**GNSS2TWS: An open-source Matlab tool for inferring daily terrestrial water storage changes using GNSS vertical data**

***User Manual***

Written by Zhongshan Jiang

Southwest Jiaotong University

February 2022

# 1 Installation

The primary purpose of this code is to allow the inversion of time series of surface vertical displacement for daily terrestrial water storage (TWS) changes in a specific study area. We assume that the potential users are familiar with the Matlab language programming, PCA decomposition, inversion method, and mass loading theory. The code has been written in MATLAB (version 2018b) and has been tested on Windows (version 10) operating systems.

Readers can download the complete software (including the code and one example (all input/output data)) from the GPS Toolbox website (<https://geodesy.noaa.gov/gps-toolbox/gnss2tws.shtml>) and we also release this open-source tool (including the code and one example (all input data)) on GitHub (<https://github.com/jzshhh/gnss2tws>). Readers need to decompress the code into a local folder and decompress the file 'gnss.zip' in the folder 'data/' to run this example. There are no additional requirements except for the MATLAB language running environment. In this public edition 1.0 of GNSS2TWS, we provide an example to investigate the spatiotemporal TWS changes in the Pacific Northwest Basin, northwest of the United States. Readers can directly run this example without additional operations.

# 2 Software introduction

2.1 Structure of GNSS2TWS

The folder of ‘gnss2tws’ includes 3 subfolders (‘code/’, ‘data/’, and ‘result/’) and 2 Matlab scripts (‘gnss2ewh\_main.m’ and ‘load\_scenario.m’) (see Figure 1).

(1) The subfolder ‘code/’ includes all sub-routines for the inversion code, there are 9 subfolders (‘Load Data/’, ‘Decomposition/’, ‘Create grids/’, ‘Laplacian/’, ‘Greens function’, ‘Inversion/’, ‘Reconstitution/’, ‘Plotting/’, and ‘Tools/’. Each subfolder saves the corresponding main driver functions and related sub-routines.

(2) The subfolder ‘data /’ includes all datasets, including GNSS data associated with hydrological cycles, station information files, and other auxiliary files (e.g., boundary file of the study area).

(3) The subfolder ‘‘result/’ saves all output results and figures for the final analysis and instruction.

(4) The ‘load\_scenario.m’ file saves all parameters and most of them need some adjustments to match specific study cases.

(5) The ‘gnss2ewh\_main.m’ file is the main program of the ‘gnss2tws’.



Figure 1. The directory structure of the inversion code ‘gnss2tws’.

2.2 Program flow of GNSS2TWS

The GNSS2TWS is an open-source Matlab-based tool for inferring daily terrestrial water storage changes. This software applies the PCA dimensionality-reduction technology to realize the time-varying inversion of vertical position time series in a dense GNSS network. This software is programmed by applying a similar methodology to PCAIM (Kositsky and Avouac, 2010), which recovers the spatiotemporal evolution of the sources (e.g., slip on a fault or magmatic inflation) using surface displacements. Here, we only briefly summarize the program flow of this inversion strategy (see Figure 2):

**1) Initializing scenario**

This step aims to initialize relevant parameters used throughout the code, including the configuration of directories and files for datasets and the setup of parameters for the inversion model.

**2) Loading data**

This step imports known data types and station information into the program for further processing. Each vertical position time series is placed in a column of the observation matrix ( and indicate the numbers of epochs and stations, and missing values are assigned as NaN).

**3) Linear decomposition**

This step would call Matlab's built-in function “pca.m” and the alternating least squares (ALS) algorithm is used to decompose the observation matrix with missing values. This decomposition yields two groups of matrixes referred to as spatial and temporal functions (see Eq. 1). The filtered data are equivalent to the sum of several linearly uncorrelated principal components (PCs) and the number of PCs is usually selected when increasing the number of PCs does not noticeably improve the fit to the raw data.

(1)

**4) Making regional grids**

There are three key steps, including the construction of a discrete study area, generation of a Laplacian matrix using a 2-D discrete kernel of L4 (), and calculation of Green's functions for the equivalent water height (EWH) model using the load Love numbers of Wang et al. (2012).

