InSARFlow: A high-performance program for time series InSAR processing and analysis of land deformation

Phong V.V. Le[a,], Luyen K. Bui[b], Hai V. Pham[c], Anh N. Tran[a], Giang Nguyen-Van[d], Chien V. Pham[d], Phuong A. Tran[e]

[a] *Faculty of Hydrology Meteorology and Oceanography, VNU University of Science, Vietnam National University, Hanoi, Vietnam*
[b] *Department of Spatial Sciences, Curtin University, Perth, WA, Australia*
[c] *Division of Hydrologic Sciences, Desert Research Institute, Las Vegas, NV 89119, USA*
[d] *Faculty of Water Resources Engineering, Thuyloi University, Hanoi, Vietnam*
[e] *Water Research Institute, Hanoi, Vietnam*

Abstract

`InSARFlow` is a highly scalable, parallel software tool for land deformation and subsidence modeling. The program integrates `ISCE` and `GIAnT` software on a high-performance parallel framework for processing large-scale networks of interferograms and time series InSAR analysis. Its parallel computing engine is based on the distributed memory architecture using MPI. Written in Python, `InSARFlow` provides an easy-to-use platform for InSAR processing on high-end parallel supercomputers for the community.

Keywords: Land deformation, InSAR, parallel computing, python

---

Nr.	Code metadata description	Please fill in this column
C1	Current code version	v1.0
C2	Permanent link to code/repository used for this code version	`https://github.com/levuvietphong/InSARFlow`
C3	Code Ocean compute capsule	NA
C4	Legal Code License	GNU GPL v3.0
C5	Code versioning system used	git
C6	Software code languages, tools, and services used	Python 3.x
C7	Compilation requirements, operating environments & dependencies	Standard Python 3 installation
C8	If available Link to developer documentation/manual	`https://github.com/levuvietphong/InSARFlow`
C9	Support email for questions	levuvietphong@gmail.com

Table 1: Code metadata

1. Motivation and significance

Land subsidence, caused by the compaction of susceptible aquifer systems, is a significant global problem that affects many regions around the world [1–5]. Recently, time series interferometric synthetic aperture radar (InSAR) has offered a powerful tool for extracting the temporal evolution of surface deformation from a set of repeated SAR images [6–11]. With the lauch of new SAR satellites (i.e. ALOS-2, Sentinel-1, etc), this radar remote sensing technique is capable of mapping a wide spatial coverage ($\sim$100s km^2) of land deformation with high vertical accuracy ($\sim$1 cm) and fine spatial resolution ($\sim$10 m or less) that is never possible before with techniques like GPS and leveling [12–14].

InSAR relies on the estimation of the phase difference or interferograms between complex-valued SAR images [15]. Error corrected and unwrapped interferograms are used to estimate land deformation rates using time series analysis algorithms, i.e. the persistent scatterer [16, 17] or distributed scatterer [18, 19] methods. For long-term subsidence detection, in order to limit temporal decorrelation, we must use many SAR images acquired at different times [20]. As a result, land subsidence modeling often requires processing a large number of interferograms.

The ever-increasing computing power of high-end parallel supercomputers is enabling the study of InSAR-based land deformation encompassing unprecedentedly large spatiotemporal scales. However, it also poses enormous algorithmic and computational challenges for scaling up the simulations. In this paper, we present a high-performance, python-based software (`InSARFlow`) for land subsidence modeling. `InSARFlow` is unique in that it integrates InSAR Scientific Computing Environment (`ISCE`) [21] and Generic InSAR Analysis Toolbox (`GIAnT`) [22] toolboxes on a MPI-parallel structure for massive interferogram processing. This approach provides good scalability and acceleration of computation on parallel supercomputers. `InSARFlow` also facilitates data processing and transition from `ISCE` to `GIAnT`.

2. Software description

`InSARFlow` was designed to significantly reduce the amount of work and time to process SAR datasets and model land deformation. The software is written in Python due to its popularity and the availability of reliable packages (`ISCE` and `GIAnT`) for InSAR processing in the script language. `InSARFlow` offers parallel capabilities and additional helpful functions for InSAR time series processing.

