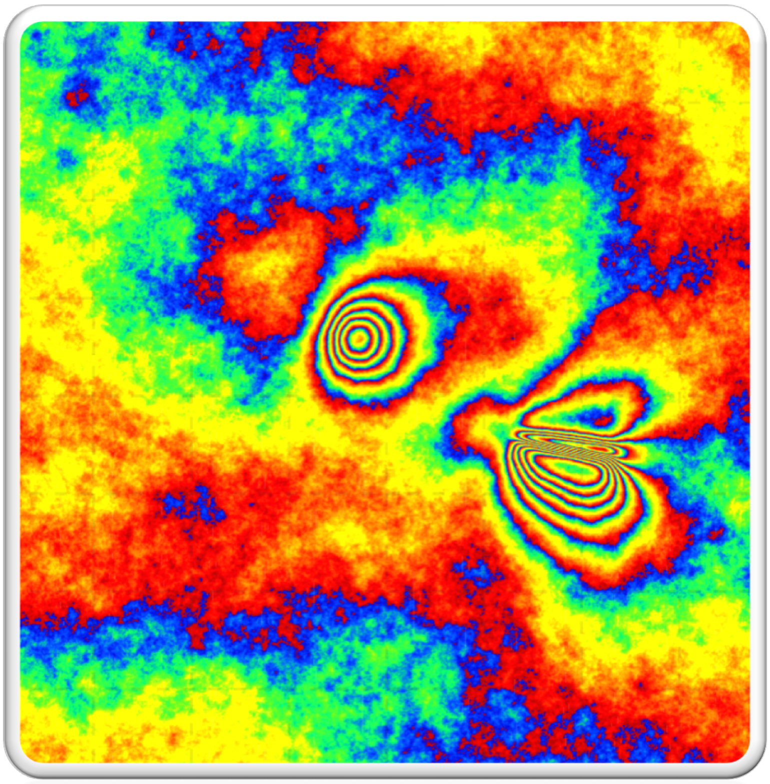
CENTRE FOR OBSERVATION & MODELLING OF EARTHQUAKES, VOLCANOES & TECTONICS – UNIVERSITY OF LEEDS

GBIS

Geodetic Bayesian Inversion Software

– Version 1.1 –

USER MANUAL



Author: Marco Bagnardi
8 August, 2018

COMET

UNIVERSITY OF LEEDS

Contents

1. INTRODUCTION **3**

2. INSTALLATION **4**

3. DATA PREPARATION **5**
3.1 GPS Data 5
3.2 InSAR Data 6

4. INPUT FILE **7**
4.1 Reference point and area of interest 7
4.2 InSAR data 7
4.3 GPS data 8
4.4 Model parameters 8
4.4.1 Point source 9
4.4.2 Finite spherical source 9
4.4.3 Prolate spheroid source 10
4.4.4 Penny-shaped sill-like source 10
4.4.5 Dipping dike with uniform opening 10
4.4.6 Horizontal rectangular sill with uniform opening 11
4.4.7 Dipping fault with uniform slip 11

5. DATA ERRORS **14**
5.1 InSAR data semi-variogram 14

6. INVERSION **16**
6.1 Starting the inversion 16
6.2 Initial plots and data check 17
6.3 During the inversion 17

7. FINAL REPORT **18**
7.1 Inversion summary 18
7.2 Convergence plots (Figure 1) 18
7.3 Posterior density functions plot (Figure 2) 18
7.4 Joint probabilities plot (optional) 18
7.5 Data, model, and residual comparison (optional) 18

8. OUTPUT FILE 20

References **21**

Appendix A: Exercise **22**

1. INTRODUCTION

The open-source ***Geodetic Bayesian Inversion Software*** (***GBIS***, Version 1.1) allows the user to perform the inversion of Interferometric Synthetic Aperture Radar (InSAR) and/or Global Positioning System (GPS) data to estimate deformation source parameters. The inversion software uses a Markov-chain Monte Carlo algorithm, incorporating the Metropolis-Hastings algorithm [e.g., *Hastings*, 1970; *Mosegaard and Tarantola*, 1995], to find the posterior probability distribution of the different source parameters. A detailed explanation of the inversion approach is provided in *Bagnardi and Hooper* [2018].

The current version of *GBIS* includes analytical forward models for magmatic sources of different geometry (e.g., point source [*Mogi*, 1958], finite spherical source [*McTigue*, 1987], prolate spheroid source [*Yang et al*., 1988], penny-shaped sill-like source [*Fialko et al*., 2001], and dipping dike with uniform opening [*Okada*, 1985]) and for dipping faults with uniform slip [*Okada*, 1985], embedded in a isotropic elastic half-space. However, the software architecture allows the user to easily add any other analytical or numerical forward models to calculate displacements at the surface.

GBIS is written in *Matlab*® and uses a series of third-party functions to perform specific steps for data pre-processing, subsampling, and includes forward models from the *dMODELS* software package [*Battaglia et al.*, 2013].

The software offers a series of pre- and post-processing tools aimed at estimating errors in InSAR data, and at graphically displaying the inversion results.

GBIS is made available for non-commercial applications only and can be downloaded from `http://comet.nerc.ac.uk/gbis`

A Google discussion group is available at:
`https://groups.google.com/forum/#!forum/gbisinfo`

When using this software please reference *Bagnardi and Hooper* [2018]:

Bagnardi M. & Hooper A, (2018). Inversion of surface deformation data for rapid estimates of source parameters and uncertainties: A Bayesian approach. *Geochemistry, Geophysics, Geosystems*, 19. https://doi.org/10.1029/2018GC007585

Throughout this manual, commands to be entered on the *Matlab* command line are in `blue` and preceded by the symbol `>>`, screen outputs are in `grey`. Specific portions of the command that require modification by the user are in `red`.

