obtain the exact location of the higher pixel but not with the same intensity. We can look at the values for the first 100 columns that will be similar to the previous case⁵.

Discussing the shape, we argue about the smoothness or roughness of the images. Clearly the Tikhonov regularization is the one which has smooth lines in the columns from 100 to 251. However, the roughness of the others methods has the vantage that the "vertical" lines are more hidden, or not so noticed as in Tikhonov regularization.

The main disadvantage in Tikhonov regularization is the fact that the pixel with higher intensity (in the origi-

nal image) has a lower intensity than the center of the circle in the first 100 columns. In other hand, MATLAB inversion is the method where the pixel located at line 16 and column 181 has a bigger difference in the values from the area in the firsts columns.

Now, the real question is: How do it change the detection of the target when 1% of noise is introduced in the scenario? The results obtained can be found in Figure 7.

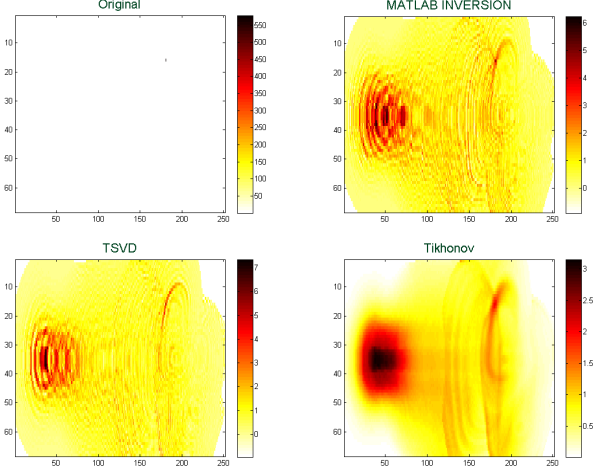


Figure 7: σ_{SWATH}^0 original, and recovering using MATLAB inversion, TSVD and Tikhonov, in target scenario introducing 1% of noise.

Except from Tikhonov regularization, the rest of the methods present rough lines in the recovered images. Tikhonov seems to be not touched by the noise, what happened before in the previous scenarios. However, the issue is that the value of the target pixel is smaller than the values in the circle in the left side of the image, as in the case without noise. This regularization method is also the only one that does not have negative values.

The recovered shapes in the other methods (apart from Tikhonov regularization) are rough, from the introduction of noise. Looking at the methods, the TSVD presents "vertical" lines more soft than MATLAB inversion, where the "vertical" lines have an orange color.

Looking at the first 100 columns, the mutual region with the wind scenario, Tikhonov is the method which better detects the shape without the holes in the red/orange regions, what happened in MATLAB inversion and TSVD.

For the detection of the target, all the methods noticed the presence of a pixel with a different behaviour. TSVD seems to be the better method in the detection, the pixel has a higher value, and the pixels around it have intensities not to strong. This problem, having higher values in the target and in the pixels around it, is present in all methods.

⁵Recall that, in wind case, the maximum value was near 3.5.

iv. Building a new grid

In Table 2, we can find the results for the error between Y_t and the generated measurements using a specific method and grid. In bold we present the best value for each case based on the grid. The reference to a grid, for example, 2×4 means that one pixel in the new mesh has respect to 2×4 pixels in the original image.

Table 2: Error between the original image and the recovered ones using the new grids, for nominal ($\times 10^{-23}$), wind ($\times 10^{-24}$) and target ($\times 10^{-24}$) scenarios.

Scenario	grid	without noise			with noise		
		MATLAB	TSVD	Tikhonov	MATLAB	TSVD	Tikhonov
Nominal	1×1	7.4221	6.4385	6.2410	7.4630	6.4483	6.2416
	2×4	7.4193	6.4341	6.2386	7.4609	6.4441	6.2392
	4×8	7.4076	6.4177	6.2287	7.4495	7.4159	6.2293
Wind	1×1	3.2505	3.2499	3.1988	3.2225	3.2565	3.1974
	2×4	3.2538	3.2532	3.2020	3.2244	3.2598	3.2005
	4×8	3.2649	3.2640	3.2130	3.2337	3.2701	3.2117
Target	1×1	3.9597	3.9590	3.8830	3.9718	3.9745	3.8810
	2×4	3.9720	3.9712	3.8928	3.9821	3.9861	3.8907
	4×8	3.9898	3.9886	3.9137	4.0043	3.9954	3.9114

Summarizing Table 2, in the Wind and Target scenarios to build a grid is not profitable, however in the Nominal scenario, we get a smaller error between Y_t and Y_t^{GRID} . Hereafter, for nominal scenario, we will use the recovered matrix in the grid 4×8 , which in practice is a matrix with dimension 17×32 .

