Flight Path Reconstruction from SAR Images and Spotlight SAR Data

Aron Sommer, Leibniz Universität Hannover, sommer@tnt.uni-hannover.de, Germany
Jörn Ostermann, Leibniz Universität Hannover, office@tnt.uni-hannover.de, Germany

Abstract

In regular airborne Synthetic Aperture Radar applications the quality of the processed image depends on the accuracy of the flight path measured by an inertial navigation system. Usually, only approximated flight paths are available. Autofocus techniques sharpen the resulting images, but do not correct the flight path. In this paper we present a method to reconstruct the exact flight path from the focused image and the range compressed data. We show that the flight path can be reconstructed precisely for a wide antenna beam of more than $20°$.

1 Introduction

In an airborne Synthetic Aperture Radar (SAR) scenario, inaccuracies in the measured flight path cause phase errors in data. These errors lead to an unfocused image. To generate an optimally focused SAR image, the actual flight path has to be known precisely. Typically, the flight path is measured by an inertial navigation systems (INS) mounted on the platform. However, even expensive, accurate INS units yield imprecisely approximated flight paths due to thermal noise and accumulation errors.

Since only approximated flight paths are available, SAR autofocus techniques were developed to improve image quality without the knowledge of the exact flight path. Almost all of these methods correct phase errors in the data. However, they do not correct the measurement errors of the INS unit. Hence, after the total processing the real flight path is still unknown. The phase corrections, which are estimated by an autofocus method, cannot easily be used to correct the flight path. The 3D position error cannot be estimated from the 1D phase correction, which is in principal the difference between the real flight path and the measured flight path projected onto the evaluation plane.

In this paper, we show how to estimate real flight paths from range compressed data and already focused images. We do not propose a new autofocus technique.

Our flight path reconstruction method avoids techniques like prominent point processing, since problems arise if no symmetric point reflector is in the scene. Furthermore, we avoid image sharpness functions, because finding an optimal sharpness function of SAR images is still under research.
Instead of these common techniques we use the entire focused image as a representation of the real reflectivity function so that even asymmetric point targets do not cause problems.

One limiting factor is the azimuth beam width and the evaluation beam width. To reconstruct the flight path exactly, both angles must be approximately greater than $20°$. Otherwise there is not enough information in the data to distinguish between altitude, range and velocity of the airplane.

This paper is organized as follows. The signal model as well as the data generation process are described in Section 2. From this signal model we derive the algorithm, which reconstructs the flight path as a post processing step in Section 3. The proposed algorithm is based on a Newton type method with nonlinear Tikhonov regularization to compute the flight path iteratively. Section 4 gives three numerical examples to show the performance of this proposed algorithm. Section 5 draws the conclusion.

2 Signal Model

Assuming the airplane flies with arbitrary velocity v along a flight path $\gamma : \mathcal{L} \to \mathbb{R}^3$, which is parametrized by the slow time $s \in \mathcal{L} := [0, T_a]$. Here T_a is the synthetic aperture time corresponding to the synthetic aperture length. The actual flight path γ is modeled by a straight line $\overline{\gamma}$ added by some curves $\widetilde{\gamma}$, which are generated by a 3D colored noise function. We use additive Gaussian white noise and a Hann window lowpass-filter in x, y and z to model $\widetilde{\gamma}$. Thus, the real flight path γ is generated by

$$\gamma(s) = \overline{\gamma}(s) + \widetilde{\gamma}(s), \qquad s \in \mathcal{L}.$$

Moreover, let the transmitted chirp pulses p [2], which vary with fast time t, be

$$p(t) = \mathrm{rect}(\tfrac{t}{T}) \exp(-2\pi i f_c t) \exp(i\pi\kappa t^2), \qquad t \in \mathcal{T},$$

with carrier frequency f_c, chirp rate κ, pulse duration T and sampling time interval $\mathcal{T}$. We assume that the ground is a flat surface at height $z \equiv 0$. The reflectivity function $\mathcal{V}$ represents the scatterer in the illuminated area.

