StaMPS/MTI Manual

Version 3.2

Andy Hooper

4th November, 2009

Delft Institute of Earth Observation and Space Systems
Delft University of Technology
Kluyverweg 1, 2629 HS, Delft
The Netherlands

Contents

Chapter 1

Introduction

StaMPS/MTI is made available for non-commercial applications only.

StaMPS (Stanford Method for Persistent Scatterers) is a software package that implements an InSAR persistent scatterer (PS) method developed to work even in terrains devoid of man-made structures and/or undergoing non-steady deformation. StaMPS/MTI (Multi-Temporal InSAR) is an extended version of StaMPS that also includes a small baseline method and a combined multi-temporal InSAR method. The original development of StaMPS was undertaken at Stanford University, but subsequent development of StaMPS and StaMPS/MTI has taken place at the University of Iceland and Delft University of Technology. There are also contributions from users of the package based at other institutions.

This manual provides a guide to running StaMPS/MTI, but does not explain all the processing. For some details on the inner workings, see **?????**.

A user group is also maintained at `http://groups-beta.google.com/group/mainsar`. If you have a query, check the discussion threads there and, if not resolved, submit your question to the group.

There are two pre-processing steps before getting to the PS/MTI processing proper. The first is to focus the raw data (if required), and the second is to form interferograms from single-look complex (SLC) images. ROI_PAC is used for the focusing and Doris for interferogram formation. If starting with CEOS format SLC images, rather than raw data, the focusing step is skipped and the images are imported directly into Doris. Currently, support is provided for processing ERS, Envisat or ALOS data, if starting with raw data, and for ERS, Envisat or RADARSAT-1 data, if starting with CEOS SLCs.

Both 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/MTI friendly.

The PS/MTI processing itself includes C++ programs and matlab scripts to identify coherent pixels, and to extract the deformation signal for these pixels. Typing `help` followed by the name of the matlab script provides a brief description of the processing.

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

Chapter 2

Installation

Install StaMPS/MTI:

```
tar -xvf StaMPS_v3.2.tar
cd StaMPS_v3.2/src
make
make install
```

2.1 Configuration

Edit StaMPS_CONFIG.tcsh or StaMPS_CONFIG.bash (depending on which shell you prefer to use) to point to the correct directories for your set-up (you will need additional programs installed, see below).

```
source StaMPS_CONFIG.xxxx
```

This must be done whenever a new terminal is opened. You might want to add this line to your .cshrc or .bashrc file so that this is done automatically.

2.2 Data display

A program called dismph (to display complex data) is included in the package but not compiled by default. It uses X11 Open Motif. This is installed as standard under many unix/linux operating systems, 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. To compile dismph, run

make dismph.

Alternatively another display program may be used instead of dismph, e.g.,
xv (http://www.trilon.com/xv) or OpenEV (http://fwtools.maptools.org/).

2.3 ROI_PAC

Details on installing and running ROI_PAC (if needed) can be found at:
`http://roipac.org/ROI_PAC`.

For ERS, ROI_PAC requires the program getorb which must be installed in the `ROI_PAC/INT_BIN` directory and can be downloaded from:
`http://www.deos.tudelft.nl/ers/precorbs/tools/getorb_pack.shtml`.

ODR and arclist files containing the orbit information used by getorb can be downloaded from:
`http://www.deos.tudelft.nl/ers/precorbs/orbits/`.
These files should be stored in directories `.../ODR/XXXX` where `XXXX` is `ERS1` or `ERS2`. For Envisat data, getorb can also be used, in which case `XXXX` should be `Envisat`. Alternatively, use the ESA DORIS (the tracking system on Envisat, not to be confused with the Doris interferometry software) orbits. This is a better option at the date of writing as the ODR files have not been updated since the beginning of 2008.

2.4 Doris

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

2.5 Triangle

The Triangle program is used for Delaunay triangulation and can be found at:
`http://www.cs.cmu.edu/~quake/triangle.html`.

2.6 Snaphu

The optimisation routines of snaphu are used by the 3-D unwrapping code and can be downloaded from:
`http://www-star.stanford.edu/sar_group/snaphu`.