In terms of the methods, the Tikhonov regularization is the method that provides a smaller value in the errors using the Morozov criteria.

The nominal scenario was the only one that provided a change of mesh (or grid) by the Morozov criteria. The results with the grid are presented in the following, in Figure 8.

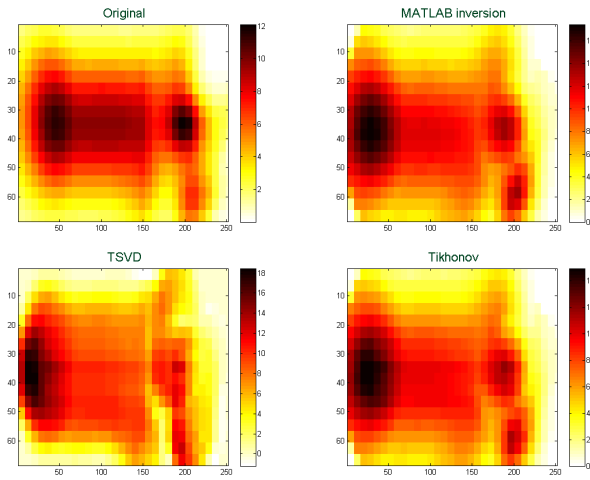


Figure 8: σ_{SWATH}^0 original, and recovering using MATLAB inversion, TSVD and Tikhonov using a new grid, in nominal scenario without introducing noise.

The main difference from the case without grid is that the left shift in the methods with respect to the original image is not so intense here.

In the case with noise, in Figure 9, the influence of using a new mesh is more evident.

Here the shapes in MATLAB inversion are more identical to the original image, comparing with the case without the use of a new grid.

The other obvious vantage is that some of the non positive values disappear.

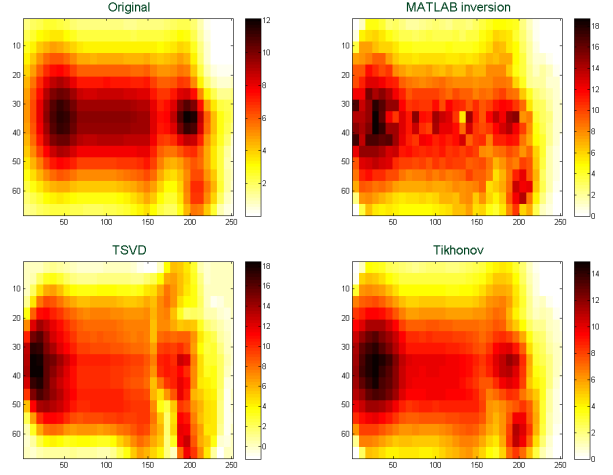


Figure 9: σ_{SWATH}^0 original, and recovering using MATLAB inversion, TSVD and Tikhonov using a new grid, in nominal scenario introducing 1% of noise.

v. Adjustment by mean value

The results about the adjustment by mean can be found below in Table 3. In the most cases, the relative error decrease if we use this adjustment by the mean of the original image, the exception is the TSVD method with noise, in the wind scenario. Regarding the target scenario, the difference is smaller and the relative error increase in a smaller scale, which originate from the type of the scenario.

Table 3: Relative error obtained with the original recovering images and with the adjustment by the mean values.

	Without noise			With noise			
	MATLAB	TSVD	Tikhonov	MATLAB	TSVD	Tikhonov	
Nominal	0.5346	0.5009	0.3828	0.8557	0.5026	0.3817	original
	0.2595	0.3111	0.2698	0.2730	0.3108	0.2688	adjusted
Wind	0.2854	0.2862	0.2282	0.6005	0.2927	0.2288	original
	0.1701	0.1692	0.1923	0.4431	0.3520	0.1914	adjusted
Target	0.9769	0.9775	0.9813	0.9813	0.9813	0.9813	original
	0.9772	0.9777	0.9813	0.9828	0.9829	0.9813	adjusted

In the above table we have also the relative error of the methods, without using the procedure of the grid or an adjustment by the mean.