Thus, the signal model given by Cheney [1] of the range compressed data is

$$d(t,s) = \int_{\Omega} A(t,s,\boldsymbol{x})\,\mathcal{V}(\boldsymbol{x})\,\text{si}\left(\pi\kappa T\left(t - \frac{2||\boldsymbol{\gamma}(s)-\boldsymbol{x}||_2}{c}\right)\right)$$
$$\cdot \exp\left(-2\pi i f_c \frac{2||\boldsymbol{\gamma}(s)-\boldsymbol{x}||_2}{c}\right) \mathrm{d}\boldsymbol{x}, \qquad (1)$$

where c is the speed of light, A the antenna beam pattern and Ω the illuminated scene. Note that in a spotlight scenario the antenna beam pattern A can be neglected, because all targets are in the illuminated area Ω.

3 Flight Path Reconstruction

In this section we firstly introduce the flight path reconstruction problem in terms of a nonlinear inverse problem. We derive the algorithm from the nonlinear system equation by applying a Newton type method including a regularization technique. In the second part of this section we describe the implementation of the algorithm.

3.1 Problem Formulation

Let $\mathcal{V}$ be the known reflectivity function, d the range compressed data and $\boldsymbol{\gamma}$ the unknown, actual flight path, from which only an initial approximation $\boldsymbol{\gamma}_0$ is available. The system equation (1) reads in operator form

$$\Lambda_{\mathcal{V}}(\boldsymbol{\gamma}) = d, \qquad (2)$$

where $\Lambda_{\mathcal{V}}$ is the forward operator including the true reflectivity $\mathcal{V}$. To find $\boldsymbol{\gamma}$ from d and $\mathcal{V}$ the nonlinear problem (2) has to be solved. Following the idea of Newton type methods, we firstly linearize equation (2) by applying the Taylor approximation for operators [6]:

$$\Lambda_{\mathcal{V}}(\boldsymbol{\gamma}_0) + \partial_{\gamma}\Lambda_{\mathcal{V}}(\boldsymbol{\gamma}_0)\left(\boldsymbol{\gamma} - \boldsymbol{\gamma}_0\right) + \varepsilon = d. \qquad (3)$$

The derivative $\partial_{\gamma}\Lambda_{\mathcal{V}}(\boldsymbol{\gamma}_0)$ of the forward operator $\Lambda_{\mathcal{V}}$ of the flight path $\boldsymbol{\gamma}$ is generally a Frechet Derivative [7]. We define $h = \boldsymbol{\gamma} - \boldsymbol{\gamma}_0$ to be the difference between the unknown flight path $\boldsymbol{\gamma}$ and an initial solution $\boldsymbol{\gamma}_0$. Since we are only interested in the first order Tailor approximation, we neglect the approximation error ε. From equation (3) we find the linearized operator equation

$$\partial_{\gamma}\Lambda_{\mathcal{V}}(\boldsymbol{\gamma}_0)h = w. \qquad (4)$$

Here $w := d - \Lambda_{\mathcal{V}}(\boldsymbol{\gamma}_0)$ is the right-hand side containing the range compressed data d and the synthetically generated data $\Lambda_{\mathcal{V}}(\boldsymbol{\gamma}_0)$ along the initial flight path $\boldsymbol{\gamma}_0$. Remind that the synthetic data $\Lambda_{\mathcal{V}}(\boldsymbol{\gamma}_0)$ can be assembled by the system equation (1).

Moreover, the derivative $\partial_{\gamma}\Lambda_{\mathcal{V}}(\boldsymbol{\gamma}_0)$ is directly derived from the system equation (1):

