StaMPS (Stanford Method for PS) Manual

Andy Hooper
Nordic Volcanological Centre
Institute of Earth Sciences
University of Iceland
Askja, 101 Reykjavik

Version 2.1, June 1st, 2007

1 Introduction

This manual provides a guide to running StaMPS, but does not attempt to explain all the processing. For some details on the inner workings, see Hooper et al. (2007), Hooper and Zebker (2007) and Hooper, Ph.D Thesis.

There are two pre-processing steps before getting to the PS processing proper. The first is to focus the raw data, and the second is to form interferograms. ROI_PAC is used for focusing and DORIS for interferogram formation. If starting with SLC images, rather than raw data, the focusing step can be skipped and the images imported directly into DORIS.

Both the ROI_PAC and DORIS processing are non-standard and various shell scripts, matlab scripts and programs are included in this package to produce interferograms that are PS friendly.

The PS processing itself includes C++ programs and matlab scripts to identify PS pixels, and to extract the deformation signal for these pixels. Typing help followed by the name of the matlab script should, in most cases, provide a brief description of the processing.

Throughout this manual, commands to be entered on the command line are in **bold blue** and entries that are specific to the data set being processed and may require modification are in **bold red**. The presence of >> before a command indicates that the command is a matlab script.

2 Environment

Install StaMPS:
tar -xvf StaMPS_v2.1.tar
cd StaMPS/src
make
make install

dismph is a complex phase display program complied in the install that uses X11 Open Motif. This is installed as standard under most unix/linux operating systems (I think), and can be installed on

OS X systems from `http://fink.sourceforge.net/` - copy include/Xm to /usr/X11R6/include and libXm.a to /usr/X11R6/lib. Alternatively another display program may be used instead of dismph.

Details on installing ROI_PAC (if needed) can be found here: `http://www.openchannelfoundation.org/projects/ROI_PAC`

Details on installing DORIS can be found here: `http://enterprise.lr.tudelft.nl/doris/`

The Triangle program is used in 3-D phase unwrapping of PS pixels and can be found here: `http://www.cs.cmu.edu/~quake/triangle.html`

Edit the first 6 lines of StaMPS_CONFIG to point to the correct directories in your case.
source StaMPS_CONFIG

3 Create SLCs (using ROI_PAC)

If you already have CEOS level 1 type SLCs skip this section.

Scripts have been updated to run with ROI_PAC version 2.3. If you have version 2.2 installed, you should run make_slcs_ers_V2.2, make_slcs_envi_V2.2, and remake_slcs_V2.2, instead of the current versions.

In your processing directory:
mkdir SLC
cd SLC
cp $MY_SCR/roi.proc .
mkdir yyyymmdd for each scene and put raw data (IMAGERYyyymmdd) and leader file (SAR-LEADERyyyymmdd) here, or the ASA_file for Envisat data.

Choose a master based on minimizing perpendicular, Doppler and temporal baselines (see Hooper et al., 2007). Substitute your master date in the format yyyymmdd wherever master_date appears below.

While in the SLC directory:

echo master_date > make_slcs.list
make_slcs_ers or **make_slcs_envi**
cd master_date
multilook master_date.slc 6312 (or whatever is the filewidth)
dismph master_date.slc.4looks 1578 (former filewidth/4)

Choose area of interest and note first and last azimuth line numbers (multiplying by 20) e.g. 15000 and 20000 and first and last range pixel numbers (multiplying by 4) e.g. 2400 and 3400

cd .. (back to SLC directory)
vi roi.proc (or use your favorite editor) and change the following:
`before_z_ext = -12500` -(first az line minus 2500) N.B., include the minus sign
`number_of_patches = 3` ((last line - first line + 2500)/3000 rounded up)
`mean_pixel_rng = 2900` (mean range pixel of area of interest)

remake_slcs

cd master_date
multilook master_date.slc 6312 (or whatever is the filewidth)
dismph master_date.slc.4looks 1578 (former filewidth/4)

Find your area of interest and note the new first and last azimuth line numbers (multiplying by 20).

cp $MY_SCR/master_crop.in .
vi master_crop.in and update the crop area. `first_l` and `last_l` are the first and last azimuth line numbers, `first_p` and `last_p` are the first and last range pixels. To ensure round numbers, `first_l` & `first_p` should end in a `1` and `last_l` & `last_p` should end in a `0`.
step_master_setup
cd .. (back to SLC directory)

\ls -d [1,2]* > make_slcs.list
vi make_slcs.list and remove the entry for **master_date**
make_slcs_ers or **make_slcs_envi**

If needed, **remake_slcs** will recreate SLCs for all entries in make_slcs.list without rerunning the make_raw.pl step of ROI_PAC.

