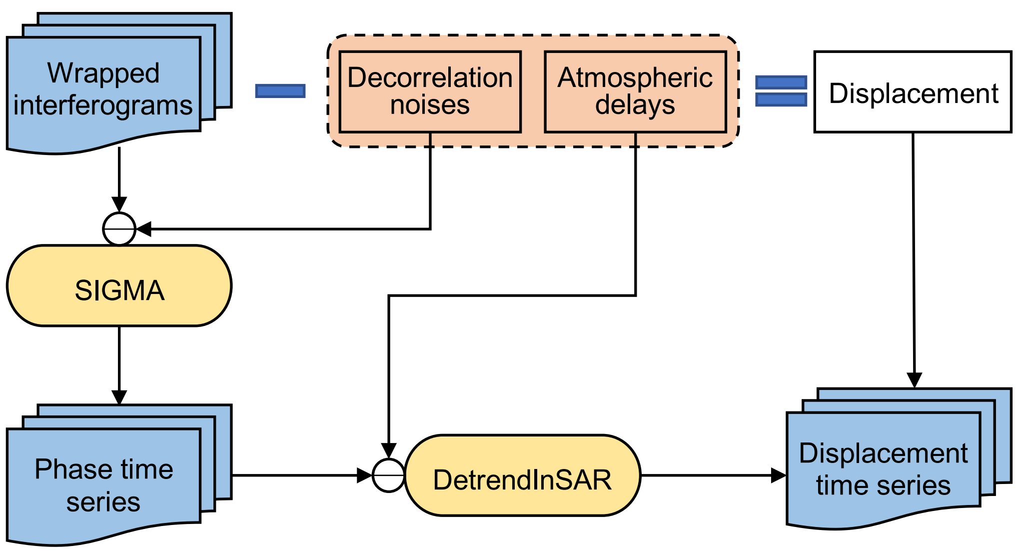
Manual for SIGMA & DetrendInSAR

Version 1.0



Jihong Liu, Jun Hu, Roland Bürgmann, Zhiwei Li, Sigurjón Jónsson

April 29th, 2024

中南大學
CENTRAL SOUTH UNIVERSITY

جامعة الملك عبدالله
للعلوم والتقنية
King Abdullah University of
Science and Technology

Berkeley
UNIVERSITY OF CALIFORNIA

Contents

Introduction 1

SIGMA Method 2

2.1 Installation 2

2.2 Input preparation 2

2.3 Run SIGMA 4

2.4 Output 5

DetrendInSAR Method 5

3.1 Installation 5

3.2 Input preparation 5

3.3 Run DetrendInSAR 6

3.4 Output 7

3.5 Main parameters 7

3.5.1 Select a polygon with significant deformations 7

3.5.2 The range for modeling trend and DEM-correlated components 8

3.5.3 The range for constraining temporal displacements 8

3.5.4 Downsample degree 9

3.5.5 Including GNSS data in DetrendInSAR model 9

Change history 10

3.1 Version 1.0 10

Chapter 1

Introduction

SIGMA, Strain-model based InSAR for Geo-hazards' Monitoring Approach, aims to decrease the decorrelation noise in DInSAR interograms based on strain model, and to obtain high-quality InSAR phase time series.

DetrendInSAR, DEcrease both the TRENd and Dem-correlated components in InSAR time series, aims to decrease the atmospheric delays and orbital errors in InSAR phase time series (SIGMA output) based on the spatiotemporal characteristics of different components, and to obtain high-quality InSAR displacement time series.

These two methods serve as the post-processing procedure for InSAR displacement measurement based on unfiltered DInSAR interferograms (generated using other popular software). All processes are conducted in Matlab software, and detailed introduction can be found in the following.

Note that these two methods can be used independently.

The code can be downloaded from [Zenodo](), and please reference Liu et al. (2021, 2023):

Liu, J., Hu, J., Bürgmann, R., Li, Z., Sun, Q., & Ma, Z. (2021). A Strain-Model Based InSAR Time Series Method and Its Application to The Geysers Geothermal Field, California. Journal of Geophysical Research: Solid Earth, 126(8), e2021JB021939. https://doi.org/10.1029/2021JB021939

Liu J., Hu J., Bürgmann R., Li Z., & Jónsson S. (2024). Mitigating Atmospheric Delays in InSAR Time Series: The DetrendInSAR Method and Its Validation. Journal of Geophysical Research: Solid Earth, 129, e2024JB028920. https://doi.org/10.1029/2024JB028920

If you have any query, please feel free to contact me (liujihong@csu.edu.cn, be sure introducing yourself in the email 😊).