$$\partial_{\gamma}\Lambda_{\mathcal{V}}(\boldsymbol{\gamma}_0) =$$
$$\int_{\Omega} \mathcal{V}(\boldsymbol{x})\left(\frac{\sin\left(t - \frac{2||\boldsymbol{\gamma}_0(s)-\boldsymbol{x}||_2}{c}\right)}{\left(t - \frac{2||\boldsymbol{\gamma}_0(s)-\boldsymbol{x}||_2}{c}\right)^2} - \frac{\cos\left(t - \frac{2||\boldsymbol{\gamma}_0(s)-\boldsymbol{x}||_2}{c}\right)}{t - \frac{2||\boldsymbol{\gamma}_0(s)-\boldsymbol{x}||_2}{c}}\right)$$
$$\cdot \frac{2}{c} \exp\left(\frac{-4\pi i f_c}{c}||\boldsymbol{\gamma}_0(s)-\boldsymbol{x}||_2\right)\frac{\boldsymbol{\gamma}_0(s)-\boldsymbol{x}}{||\boldsymbol{\gamma}_0(s)-\boldsymbol{x}||_2} \mathrm{d}\boldsymbol{x} \qquad (5)$$
$$+ \int_{\Omega} \mathcal{V}(\boldsymbol{x})\,\text{si}\left(\pi\kappa T\left(t - \frac{2||\boldsymbol{\gamma}_0(s)-\boldsymbol{x}||_2}{c}\right)\right)$$
$$\cdot \frac{-4\pi i f_c}{c} \exp\left(\frac{-4\pi i f_c}{c}||\boldsymbol{\gamma}_0(s)-\boldsymbol{x}||_2\right)\frac{\boldsymbol{\gamma}_0(s)-\boldsymbol{x}}{||\boldsymbol{\gamma}_0(s)-\boldsymbol{x}||_2} \mathrm{d}\boldsymbol{x}$$
$$=: \partial_{\gamma}^c\Lambda_{\mathcal{V}}(\boldsymbol{\gamma}_0) + \partial_{\gamma}^f\Lambda_{\mathcal{V}}(\boldsymbol{\gamma}_0).$$

The first integral denoted by $\partial_{\gamma}^c\Lambda_{\mathcal{V}}(\boldsymbol{\gamma}_0)$ describes the coarse flight path reconstruction, where the derivative of the si-function mainly dominates the integral. The second integral denoted by $\partial_{\gamma}^f\Lambda_{\mathcal{V}}(\boldsymbol{\gamma}_0)$ describes the fine flight path derivative, where the phase information is used to correct small motion errors.

So far, we explained all components of equation (4). Now we show how to solve it. Since equation (4) is the linearization of the nonlinear equation (2) it has to be solved iteratively, which is the idea of a Newton type method. Since equation (4) describes a linear inverse problem, which is in general a challenging task, we apply the Tikhonov regularization method [3] to ensure that the solution is well defined. So we summarize the basic idea of the proposed flight path reconstruction algorithm:

$$\boldsymbol{\gamma}_{i+1} = \boldsymbol{\gamma}_i + h_{\alpha,i},$$
$$h_{\alpha,i} = \arg\min_h \left(||\partial_{\gamma}\Lambda_{\mathcal{V}}(\boldsymbol{\gamma}_i)h - w_i||_2 + \alpha||Dh||_2\right), \qquad (6)$$
$$w_i = d - \Lambda_{\mathcal{V}}(\boldsymbol{\gamma}_i).$$

In every iteration of the algorithm we compute the residual data w_i, solve the regularized equation (6) and update the flight path.
The flight path difference $h_{\alpha,i}$ depends on the regularization parameter α. The purpose of α is to find a compromise between accuracy and stability and has to be chosen advisedly. How to find a suitable α is mentioned in the next section. As penalty operator D we use a finite difference operator. This operator includes the a priori information that the flight path must be a smooth, continuous curve in $\mathbb{R}^3$.

One can show that solving equation (6) is equivalent to solve the following problem:

$$\boldsymbol{\gamma}_{i+1} = \boldsymbol{\gamma}_i + h_{\alpha,i},$$
$$h_{\alpha,i} = \left(\partial_{\gamma}\Lambda_{\mathcal{V}}(\boldsymbol{\gamma}_i)^{\top}\partial_{\gamma}\Lambda_{\mathcal{V}}(\boldsymbol{\gamma}_i) + \alpha D\right)^{-1} \qquad (7)$$
$$\cdot \partial_{\gamma}\Lambda_{\mathcal{V}}(\boldsymbol{\gamma}_i)^{\top} w_i,$$
$$w_i = d - \Lambda_{\mathcal{V}}(\boldsymbol{\gamma}_i).$$

Finally, this is the core of the proposed flight path reconstruction algorithm and the basis of our implementation.

3.2 Implementation

In the previous section we derived the core of the flight path reconstruction algorithm (7) from the nonlinear system equation (2). The key idea is to update the flight path iteratively so that it converges to the actual flight path. To compute the update $h_{\alpha,i}$, a linearized inverse problem has to be solved in every iteration.