**5) Inverting linear components**

This step inverts each component of spatial functions for a corresponding gridded EWH component in the study area. To the component, the EWH distribution can be written as follows:

(2)

**6) Calculating model predictions**

This step uses the total time-varying EWH distribution to estimate the predicted displacement time series at GNSS stations. Estimated displacement time series of our inversion results can be expressed as follows:

(3)

where denotes the total time-varying water distribution,  is the number of subpatches.

**7) Plotting and displaying results**

This step plots and displays inversion results, including figures of spatial and temporal functions for each component, maps of annual amplitudes for vertical crustal displacements and EWH, a figure of regional averaged EWH time series, and figures of GNSS observed and predicted time series at all stations.



Figure 2. Program flow and main driver functions of GNSS2TWS.

2.3 Tools

In the folder of ‘code/tools/’, we provide several practical scripts that are useful for our inversion modeling. These scripts require some modifications according to your experiment.

**1) Making extended boundary**

The script ‘making\_extended\_boundary.m’ calls the Matlab built-in function ‘polybuffer.m’ to generate a buffer with any radiuses. Readers can manually extract the extended boundary according to the actual situation.

**2) Determine smoothing factor**

The smoothing factor, adjusting the relative weight between model roughness and data misfit, should be determined for the final inversion model. We provide one script (‘GCV\_smooth\_factor.m’) to determine the smoothing factor, which is based on the statistical method of cross-validation (Jiang et al., 2021; Matthews and Segall, 1993).

**3) Perform checkerboard test**

Checkerboard sensitivity tests are useful to investigate the spatial resolution of the inversion results for regional water storage changes. Readers can revise the script ‘checkerboard\_test.m’ to make it available for their scenes of interest.

# 3 Example

To demonstrate the performance of GNSS2TWS, we investigate the spatiotemporal TWS changes by inverting daily GNSS vertical crustal displacement in the Pacific Northwest Basin, the northwest United States.

3.1 Preparing data

**1) GNSS data**

* **Download GNSS data**

The GNSS time series in the northwest United States are downloaded from the Nevada Geodetic Laboratory, University of Nevada, Reno, United States (http://geodesy.unr.edu/). The daily solutions at 213 stations are chosen according to data continuity, uncertainty, length, and deformation characteristic (e.g., poroelastic and elastic response).

* **Correcting NTAL and NTOL effects**

We first subtract the vertical motions due to non-tidal oceanic and atmospheric loading effects using the environmental loading products released by the German Center for Geoscience (<http://esmdata.gfz-potsdam.de:8080/repository>).

* **Extracting hydrological loading displacement**

The GNSS vertical time series are then modeled with a linear trend, annual and biannual motions, and offsets. We estimate all parameters based on a least-squares fitting method and remove the long-term linear trend and offsets to obtain vertical position time series dominated by seasonal hydrological loads. The Matlab code to extract surface displacement associated with water cycles is also released on GitHub (named lsf, https://github.com/jzshhh/lsf), which is a modified version of Tsview software (<http://www-gpsg.mit.edu/~tah/GGMatlab/#_tsview>) and is suitable for batch processing of GNSS time series.

* **Saving GNSS data recognized by software**

The residual time series at each station are saved in a separate ‘\*.up” file with a 4-char station name. The format of each row in the ‘\*.up’ file is “date (yyyymmdd) up (m) sig\_up (m)”. All files are saved in the folder of ‘data/gnss/’.

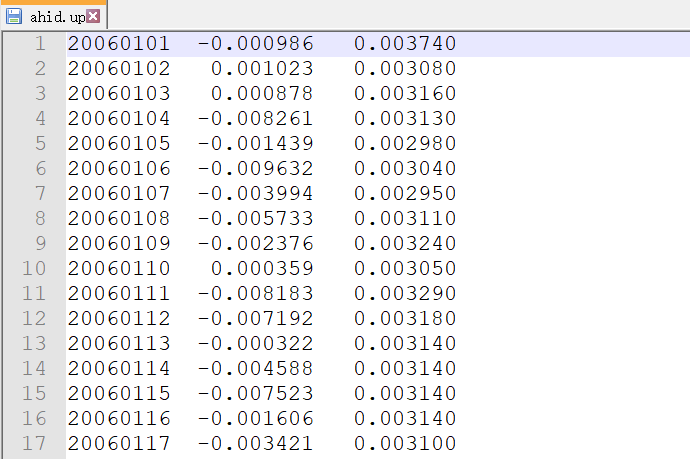


Figure 3. Example of data records in each ‘\*.up’ file.