Chapter 3

Create SLCs (using ROI_PAC)

Currently support is provided for raw data from ERS, Envisat and ALOS satellites.

If you are starting with SLC data rather than raw data, skip this section.

Scripts have been updated to run with ROI_PAC version 2.3 or 3.0. If you have the ROI_PAC version 2.2 installed, you should run the versions of the SLC generation scripts with the suffix `_V2.2` attached.

In your processing directory:
`mkdir SLC`
`cd SLC`
`mkdir yyyymmdd` for each scene and within this directory create symbolic links to the raw data (and leader files if they exist). For ERS data the link names must be `IMAGERYyyyymmdd` and `SARLEADERyyyymmdd` (be sure to use 4 digit, not 2 digit, years), but for Envisat and ALOS data the link names should be the same as the original names.

Choose a master based on minimising perpendicular, Doppler and temporal baselines (see **?**). If you do not know the baseline data, choose a preliminary master based only on time and follow this manual as far as the `make_coarse` step in Section **??**. After running `make_coarse`, enter `grep Bperp */coreg.out` in the `INSAR_master_date` directory to see the perpendicular baselines with respect to the selected master (enter `grep f_DC_con */slave.res` in the `SLC` directory for the Doppler centroids). A new master can then be picked if necessary. If this is the case, edit `master_crop.in` for the new master, run `step_master_setup`, then go directly to Section **??**.

Substitute your chosen master date in the format `yyyymmdd` wherever `master_date` appears below.

From the SLC directory:

`cd master_date`

To use ODR orbits:
`step_slc_ers` or `step_slc_envi` (use `_V2.2` suffix if necessary)

To use DORIS orbits (Envisat only):
`step_slc_envi_vor`
For ALOS data:
`step_slc_alos`
Take note of the number of azimuth/range looks and the multilooked filewidth.

`dismph image.slc.Xlooks.raw 1294` (range looks, multilooked filewidth) or
`xv image.slc.Xlooks.ras`

Choose your region of interest and note the first/last azimuth lines and first/last range pixels. You should try to pick an area that is included in all the slave images too, which may not be the case if you choose an area too close to the image edge. Multiply the line numbers by the number of azimuth looks and the pixel numbers by the number of range looks to get line/pixel numbers referenced to the SLC.

Edit `../roi.proc` and change the following:
`ymin = 14000` (first azimuth line minus 1000)
`ymax = 21000` (last azimuth line plus 1000). You will also need to uncomment this line
`mean_pixel_rng = 2900` (range pixel of middle of region of interest)

`step_slc_alos`, `step_slc_ers` or `step_slc_envi` (attach `_V2.2` suffix if necessary).
`dismph image.slc.Xlooks.raw 1294` (range looks, multilooked filewidth) or
`xv image.slc.Xlooks.ras`

Find your region of interest again and note the new first and last azimuth line numbers (multiplying by the number of azimuth looks) .

`cp $MY_SCR/master_crop.in .`
Edit `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.

`step_master_setup`
`cd ..` (back to SLC directory)

`make_slcs_alos`, `make_slcs_ers`, `make_slcs_envi` (attach `_V2.2` suffix if necessary), or `make_slcs_envi_vor`. This will create SLCs for all directories listed in `make_slcs.list`, which by default contains all data directories except the master.

It may become apparent later that one or more scenes are offset from the master by so much that the focused image does not include the entire 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 `make_slcs_XXXX`.

Chapter 4

Processing CEOS SLCs (Level 1 product)

Currently, scripts exist for processing SLCs from ERS, Envisat and RADARSAT-1 satellites.

If you created SLCs with ROI_PAC, skip this section.

In your processing directory:

```
mkdir SLC
cd SLC
```

`mkdir yyyymmdd` for each scene and within this directory create symbolic links to the raw data (and leader/volume files if they exist). For ERS and RADARSAT-1 data the link names must be `DAT_01.001` and `LEA_01.001` and `VDF_DAT.001`. For Envisat data the link name must be `image.slc`.