Chapter 2

SIGMA Method

2.1 Installation

The SIGMA package is implemented in Matlab language, and we recommend MATLAB 2018a (or higher version) to operate this package. When using the SIGMA, a new project folder should be created to include the folder `SIGMAInSAR_code` and the main script `SIGMAInSAR_main.m`. The folder `SIGMAInSAR_code` includes the matlab functions of the SIGMA package, and the matlab script `SIGMAInSAR_main.m` outside the folder `SIGMAInSAR_code` is the main function to run the SIGMA method.

Users can directly run the main matlab script `SIGMAInSAR_main.m` to install this package, which is basically realized by using `addpath` function.

2.2 Input preparation

The input of SIGMA includes:

(1) Multi-looked unfiltered DInSAR interferogram files in radar coordination

Each file indicates an interferogram, i.e., a `row*col` sized complex double matrix; In general, these interferograms are with short spatiotemporal baselines to keep high coherence;

(2) The coherence file for each interferogram

A `row*col` double matrix;

(3) Multi-looked SAR amplitude files

The amplitude of multi-looked SLC images, which is used to mask extreme low coherence area;

(4) The short baseline file

`nifg*4`, `nifg` is the number of short-baseline interferograms, `4` columns correspond to primary date, secondary date, spatial baseline, and temporal baseline (a text file);

(5) One SLC/MLI parameter file

A text file, includes some basic parameter used in the following analysis; if you don't have this file, just manually input the corresponding values in `SIGMAInSAR_main.m`.

(6) DEM file in radar coordination

A `row*col` sized double matrix;

(7) Longitude and latitude files in radar coordination

A `row*col` sized double matrix, to indicate the geographic coordinate of each pixel;

(8) Incident and azimuth angle file in radar coordination

A `row*col` sized double matrix, to indicate the look direction of InSAR data at each pixel;

These input files are also introduced in the main matlab script `SIGMAInSAR_main.m`. (1)-(3), (6)-(8) can be read/written using `freadbkB.m/fwritebkB.m` in folder `SIGMAInSAR_code`. To better illustrate the parameter setting when loading input file in `SIGMAInSAR_main.m`, here shows the input file configuration during my pre-processing procedures. You may use different pre-processing software from me, just import your own corresponding dataset into MATLAB, than use `fwritebkB.m` function to write the files required.