Typing “help” followed by the name of the function (for example, `>> help GBISrun`) provides a brief description of the function and of the input/output parameters.

COMET UNIVERSITY OF LEEDS

2. INSTALLATION

After downloading the *GBIS* compressed file (`GBIS_V1.1.zip`), un-compress it in your preferred location.

Once uncompressed, you should find the following directories and files:

```
/Example
/GBIS
GBIS_User_manual_V1.1.pdf
```

After opening *Matlab*, add the `GBIS/` directory and subdirectories to your path:

```
>> addpath(genpath('/path/to/your/local/directory/GBIS/'))
```

You can test if the path is set correctly by using the following function:

```
>> testGBISPath
```

If the path is set correctly, the following message should appear on the screen:

```
/ yourpath /GBIS/GBISrun.m
/ yourpath /GBIS/PrepareData/loadInsarData.m
/ yourpath /GBIS/DeformationSources/mogi/mogi.m
/ yourpath /GBIS/PlotData/plotInsarWrapped.m
/ yourpath /GBIS/PrepareReport/generateFinalReport.m
/ yourpath /GBIS/RunInversion/runInversion.m
/ yourpath /GBIS/ThirdPartyScripts/local2llh.m
/ yourpath /GBIS/Variogram/fitVariogram.m

If you see this message, you are ready to go!
```

You are now ready to use *GBIS* with your own data or with the provided synthetic dataset.

Note: *GBIS V1.1* was developed using *Matlab 2015b*.

3. DATA PREPARATION

GBIS requires data to be formatted following specific structures. We do not provide scripts to convert from other formats to the *GBIS* format since they are specific to the software used for data processing.

3.1 GPS DATA

GPS data must be provided in a single ASCII text file (for example, `GPS_Data.txt`). The file must have 8 columns separated by a single space, and each row must correspond to a single GPS site. The 8 columns are:

1- Longitude in decimal degrees, positive if East and negative if West.
Example: `-155.04005` for longitude 155°.04005 West.

2- Latitude in decimal degrees, positive if North and negative if South.
Example: `19.41875` for latitude 19°.41875 North.

3- Horizontal displacement in North-South direction in millimetres, positive if North and negative if South.
Example: `10.18` for 10.18 mm towards the North.

4- Estimated error in North-South horizontal displacement in millimetres, normally expressed as one standard deviation.
Example: `0.35` for 0.35 mm.

5- Horizontal displacement in East-West direction in millimetres, positive if East and negative if West.
Example: `5.86` for 5.86 mm towards the East.

6- Estimated error in East-West horizontal displacement in millimetres, normally expressed as one standard deviation.
Example: `0.35` for 0.35 mm.

7- Vertical displacement in millimetres, positive if upward and negative if downward.
Example: `1.66` for 1.66 mm upward.

8- Estimated error in vertical displacement in millimetres, normally expressed as one standard deviation.
Example: `0.35` for 0.35 mm.

Example, 3 GPS stations:

```
-155.04005 19.41875 10.18 0.35 5.86 0.35 1.66 0.35
-155.22762 19.43737 -17.36 0.22 -21.36 0.22 -13.74 0.22
-155.21762 19.39626 20.65 0.67 -37.41 0.67 -15.53 0.67
```

3.2 InSAR DATA

InSAR data must be provided as *Matlab* `*.mat` files (for example, `CSK20160614-20160625asc.mat`), one for each InSAR dataset. The name of the file will be displayed in plots and printed to the screen. Meaningful names should therefore be preferred (e.g., sensor, dates, orbit, etc.).
The files must contain 5 Nx1 vectors with single or double precision, where N is the number of data points. <u>Remove all "0" and "NaN" values.</u> The 5 vectors must be named as follows and contain the relative value for each data point:

> `Lon`
> Longitude in decimal degrees, positive if East and negative if West.
> Example: `-155.04005` for longitude 155°.04005 West.
>
> `Lat`
> Latitude in decimal degrees, positive if North and negative if South.
> Example: `19.41875` for latitude 19°.41875 North.
>
> `Phase`
> Surface displacement expressed as interferometric phase change in radians. <u>Important</u>: phase change is positive when away from the satellite and negative when towards the satellite!).
> Example: `13.2860` for 13.2860 radians towards the satellite (if SAR sensor is C-band, wavelength 0.056 m, this would represent -0.0592 m of line-of-sight displacement, which is negative when away from the satellite and positive when towards the satellite).
>
> `Inc`
> Line-of-sight incidence angle in decimal degrees.
> Example: `23.00` for 23°.
>
> `Heading`
> Satellite heading angle in decimal degrees.
> Example: `-11.00` for -11° or 349° (ascending orbit).

To improve the speed of the inversion, the visualization of data, and to minimise the file size, it is important to keep the size of the InSAR data vectors (`Lon, Lat, Phase, Inc`, and `Heading`) to less than 1,000,000 data points. InSAR data are largely subsampled before the inversion and it is therefore unnecessary to have very large raw datasets. The data vector reduction can be obtained through regular subsampling of the raw data. For example, use:

```
>> Phase = Phase(1:5:end);
```

to retain one data point every five raw data points.

<u>Important</u>: make sure all vectors have the same length (Nx1, where N is the number of data points).

4. INPUT FILE

GBIS reads a set of information relative to data and model parameters from an ASCII text file referred to as the *input file* (e.g., `VolcanoExercise.inp`). The parameters included in the input file do not usually vary much between different inversions for a given set of data and models.

Important: parameter names should not be changed since these are used by the software to name variables and structures. Here follows a description of each required parameter that must be included in the input file.