2.1. Software Architecture

The implementation of `InSARFlow` is very simple that takes parameters from a single configuration file. The standard steps are: (i) download and process SAR images, (ii) construct a connected network among these images, (iii) calculate and unwrap interferograms in parallel, and (iv) post-process and perform time analysis for land deformation. Figure 1 shows the overview and main components of `InSARFlow`, including:

- `InSARFlowObjs.py`: provides classes to define and control model options for different SAR data types.
- `InSARFlowFuncs.py`: contains key functions to download automatically SAR data, form an optimal connected network of images, and process a large number of interferograms in parallel using MPI.
- `InSARFlowGIAnT.py`: provides a tool to process ISCE outputs and run InSAR time series analysis in `GIAnT`.
- `InSARFlowTools.py`: provides tools for time series post-processing and visualization.

Currently, `InSARFlow` supports ALOS-PALSAR (L-band) and Sentinel-1 (C-band) datasets that are global coverage and free available.

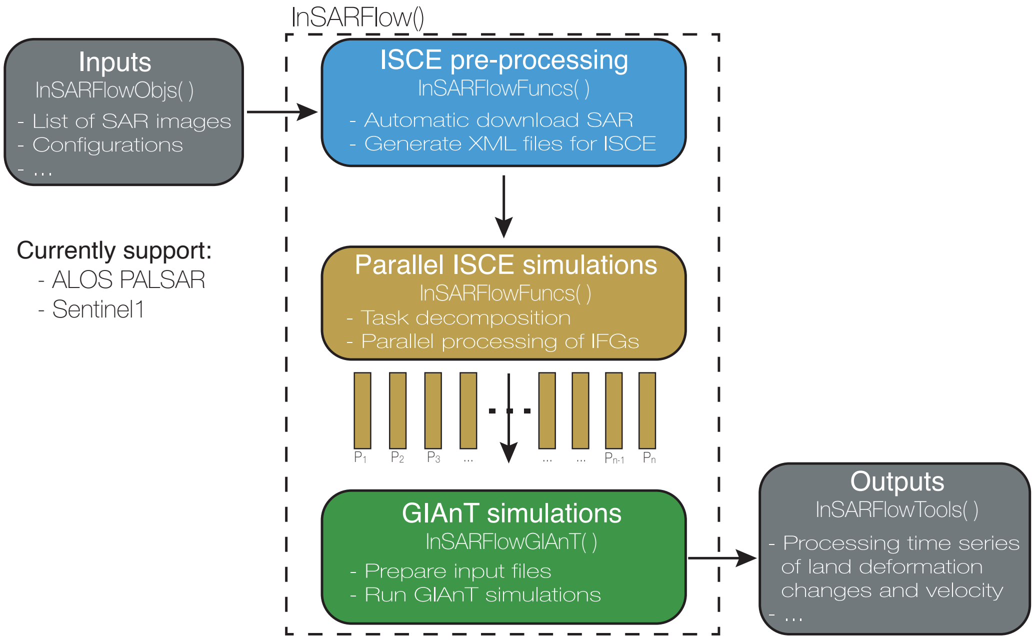

Figure 1: Overview of `InSARFlow` structure and functionality

2.2. Software Functionalities

There are three main functionalities of `InSARFlow`. First, it aims to assist users to download and pre-process SAR images automatically. Given the huge volume of datasets required for processing, these processes are often time-consuming and error-prone. All images are cropped to obtain a common region before generating the interferograms. Second, `InSARFlow` develops an optimal connected network of SAR images (see Figure 2) for interferogram processing in parallel. Specifically, satellite images are considered as nodes, and the links among nodes are edges. Two nodes are linked to each other if (i) their perpendicular baseline (the distance between two acquisition spots) and temporal baseline (the time difference between two acquisitions) are smaller than a given threshold and (ii) degree of connectivity at each node is larger than a user-defined value. The network optimization in `InSARFlow` is performed using `networkX` package. Finally, interferogram processing is parallelized based on task decomposition using MPI. Because the calculation of each interferogram is independent, running `ISCE` for many interferogram can be implemented very efficiently in parallel. Figure 3 show the strong-scaling test of `InSARFlow` as a function of computing nodes used. The result show that time-to-resolution is reduced by a factor of 12.5 running on 16 nodes compared to 1 node, representing descent scaling for InSARFlow applications. The code is parallelized using `mpi4py` package.