**2) Boundary file**

* **Download boundary dataset**

The watershed boundary dataset in the contiguous United States is from the U.S. Geological Survey (USGS, <https://www.usgs.gov/core-science-systems/ngp/national-hydrography/watershed-boundary-dataset>). We extract the boundary data of the Pacific Northwest Basin (blue lines in Figure 4).

* **Extend boundary dataset**

To reduce inversion artifacts near the edges of the inversion model, the inland frontier of the study area is extended with a 2.5-degree buffer according to Fu et al. (2015) and the western boundary adjacent to the Pacific is slightly widened 0.25° to avoid unrealistic values along the edge. The extended boundary dataset is saved in the file “PNEB\_border\_buffer.dat”.

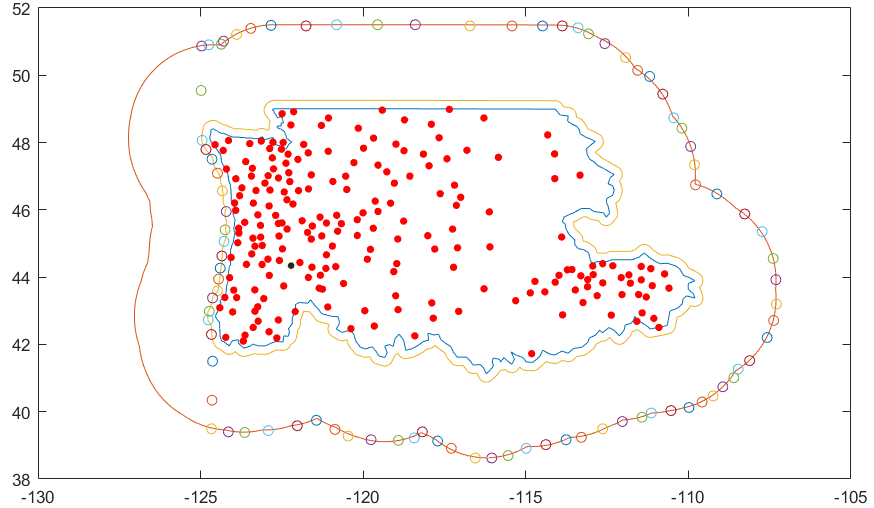


Figure 4. Boundary files of the Pacific Northwest Basin. The blue line is the actual boundary. The outer and inner orange lines are boundaries extended with and buffers, respectively. The circles are manually selected boundaries used in the inversion model. Red solid circles are the locations of GNSS stations.

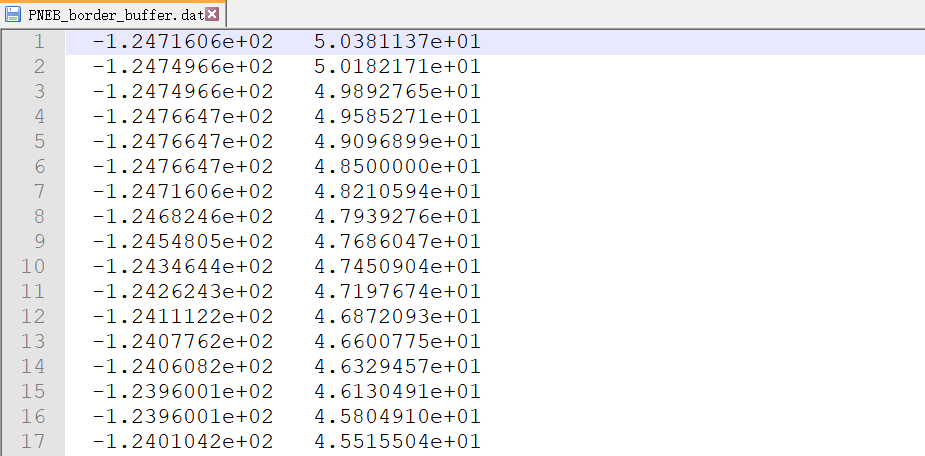


Figure 5. Data records in the extended boundary file (‘PNEB\_border\_buffer.dat’).

**3) Station information file**

The station information is saved in the file ‘sites.info’, the format in each row is ‘name, longitude, latitude, and elevation in meter’. Note that all GNSS data saved in the directory “data/gnss/” should have station information.