4.1 Reference point and area of interest:

```
geo.referencePoint = [-155.23; 19.38];
```
Provides the reference point for the local Cartesian coordinate system (Longitude and Latitude are in decimal degrees).

```
geo.boundingBox = [-156.00; 20.00; -154.00; 18.00;];
```
Provides the coordinates (Longitude and Latitude in decimal degrees) of the upper left and lower right corners of a rectangular area of interest. This can be smaller than the extent of the entire dataset or larger if all data points should be used.

4.2 InSAR data:

For each InSAR dataset the following parameters must be provided:

```
insarID = 1;
```
Assigns an ID number to each InSAR dataset and is used when calling the function *GBISrun* (see Paragraph 6). This number should be unique to each dataset.

```
insar{insarID}.dataPath = '/Path/to/data/file/S1A20160615-20160627dsc.mat';
```
Provides the full path to the InSAR dataset `*.mat` file.

```
insar{insarID}.wavelength = 0.056;
```
Provides the sensor wavelength in meters used to convert phase change into line-of-sight displacement (e.g., 0.056 m for C-band sensors, 0.031 m for X-band sensors, and 0.023 m for L-band sensors).

```
insar{insarID}.constOffset = 'y';
```
Flag to indicate if a constant offset in the line-of-sight direction should be estimated during the inversion. Set it to `'y'` to estimate it, to `'n'` to not estimate it.

```
insar{insarID}.rampFlag = 'y';
```
Flag to indicate if a linear ramp (plane) across the entire dataset should be estimated during the inversion to simulate an orbital residual. Set it to `'y'` to estimate it, to `'n'` to not estimate it.

```
insar{insarID}.sillExp = 6.7e-06;
```
Sill value in meters of the exponential fit to the experimental variogram (see Paragraph 5.1).

```
insar{insarID}.range = 11700;
```
Range value in meters of the exponential fit to the experimental variogram (see Paragraph 5.1).

```
insar{insarID}.nugget = 1.6e-08;
```
Nugget value in meters of the exponential fit to the experimental variogram (see Paragraph 5.1).

```
insar{insarID}.quadtreeThresh = 0.008^2;
```
Threshold variance in meters used for the Quadtree subsampling. See *Jónsson et al.* [2002] for more information on the meaning of this parameter.

4.3 GPS data:

```
gps.dataPath = '/Path/to/data/file/GPS_data.txt';
```
Provides the full path to the GPS dataset ASCII `*.txt` file.

```
gps.weight = 1/1;
```
Weight of GPS data vs. InSAR data in joint inversion. All data is overall "weighted" by the associated data error expressed in the variance-covariance matrix and this parameter should be normally set to 1/1. However, there may be some cases where an even greater weight is wished for the GPS data. This can be achieved by changing the ratio from 1/1 to, for example, 10/1 (10 for GPS, 1 for InSAR). The weighting factor is considered as follows:

$$\text{Misfit function}_{GPS} = (Gm - d)^T * \Sigma_d^{-1} * (Gm - d) * W$$
$$\text{Misfit function}_{InSAR} = (Gm - d)^T * \Sigma_d^{-1} * (Gm - d) * 1$$

where d is the vector of measurements, G is a matrix of Green's functions mapping the effect of each deformation source (m is the set of source parameters) to displacements, Σ_d^{-1} is the inverse of variance-covariance matrix for the measurements, and W is the weighting factor.

4.4 Model parameters:

```
modelInput.nu = 0.25;
```
Poisson's ratio of the elastic half-space.

For each forward model (here defined by the generic substructure '`modeltype`'; make sure to use the appropriate model name in the input file!), there are four sets of parameters to set:

```
modelInput.modeltype.lower = [ m ];
```
For each model parameter, this is the lower bound of the prior uniform distribution (e.g., minimum realistic value that the parameter can assume).

`modelInput.modeltype.upper = [ m ];`
For each model parameter, this is the upper bound of the prior uniform distribution (e.g., maximum realistic value that the parameter can assume).

`modelInput.modeltype.start = [ m ];`
For each model parameter, this is the starting value from which to start the inversion. This becomes important if the initial Simulated Annealing phase is switched off (see Paragraph 6, `skipSimulatedAnnealing`). These values must fall within the upper and lower bound.

`modelInput.modeltype.step = [ m ];`
For each model parameter, this value sets the initial step-size in the random walk during the inversion. However, this value is adjusted during the inversion to achieve the optimal rejection rate (~ 77%).

In the current version of *GBIS*, these are the possible forward models and associated model parameters `m`:

4.4.1 Point source [Mogi, 1958]:
`modeltype: mogi`
`m: [X-center; Y-center; Z-center; DV;];`

where `X-center` = X-coordinate of point source in meters;
`Y-center` = Y-coordinate of point source in meters;
`Z-center` = depth of point source in meters (positive downward);
`DV` = volume change in cubic meters.