Here we want to give an overview of the implementation of the proposed algorithm. The algorithm is divided into two parts: Firstly, the coarse part does a registration of the flight path to less than one half of the wavelength λ. Secondly, the fine part uses the phase information to reconstruct the unknown flight path accurately.

The following pseudocode shows the process of the proposed algorithm, more precisely the sequential two parts of them:

Algorithm in pseudocode

Input: $d, \mathcal{V}, \boldsymbol{\gamma}_0$

Output: $\boldsymbol{\gamma}$

01: Load data d

02: Load focused image $\mathcal{V}$

03: Load initial flight path $\boldsymbol{\gamma}_0$

04: Set $i = 0$

05: **while** $\| \boldsymbol{\gamma}_{i+1} - \boldsymbol{\gamma}_i \|_2 < \varepsilon_0^c$ **do**

06:     Assemble $\partial_{\boldsymbol{\gamma}}^c \Lambda_{\mathcal{V}}(\boldsymbol{\gamma}_i)$ by equation (5)

07:     Assemble $\Lambda_{\mathcal{V}}(\boldsymbol{\gamma}_i)$ by equation (1)

08:     $w_i = d - \Lambda_{\mathcal{V}}(\boldsymbol{\gamma}_i)$

09:     Compute $h_{\alpha,i}$ by equation (7)

10:     $\boldsymbol{\gamma}_{i+1} = \boldsymbol{\gamma}_i + h_{\alpha,i}$

11:     $i = i + 1$

12: **end**

13: **while** $\| \boldsymbol{\gamma}_{i+1} - \boldsymbol{\gamma}_i \|_2 < \varepsilon_0^f$ **do**

14:     Assemble $\partial_{\boldsymbol{\gamma}}^f \Lambda_{\mathcal{V}}(\boldsymbol{\gamma}_i)$ by equation (5)

15:     Assemble $\Lambda_{\mathcal{V}}(\boldsymbol{\gamma}_i)$ by equation (1)

16:     $w_i = d - \Lambda_{\mathcal{V}}(\boldsymbol{\gamma}_i)$

17:     Compute $h_{\alpha,i}$ by equation (7)

18:     $\boldsymbol{\gamma}_{i+1} = \boldsymbol{\gamma}_i + h_{\alpha,i}$

19:     $i = i + 1$

20: **end**

21: **return** $\boldsymbol{\gamma} = \boldsymbol{\gamma}_{i+1}$

The regularization parameter α depends mainly on the eigenvalues of the linearized operator $\partial_{\boldsymbol{\gamma}} \Lambda_{\mathcal{V}}(\boldsymbol{\gamma}_i)$. To avoid an computationally expensive eigenvalue analysis, the following approximation is mostly sufficient:

$$\alpha = \text{mean}\big(\text{diag}\big(\partial_{\boldsymbol{\gamma}} \Lambda_{\mathcal{V}}(\boldsymbol{\gamma}_i)^\top \partial_{\boldsymbol{\gamma}} \Lambda_{\mathcal{V}}(\boldsymbol{\gamma}_i)\big)\big).$$

In other words, α is the mean value of all entries on the main diagonal of the regularized operator.

For both termination conditions, $\| \boldsymbol{\gamma}_{i+1} - \boldsymbol{\gamma}_i \|_2 < \varepsilon_0^c$ for the coarse part and $\| \boldsymbol{\gamma}_{i+1} - \boldsymbol{\gamma}_i \|_2 < \varepsilon_0^f$ for the fine part, we chose the values $\varepsilon_0^c = 1$ mm and $\varepsilon_0^f = 0.01$ mm. These values describes the lower bounds of the mean flight path update distance.

One main benefit of this algorithm is that in every iteration step the whole flight path is reconstruction at once. A pulse by pulse computation would result is a non-smooth, discontinuous flight path.

4 Simulation Results

To show the capabilities and the limitations of the proposed flight path reconstruction algorithm, we give two synthetic examples and one real data example using the public Gotcha Dataset [5]. In the first two we simulate an X-band SAR system with parameters listed in **Table 1**. The distance from the flight path center to the spot center is approximately $r_0 = 1000$ m. The depression angle is $\theta_D = 45°$.

