# Release Notes GAMMA Software, 20240701

Urs Wegmüller, Christophe Magnard, Charles Werner, Andreas Wiesmann
Gamma Remote Sensing AG
Worbstrasse 225, CH-3073 Gümligen
http://www.gamma-rs.ch
1-Jul-2024

## Introduction

This information is provided to users of the GAMMA software. It is also available online at https://www.gamma-rs.ch/uploads/media/GAMMA_Software_upgrade_information.pdf.

This release of the Gamma software includes new programs that provide new capability, additional features to existing programs and bug fixes.

## Gamma Software on Linux, macOS, and Windows

The Gamma software has been compiled and tested on Linux (different distributions), Apple macOS Sonoma (14.5), and Windows 10 and 11. Computationally intensive programs such as used in co-registration and resampling and geocoding have been parallelized using the OpenMP API built into the GCC compiler. Processing speed on Linux, macOS, and Windows systems is comparable.

### *Linux Distribution:*

The Gamma software is developed on Ubuntu 22.04 LTS 64-bit Linux and is tested extensively with this distribution. The Gamma software is also available for Ubuntu 24.04 LTS.

Versions of the Software will also be uploaded for RHEL7 based on CentOS7, RHEL8 based on Rocky Linux 8, and RHEL9 based on Rocky Linux 9.

Note for ***CentOS7***: Some Gamma Software programs now require a GDAL version newer than the version provided in CentOS7 standard repository. Hence, newer versions of GDAL and PROJ have to be installed. The installation method is described in the installation instructions (INSTALL_linux.html).

Announcement: This is the last time RHEL7 / CentOS7 will be supported.

Note that Red Hat has ended support for CentOS8 at the end of 2021. Consequently, the Gamma software built for RHEL8 is now built on Rocky Linux 8.

For installation instructions for the binary LINUX distributions see the HTML file INSTALL_linux.html (provided with the distribution E-mail or found in the main directory of the distribution).

### *Apple MacOS Distribution:*

The software in this version has been compiled using macOS Sonoma (14.5). You will need to install libraries such as GDAL using MacPorts. The build uses GCC 13 compiler. Version are available for Apple Silicon (M1/M2) and Intel x86_64 processors.

For installation instructions for the binary macOS distributions see the HTML file INSTALL_macOS.html (provided with the distribution E-mail or found in the main directory of the distribution).

***Windows Distribution:***

The Windows version of the Gamma software is compiled with 64-bit support and multi-threaded. The build uses the MINGW64 GCC 14 compiler.

For installation instructions for the binary Windows distributions see the HTML file INSTALL_win64.html (provided with the distribution E-mail or found in the main directory of the distribution). Notice that installing the latest GAMMA_LOCAL_w64 version is mandatory because a new GCC compiler and new libraries were used to build the software. Furthermore, the .bashrc file needs to be updated following the installation instructions.

The Gamma Plugin for ArcGIS is available in all Windows distributions that include the *GEO* or *ISP/DIFF&GEO* modules. Full functionality requires access to the *LAT* module.

On both Windows 10 and 11, it is also possible to install the Windows Subsystem for Linux (WSL2) and run a Linux distribution of the Gamma software on that environment. Instructions for this setup are available in the HTML file INSTALL_wsl.html located in the main directory of the distribution.

**Documentation and Program List**

The Gamma documentation browser is an HTML based system for viewing the web pages and pdf documents. The documentation browser includes for each module a Contents sidebar on the right side of the screen and a search functionality. The main Gamma documentation browser page *Gamma_documentation.html* is found in the main software directory.

The program *gamma_doc* facilitates the access to the documentation related to a given module or program:

| | |
|---|---|
| *gamma_doc* | Opens the main page of the Gamma documentation browser and shows the program list. |
| *gamma_doc DIFF* | Opens the DIFF&GEO documentation. |
| *gamma_doc gc_map2* | Opens the reference manual web page for *gc_map2*. |

Further information related to the GAMMA Software is available online:

General information:
gamma-rs.ch/uploads/media/GAMMA_Software_information.pdf
Technical reports, conference and journal papers:
gamma-rs.ch/uploads/media/GAMMA_Software_references.pdf
Release notes / upgrade information:
gamma-rs.ch/uploads/media/GAMMA_Software_upgrade_information.pdf

In case the program list is incomplete, run the python script program_list.py after successful installation of the Gamma Software in the main folder of the Gamma Software distribution:

```
./program_list.py Gamma_documentation_base.html Gamma_documentation_contents_sidebar.html -a
```

**Python and Matlab wrappers**

The Gamma Software is integrated into Python and Matlab through wrappers. Gamma Software program calls become Python / Matlab function calls where command line arguments can be used as function arguments, and system outputs can be stored in variables or written to log files. Binary images, point lists and data, parameter files, tab files, can be easily read, inspected, and written using additional functions provided with the wrappers.

The *py_gamma* Python module permits a smooth usage of the Gamma Software within Python scripts as well as within a Python Interactive Development Environment (IDE) such as Spyder or PyCharm or using Jupyter Notebooks. Using *py_gamma*, function arguments can be entered either as positional arguments or as keyword arguments, with the Gamma command line parameter names becoming the keyword names.