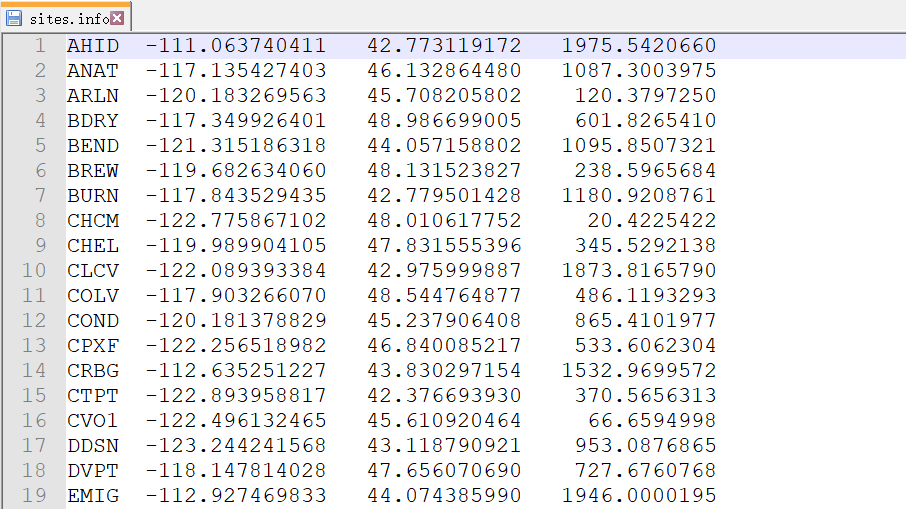


Figure 6. Data records in the station information file (‘sites.info’).

3.2 Performing inversion

After preparing all files, we should set up some parameters for our specified model, click Matlab’s Run button, and wait for the result. Here, we will show how to configure the file “load\_scenario.m” and demonstrate some key steps of our inversion model.

**1) Parameter setting**

First, we need to configure the parameters in the file “load\_scenario.m”, which includes information about ‘Constants’, ‘Study period’, ‘GNSS data’, ‘PCA decomposition’, ‘Study area’, ‘Inversion’, and ‘Plotting’. The example of the file “load\_scenario.m” is shown in Figure 7.

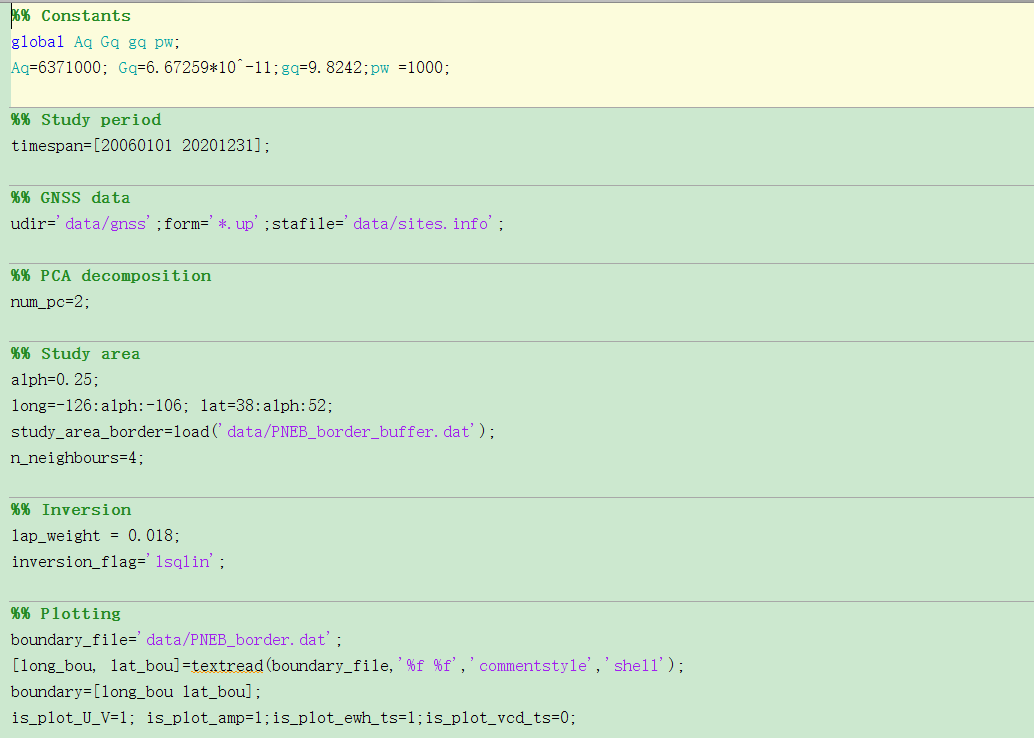


Figure 7. Parameters that need to configure in the file “load\_scenario.m”.