Below, it is shown that for a wide antenna beam the flight path can be reconstructed with high precision, whereas in the narrow beam case the accuracy decreases. The reason is that the small spot yields not enough information to separate range and altitude. In the third example we show that deviations up to 1 m can be reconstructed for the Gotcha Dataset [5] with real data. For the initial flight path $\boldsymbol{\gamma}_0$ we use the straight line $\overline{\boldsymbol{\gamma}}$.

Before the example are presented in detail, we define the azimuth beam width as the antenna beam width in horizontal orientation. The vertical antenna beam width is called evaluation beam width.

Table 1: Parameters of simulated X-band SAR system.

Parameter	Value
Carrier frequency f_c	9.6 GHz
Wave length λ	3 cm
Pulse bandwidth B	100 MHz
Pulse duration T	6 µs
Sampling frequency f_s	200 MHz
Mean platform velocity v_0	100 m/s
Synthetic aperture length L	400 m
Pulse repetition frequency prf	200 Hz
Synthetic aperture time T_a	2 s
Squint angle θ_S	0°
Depression angle θ_D	45°

1) Wide beam:

In this first synthetic example we simulate a wide antenna beam by placing four point targets in the scene. Two targets are 200 m away from the spot center, two are 100 m, see **Figure 1**. This 200 m×400 m illumination area models approximately a $\theta_{az} = 20°$ azimuth beam and a $\theta_{ev} = 20°$ evaluation beam. To demonstrate that the proposed method works even perfectly with asymmetric point targets, we simulate each single target by three single point targets in a row, see **Figure 1**. The data is generated by equation (1) added by 10 dB Gaussian white noise along the actual flight path shown in **Figure 2** as a dashed, blue line (partly overlaid by the red line, which represents the reconstructed flight path).
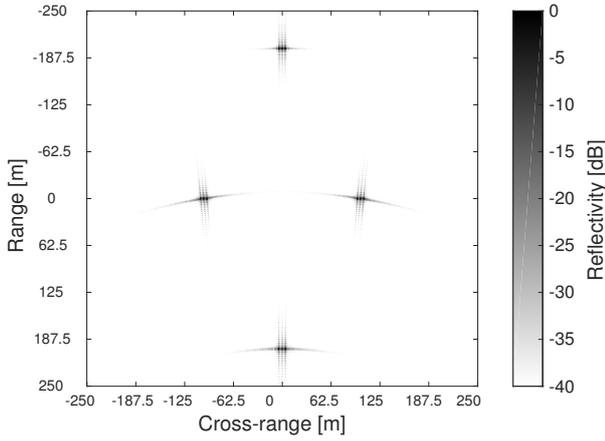
Figure 1: Four asymmetric point targets are placed $100\,\mathrm{m}$ and $200\,\mathrm{m}$ away from the spot center to simulate a wide antenna beam. This optimal focused image is processed by the global Backprojection algorithm using the actual flight path.

The result of the flight path reconstruction algorithm is shown in **Figure 2**. It shows, that the azimuth, range and altitude-component of the actual flight path are reconstructed precisely. This means that measurement errors of the INS unit can be corrected in the wide beam case up to half of the wavelength. The usage of asymmetric point reflectors shows the advantage of the inclusion of the entire image.
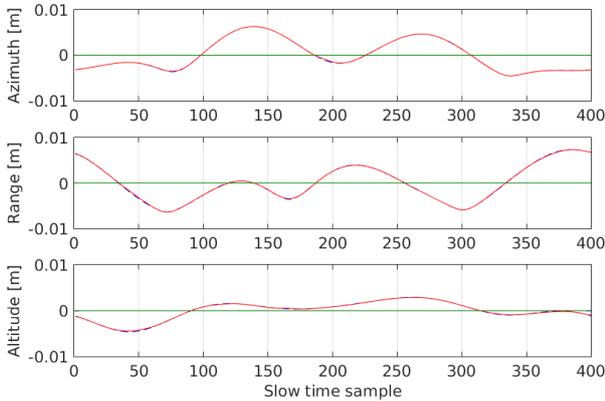
Figure 2: Simulated flight path in the case of a $60°$ antenna beam in azimuth. The dashed, blue line is the actual flight path γ. The continuous red line is the reconstructed flight path and the dotted green line is the initial solution γ_0.