Choose a master based on minimising perpendicular, Doppler and temporal baselines (see **?**). Substitute your master date in the format `yyyymmdd` wherever `master_date` appears below.

```
cd master_date
step_read_whole_XXX
```

(where `XXX` is '`ERS`', '`Envisat`' or '`RSAT`')

Take note of the number of azimuth/range looks and the multilooked filewidth.

`dismph image.slc.Xlooks.raw 1294` (range looks, multilooked filewidth) or `xv image.slc.Xlooks.ras`

4.1 Option 1: Specify by latitude and longitude

edit `master_crop_geo.in` in the SLC directory, and specify your area of interest.

```
cd master_date
step_master_read_geo
cd ..
```

```
make_read_geo
```

4.2 Option 2: Specify by line/pixel number

Choose area of interest and note first/last azimuth line and first/last range pixel. You should try to pick an area that is included in all the slave images too, which may not be the case if you choose an area too close to the image edge. Multiply the line numbers by the number of azimuth looks and the pixel numbers by the number of range looks to get numbers referenced to the SLC.

```
cp $MY_SCR/master_crop.in .
```

Edit `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.

```
step_master_read
```

`cd ..` (back to SLC directory)
`make_read` This will read and crop SLCs from all directories listed in `make_slcs.list`, which by default contains all SLC directories except the master.

Chapter 5

Create IFGs (using DORIS)

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

Place your DEM in this directory.

`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 (to the nearest degree is fine):

```
HEADING    -167
```

`step_master_orbit_ODR` (only run if using precise ODR orbits)

edit `timing.dorisin` and update the following fields based on the DEM you are using:

```
SAM_IN_FORMAT real4
SAM_IN_DEM    /data/T156/DEM/dem_data.flt
SAM_IN_SIZE   4801 4801  // rows cols
SAM_IN_DELTA  0.000833333 0.000833333  // posting in degrees
SAM_IN_UL     13 42  // lat and lon of upper left
SAM_IN_NODATA -9999
```

`step_master_timing`
This step can be run alongside `make_orbits`, `make_coarse` and `make_coreg`. This step was new in version 3.1, and replaces the former StaMPS codes for DEM offset correction with new code in Doris v4.0. Offsets are calculated for 30 (by default) different windows, which are printed to the screen at the end. Check that the selected offsets are consistent with the mode values of for the windows. If not, change the number of windows in `timing.dorisin` to a higher number, comment out the `M_SIMAMP` process and rerun the step. If still not successful, delete the output from the timing step in master.res in the INSAR_master_date directory and ALL subdirectories, and calculate the DEM offset manually (see Section **??**).

5.1 Bulk Processing

In the INSAR_master_date directory:

`make_orbits`
This creates a subdirectory for each slave image. The default is to treat all images in the SLC directory except the master as slave images. If a different set of slave images is required, create a file named `slcs.list` listing the directories containing the images you wish to include, before running `make_orbits`. Precise orbits are extracted from the ODR files if they are found (ERS and Envisat only).

`make_coarse`
This creates a `coreg.out` file in each slave subdirectory. The last 32 lines of each `coreg.out` file is output to the terminal at the end. Check the following values for each `coreg.out` file:

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

These values should be approximately the modal values from the data below them. If this is not the case and the values are wrong by more than a couple of pixels, you should edit the relevant `coreg.out` file and correct the values.

Optionally, you can also check that the master crop is included within each slave image (only the parts of the master crop that are in ALL slaves will be considered in the later times series processing). Look for the highest and lowest values of `Coarse_correlation_translation_lines` and `Coarse_correlation_translation_pixels`. Add the translations to the master crop range (in `master.res`) and make sure the corresponding slave crop contains the translated values (in `slave.res`). If not, you should adjust either the master crop or the relevant slave crops. If you adjust the master crop, it is easiest to delete the whole `INSAR_master_date` directory and recreate it with `step_master_setup` (raw data) or `step_master_read/step_master_read_geo` (CEOS SLC data). If you adjust slave crops (by rerunning `step_slc_XXX` or `step_read`), you need run `step_orbit` in the slave subdirectory only for those slaves that have been adjusted.

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.