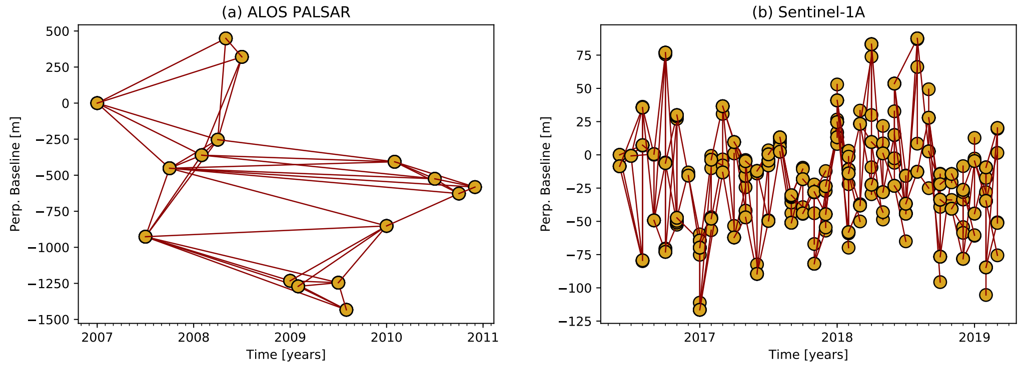

Figure 2: Network of interferograms for (a) 17 ALOS-PALSAR images from late 2006 to 2011 and (b) 81 Sentinel1-A images from mid 2016 to 2019. The x-axis show the time that images are acquired. The y-axis shows the perpendicular baseline with respect to the first image.

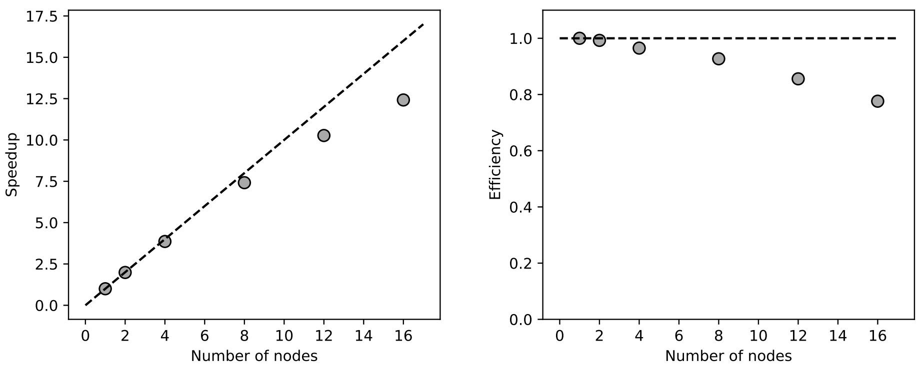

Figure 3: Strong scaling for speedup (left) and efficiency (right) of parallel code in `InSARFlow`. Dash line represents linear speedup (ideal) case.

3. Illustrative Examples

To demonstrate the performance of `InSARFlow`, two examples for land subsidence in the Mekong Delta using both ALOS-PALSAR and Sentinel-1A are provided along with the source code. In order to run InSARFlow, there are two mandatory files that must be presented in the correct structure: A configuration file (.cfg) showing all the options and a comma-separated values (.csv) file containing a list of SAR images for processing. Figure 4 shows two interferograms of the upper Mekong Delta using Sentinel-1A between several acquisition dates (Aug 27, 2017 and Oct 2, 2017; Nov 26, 2018 and Jan 25, 2019).

Listing 1: Running ALOS-PASAR example in `InSARFlow` from command line

```
#!/bin/bash
$ InSARFlow.py -c Mekong_ALOS.cfg
```