* **Constants**

There are several global variables with constants for the Earth’s average radius (Aq=6371000 m), universal gravitational constant (Gq=6.67259\*10^-11 N·m²/kg²); gravitational acceleration (gq=9.8242 m/s2), and water density (pw =1000 kg/m3).

* **Study period**

The study period should be set for reading GNSS data. In this example, our study period is from 2006-01-01 to 2020-12-31, so we set “timespan=[20060101 20201231];”.

* **GNSS data**

The path of GNSS data, file extension and GNSS site information file should be correctly located, e.g., “udir='data/gnss'; form='\*.up'; stadir='data/sites.info';”.

* **PCA decomposition**

This aims to set up the number of selected PCs for inferring principal EWH components. The number is generally determined when the improvement of fit to the GNSS data is insignificant with more PCs. e.g., “num\_pc=2;”.

* **Study area**

This step aims to setup parameters for the discrete research area, which is used for the calculation of Green’s functions and Laplacian matrix. There are several parameters:

‘alph’ represents grid spacing, e.g., alph=0.25;

‘long’ indicates longitude range, e.g., long=-126:alph:-106;

‘lat’ indicates latitude range, e.g., lat=38:alph:52;

‘study\_area\_border’ indicates the extended boundary data, e.g., study\_area\_border = load('data/PNEB\_border\_buffer.dat');

‘n\_neighbours’ is number of neighbor points used for calculating Laplacian matrix, e.g., n\_neighbours=4;

* **Inversion**

Two parameters need to be setup for inversion.

‘lap\_weight’ shows the smoothing factor, e.g., lap\_weight = 0.018;

‘inversion\_flag’ shows the selected inversion method, e.g., inversion\_flag='lsqlin';

* **Plotting**

The below parameters are configured for plotting (note that 1 means Yes and 0 means NO):

boundary\_file='data/PNEB\_border.dat';

[long\_bou, lat\_bou]=textread(boundary\_file,'%f %f','commentstyle','shell');

boundary=[long\_bou lat\_bou];

is\_plot\_U\_V=1; is\_plot\_amp=1;is\_plot\_ewh\_ts=1;is\_plot\_vcd\_ts=0;

**2) Key steps of our inversion model**

* **Loading GNSS data**

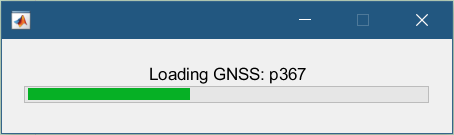


Figure 8. Loading GNSS data.

* **PCA decomposition**



Figure 9. PCA decomposition of GNSS data.

* **4) Calculating Green’s function**

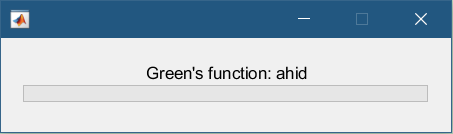


Figure 10. Calculating Green’s function matrix. It may take a long time for the first run.

* **Calculating Laplacian matrix**

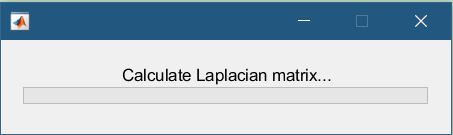


Figure 11. Calculating Laplacian matrix.

* **Invert component**

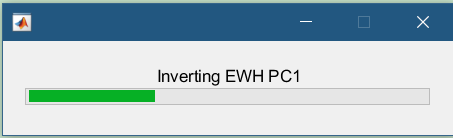


Figure 12. Inverting each component for EWH.

* **Create predictions**



Figure 13. Obtaining total time-varying EWH distribution and estimating predicted displacement time series at GNSS stations.