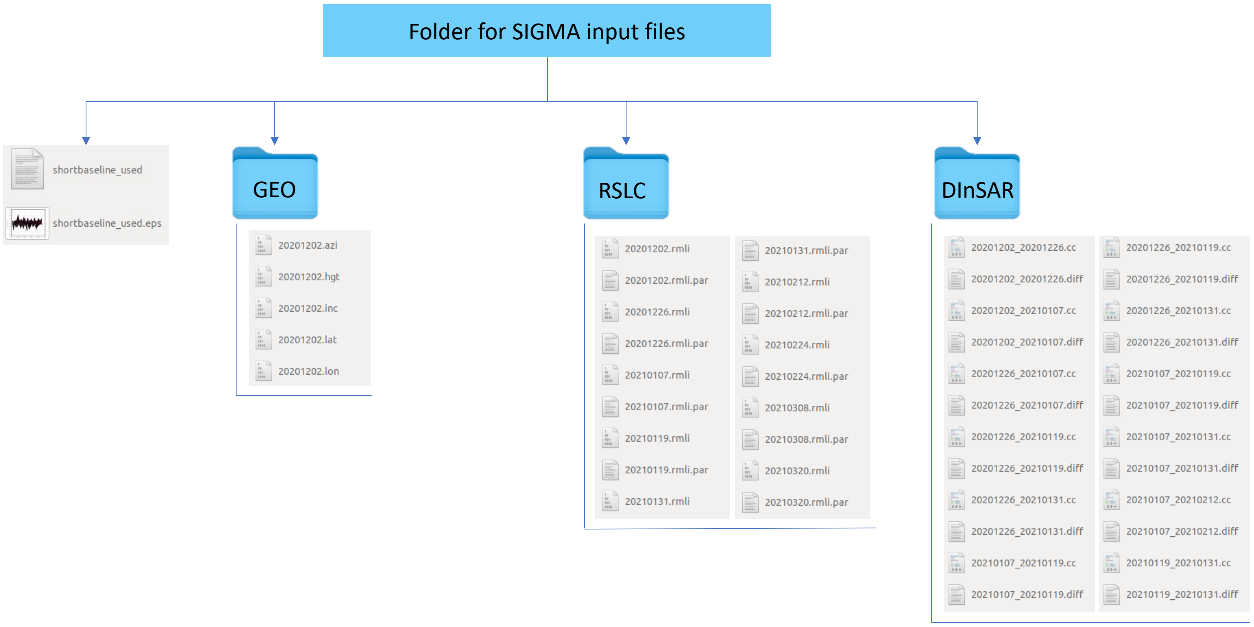


Figure 1. Example of the input files for SIGMA method

2.3 Run SIGMA

(1) Prepare the input as section 2.2;

(2) Configure the input parameters (Lines 16-82 in `SIGMAInSAR_main.m`, apart from the file path parameters, others usually use the default setting).

(3) Reset the threshold of `thrPS` in Lines 172-177, and iteratively run Lines 168-239 to select satisfied PS/DS candidate coverage.

(4) After (3), just run the following code to obtain the final InSAR phase time series.

Note

SIGMA method aims to decrease the decorrelation noises with spatially high frequency, thus the atmospheric delays with spatially low frequency will be reserved in the final InSAR phase time series. In this case, the following DetrendInSAR method can be used to suppress the atmospheric delays and achieve InSAR displacement time series measurement. In the other hand, severe decorrelation noises will result in nonnegligible phase closures and fake temporal cumulation, which is especially obvious in areas with dense vegetation coverage. Further investigations on mitigating such heavy decorrelation noises are required.

2.4 Output

The main output is the InSAR phase time series `defoTS` (here using “`defo`” term just means the unit being meter).

Chapter 3

DetrendInSAR Method

3.1 Installation

The DetrendInSAR package is implemented in Matlab language, and we recommend MATLAB 2018a (or higher version) to operate this package. When using the DetrendInSAR, a new project folder should be created to include the folder `DetrendInSAR_code` and the main script `DetrendInSAR_main.m`. The folder `DetrendInSAR_code` includes the matlab functions of the DetrendInSAR package, and the matlab script `DetrendInSAR_main.m` outside the folder `DetrendInSAR_code` is the main function to run the DetrendInSAR.

Users can directly run the main matlab script `DetrendInSAR_main.m` to install this package, which is basically realized by using `addpath` function.

3.2 Input preparation

The input of DetrendInSAR includes:

(1) the InSAR displacement time series `data`, i.e., a 3D matrix with each layer represents the cumulative displacement at the exact date, with size of `row*col*nslc`; this matrix could be the main output `defoTS` from SIGMA method;

(2) the Digital Elevation Model `dem`, i.e., a 2D matrix represents the DEM data, with size of `row*col`;

(3) the date (yyyymmdd) of the time series `SLC`, i.e., a 1D vector represents the date of each layer in `data`, with size of `nslc*1`;

(4) the relative perpendicular baseline of each date `perpBase` (optional, if no this data `perpBase=[]`), i.e., a 1D vector, with size of `nslc*1`;

(5) the gps time series used to assist the DetrendInSAR correction `gps` (optional, if no this data `gps=[]`), with size of `ngps`*(`nslc`+2), each row is [`row_gps`, `col_gps`, `gps displacement time series at each data acquisition`], [`row_gps`, `col_gps`] is the gps location in `dem` matrix, if no data for some acquisitions at gps station, fill the corresponding position with `nan`.

No matter what preprocessing software is used, the only input for the DetrendInSAR are the aforementioned five matrixes.

3.3 Run DetrendInSAR

(1) Prepare the input as section 1.3;

(2) Users should manually select a polygon to encompass the deformation area. See Line 34 in `DetrendInSAR_main.m` for how to select this polygon, and see section 2.1 for more information about this polygon. The information of this polygon will be stored as '`maskidx.mat`' automatically after the first selection of this polygon, if you want to update the polygon information, please delete the file '`maskidx.mat`'.

(3) Just wait until the scripts finish (be patient, 4000-5000 iterations are required for calculating the unknowns by lsmr), about 1 hour for the whole process with the DEMO case under MacOS 13.2.1, Apple M1 Pro, 16GB, 10 Cores, MATLAB R2022b.

Note

It's still a challenging task to correct atmospheric delays in InSAR, and our contribution aims to enrich the possible solutions in this topic. Of course, the current default parameters in DetrendInSAR cannot realize the optimal correction for all cases, hence there are some parameters in function `DetrendInSAR_downsample.m` that can be modulated according to different cases. Details about these parameters are introduced in Chapter 2, and if you have any questions, please feel free to contact me (make sure introducing yourself in the email).