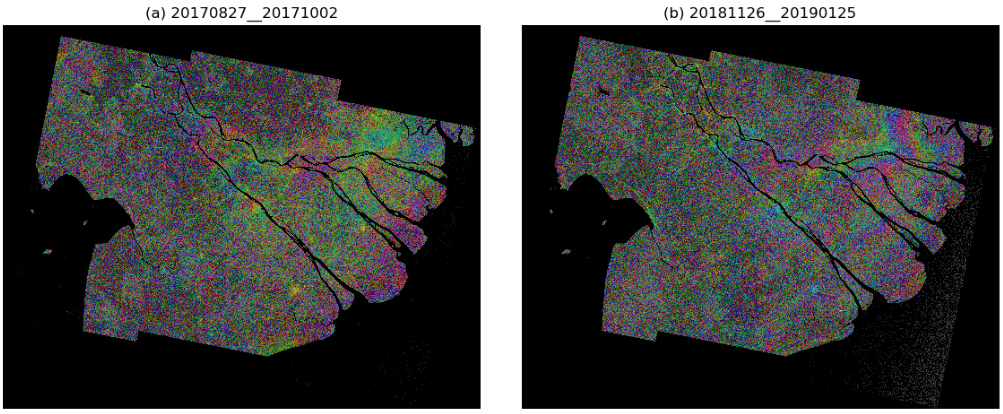

Figure 4: Examples of filtered and geocoded interferograms using Sentinel1-A in the Mekong Delta for (a) August 27, 2017 vs October 2, 2017 and (b) November 26, 2018 and January 25, 2019. In each interferogram, color represents phase difference between two SAR images used.

4. Impact

Two major impacts of the InSARFlow software are to: (i) facilitate land deformation modeling by the integration of interferogram processing and time series analysis into a single framework and (ii) provide an open-source, scalable platform for large-scale InSAR processing on massively parallel supercomputers. New research questions can be pursued in the context of high-performance computing for processing InSAR data.

5. Conclusions

`InSARFlow` is a high-performance and easy-to-use Python program for land subsidence modeling. It contains a variety of functions for InSAR images processing, network optimization, as well as being MPI-parallelized by the task decomposition. It is designed to reduce the amount of work and time to study large-scale land subsidence using both old and new SAR satellite datasets. It has also been rigorously tested during multiple research projects for different SAR images on machines of all sizes, from desktop to high-end parallel computing platforms.

6. Conflict of Interest

The authors declared that they had no conflicts of interest with respect to their authorship or the publication of this article.

Acknowledgements

This research is funded by Vietnam National Foundation for Science and Technology Development (NAFOSTED) under grant number 105.06-2017.320.

References

[1] H. Z. Abidin, R. Djaja, D. Darmawan, S. Hadi, A. Akbar, H. Rajiyowiryono, Y. Sudibyo, I. Meilano, M. A. Kasuma, J. Kahar, C. Subarya, Land subsidence of jakarta (indonesia) and its geodetic monitoring system, Natural Hazards 23 (2) (2001) 365–387. doi:10.1023/A:1011144602064.

[2] J. Hoffmann, H. A. Zebker, D. L. Galloway, F. Amelung, Seasonal subsidence and rebound in las vegas valley, nevada, observed by synthetic aperture radar interferometry, Water Resources Research 37 (6) (2001) 1551–1566. doi:10.1029/2000WR900404.
URL `https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2000WR900404`

[3] D. L. Galloway, T. J. Burbey, Review: Regional land subsidence accompanying groundwater extraction, Hydrogeology Journal 19 (8) (2011) 1459–1486. doi:10.1007/s10040-011-0775-5.

[4] S. A. Higgins, I. Overeem, M. S. Steckler, J. P. M. Syvitski, L. Seeber, S. H. Akhter, Insar measurements of compaction and subsidence in the ganges-brahmaputra delta, bangladesh, Journal of Geophysical Research: Earth Surface 119 (8) (2014) 1768–1781. doi:10.1002/2014JF003117.
URL `https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1002/2014JF003117`

[5] K. D. Murray, R. B. Lohman, Short-lived pause in central california subsidence after heavy winter precipitation of 2017, Science Advances 4 (8). doi:10.1126/sciadv.aar8144.
URL `https://advances.sciencemag.org/content/4/8/eaar8144`

[6] R. Bamler, P. Hartl, , Inverse Problems 14 (4) (1998) R1–R54. doi:10.1088/0266-5611/14/4/001.
URL https://doi.org/10.1088%2F0266-5611%2F14%2F4%2F001

[7] D. Massonnet, K. L. Feigl, Radar interferometry and its application to changes in the earth's surface, Reviews of Geophysics 36 (4) (1998) 441–500. doi:10.1029/97RG03139.
URL https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/97RG03139

[8] P. A. Rosen, S. Hensley, I. R. Joughin, F. K. Li, S. N. Madsen, E. Rodriguez, R. M. Goldstein, Synthetic aperture radar interferometry, Proceedings of the IEEE 88 (3) (2000) 333–382. doi:10.1109/5.838084.