It may become apparent later that one or more scenes are offset from the master by so much that the focussed image does not include the cropped master image. In which case copy roi.proc from the SLC directory to the relevant yyyymmdd directory, rename it yyyymmdd.proc and edit it so that the the part processed includes the cropped master image. Edit make_slcs.list to leave only scenes that need recreating and run remake_slcs.

4 CEOS SLC (Level 1 product)

If you have created SLCs with ROI_PAC, skip this section.

mkdir SLC
cd SLC
mkdir yyyymmdd for each scene and put the corresponding ASA_IMP_1P file in each directory, renamed to image.slc

Choose a master based on minimizing perpendicular, Doppler and temporal baselines (see Hooper et al., 2007). Substitute your master date in the format yyyymmdd wherever master_date appears below.

cd master_date
step_master_read_whole
dismph image.slc.4looks 1294 (filewidth/4)

Choose area of interest and note first and last azimuth line numbers (multiplying by 20) e.g. 15000 and 20000 and first and last range pixel numbers (multiplying by 4) e.g. 2400 and 3400

cp $MY_SCR/master_crop.in .
vi master_crop.in and update the crop area. `first_l` and `last_l` are the first and last azimuth line numbers, `first_p` and `last_p` are the first and last range pixels. To ensure round numbers, `first_l` & `first_p` should end in a `1` and `last_l` & `last_p` should end in a `0`.

step_master_read

cd .. (back to SLC directory)
\ls -d [1,2]* > make_slcs.list
vi make_slcs.list and remove the entry for **master_date**
make_read

5 Create IFGs (using DORIS)

In the same directory where SLC and INSAR_master_date reside:
mkdir DEM

Place your DEM in this directory. Resolution should be of order 20m.

cd INSAR_master_date

If the SLCs weren't created by ROI_PAC create **master_date**.slc.rsc with the following line, substituting the correct heading:

```
HEADING     -167.5299818
```

vi dem.dorisin and update the following fields:

```
CRD_IN_FORMAT real4
CRD_IN_DEM    /data/T156/DEM/NED_THIRD_ARCSEC.flt
CRD_IN_SIZE   5222 6070  // rows cols
CRD_IN_DELTA  0.000092592574951594  // posting in degrees
CRD_IN_UL     46.42181 -122.46252  // lat and lon of upper left
CRD_IN_NODATA -9999
```

step_master_orbit

\ls -d ../SLC/[1,2]* > slcs.list (N.B. If the SLC directories are not in the same directory as INSAR_**master_date**, the full path must be given)
vi slcs.list and remove **master_date**
make_orbits

Check output from the `_Start_coarse_correl` step in coreg.out in the yyyymmdd subdirectories that are created:

```
Coarse_correlation_translation_lines:  -76
Coarse_correlation_translation_pixels:  -1
```

These values should be approximately the median values from the data below them. Since DORIS v 3.17, this is sometimes not the case and if wrong by more than a couple of pixels, you should manually change them.

It may also be the case that there is a timing error in the orbit info and the approximate values in `_Start_coarse_orbits` are too far from the real values for coarse correlation to work. In this case, estimate the coarse offsets yourself (look at the SLCs), update them in `_Start_coarse_orbits` in coreg.out and rerun just the DORIS COARSECORR step.

5.1 Manual DEM offset correction

Choose a slave close in time and space. In the **yyyymmdd** subdirectory for the chosen slave:
step_coreg
step_dem
step_resample
step_ifg

matlab -nojvm -nosplash
>>**make_amp_dem(az_down,rg_right,red_contrast,green_brightness)**

- this will combine amplitude (red) and DEM (green).
- the first time, enter **make_amp_dem(0,0)**. Estimate by eye the offset of the DEM in az and range, and rerun until 'good' fit.
- you can adjust **red_contrast** and **green_brightness** (default 0.5 and 1) to vary contrast between amplitude image and DEM.
- Two offsets are output. Once happy with the fit, update these values in dem.dorisin and geocode.dorisin in the INSAR_**master_date** directory.