V. CONCLUSIONS

In a general way, the goal presented in the beginning of this thesis was accomplished, recovering a good approximation of σ^0 .

Considering the three scenarios, the wind scenario was the one where we reach the better results and the nominal was the scenario with the worse results, having been recovered the original shapes, but shifted to the left side. In the target scenario, the main goal was recover the localization of the pixel with a higher or different intensity of σ^0 , that was reached with all the methods, with and without noise. The localization was found by the methods, but the intensity wasn't possible to recover due to the ambiguity problem.

Although the ambiguity problem mentioned, occurring in this type of scenarios, frequent in GNSS-R, joining the snapshots of σ^0 was computed to overcome this problem in all the scenarios. This improves the results, and it is a simple way to face the problem.

The numerical method that better provide the inversions in all the cases was Tikhonov regularization. This method does not present the impact of noise, when introduced, in all the scenarios, and with all the adjustment presented.

Concerning the introduction of a new grid, we concluded that it can improve the results when working with results as the obtained in the Nominal scenario, where the shapes are the correct ones, but the image presents a shift to the left side when comparing with the original image. Building this new grid does not resolve the problem although it improves the results.

The adjustment by mean value has the disadvantage is that we need to know an a priori information, that could be replaced by the reference values. The main advantage is that could really improve the results, as in the case of adjustment by the mean value, the relative error decrease using this adjustment. Thereby, we consider that adjustment by mean value is a really good choice to implement in the results that present the correct shapes but not the correct values.

This work is only the beginning of what could be done for this type of scenarios. Using the recovering methods and the results, the next step could be the physical interpretation of the results. This means read the values in the recovered images and do the translation to the physical meaning of this value, in practice this is to understand what type of grounding we are facing: water, ice, snow, etc.

In terms of the methods used, as was said before, Tikhonov regularization was the method that really deal with the introducing of noise. In this work we use Tikhonov regularization and then use MATLAB inversion (the standard inversion for the software used). It could be interesting to study the impact of using Tikhonov regularization combined with other inversion technique.

Regarding the ambiguity problem, it would be interesting to study the work presented here without this problem. This could be possible if the scenarios were simulated imposing the spatial area that the receiver antenna must point to, as done in [9].

Another option that could be done in the future is to introduce techniques of smoothness in the cases with noise. This can be an option to reduce the impact of noise in the majority of the methods used here, since the Tikhonov regularization was the only method which handled the introduction of noise.

REFERENCES

- [1] R. Giulio. A Brief Introduction to Remote Sensing Using GNSS Reflections. *IEEE Geos. and Rem. Sens. Society Newsletter*, 2006.
- [2] J. Shuanggen, E. Cardellach, F. Xie. *GNSS Remote Sensing - Theory, Methods and Applications*, volume 19. Springer, 2014.
- [3] V. U. Zavorotny, S. Gleason, E. Cardellach, A. Camps. Tutorial on Remote Sensing Using GNSS Bistatic Radar of Opportunity. *IEEE Geos. and Rem. Sens. Magazine*, 2014.
- [4] J. Perfeito. Numerical Reconstructions from Reflected Global Navigation Satellite Data. Instituto Superior Técnico, 2017.
- [5] R. Kress. *Linear Integral Equations (3rd edition)*. Springer, 2014.
- [6] V. Isakov. *Inverse Problems for Partial Differential Equations (2nd edition)*. Springer, 2006.
- [7] C. Alves. Problemas Inversos em Equações Diferenciais Parciais (in portuguese), Instituto Superior Técnico, 2011.
- [8] D. Colton, R. Kress. *Inverse Acoustic and Electromagnetic Scattering Theory (3rd edition)*. Springer, 2013.
- [9] E. Valencia, A. Camps, J. F. Marchan-Hernandez, H. Park, X. Bosch-Lluis, N. Rodriguez-Alvarez, I. Ramos-Perez. Ocean Surface's Scattering Coefficient Retrieval by Delay-Doppler Map Inversion. *IEEE Geosci. Remote Sens. Lett.*, 8(4):750–755, 2011.