Example:
```
modelInput.mogi.lower = [-8000; -2000;   100; -1e9; ];
modelInput.mogi.upper = [ 8000;  9000; 10000; -1e2; ];
modelInput.mogi.start = [    0;     0;  3000; -1e6; ];
modelInput.mogi.step =  [  100;   100;   100; 1e04; ];
```

4.4.2 Finite spherical source [McTigue, 1987]:
`modeltype: mctigue`
`m: [X-center; Y-center; Z-center; radius; DP/mu;];`

where `X-center` = X-coordinate of source center in meters;
`Y-center` = Y-coordinate of source center in meters;
`Z-center` = depth of source center in meters (positive downward);
`radius` = radius of the sphere in meters;
`DP/mu` = dimensionless excess pressure (pressure change/shear modulus);

4.4.3 Prolate spheroid source [Yang et al., 1988]

`modeltype: yang`

`m = [X-center; Y-center; Z-center; radius; A; B/A; Strike; Plunge; DP/mu;];`

where

`X-center =`		X-coordinate of source center in meters;
`Y-center =`		Y-coordinate of source center in meters;
`Z-center =`		depth of source center in meters (positive downward);
`A =`		length of major semi-axis in meters;
`B/A =`		dimensionless aspect ratio between semi-axes (minor/major); the ratio must be between 0 and 1 (1 = spherical);
`Strike =`		strike angle of major semi-axis with respect to North (0°/360° = N; 90°=E; 180°=S; 270°=W) in degrees;
`Plunge =`		inclination angle of major semi-axis with respect to horizontal (0° = horizontal; 90° = vertical) in degrees;
`DP/mu =`		dimensionless excess pressure (pressure change/shear modulus);

4.4.4 Penny-shaped sill-like source [Fialko et al., 2001]

`modeltype: penny`

`m = [X-center; Y-center; Z-center; radius; DP/mu;];`

where

`X-center =`		X-coordinate of source center in meters;
`Y-center =`		Y-coordinate of source center in meters;
`Z-center =`		depth of source center in meters (positive downward);
`radius =`		radius of the source in meters;
`DP/mu =`		dimensionless excess pressure (pressure change/shear modulus);

4.4.5 Dipping dike with uniform opening [Okada, 1985]

`modeltype: dike`

`m: [Length; Width; Z-center; Dip; Strike; X-center; Y-center; Opening;]`

where

`Length =`		first dimension (horizontal edge) of rectangular source in meters;
`Width =`		second dimension of rectangular source in meters;
`Z-center =`		depth of one edge of rectangular source in meters (positive downwards);
`Dip =`		dip angle with respect to horizontal (0° = horizontal; -90° or 90° = vertical) in degrees;
`Strike =`		strike angle of horizontal edge with respect to North (0°/360° = N; 90°=E; 180°=S; 270°=W) in degrees;
`X-center =`		X-coordinate of horizontal edge mid-point in meters;
`Y-center =`		Y-coordinate of horizontal edge mid-point in meters;

`Opening =` opening of dislocation plane (e.g., dike thickness) in meters;

4.4.6 Horizontal rectangular sill with uniform opening [Okada, 1985]

`modeltype: sill`
`m: [Length; Width; Z-center; Strike; X-center; Y-center; Opening;]`

where parameters are the same as in `dike` but where `Dip` is set to 0 degrees.

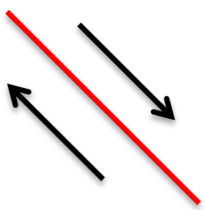
4.4.7 Dipping fault with uniform slip [Okada, 1985]

`modeltype: fault`
`m: [Length; Width; Z-center; Dip; Strike; X-center; Y-center; SS; DS]`

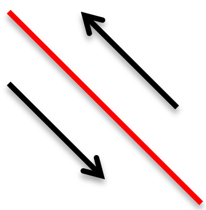
where	`Length =`	first dimension (horizontal edge) of rectangular source in meters;
	`Width =`	second dimension of rectangular source in meters;
	`Z-center =`	depth of one edge of rectangular source in meters (positive downwards);
	`Dip =`	dip angle with respect to horizontal (0° = horizontal; -90° or 90° = vertical) in degrees;
	`Strike =`	strike angle of horizontal edge with respect to North (0°/360° = N; 90°=E; 180°=S; 270°=W) in degrees;
	`X-center =`	X-coordinate of horizontal edge mid-point in meters;
	`Y-center =`	Y-coordinate of horizontal edge mid-point in meters;
	`SS =`	strike-slip component of total slip in meters;
	`DS =`	dip-slip component of total slip in meters;

<u>For all rectangular dislocation sources, the sign convention is:</u>

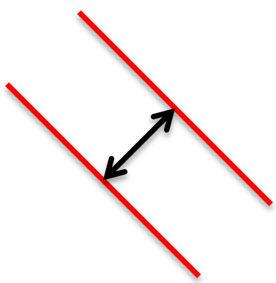
Positive slip



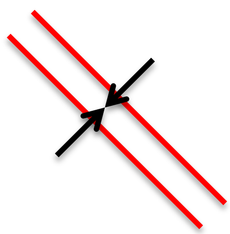
Negative slip



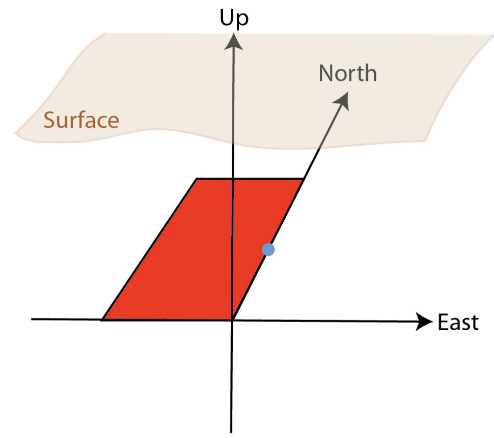
Positive opening



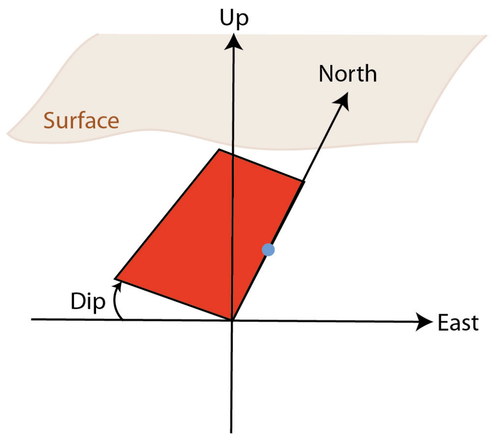
Negative opening (closure/contraction)



In the following cartoon (modified after Fault Mechanics Group, Stanford University, *https://pangea.stanford.edu/*), the blue dot marks the mid-point of one (along-strike) edge of the rectangular plane (identified by `X-center` and `Y-center`).