3.3 Displaying inversion results

* **Determinations of smoothing factor**

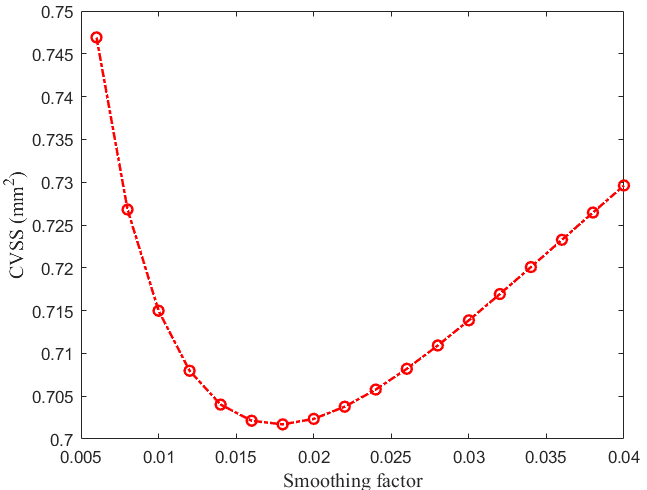


Figure 14. The relation between the sum of squared residuals from cross-validation (CVSS) and smoothing factor. The preferred smoothing factor of 0.018 is determined when obtaining the smallest value of CVSS. Note that it would take a long time to determine the preferred smoothing factor.

* **Temporal and spatial functions**

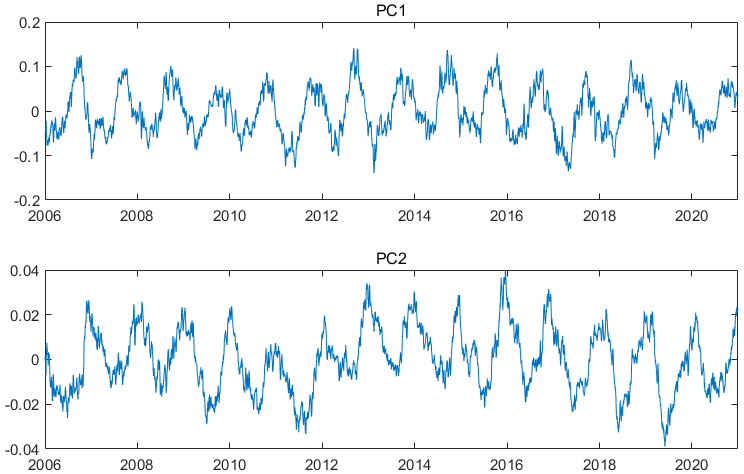


Figure 15. Temporal functions of the first two PCs.

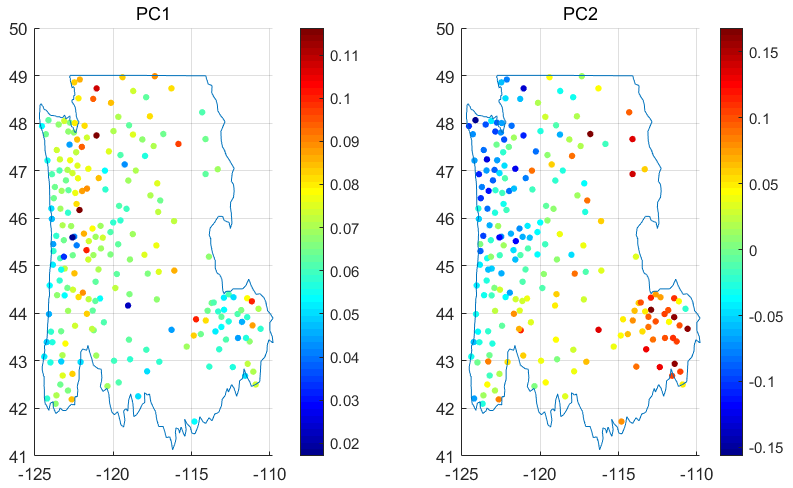


Figure 16. Spatial functions of the first two PCs.