3.4 Output

The main output is the corrected InSAR displacement time series `data1`.

3.5 Main parameters

Here presents the main parameters that should be defined by users or would affect the performance of DetrendInSAR, and some tips of how to set them are also presented.

3.5.1 Select a polygon with significant deformations

To distinguish the non-deformation area from the deformation area, users should select a polygon to cover the area with significant deformations, i.e., the `selfigbypoly` function in `DetrendInSAR_main.m`. In this case, the area outside this polygon would be considered as non-deformation so that the constraint of "deformation=0" would be added to the DetrendInSAR model at those non-deformation pixels.

For the input demo data, the polygon information exists and is saved as "`maskidx.mat`", if user start a new case, please delete this .mat file, and it's required to select this polygon by yourself when running `DetrendInSAR_main.m`. After the first selection of this polygon, the polygon information is automatically saved as "`maskidx.mat`", and you should delete this file if you want to change the polygon information.

It's better to select a tense polygon to cover only the deformation area, which means that the polygon is as small as possible to include exact the range of deformation area. This will make the "far-field" area much "clear". Of course, it's hard to know the deformation area exactly, users can select a relatively larger polygon, which include part of non-deformation area, to make sure that the constraint of "deformation=0" would not be used at those deforming pixels by mistake.

Of course, it's also feasible for select the polygon as large as possible, in this case, the constraint of "deformation=0" is not used in DetrendInSAR process.

Another parameter to affect the non-deformation area result is the weight of the constraint "deformation=0". This parameter is `weight_eq0` in Line 77 of the function `DetrendInSAR_downsample.m`, which indicate how strong the constraint of "deformation=0" works in the DetrendInSAR model. Large values of `weight_eq0` will result in clearer field in the non-deformation area, i.e., the result value of non-deformation area will close to zero, and vice versa.

3.5.2 The range for modeling trend and DEM-correlated components

The related parameters are in Lines 87-99 in "`DetrendInSAR_downsample.m`". `corrdis*` means the correlative distance of signals calculated by the function of cal_correlate_distance.m, and the trend and DEM-correlated components are modelled in this distance. `win2_*` means that there will be (`win2_*`+1)^2 pixels for modeling the central point, but the adjacent distance of these pixels used for modeling is `dsamp_win2_*`. Large value of `win2_*` will increase the computational burden.

Based on our numerous case studies, it's recommended to use the default parameters. However, users can also adjust them. Larger values of `corrdis*` will result in large window size for modeling the trend and DEM-correlated components. It's found different parameter settings of `corrdis*` are very similar, however it seems that smaller value of `corrdis*` would result in somehow underestimate of the final displacement, vice versa.

3.5.3 The range for constraining temporal displacements

The related parameters are in Lines 106-108 in `DetrendInSAR_downsample.m`. Generally, it's recommended to use the default parameters.

`WT`: the weight of temporal constraint. Larger values mean smoother displacement time series, and vice versa.

`win2_tempPoly`: the number of acquisitions used for constraining the deformation of the central acquisition. Larger values mean smoother displacement time series, and vice versa.

`ord_tempPoly`: the order of temporal polygons. 1 for simple temporal behavior (e.g., linear, log) and 3 for complicated behavior (e.g., period).

3.5.4 Downsample degree

In order to accelerate the computation process of DetrendInSAR, the original data is downsampled in DetrendInSAR_downsample.m. The downsample degree is controlled by the parameter `ncount` in Line 57 of `DetrendInSAR_downsample.m`. Too large values mean much slow computation process. If you have a very powerful workstation, you can try any large value you want. For me, I prefer `ncount`=100.

3.5.5 Including GNSS data in DetrendInSAR model

The DetrendInSAR package is able to include GNSS data to assist the DetrendInSAR modeling. The required information of GNSS data is the position and the cumulative displacements at the SAR acquisition time, and see the parameter introduction of `gps` in `DetrendInSAR_main.m` for the detailed information.

One parameter to adjust how strong the GNSS data control the result is `weigh_gps` in Line 81 of `DetrendInSAR_downsample.m`. Large values of `weigh_gps` means strong constrain of the GNSS data to the result, vice versa. Users can use the default setting or try other values to see the performance.

Chapter 3

Change history

3.1 Version 1.0

2024/04/29: Initial beta release.