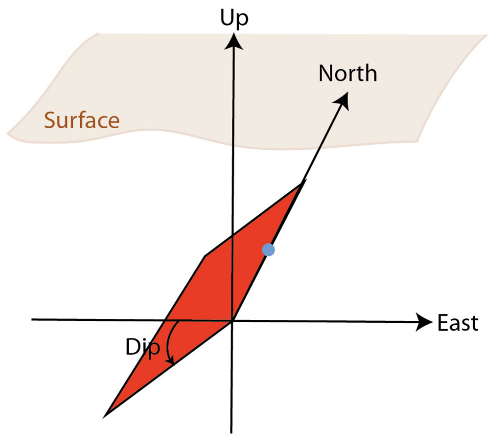


Fault/dike striking North-South (`Strike = 0`) and horizontal (δ, `Dip = 0`).



The same fault/dike, now with positive dip (`Dip = 30`); geologically speaking, this plane would be described as dipping to the East.

Note that with the positive dip, the blue dot now marks the mid-point of the <u>lower</u> edge of the plane.



The same fault/dike has now negative dip (`Dip = -30`); geologically speaking, this plane would be described as dipping to the West.

Note that with the negative dip, the blue dot now marks the mid-point of the <u>top</u> edge of the plane.

The possibility to specify the coordinates of either the top edge mid-point or the lower edge mid-point is useful for the inversion of the fault geometry. In this way you can prevent the inversion algorithm from returning models that extend above or below a certain point. For example, you can invert for the width of the fault while preventing it from becoming shallower than a certain depth that you choose (e.g., the surface).

5. DATA ERRORS

GBIS considers errors in the data to build the variance-covariance matrix for all data points.

For GPS data, the algorithm assumes no spatial correlation of the errors (spatial covariance = 0) between stations, and only considers the variance of each displacement component.

5.1 InSAR data semi-variogram

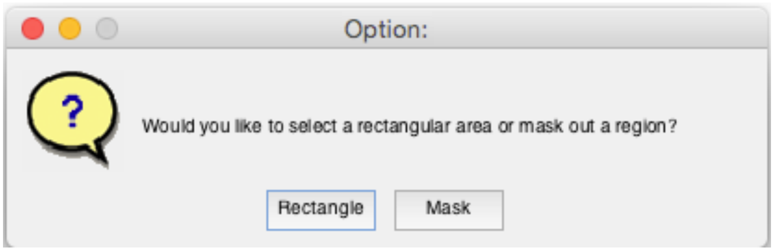
For InSAR data, the variance-covariance matrix is built assuming that errors in the data can be simulated using an exponential function with nugget, fitted to the isotropic experimental semi-variogram [e.g., *Webster and Oliver*, 2007].

To calculate the experimental semi-variogram and the exponential fit you can use the following function:

```
>> fitVariogram('/Path/to/data/file/InSARData.mat',
wavelength)
```

where you must specify the `*.mat` file containing InSAR data and the satellite radar wavelength in meters (e.g., `0.056` for C-band sensors).

The function will display the selected InSAR dataset wrapped at the specified wavelength and ask if you would like to select a rectangular area or mask out a region.



In the first case (`Rectangle`), you can select a rectangular region by clicking and dragging with the mouse over the displayed InSAR image. This area will be used to calculate the experimental semi-variogram and therefore should not include any actively deforming area and be larger than the maximum distance at which noise may be spatially correlated (e.g., > 20 km).

In the second case (`Mask`), by clicking with the mouse over the displayed InSAR image, you can draw a polygon that includes the area to be masked out (e.g., area showing surface deformation). To close the polygon, click on the first vertex. The experimental semi-variogram will be then calculated over the entire image, excluding the masked region.

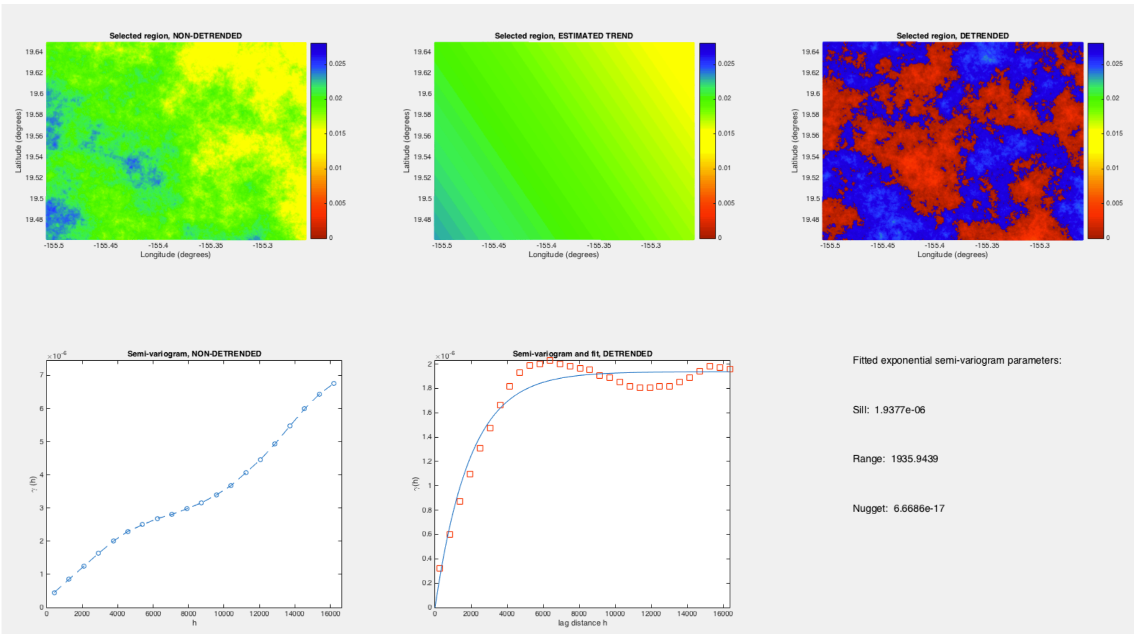
In both cases, the function will first remove a linear ramp from the data, will then estimate the experimental semi-variogram on the de-trended dataset, and finally fit an exponential function with nugget.

