ISCE to StaMPS Manual

Version 1.0.1

Author: David Bekaert

Jet Propulsion Laboratory, California Institute of Technology
United States Government Sponsorship acknowledged

20 April, 2018

Contents

Chapter 1

DISCLAIMER

This manual and the included scripts call routines of the JPL/Caltech ISCE software to prepare outputs of the ISCE stack processort for ingestion in the open source StaMPS package. It is assumed that the user has already generated the coregisterd stack using the ”SLC” workflow of the stack processors included ISCE contrib folder. At this time, the ISCE to StaMPS package supports only outputs from the topsStack processor.

In additon to citing ISCE and the stack processor, cite the ISCE to StaMPS as:

D. P. S. Bekaert et al. Spaceborne Synthetic Aperture Radar Survey of Subsidence in Hampton Roads, Virginia (USA), Scientific Reports (2017). DOI: 10.1038/s41598-017-15309-5

The provided scripts as part of this ISCE to StaMPS conversion package, are included into the contrib of ISCE and are not supported as part of official ISCE applications. The provided scripts only set-up the assumed StaMPS processing structure. Users should refer back to StaMPS for actual time-series processing. The scripts are provided to you ”as is” with no warranties of corrections. Use at your own risk.

Author: David Bekaert
Organization: Jet Propulsion Laboratory, California Institute of Technology
United States Government Sponsorship acknowledged.

Chapter 2

Installation

The provided scripts call ISCE and StaMPS functionality, both packages should be installed. In addition the bin folder should be added to your PATH.

Chapter 3

Introduction

A set of shell scripts are provided that set-up the StaMPS directory structure using outputs from the ISCE stack processors. Outputs from stripmapApp.py and topsApp.py are not supported as these are pair-by-pair processing apps in ISCE. Only outputs from the topsStack and stripmapStack processors are supported in which the user should have selected the ”SLC” workflow.

The SLC workflow outputs a stack of coregisted SLC’s, baselines grids for each SLC, and geometry files (e.g., los, lon, lat, hgt). Each of these files should a corresponding xml file. In case xml files are needed, use the `gdal2isce_xml.py` script included in ISCE package to generate them.

The following structure is assumed:

- SLC’s are organized as `yyyymmdd`/`yyyymmdd.slcSLC_suffix`, where the `SLC_suffix` is an optional suffix string.
- Baselines are organized as `master_slave`/`master_slave` or `master_slave`/`master_slave`.txt, where `master` and `slave` are in `yyyymmdd` format.
- Geom data are stored together as `lon.rdrgeom_suffix`, `lat.rdrgeom_suffix`, `hgt.rdrgeom_suffix` and `los.rdrgeom_suffix`, where the `geom_suffix` is an optional suffix string.

Chapter 4

Setting-up StaMPS processing directories

Generate a file called "input_file" at the location where the INSAR_master_date needs to be generated. The following parameters should be contained in the input_file. Required parameters are indicated with (R), optional parameters with (O). The master_date can be chosen based on minimizing perpendicular and temporal baseline. The baselines will be automatically re-inverted to the chosen master.

Parameter Name	**Value**	**Description: R=required,O=optional**
source_data	slc_stack	Do not change (R)
slc_stack_path	PATH_SLC	Path to the SLC stack (R)
slc_stack_master	yyyymmdd	Master date as to be used in StaMPS processing. Does not need to be the same as the master used for generating the coregistered stack. (R)
slc_stack_geom_path	PATH_geom	Path to the geometry directory. (R)
slc_stack_baseline_path	PATH_baseline	Path to the baselines directory (R)
range_looks	40	Number of range looks multi-looked products (R)
azimuth_looks	10	Number of azimuth look multi-looked products (O). azimuth_looks overrules aspect_ratio.
aspect_ratio	4	Aspect ratio multi-looked products (R)
lambda	0.056	Wavelength in meter units (R)
slc_suffix	.full	Suffix string for SLC's (O)
geom_suffix	.full	Suffix string for geom files (O)

To generate the single master setup type:

```
make_single_master_stack_isce
```

4.1 PS Processing

If you plan to run the StaMPS PS processing, proceed directly to the ”PS processing” chapter of the StaMPS manual. Unlike the mt_prep command detailed in the StaMPS manual use `mt_prep_isce` instead.

4.2 SB Processing

If you plan to only run the StaMPS Small Baseline processing, it will save time and disk space to turn the hard-coded flag for the SM interferogram and amplitude file generation to ’n’ in the `make_single_master_stack_isce` code. Even when not generating the interferograms or amplitude files, the required information for the SMALL_BASELINES processing will be extracted.

If you did not run the STAMPS PS processing in the `INSAR_master_date` directory, then first load baseline info into the matlab workspace. If you already ran the PS processing, you can skip this step:

```
mt_extract_info_isce
matlab
>>ps_load_info
```

To determine which small baseline interferograms to make, in the `INSAR_master_date` directory run one of the `sb_find*.m` codes included in StaMPS, e.g.,:

```
matlab
>>sb_find (rho_min , Ddiff_max , Bdiff_max)
```

Adjust the input parameters according to your data set, where `rho_min` is the minimum coherence based on the maximum temporal `Ddiff_max` and perpendicular `Bdiff_max` decorrelated baselines. There should be no isolated clusters of images. More connections can be made by reducing `rho_min`, or individual connections can be added by editing `small_baselines.list`, which is created by `sb_find`. The connections in `small_baselines.list` can then be plotted with the default StaMPS program:

`>>plot_sb_baselines`.

To create the small baseline interferograms listed in `small_baselines.list` in the `INSAR_master_date` directory, run:

`make_small_baselines_isce`.

Once completed, proceed directly to ”Small Baseline processing” chapter of the StaMPS manual. Unlike the mt_prep command detailed in the StaMPS manual use `mt_prep_isce` instead.

Chapter 5

Additional scripts

If mdx is intalled, the following command can be used to make snapshots of the interferograms:
`get_quickview_isce isce_minrefdem.int`