2) Narrow beam:
In the second example we simulate an azimuth beam of $\theta_{\mathrm{az}} = 5°$ and an evaluation beam of $\theta_{\mathrm{ev}} = 2°$ as approximately used in common SAR systems [4]. To do so, we put four single point targets each $20\,\mathrm{m}$ away from the spot center, see **Figure 3**. All other parameters are still the same as in example 1).
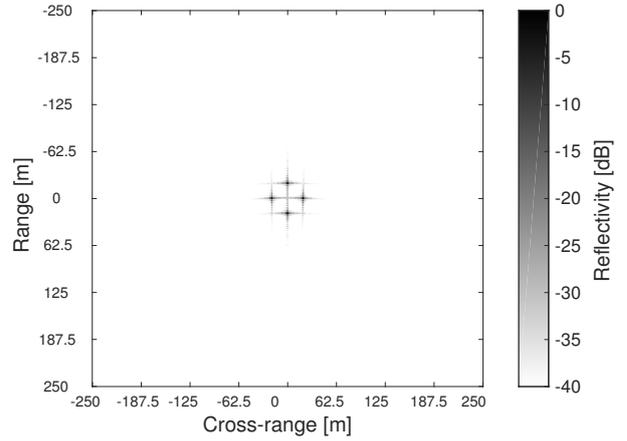

Figure 3: This well focused image is processed by the global Backprojection algorithm. Four symmetric point targets are placed each $20\,\mathrm{m}$ away from the spot center to simulate this narrow antenna beam. Here the new reconstructed flight path is used, which is different from the actual flight path, see **Figure 4**.

The results in **Figure 4** show that the azimuth component of the position error, which is basically the acceleration, is well reconstructed. However, the range and altitude components do not converge as good as the azimuth to the real flight path. The reason is that the phase history in the narrow beam case contains not enough information to separate the range from the altitude component. Small deviations in range produces the same phase history as small altitude deviations. However, this new flight path fits to the raw data, so that an repeated image processing of the range compressed data using this new flight path would result in a focused image similar to **Figure 3**. Finally, this shows that 3D position errors can be approximated by 2D phase errors in the narrow beam case.
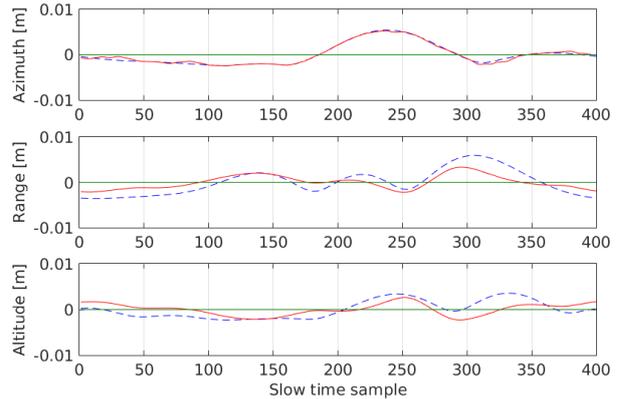

Figure 4: This is the case of an $5°$ antenna beam angle in azimuth. The azimuth, range and altitude deviations of the straight line $\overline{\gamma}$. The dashed, blue line is the actual flight path γ. The continuous red line is the reconstructed flight path and the dotted green line is the initial solution γ_0.
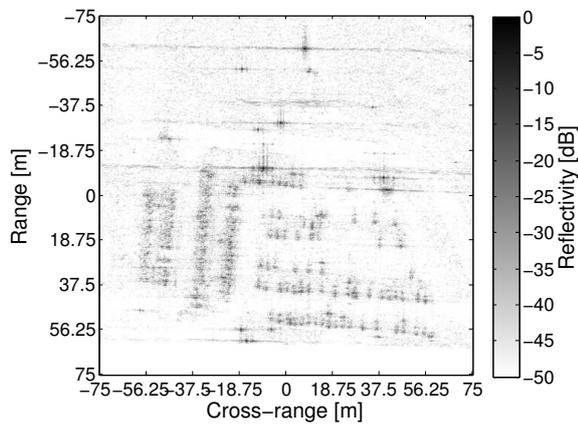
Figure 5: Gotcha parking space [5] processed by the global Backprojection algorithm. The reason of the coarse spacial resolution is the short $2°$ circular aperture containing only 234 pluses. The flight path in **Figure 6** is computed from the reflectivity function seen in this image.