The function generates a second image in which are shown:

- the original dataset (the rectangular area or the masked full image);
- the estimated linear ramp, removed from the data;
- the de-trended dataset;
- the experimental semi-variogram calculated on the original dataset;
- the experimental semi-variogram calculated on the de-trended dataset (red squares) and the exponential fit (blue line);
- the parameters of the best-fitting exponential function with nugget (`Sill`, `Range`, `Nugget`). These parameters are also printed in the *Matlab* Command Window and should be copied and pasted in the input file for each InSAR dataset.

For example:

```
insar{insarID}.sillExp = 1.94e-06;
insar{insarID}.range = 1936;
insar{insarID}.nugget = 7e-17;
```



It is recommended to repeat this step multiple times to fully explore the dataset and best characterise the spatially correlated noise in the data.

6. INVERSION

6.1 Starting the inversion

To run the inversion, use the `GBISrun` function as follows:

```
>> GBISrun(inputFileName, insarDataCode, gpsDataFlag,
modelCode, nRuns, skipSimulatedAnnealing)
```

where:

- `inputFileName` is the name and extension of the input file between single quotations (for example, `'VolcanoExercise.inp'`);

- `insarDataCode` contains the ID number of the InSAR data to use (for example, `[1,3]` to use InSAR data with `insarID = 1` and `3`). Leave empty (e.g., `[]`) if no InSAR data should be used in the inversion;

- `gpsDataFlag` is a flag indicating if GPS data should be used in the inversion. Set it to `'y'` to use GPS data or to `'n'` if GPS data should not be used.

- `modelCode` indicates which forward models should be used. The current version of *GBIS* offers the following models:

  `'M'` for *Mogi* source (point source, [*Mogi*, 1958]);
  `'T'` for *McTigue* source (finite sphere, [*McTigue*, 1987]);
  `'Y'` for *Yang* source (prolate spheroid, [*Yang et al.*, 1988]);
  `'S'` for rectangular horizontal sill with uniform opening [*Okada*, 1985];
  `'P'` for penny-shaped crack [*Fialko et al.*, 2001];
  `'D'` for rectangular dipping dike with uniform opening [*Okada*, 1985];
  `'F'` for rectangular dipping fault with uniform slip [*Okada*, 1985];

  Any combination of the above mentioned sources is possible (for example, `'MD'` to invert for a *Mogi* source and a dipping dike);

- `nRuns` is the number of iterations (samples) to be performed (for example, `1e06`);

- `skipSimulatedAnnealing` is a flag indicating if the initial phase during which Simulated Annealing is used to estimate the starting model parameters should be skipped (e.g., `'y'`) or not (e.g., `'n'`).

For example:

```
>> GBISrun('VolcanoExercise.inp',[1,2],'y','MD',1e06, 'n')
```

6.2 Initial plots and data check

Before performing the inversion, *GBIS* generates several plots and prints useful information in the *Matlab Command Window* to check whether the data is formatted correctly.

Plots:

1. For each InSAR dataset:
   – wrapped InSAR data;
   – unwrapped InSAR data;
   – subsampled data using *Quadtree*.

2. GPS horizontal displacements.

Matlab Command Window:

1. Name of the output directory;
2. Full path and name of the InSAR data `*.mat` files used in the inversion;
3. Mean heading angle for the InSAR dataset;
4. Mean incidence angle for the InSAR dataset;
5. Maximum and minimum line-of-sight displacements in the InSAR dataset;
6. Full path and name of the GPS data `*.txt` file used in the inversion;
7. Number of GPS sites found in the `*.txt` file.

If everything is correctly formatted and the data looks as expected, the user will see the following message and must press a key to continue:

```
#################    Press any key to continue ...
```

6.3 During the inversion

Throughout the duration of the inversion some information is printed to the screen in the *Matlab Command Window*:
- forward models in use;
- total number of iterations performed (`model trials`);
- current optimal probability (this number should decrease throughout the inversion);
- number of model trials rejected and rejection rate. The rejection rate should be ~77% if the inversion is properly functioning, if much higher or lower there may be something not set correctly;
- optimal model parameters after the current number of iterations. These values can be used to check the progress of the inversion.

During the inversion, *GBIS* generates and constantly updates the `temporary.mat` file in the output directory, which contains the inversion results up to that point. This file is deleted once the inversion is completed and substituted by the `invert_****.mat` file.