In the **yyyymmdd** directory:
step_dem
>>**make_amp_dem(0,0)** and check that the fit is correct.

5.2 Bulk Processing

In the INSAR_**master_date** directory:

make_coreg (long runtime)
By default all images with baseline < 100m are coregistered directly to the master and those with larger baselines are coregistered to the 3 closest slave images with a smaller baseline. These default values can be changed by copying $DORIS_SCR/make_coreg to INSAR_master_date, editing the values at the top and running **./make_coreg**.

If rerunning, make_coreg does not re-coregister scenes that have already been processed. If this is required, delete the corresponding CPM_Data.**n1**.**n2** files in the coreg subdirectory, where **n1** and **n2** refer to the order of the two coregistered scenes in make_coreg.list (or 0 for the master), or delete the entire coreg subdirectory to re-coregister all scenes.

Also by default, all cross-correlations with coherence greater than 0.3 are selected initially by DORIS. This value can be increased (by editing coreg.dorisin) if there is generally good coherence to make run times faster or decreased if coherence is generally particularly bad, though I generally find that any cross-correlation with coherence below 0.12 is never correct.

make_dems (long runtime)
make_resample or **make_filtazi_resample** The former does not filter in azimuth, and is the recommended approach for PS processing. The latter filters in azimuth before resampling.

make_ifgs
xv */*.ras and check that each interferogram looks OK

5.3 Rerunning Steps

make_orbits processes all images listed in slcs.list

make_dems, make_resample, make_filtazi_resample and make_ifgs process all images listed in make_ifgs.list, which is output by make_orbits, but can be edited to add or drop images.

make_coreg processes all images listed in make_coreg.list in the coreg subdirectory. Extra images can be added to the bottom of this file, but no lines should ever be deleted, as n1 and n2 in CPM_Data.n1.n2 files refer to the order of make_coreg.list

The following individual steps can be rerun in the individual **yyyymmdd** subdirectories of INSAR_**master_date**:
step_coreg coregisters the slave image directly to the master (may be different to results from make_coreg which includes slave-slave coregistration).
step_resample resamples the slave image.
step_filtazi_resample filters both master and slave in azimuth and resamples the slave.
step_dem creates the simulated dem interferogram.
step_ifg creates the interferogram.

5.4 Geocode

cd to your favorite interferogram directory and run:
step_geo (calculates the latitude and longitude of each pixel, only needs to be run once)

5.5 Possible reasons for DORIS SIGERV error

- master.res or slave.res (as specified in dorisin file) is missing
- orbits are missing from master.res or slave.res
- higher order coefficients in coregpm are too large - makes resampling impossible
- slave SLC doesn't completely overlap the master cropped SLC. See discussion above on recreating the SLC using ROI_PAC.

6 PS Processing

In the INSAR_**master_date** directory run
ps_prep 0.4 3 2 50 200 where

0.4 = amplitude dispersion
3 = number of patches in range (default 1)
2 = number of patches in azimuth, (default 1)
50 = overlapping pixels between patches in range (default 50)
200 = overlapping pixels between patches in azimuth (default 200)

The number of patches you choose will depend on the size of your area and the memory on your computer. Generally, patches should be < 5 million SLC pixels.

matlab -nojvm -nosplash
>>**ps_parms_default**

You can modify the default parameters using >>**setparm** (refer to my thesis for meaning of most parameters).

>>**stamps**

The default is to run all steps. A subset of steps can also be selected - see >>**help stamps** for details.

Steps 1 to 7 run by default on individual patches. Step 8 merges the patches into one patch and Step 9 runs on the merged patch. After merging, it is also possible to run Step 6 and/or 7 on the merged patch by setting the patch_flag to 'n', e.g.,
>>**stamps(6,7,'n')**
This should in general produce more reliable results, but will take longer to run.