`make_coreg` (long runtime)
By default all images with baseline $<$ 100 m 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` (`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. If there is generally good coherence, this value can be increased (by editing `coreg.dorisin` in the `INSAR_master_date` directory) to make run times faster or, if coherence is particularly bad, the value can be decreased, though any cross-correlation with coherence below `0.12` is usually never correct.

When `make_coreg` has finished, check the size of the `CPM_Data` files in the `coreg` directory (`ls -l CPM_Data*`). View any which are around 1000 bytes or less, and if there are 12 lines or less, delete the file, as the coregistration for this pair has failed. After deletion, the inversion step must be rerun by entering `update_coreg` within the `INSAR_master_date/coreg` directory. Note that if all `CPM_Data` files associated with a particular slave are deleted, then there is a problem with that slave image, e.g. it is badly focused. Resolve the problem, then rerun `make_coreg` to recreate the `CPM_Data` files for this slave.

`make_dems` (long runtime - can run alongside `make_coreg`).

`make_resample`
After running, check the sizes of the resampled SLC images (`ls -l */*.slc`). They should be all identical. For any that differ, the slave crop does not include the entire master crop, probably due to a problem with coregistration for that slave. After resolving the problem(s), run `step_resample` in the slave subdirectory for the problem slaves only.

`make_ifgs`

`xv */*dem_Xl.ras` (`X` is number of range looks) and check that each interferogram looks OK (i.e., the amplitude looks reasonable and there is at least a little coherence apparent in the phase)

Finally, in (only) one of the slave directories run:
`step_geo` (calculates the latitude and longitude of each pixel)

5.2 Manual DEM offset correction

This step need only be run if `step_master_timing` failed. It should be run after `make_coarse` and before `make_dems`.

Choose a slave close in time and space. In the `yyyymmdd` subdirectory for the chosen slave:
`step_coreg`
`step_dem` (can be run alongside step_coreg)
`step_resample`
`step_ifg`

```
matlab -nojvm -nosplash
>>calc_dem_offset
```

This estimates the offset of the DEM range slope from the interferogram amplitude and displays the best-fitting result (DEM slope in blue, amplitude in red). Check that the offset is reasonable by zooming in on a few places. If not use `>>plot_amp_dem(dem_down,dem_right)` to adjust the offsets in azimuth and range until a better fit is achieved.

You can adjust `red_contrast` and `blue_brightness` (default 0.5 and 1) to vary contrast between amplitude image and DEM (see `>>help plot_amp_dem`)

Once happy with the fit, update the values for M_RG_T_ERROR and M_AZ_T_ERROR in dem.dorisin and geocode.dorisin (in the INSAR`_master_date` directory), by adding the values output by calc_dem_offset or plot_amp_dem.

5.3 Re-running Steps

`make_orbits` processes all images listed in `slcs.list`. Delete `slcs.list` to process all slave SLC images in the main `SLC` directory.

`make_coarse` processes all slave subdirectories in `make_ifgs.list`. Delete `make_ifgs.list` to process all subdirectories containing a `slave.res` file.

`make_dems, make_resample and make_ifgs` process all slave directories in `make_ifgs.list`. Delete `make_ifgs.list` to process all directories containing a `coreg.out` file.

`make_coreg` processes all slave subdirectories listed in `make_coreg.list` (in the `coreg` subdirectory), which is initially a copy from `make_ifgs.list`. Extra images can be added to the bottom of this file, but no lines should ever be deleted, as `n1` and `n2` in the `CPM_Data.n1.n2` files refer to the order of the files listed in `make_coreg.list`.

The following individual steps can be rerun in the individual `yyyymmdd` subdirectories of INSAR`_master_date`:
`step_orbit` extracts orbit info.
`step_coarse` coregisters coarsely.
`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_dem` creates the simulated dem interferogram.
`step_ifg` creates the final interferogram.

5.4 Possible reasons for Doris SIGERV error

- `master.res` or `slave.res` (as specified in the `.dorisin` file being run) 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 overlap the master cropped SLC.