* **Annual amplitudes of VCD and EWH**

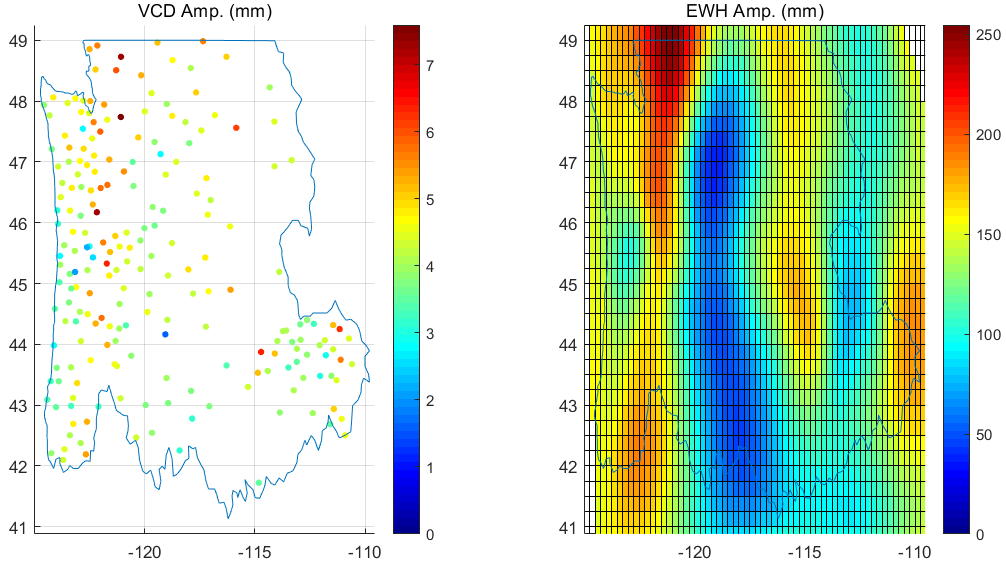


Figure 17. Annual amplitudes of VCD (left) and EWH (right)

* **Regional-averaged EWH time series**

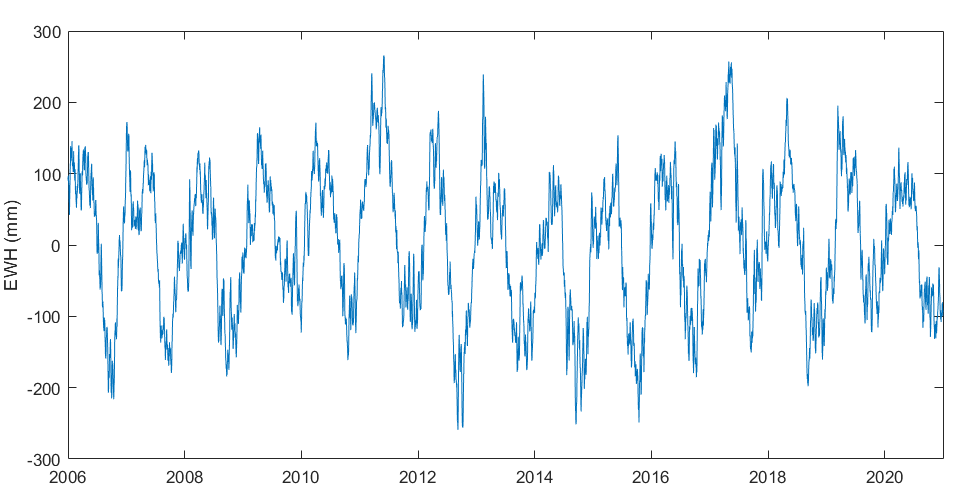


Figure 18. Regional-averaged EWH time series.

* **GNSS VCD time series**

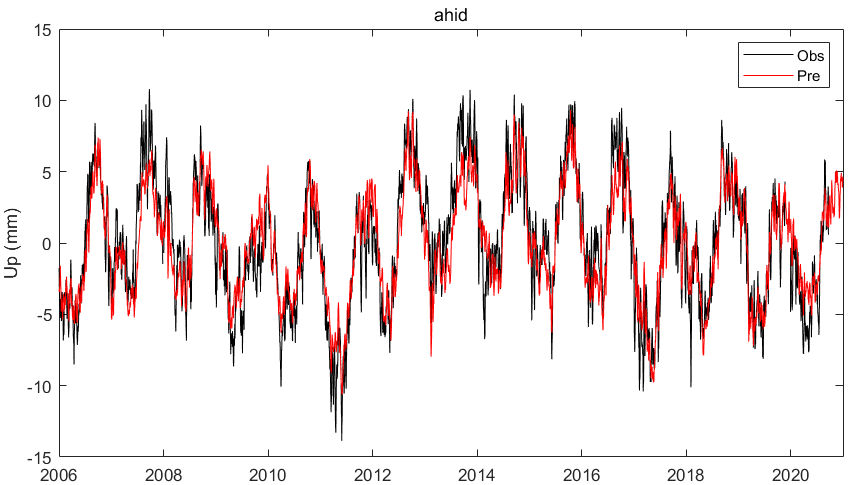


Figure 19. GNSS observed (black line) and predicted (red line) time series at the station of AHID.