3) Real data with narrow beam:
The third example demonstrates the results of the coarse reconstruction applied to the Gotcha-Public Released Dataset [5]. This dataset provides an X-band circular SAR system with $640\,\text{MHz}$ bandwidth, $45°$ depression angle and a $150\,\text{m} \times 150\,\text{m}$ spot, see **Figure 5**. Here, the azimuth beam angle θ_{az} and the evaluation beam angle θ_{ev} are approximately $1°$. The spot is about $r_0 = 10\,000\,\text{m}$ away from the flight path. In this dataset the true flight path is available, so that measurement inaccuracies of the INS unit are created artificially. We choose the initial solution γ_0 to be a straight flight path, see **Figure 6**.

The reconstructed range and altitude component of the real circular trajectory are shown in **Figure 6**. Like in example two, not the real flight path is reconstructed, but the processing of the range compressed data with the new flight path would yield to a well focused image.
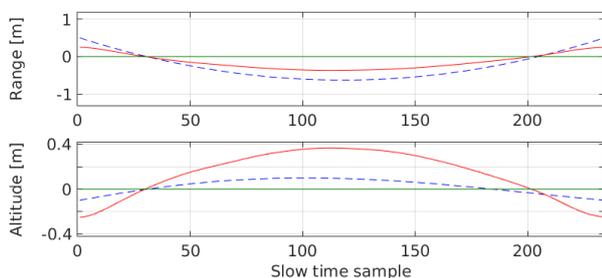

Figure 6: Range and altitude of the flight path of the Gotcha dataset [5]. Starting the proposed algorithm with the green line ends up to the red line, which is the projection of the true flight path, drawn by the blue line, onto the evaluation plane.

5 Conclusion

An algorithm to reconstruct the actual flight paths from range compressed data and focused SAR images is presented in this paper. In general, the proposed method can be used to correct the measurement errors of an inertial navigation system as a post processing step after the SAR image has been processed and focused by an autofocus method.

We compute the real flight path iteratively by a nonlinear Tikhonov regularization. Our algorithm does not use a prominent point processing or a sharpness function. The coarse part of this algorithm can handle deviations of more than $1\,\text{m}$. The fine part uses the phase information to correct the flight path within half a wavelength.

Our numerical examples show that in the case of a wide antenna beam pattern greater than approximately $20°$ the path of the antenna can be reconstructed precisely. If the beam is narrower, only an 2D version of the path is computed. However, this 2D path is a good approximation, because using this path in the Backprojection algorithm yields well focused images.

References

[1] M. Cheney, B. Borden, *Fundamentals of Radar Imaging*, SIAM, Philadelphia, 2009.

[2] I. A. Cumming, F. H. Wong, *Digital Processing of Synthetic Aperture Radar Data - Algorithms and Implementation*, 1-th edition, Artech House, Boston, 2005.

[3] A. N. Tikhonov, *Solution of incorrectly formulated problems and the regularization method*, Soviet Math. Dokl., vol. 5, pp. 1035-1038, 1963.

[4] S. Stanko et al., *SAR with MIRANDA - Millimeterwave Radar using Analog and Digital Approach*, EuRAD, pp. 214-217, Manchester, 2011.

[5] C. Casteel, L. Gorham, M. Minardi, S. Scarborough, K. Naidu, U. Majumber, *A challenge problem for 2-D/3-D imaging of targets from a volumetric data set in an urban environment*, Proc. SPIE - Algorithms Synthetic Aperture Radar Imagery XIV, vol. 6568, p. 65680D, 2007.

[6] H. W. Engl, M. Hanke, A. Neubauer, *Regularization of Inverse Problems*, Mathematics and Its Applications, vol. 375, Kluwer Academic Publishers, 1996.

[7] D. Behmardi E. D. Nayeri, *Introduction of Fréchet and Gâteaux Derivative*, Applied Mathematical Science, Vol. 2, no. 20, pp. 975 - 980, 2008.