5.5 Disk Space

Many intermediate files are produced and disk space requirements are therefore large (approximately 12.5 GB per image, if the whole image area is processed). Once step 1 of `stamps` matlab script has been run, the following may be run in `INSAR_master_date` to free up space (only easily recreatable files are deleted):

`make_clean_ifgs` To recreate the files deleted by this script, run `make_ifgs` in the `INSAR_master_date` directory.

`make_clean_resample` To recreate the files deleted by this script, run `make_resample` in the `INSAR_master_date` directory.

`make_clean_raw` (if you created SLCs using ROI_PAC). To recreate files deleted by this script, run `make_slcs_XXXX` in the `SLC` directory.

Chapter 6

PS Processing

First, create single master interferograms by following Chapter **??**.

In the `INSAR_master_date` directory run
`mt_prep 0.4 3 2 50 200` where

`0.4` = amplitude dispersion (0.4-0.42 are reasonable values)
`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 containing $<$ 5 million SLC pixels are OK.

The parameters that control the processing are set to default values which you can view with:

```
matlab
>>getparm
```

You can modify any parameters from the default using

```
>>setparm(‘param_name’,param_value)
```

Only enough characters of `param_name` to make it unique are required. Setting `param_value` to `nan` resets the parameter to the default value.

```
>>stamps
```

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

Steps 1 to 5 run by default on individual patches after which the patches are merged into one. Steps 6 to 8 run by default on the merged patch. It is also possible to run steps 6 to 8 on individual patches by setting the `patch_flag` to `‘y’`, e.g.,

```
>>stamps(6,8,‘y’)
```

6.1 Step 1: Load data

Converts the data into the formats required for PS processing and stores them in matlab workspaces.

6.2 Step 2: Estimate phase noise

This is an iterative step that estimates the phase noise value for each candidate pixel in every interferogram. Processing is controlled by the following parameters:

Parameter Name	Default	Description
`max_topo_err`	`10`	Maximum uncorrelated DEM error (in m). Pixels with uncorrelated DEM error greater than this will not be picked (this includes error due to the phase center of the resolution element being offset from the middle of the pixel in range). Setting this higher, however, increases the mean γ value (coherence-like measure, see **?**) of pixels that have random phase.
`filter_grid_size`	`50`	Pixel size of grid (in m). Candidate pixels are resampled to a grid with this spacing before filtering to determine the spatially-correlated phase.
`filter_weighting`	`'P-square'`	Weighting scheme (PS probability squared), the other possibility being `'SNR'`. Candidate pixels are weighted during resampling according to this scheme.
`clap_win`	`32`	CLAP (Combined Low-pass and Adaptive Phase) filter window size (**?**). Together with `filter_grid_size`, determines the area included in the spatially-correlated phase estimation.
`clap_low_pass_wavelength`	`800`	CLAP filter low-pass contribution cut-off spatial wavelength (in m). Wavelengths longer than this are passed.
`clap_alpha`	`1`	CLAP α term. Together with the β term, determines the relative contribution of the low-pass and adaptive phase elements to the CLAP filter.
`clap_beta`	`0.3`	CLAP β term
`gamma_change_convergence`	`0.005`	Threshold for change in change in mean value of γ (coherence-like measure). Determines when convergence is reached and iteration ceases.

6.3 Step 3: PS selection

Pixels are selected on the basis of their noise characteristics. Processing is controlled by the following parameters:

Parameter Name	Default	Description
`select_method`	`'DENSITY'`	Other option `'PERCENT'`.
`density_rand`	`20`	Maximum acceptable spatial density (per km^2) of selected pixels with random phase. Only applies if `select_method` is set to `'DENSITY'`. At this stage we can usually accept a high density, as most random-phase pixels will be dropped in the next step.
`percent_rand`	`20`	Maximum acceptable percentage of selected pixels with random phase. Only applies if `select_method` is set to `'PERCENT'`

6.4 Step 4: PS weeding

Pixels selected in the previous step are weeded, dropping those that are due to signal contribution from neighbouring ground resolution elements and those deemed too noisy. Data for the selected pixels are stored in new workspaces. Processing is controlled by the following parameters:

Parameter Name	Default	Description
`weed_alpha`	`8`	Smoothing parameter for estimating phase noise distribution for each pair of neighbouring pixels. The time series phase for each pair is smoothed using a Gaussian window with standard deviation 1/`weed_alpha`. The original phase minus the smoothed phase is assumed to be noise.
`weed_standard_dev`	`1.0`	Threshold standard deviation. For each pixel, the phase noise standard deviation for all pixel pairs including the pixel is calculated, If the minimum standard deviation is greater than the threshold, the pixel is dropped. If set to `10`, no noise-based weeding is performed

6.5 Step 5: Phase correction

The wrapped phase of the selected pixels is corrected for spatially-uncorrelated look angle (DEM) error. At the end of this step the patches are merged. Processing is controlled by the following parameters:

Parameter Name	Default	Description
`merge_resample_size`	`0`	Coarser posting (in m) to resample to. If set to `0`, no resampling is applied.

Check the wrapped phase of the selected pixels after running this step, e.g.,

```
>>ps_plot('w')
```

In terms of reprocessing, the first parameter to play with is `weed_standard_dev`. If it looks like too many noisy pixels are being chosen, the value can be reduced. If very few pixels are chosen, the value can be increased.

If still too few pixels are being selected such that any signal is generally undersampled, variation of Step 2 parameters can be tried. The number of initial candidates can also be increased by setting the amplitude dispersion higher in `mt_prep`.

6.6 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 neighbouring 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.
unwrap_gold_alpha	0.8	Value of α for Goldstein filter.

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

```
>>scla_reset
```

(This subtraction of SCLA and master AOE has not however 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 pixel, 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 have been unwrapped OK by running Step 7. If Step 6 is re-run after Step 7 has been run, the SCLA error phase is temporarily subtracted from the wrapped phase before unwrapping. The unwrapping accuracy is further improved by also temporarily subtracting the atmosphere and orbit error (AOE) phase of the master image, present in all the interferograms, which is also estimated in Step 7.

6.7 Step 7: Estimate spatially-correlated look angle error

Spatially-uncorrelated look angle (SULA) error was calculated in Step 3 and removed in Step 5. In Step 7, spatially-correlated look angle (SCLA) error is calculated which is due almost exclusively to spatially-correlated DEM error (this includes error in the DEM itself, and incorrect mapping of the DEM into radar co-ordinates). Master atmosphere and orbit error (AOE) 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.
scla_deramp	'y'	If set to 'y', a phase ramp is estimated for each interferogram. Other option is 'n'.

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

Display the estimate of master atmosphere and orbit error (AOE) phase with

```
>>ps_plot('m')
```

Display the phase ramps (if `scla_deramp` is set to `'y'`) with

```
>>ps_plot('o')
```

Unwrapped phase minus one of, or a combination of the above can be plotted with `'u-d'`, `'u-m'`, `'u-o'`, `'u-dm'`, `'u-do'`, or `'u-dmo'`.

After running Step 7, 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 (usually those with large perpendicular baselines). 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)
```

An index for interferograms with baselines smaller than, for instance, 200 m can be set with

```
>>[bperp,index]=ps_baselines(200)
```

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

Chapter 7

Small Baseline Processing

First, create single master interferograms by following Chapter **??**.

If PS processing has not been run, in the `INSAR_master_date` directory load baseline info into matlab workspaces with:

```
mt_extract_info
matlab
>>ps_load_info
```

To determine which small baseline interferograms to make, in the `INSAR_master_date` directory run:

```
matlab
>>sb_find
```

Adjust the input parameters according to your data set. 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:
`>>plot_sb_baselines`.

To create the small baseline interferograms listed in `small_baselines.list` in the `INSAR_master_date` directory, run:
`make_small_baselines` (or `make_small_baselines 1` to skip azimuth filtering)
This will create a new subdirectory called `SMALL_BASELINES` within the `INSAR_master_date` directory, containing a subdirectory for each small baseline interferogram.

Within the `SMALL_BASELINES` directory run
`mt_prep 0.6 3 2 50 200` where

`0.6`	=	amplitude difference dispersion (0.6 is reasonable)
`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)

Note that the first parameter is amplitude *difference* dispersion rather than amplitude dispersion as used for PS processing, and a higher value should be given, e.g., 0.6.

As for PS processing, small baseline MTI processing is controlled by various parameters that can be modified with `>>setparm` and processing is initiated with