6.1 Step 6: Phase Unwrapping

Processing is controlled by the following parameters:

Parameter Name	Default	Description
`unwrap_method`	'3D'	Unwrapping method.
`unwrap_ifg_index`	'all'	Index to interferograms to be unwrapped.
`unwrap_prefilter_flag`	'y'	Prefilter phase before unwrapping to reduce noise. Other option (not generally recommended) 'n'.
`unwrap_patch_phase`	'n'	Use the patch phase from Step 3 as prefiltered phase. If set to 'n' (recommended), PS phase is filtered using a Goldstein adaptive phase filter.
`unwrap_grid_size`	100	Resampling grid spacing. If `unwrap_prefilter_flag` is set to 'y', phase is resampled to a grid with this spacing.
`unwrap_gold_n_win`	32	Window size for Goldstein filter
`unwrap_time_win`	180	Smoothing window (in days) for estimating phase noise distribution for each pair of neighboring pixels. The time series phase for each pair is smoothed using a Gaussian window with standard deviation of this size. Original phase minus smoothed phase is assumed to be noise, which is used for determining probability of a phase jump between the pair in each interferogram.

Note that if re-running Step 6 and Step 7 has been run, estimates of SCLA and master atmosphere and orbit error will be subtracted before unwrapping. If you do not wish this to occur, reset these

estimates before running Step 6 with
>>**scla_reset**

Note also, however, that subtraction of SCLA and master atmosphere and orbit error has not been implemented with the `unwrap_prefilter_flag` = 'n' option.

After running Step 6, display the output with
>>**ps_plot('u')**

Check for unwrapping errors i.e., phase jumps in space which are uncorrelated in time. Unwrapping errors are more likely to occur in longer perpendicular baseline interferograms. This is for two reasons, firstly there is more noise associated with each PS, and secondly, the phase due to any spatially-correlated look angle (SCLA) error is larger, as it is proportional to perpendicular baseline. Noise is reduced by spatial filtering before unwrapping, but it is also possible to reduce the SCLA error phase by estimating the SCLA error from the interferograms that unwrap OK (Step 7). If Step 6 is re-run after Step 7 has been run, SCLA error phase is temporarily subtracted from the wrapped phase before unwrapping. Unwrapping accuracy is further improved by also temporarily subtracting the atmosphere and orbit error phase due to the master image, present in all the interferograms, which is also estimated in Step 7.

6.2 Step 7: Spatially-Correlated Look Angle Error

Spatially-uncorrelated look angle error is calculated in Step 3 and removed in Step 5. After unwrapping, spatially-correlated look angle (SCLA) error is calculated which is due to spatially-correlated DEM error (this includes error in the DEM itself, and incorrect mapping of the DEM into radar coordinates). Master atmosphere and orbit error phase is estimated simultaneously.

Processing is controlled by the following parameters:

Parameter Name	Default	Description
`recalc_index`	'all'	Index to interferograms to be used in the SCLA estimation.

Display the estimate of SCLA error with
>>**ps_plot('d')** Units are phase per m of perpendicular baseline, with 0.01 rad/m corresponding to about 12 m of DEM error for ENVISAT I2 swath.

Display the estimate of master atmosphere and orbit error phase with
>>**ps_plot('m')**

Unwrapped phase minus either or both of the above can be plotted with **'u-d'**, **'u-m'** or **'u-dm**'.

After running, check that the estimates seem reasonable, i.e., **ps_plot('u-dm')** looks generally smoother than **ps_plot('u')** (note that the default colour scales will be different). If not generally smoother, one or more interferograms has probably unwrapped incorrectly. Drop it/them from `recalc_index` and rerun Step 7, e.g., to drop the 13th and 14th interferograms,
>>**setparm('recalc_in',[1:12,15:17])**
>>**stamps(7,7)**

Once happy that all included interferograms are generally smoother, rerun Step 6. Step 6 will subtract the estimates of SCLA and master atmosphere/orbit errors, before unwrapping and add them back in afterwards (as long as `unwrap_prefilter_flag` = 'y'). If more interferograms become reliably unwrapped on re-running, add them into `recalc_index` before running Step 7. This can be

repeated until all interferograms are reliably unwrapped, or until no further improvement is seen.