[9] R. Bürgmann, P. A. Rosen, E. J. Fielding, Synthetic aperture radar interferometry to measure earth's surface topography and its deformation, Annual Review of Earth and Planetary Sciences 28 (1) (2000) 169–209. doi:10.1146/annurev.earth.28.1.169.

[10] A. Hooper, D. Bekaert, K. Spaans, M. Arıkan, Recent advances in sar interferometry time series analysis for measuring crustal deformation, Tectonophysics 514-517 (2012) 1 – 13. doi:https://doi.org/10.1016/j.tecto.2011.10.013.

[11] Z. Yunjun, H. Fattahi, F. Amelung, Small baseline insar time series analysis: Unwrapping error correction and noise reduction, Computers & Geosciences 133 (2019) 104331. doi:https://doi.org/10.1016/j.cageo.2019.104331.
URL http://www.sciencedirect.com/science/article/pii/S0098300419304194

[12] D. L. Galloway, K. W. Hudnut, S. E. Ingebritsen, S. P. Phillips, G. Peltzer, F. Rogez, P. A. Rosen, Detection of aquifer system compaction and land subsidence using interferometric synthetic aperture radar, antelope valley, mojave desert, california, Water Resources Research 34 (10) (1998) 2573–2585. doi:10.1029/98WR01285.
URL https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/98WR01285

[13] F. Amelung, D. L. Galloway, J. W. Bell, H. A. Zebker, R. J. Laczniak, Sensing the ups and downs of Las Vegas: InSAR reveals structural control of land subsidence and aquifer-system deformation, Geology 27 (6) (1999) 483–486. doi:10.1130/0091-7613(1999)027¡0483:STUADO¿2.3.CO;2.

[14] M. Motagh, Y. Djamour, T. R. Walter, H.-U. Wetzel, J. Zschau, S. Arabi, Land subsidence in mashhad valley, northeast iran: results from insar, levelling and gps, Geophysical Journal International 168 (2) (2007) 518–526. doi:10.1111/j.1365-246X.2006.03246.x.
URL `https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1365-246X.2006.03246.x`

[15] A. K. Gabriel, R. M. Goldstein, H. A. Zebker, Mapping small elevation changes over large areas: Differential radar interferometry, Journal of Geophysical Research: Solid Earth 94 (B7) (1989) 9183–9191. doi:10.1029/JB094iB07p09183.
URL `https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/JB094iB07p09183`

[16] A. Ferretti, C. Prati, F. Rocca, Permanent scatterers in sar interferometry, IEEE Transactions on Geoscience and Remote Sensing 39 (1) (2001) 8–20. doi:10.1109/36.898661.

[17] A. Hooper, H. Zebker, P. Segall, B. Kampes, A new method for measuring deformation on volcanoes and other natural terrains using insar persistent scatterers, Geophysical Research Letters 31 (23). doi:10.1029/2004GL021737.
URL `https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2004GL021737`

[18] P. Berardino, G. Fornaro, R. Lanari, E. Sansosti, A new algorithm for surface deformation monitoring based on small baseline differential sar interferograms, IEEE Transactions on Geoscience and Remote Sensing 40 (11) (2002) 2375–2383. doi:10.1109/TGRS.2002.803792.

[19] A. Ferretti, A. Fumagalli, F. Novali, C. Prati, F. Rocca, A. Rucci, A new algorithm for processing interferometric data-stacks: Squeesar, IEEE Transactions on Geoscience and Remote Sensing 49 (9) (2011) 3460–3470. doi:10.1109/TGRS.2011.2124465.

[20] H. A. Zebker, J. Villasenor, Decorrelation in interferometric radar echoes, IEEE Transactions on Geoscience and Remote Sensing 30 (5) (1992) 950–959. doi:10.1109/36.175330.

[21] P. A. Rosen, E. Gurrola, G. F. Sacco, H. Zebker, The insar scientific computing environment, in: EUSAR 2012; 9th European Conference on Synthetic Aperture Radar, 2012, pp. 730–733.

[22] P. S. Agram, R. Jolivet, B. Riel, Y. N. Lin, M. Simons, E. Hetland, M.-P. Doin, C. Lasserre, New radar interferometric time series analysis toolbox released, Eos, Transactions American Geophysical Union 94 (7) (2013) 69–70. arXiv:https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1002/2013EO070001, doi:10.1002/2013EO070001.
URL https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1002/2013EO070001