```
matlab
>>stamps
```

Step 6 includes extra processing after phase-unwrapping to retrieve the phase with respect to the original master by least-squares inversion. Step 7 includes extra processing to calculate the SCLA error from both small baseline and single master interferograms. Step 7 is controlled by the following parameters:

Parameter Name	**Default**	**Description**
sb_recalc_index	'all'	Index to small baseline interferograms to be used in the SCLA estimation that is used to improve phase-unwrapping.
recalc_index	'all'	Index to interferograms to be used in the final SCLA estimation from single master interferograms.
scla_deramp	'y'	If set to 'y', a phase ramp is estimated for each interferogram. Other option is 'n'.

As for PS processing, repeating Step 6 after running Step 7 may improve phase-unwrapping accuracy. Accuracy can also potentially be improved by setting `unwrap_method` to `'3D'` (default is `'3D_QUICK'` for small baseline processing) before running Step 6, although this will take longer to run.

Unwrapped phase of small baseline interferograms can be viewed using `ps_plot` with the `'usb'` options. SCLA error can be plotted using the `'dsb'` option. Residuals between the unwrapped phase of the small baseline interferograms and that predicted from the model values for the single master phase can be plotted with the `'rsb'` option.

The residuals for each small baseline interferogram should be visually inspected, together with the unwrapped phase for each (N.B., you will probably want to view only a few at a time using the `IFG_LIST` option of `ps_plot`) . Isolated residuals less than π in magnitude are OK, but spatially-correlated residuals indicate an unwrapping problem in one or more interferograms. When this is the case, identify which interferogram(s) are incorrectly unwrapped (N.B., one badly unwrapped interferogram can cause non-zero residuals for many interferograms) and drop them from the unwrapping process, by setting `unwrap_ifg_index` and rerunning Step 6.

Once there are no more spatially-correlated residuals, check the unwrapped phase of each of the single master interferograms with `ps_plot('u-dm')`. 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`.

Chapter 8

Combined MTI Processing

Pixels selected by both PS and small baseline methods can be combined after Step 5 has completed for both methods. Data can be combined either for the whole area or for an individual patch, depending on whether you are in the `INSAR_master_date` directory or one of the `PATCH_X` subdirectories. Run