If there is non-steady deformation present in some interferograms and, by chance, it correlates with perpendicular baseline, it can get mapped into the SCLA error. This may be evidenced as propagation of any deformation in **ps_plot('u-dm')**to all interferograms (though the sign for each will depend on the perpendicular baseline sign), or correlation of **ps_plot('d')** with **ps_plot('m')**. If you suspect this is occurring, you can attempt to remove the deformation/baseline correlation by adding or subtracting interferograms from `recalc_index`. Note that time and baseline info can be displayed with
>>**ps_info**

If some interferograms are still not reliably unwrapped, try setting `unwrap_patch_phase` to 'n' and rerunning Step 6. This will use the filtered phase of the PS pixels only, rather than that derived in Step 3 for unwrapping. Try also increasing `unwrap_grid_size` to 200 m or more. This will reduce the effects of noise by smoothing more, but do not set it higher than the distance over which you expect deformation phase to vary by about $\pi/2$. Another thing to try is dropping noisier pixels by setting `weed_standard_dev` to a lower value, and re-running from Step 4.

7 PS Plotting

The following matlab scripts plot the data in various ways (use >>**help** in matlab for options)

plot_all_ifgs	Plots all multilooked interferograms
ps_plot	Plots a series of PS phase values, on various backgrounds
ps_plot_ifg	Plots a value for each PS, on various backgrounds

You can select a reference area by setting parameters `ref_lon` and `ref_lat`. All plots will then be referenced to the mean value for this area.

8 Change History

N.B. This list is not comprehensive.

Version 1.0

- Initial beta release.

Version 1.1

- Addition of **make_resample** and **make_filtazi_resample** to give the option of filtering in azimuth. As this involves updating **master.res** differently for every image pair, a separate **master.res** is now maintained in each individual slave directory.
- Update to **make_coreg** to be more efficient (uses a different strategy for picking which images to coregister).

- Addition of **step_coreg** to allow coregistration for an individual slave image with the master image.
- Update to **make_amp_dem.m** to display the image in matlab instead of using **disrg**
- Error in **ps_load_initial.m** fixed so that individual PS bperp and look angle values are now correct.
- Addition of **ps_load_dem.m** to allow plotting of PS on shaded relief topography.
- Other tidying of code.

Version 2.0

- Processing added to enable input of CEOS Level 1 SLC data.
- **step_master_setup** added.
- Extra step added to **ps_weed.m** to drop pixels that are not correlated in time with surrounding pixels.
- Ability to process data in smaller patches added.
- Changes to way data saved, for efficiency.
- Changes to **ps_est_gamma_quick.m** to make it restartable and to make convergence criteria more reliable.
- New statistical cost function 3-D unwrapping algorithm.
- Look angle bug fixed.

Version 2.0.1

- Some bug fixes.

Version 2.0.2

- Flattening/DEM processing changed back to that in Version 1.1 (to remove a bug that was introduced).

Version 2.0.3

- Changes for compatibility with 64-bit machines.
- New scripts for working with ESA level 1 SLCs.
- Change to **ps_weed.m** to handle duplicate lat/lon assignment by DORIS.

Version 2.1

- Estimation of spatially-correlated look angle (DEM) error and master atmosphere and orbit error added (Step 7).
- Unwrapping (Step 6) now uses estimates from Step 7 if present.
- Merging of patches is now an explicit step (Step 8).
- Estimation of spatially-correlated noise is now Step 9.
- **ps_info** added

9 References

Links to PDF files for all the following references can be find at: `http://www.hi.is/~ahooper/pubs.html`

Hooper, A., P. Segall and H. Zebker, Persistent Scatterer InSAR for Crustal Deformation Analysis, with Application to Volcan Alcedo, Galapagos, *Journal of Geophysical Research*, in press, 2007.

Hooper, A., and H. Zebker, Phase unwrapping in three dimensions with application to InSAR time series, *Journal of the Optical Society of America A (Optics, Image Science and Vision)*, in press, 2007.

Hooper, A., H. Zebker, P. Segall, and B. Kampes, A New Method for Measuring Deformation on Volcanoes and Other Natural Terrains Using InSAR Persistent Scatterers, *Geophys. Res. Letters*, 31, L23611, doi:10.1029/2004GL021737, 2004

Hooper, A., Persistent Scatterer Radar Interferometry for Crustal Deformation Studies and Modeling of Volcanic Deformation, Ph.D. thesis, Stanford University, 2006.