* **Checkerboard test**

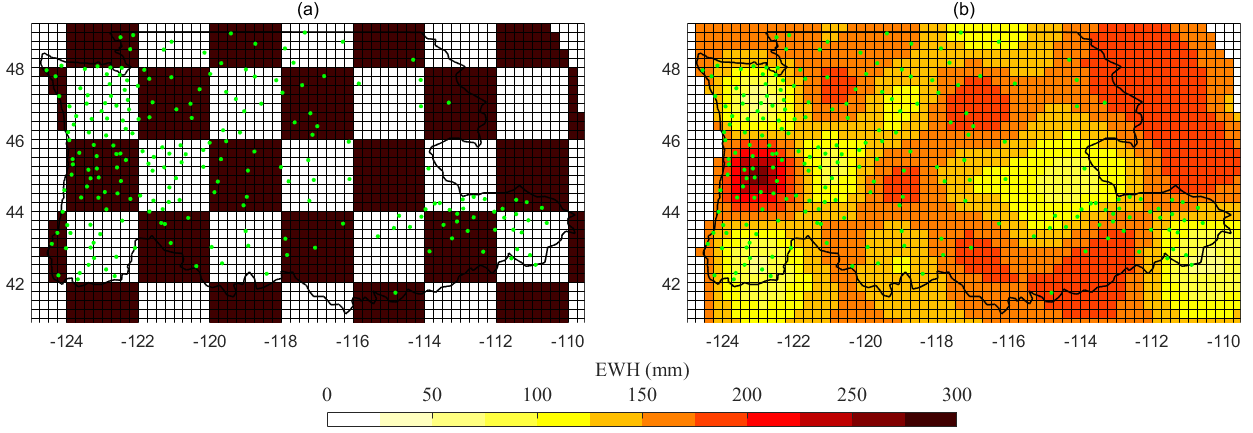


Figure 20. Checkerboard test of inversion model for GNSS equivalent water height (EWH). (a) Input EWH for the forward calculation using the distribution of GNSS stations (green dots) in this study. (b) Inversion results based on the forward vertical motions.

# 4 Contact

This is the first version of GNSS2TWS. Readers can modify the code to make it available for their applications, but please do not release it or its improved versions in another place. Please do not hesitate to contact us if you come across any bugs or have any comments, suggestions, or corrections. The contact details are as follows:

Name: Zhongshan Jiang

E-mail: jzshh@my.swjtu.edu.cn

Address: Faculty of Geosciences and Environmental Engineering, Southwest Jiaotong University, Chengdu 611756, China

References

Fu, Y., Argus, D.F., Landerer, F.W., 2015. GPS as an independent measurement to estimate terrestrial water storage variations in Washington and Oregon. Journal of Geophysical Research: Solid Earth 120, 552-566. doi:10.1002/2014jb011415

Jiang, Z., Hsu, Y.-J., Yuan, L., Huang, D., 2021. Monitoring time-varying terrestrial water storage changes using daily GNSS measurements in Yunnan, southwest China. Remote Sensing of Environment 254. doi:10.1016/j.rse.2020.112249

Kositsky, A.P., Avouac, J.P., 2010. Inverting geodetic time series with a principal component analysis-based inversion method. Journal of Geophysical Research 115. doi:10.1029/2009jb006535

Matthews, M.V., Segall, P., 1993. Estimation of depth-dependent fault slip from measured surface deformation with application to the 1906 San Francisco Earthquake. Journal of Geophysical Research: Solid Earth 98, 12153-12163. doi:10.1029/93jb00440

Wang, H., Xiang, L., Jia, L., Jiang, L., Wang, Z., Hu, B., Gao, P., 2012. Load Love numbers and Green's functions for elastic Earth models PREM, iasp91, ak135, and modified models with refined crustal structure from Crust 2.0. Computers & Geosciences 49, 190-199. doi:10.1016/j.cageo.2012.06.022