```
>>ps_sb_merge
```

This will create a new subdirectory `MERGED` in the `INSAR_master_date` directory, and a `PATCH_X` subdirectory within this, if you are merging only an individual patch.

For Step 6 onwards, processing is the same as for small baseline processing

Chapter 9

Plotting

The following matlab scripts can be used to plot the data in various ways (use `>>help` in matlab to see all options)

`>>plot_all_ifgs` plots all multilooked interferograms.

`>>ps_plot` plots values for each selected pixel, on various backgrounds, for chosen interferograms, e.g.,
`>>ps_plot('v-d',1,0,0,[1:3,7:8])` plots the mean LOS velocity corrected for DEM (SCLA) error, on a white background, calculated from the unwrapped phase of interferograms 1 to 3 and 7 to 8.

`>>ps_plot_ifg` Plots a value for each selected pixel, on various backgrounds.

To change the size of the point representing each selected SLC pixel (default 5 image pixels) change the `plot_pixel_size` parameter, e.g., `>>setparm('plot_p',9)`

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. If not set, the reference value is the mean value for the whole area.

Normal phase sign conventions apply: if the master predates the slave, positive phase implies movement *away* from the satellite.

For plotting of velocities, the units are mm/year with positive values being *towards* the satellite (in versions prior to v3.1, the sign convention for velocity was the opposite).

Data is output to ascii files with
`>>ps_output`

`plot_v.gmt` can then be run to plot mean LOS velocities on the DEM in shaded relief, using GMT. Copy (from `$STAMPS/bin`) and edit the script to adjust plotting parameters.

Chapter 10

Change History

N.B. This list is not comprehensive.

10.1 Version 1.0

- Initial beta release.

10.1.1 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.

10.2 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.

10.2.1 Version 2.0.1

- Some bug fixes.

10.2.2 Version 2.0.2

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

10.2.3 Version 2.0.3

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

10.3 Version 2.1

- Estimation of spatially-correlated look angle (DEM) error and master atmosphere and orbit error added (step 7).
- Phase-unwrapping (step 6) now uses estimates from step 7 if present.
- Merging of patches made into an explicit step (step 8). This change was reversed in Version 2.2.
- Estimation of spatially-correlated noise moved to step 9. This change was reversed in Version 2.2.
- `ps_info` added

10.4 Version 2.2

- Updated for ROI_PAC version 3.0 compatibility.
- Crop definition simplified in SLC creation.
- Extra integrity checks added.
- Manual updated with more instructions on error checking during interferogram production.
- `make_orbits` split into `make_orbits` and `make_coarse`, with corresponding `step_orbit` and `step_coarse` scripts added.
- Automatic DEM offset estimation added.
- Optional weeding of pixels with zero elevation added.
- Merging of patches now done after step 5 (implicitly).
- Unwrapping now uses snaphu optimisation routines to search for the minimum cost solution (costs still estimated by StaMPS).
- Merge of mean amplitude image added allowing `ps_plot` to plot on amplitude background for merged patches.

10.5 Version 3.0

- Small baseline and combined combined time series methods added.

10.6 Version 3.1

- Compatibility with Doris v4.0 added.
- Support for ERS and RADARSAT-1 SLCs, and ALOS raw data added.
- Option to estimate phase ramps added.
- Isolated images allowed in small baseline processing.
- Simultaneous estimation of mean velocity added to SCLA estimation.
- SCLA smoothed before subtraction for unwrapping purposes.
- Selection criteria for small baseline interferograms amended to be coherence based.
- Option added to `ps_plot.m` to plot incremental phase change for each date.
- Option added to `ps_plot.m` to plot mean velocity standard deviations.
- Sign convention for plotting mean velocities flipped.
- ps_baselines.m added.

10.7 Version 3.2

- Support for TerraSAR-X SLCs added.
- New option (and new default for small baselines) to resample to coarser sampling during merging of patches (using parameter `merge_resample_size`). This greatly reduces memory needs and processing time for Steps 6 onwards.
- New option (and the new default) to select pixels based on spatial density of random phase pixels, rather than percentage (relevant parameters: `select_method` and `density_rand`). This ensures even sampling over all patches.
- Weighting added to the inversion for single master phase from small baseline phase, using the variance-covariance of the wrapped phase.
- Precise orbit processing corrected for ERS SLCs.
- Plotting on amplitude improved.
- Magnitude raster files created automatically during read steps.
- Timing estimation more robust and easily re-runnable.
- Option added to `make_small_baselines` to skip range filtering (to skip azimuth filtering was already an option).
- Memory needs reduced during merging of patches, and combining of amplitude and dem.
- Program `display` no longer compiled by default.
- Parameter added to control amount of Goldstein filtering during unwrapping (`unwrap_gold_alpha`).
- Code to use translation lines between slave and master in slave-slave coregistration added back.
- Bug fixed that deletes timing info from master during resample.
- Bug fixed that drops master from calibration.
- Bug fixed that outputs warnings for missing process control flags.

Bibliography

Hooper, A., A multi-temporal InSAR method incorporating both persistent scatterer and small baseline approaches, *Geophys. Res. Lett.*, *35*, L16,302, 2008.

Hooper, A., and H. Zebker, Phase unwrapping in three dimensions with application to InSAR time series, *J. Opt. Soc. Amer. A*, *24*, 2737–2747, 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. Lett.*, *31*, 2004.

Hooper, A., P. Segall, and H. Zebker, Persistent scatterer InSAR for crustal deformation analysis, with application to Volcán Alcedo, Galápagos, *J. Geophys. Res.*, *112*, 2007.

Hooper, A. J., Persistent scatterer radar interferometry for crustal deformation studies and modeling of volcanic deformation, Ph.D. thesis, Stanford University, 2006.

Links to PDF files for the references above can be found at:
`http://enterprise.lr.tudelft.nl/~ahooper/pubs.html`