7. FINAL REPORT

Once the inversion is completed, *GBIS* offers the opportunity to display the inversion results and to generate an HTML report (e.g., `report_****.html`) with links to figures (saved in the `Figures` directory) and tables.
To produce the final report the user can use the following function:

```
>> generateFinalReport('invert_****.mat', burn-in)
```

where:

- `invert_****.mat` is the name of the file containing the inversion results;

- `burn-in` represents the number of initial iterations to be excluded from the computation of the posterior density functions and other statistics (e.g., mean, median, confidence interval). It is recommended to set this to 1 when generating the report for the first time and set it to a higher value once the "burn-in" period (number of iterations to reach convergence) is identified.

Note: The command should be executed from the directory containing the `invert_****.mat` file. It cannot handle an absolute path.

7.1 Inversion summary

The `generateFinalReport` function creates a `summary_****.txt` file containing the statistics calculated for each model parameter, and also prints the results in the *Matlab Command Window.*

7.2 Convergence plots (Figure 1)

This figure shows the retained value of each model parameter and its evolution throughout the inversion (the x-axis represents the number of iterations). This plot can be used to check if the inversion converged and to estimate the "burn-in" period.

7.3 Posterior density functions plot (Figure 2)

Histograms showing the posterior density functions for all source model parameters (blue bars) are shown together with the optimal values (red lines). For this plot it is important to set an appropriate "burning" period.

7.4 Joint probabilities plot (optional)

An optional plot will display the joint posterior probability distribution between pairs of model parameters. This plot is still under development and will be upgraded in future releases.

7.5 Data, model, and residual comparison (optional)

For each InSAR dataset and for the GPS horizontal displacements, *GBIS* can generate comparative plots showing the original data, the predicted displacements using the optimal solution from the inversion, and the difference

between the two (in the case of InSAR data only). InSAR data are displayed as both wrapped and unwrapped displacements.
Note that plotting of InSAR data may require several minutes.

8. OUTPUT FILE

All input parameters and inversion results are saved in the output `invert_****.mat` file. The information is saved in structures:

`geo`
Contains geographic information such as the reference point of the local Cartesian coordinate system and the limits of the area of interest.

`gps`
Contains the path and name of the GPS data file, the GPS data, variance, and inverse of the variance-covariance matrix

`inputFile`
Contains the name of the input file.

`insar`
For each InSAR dataset, contains the input parameters from the input file, the subsampled data vectors and the inverse of the variance-covariance matrix.

`insarDataCode`
ID numbers of the InSAR datasets used in the inversion.

`invpar`
Contains input parameters for the inversion, such as the number of iterations, the temperature schedule (Simulated Annealing), etc.

`invResults`
Contains the results from each iteration for the model parameters (`invResults.mKeep`) and for the likelihood function (`invResults.PKeep`). It also contains the model parameters of the optimal model (`invResults.optimalmodel`).

`model`
Contains the input model values (e.g., upper and lower bounds, start model, step size).

`modelInput`
Contains all information read from the input file (all models). It will be removed in future releases.

`nObs`
Total number of observation points (subsampled InSAR + GPS).

`saveName`
Name of the output file.

References

Bagnardi M. & Hooper A, (2018). Inversion of surface deformation data for rapid estimates of source parameters and uncertainties: A Bayesian approach. *Geochemistry, Geophysics, Geosystems*, 19. https://doi.org/10.1029/2018GC007585

Battaglia, M., Cervelli, P. F., & Murray, J. R. (2013). dMODELS: A MATLAB software package for modeling crustal deformation near active faults and volcanic centers. *Journal of Volcanology and Geothermal Research*, *254*, 1-4.

Fialko, Y., Khazan, Y., & Simons, M. (2001). Deformation due to a pressurized horizontal circular crack in an elastic half-space, with applications to volcano geodesy. *Geophysical Journal International*, *146*(1), 181-190.

Hastings, W. K. (1970). Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, *57*(1), 97-109.

Jónsson, S., Zebker, H., Segall, P., & Amelung, F. (2002). Fault slip distribution of the 1999 Mw 7.1 Hector Mine, California, earthquake, estimated from satellite radar and GPS measurements. *Bulletin of the Seismological Society of America*, *92*(4), 1377-1389.

McTigue, D. F. (1987). Elastic stress and deformation near a finite spherical magma body: resolution of the point source paradox. *Journal of Geophysical Research: Solid Earth*, *92*(B12), 12931-12940.

Mogi, K. (1958). Relations between the eruptions of various volcanoes and the deformations of the ground surfaces around them.

Mosegaard, K., & Tarantola, A. (1995). Monte Carlo sampling of solutions to inverse problems. *Journal of Geophysical Research: Solid Earth*, *100*(B7), 12431-12447.

Okada, Y. (1985). Surface deformation due to shear and tensile faults in a half-space. *Bulletin of the seismological society of America*, *75*(4), 1135-1154.

Yang, X. M., Davis, P. M., & Dieterich, J. H. (1988). Deformation from inflation of a dipping finite prolate spheroid in an elastic half-space as a model for volcanic stressing. *Journal of Geophysical Research: Solid Earth*, *93*(B5), 4249-4257.

Webster, R., & Oliver, M. A. (2007). *Geostatistics for environmental scientists*. John Wiley & Sons.

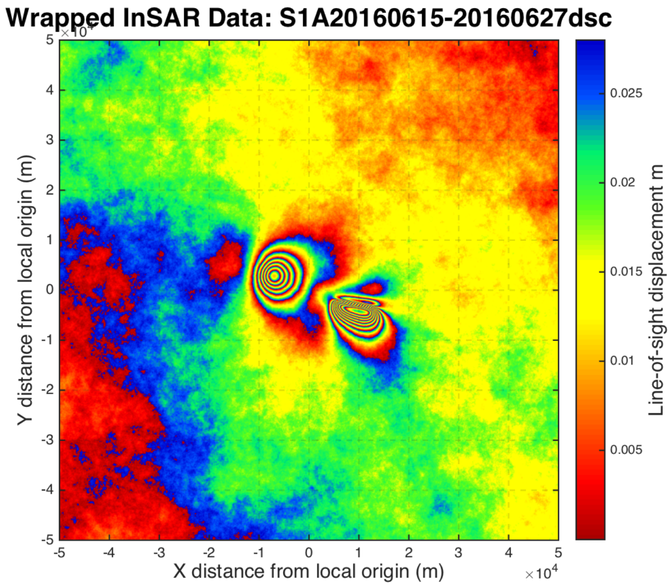
Appendix A: Exercise

Two InSAR and one GPS synthetic datasets are provided with the software for testing.

The InSAR datasets are:

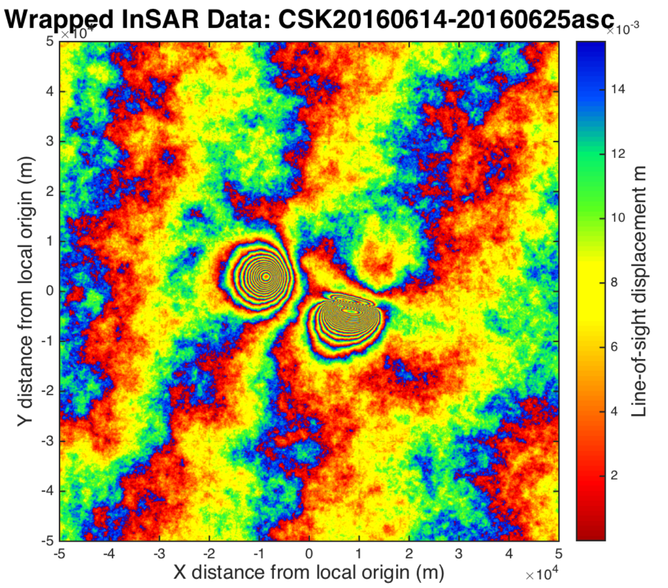
`Example/Data/S1A20160615-20160627dsc.mat`

simulating a descending Sentinel-1 interferogram (C-band)

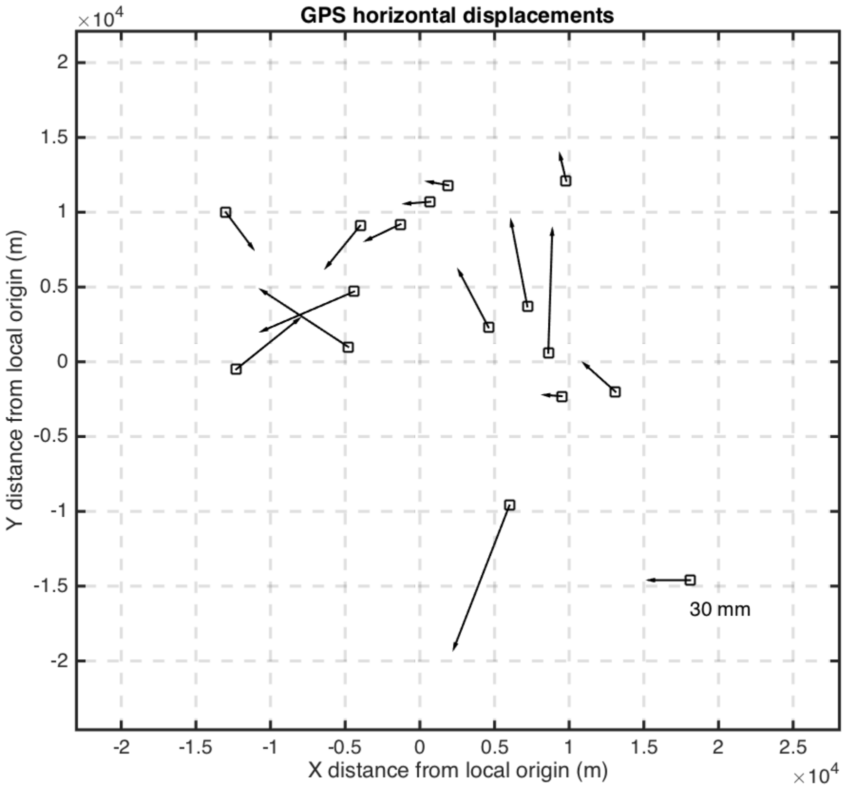


`Example/Data/ CSK20160614-20160625asc.mat`

simulating an ascending CosmoSkyMed interferogram (X-band)



The GPS dataset is composed of 15 measurement sites:
`Example/Data/ GPS_data.txt`



All datasets are generated using the same forward model simulating a deformation event at a given volcano.

The `VolcanoExercise.inp` input file provides a template/example that can be used to run the example.

You can now try to infer the sources causing the measured deformation by trying different combinations of data and models. The aim is to obtain the results reported in the following page (do not look at them until you think you are satisfied with your results!).

Synthetic forward model

Sources: finite spherical source + dipping dike (`'TD'`).

Finite spherical source model parameters (`'T'`):

```
X-center = -8000
Y-center = 3000
Z-center = 4000
radius = 1000
DP/mu = -45e06/10e09
```

Dipping dike model parameters (`'D'`):

```
Length = 8000
Width = 2500
Z-center = 1000
Dip = -65
Strike = 280
X-center = 9000
Y-center = -2500
Opening =  1.0
```

CSK InSAR data:

```
Constant offset = -0.023
Ramp-X = 4.5e-07
Ramp-Y = -1.8e-07
```

S1 InSAR data:

```
Constant offset = -0.01
Ramp-X = -1.5e-07
Ramp-Y = -1.2e-07
```