In the same way, the Matlab (and Octave) wrapper, composed of *mat_gamma* and *par_file* classes, permits a smooth usage of the Gamma Software within an interactive use of Matlab as well as within Matlab scripts.

### Gamma plugin for ArcGIS

The Gamma plugin for ArcGIS permits using some Gamma software functionalities (tools) from ArcGIS Pro (Windows only) using a convenient interface. The Gamma plugin allows to perform the following operations:

- Reading SAR data from various sensors / formats
- Detection, radiometric calibration and geocoding of SAR data
- Co-registration of SLC and MLI SAR images in slant range / azimuth geometry
- Adaptive interferometric coherence estimation
- Multi-temporal processing and filters
- Spatial filtering of 2D SAR images
- Change detection in SAR images
- Polarimetric decompositions

Using ArcGIS ModelBuilder, it is possible to generate dedicated processing chains using the Gamma tools as building blocks. It is also possible to use each Gamma tool as a Python function (ArcGIS Pro is required). Note that the LAT module is required to be able to use all the tools; without the LAT module, only a subset of the tools is available.

See also gamma-rs.ch/uploads/media/2024-1_Gamma_Plugin_for_ArcGIS_presentation.pdf.

### Hardware Recommendations

Using multi-core processors (8 or more cores) will bring substantial improvement in processing speed due to parallelization of the code base. There should be at least 8 GB RAM available for each processor core with 16 GB per core recommended. Disk storage requirements for using the Gamma Software effectively depend on the amount of input data and data products that will be produced. Based on our experience we recommend considering at least 16 TB space, especially when working with stacks of Sentinel-1 or very high-resolution data (TerraSAR-X, Cosmo-Skymed) data. The current trend towards larger data products requires substantially increased storage capacities.

### GAMMA Software Training Courses

A SAR/INSAR (MSP/ISP/DIFF&GEO/LAT) training at GAMMA (near Bern, Switzerland) is planned for 4. – 7. Nov. 2024.

A PSI (IPTA) training at GAMMA (near Bern, Switzerland) is planned for 19. – 22.Nov.2024. See also https://www.gamma-rs.ch/software/training.

## Significant Changes in the Gamma Software Modules since the end of 2023 Release

### *New script to read Sentinel-1 TOPS SLC data: read_S1_TOPS_SLC.py*

A new script was implemented to read, select and concatenate Sentinel-1 TOPS SLC data, which replaces *S1_import_SLC_from_zipfiles*.

*read_S1_TOPS_SLC.py* supports reading Sentinel-1 data from a ZIP file, a directory, a list of ZIP files, or a list of directories. The input type is automatically detected.

Bursts can be selected using a reference *burst_number_table* file, a reference *SLC_tab*, a *BURST_tab* file containing the first and last burst number for each subswath, or a *KML* file containing georeferenced polygon corner coordinates. Again, the type of the file provided for data selection is automatically detected.

For a quick examination of a Sentinel-1 data take, an option permits extracting the metadata without writing binary files.

Options permit replacing the state vectors by precise state vectors, selecting the subswaths to read, the polarization, the output format, the root name of the output files, and the output directory. An option offers the possibility to generate one KML per subswath showing the coverage of the selected bursts.

The functionalities of *read_S1_TOPS_SLC.py* also make calling *S1_BURST_tab_from_zipfile.py* unnecessary.

*read_S1_TOPS_SLC.py* also supports reading Sentinel-1 data produced using the new "burst2safe" tool of ASF (still in development). That tool allows to download only the bursts you're interested in. It also permits downloading a whole time series. (see https://github.com/forrestfwilliams/burst2safe).

**Note that *S1_import_SLC_from_zipfiles* calls in scripts will have to be replaced.** The equivalent call with *read_S1_TOPS_SLC.py* is shown but not executed when calling *S1_import_SLC_from_zipfiles*.

### *Ionosphere mitigation support for PALSAR3 and NISAR-L*

The launches of the two L-band SAR systems PALSAR3 (on-board ALOS-4) and NISAR-L (on-board NISAR) are scheduled in 2024. Both these sensors will provide, as a standard product, acquisitions consisting of an SLC for the main special band together with an additional SLC for a separated, narrower spectral band. The main objective of providing the additional narrow band SLC is to support the mitigation of ionospheric effects. Using a separated narrow spectral band provides a better sensitivity to ionospheric effects without increasing the data volume too much.

We adapted the split-spectrum ionosphere mitigation methodology so that it can support both the “old case” with a single spectral band and the “PALSAR3 and NISAR-L case” with two separate spectral bands. To test it, we generated simulated PALSAR3 data using a PALSAR2 SM1 mode data set with 80MHz bandwidth. The work is described in the open access article Wegmüller, U.; Werner, C.; Frey, O.; Magnard, C. Estimation and Compensation of the Ionospheric Path Delay Phase in PALSAR-3 and NISAR-L Interferograms. Atmosphere 2024, 15, 632. https://doi.org/10.3390/atmos15060632.

The new ISP program *ionosphere_mitigation_factors.py* permits calculating the scaling factors used in the split-spectrum methods based on the SLC parameter files. We also generated a new demo example that shows and discusses the mitigation procedure using the simulated PALSAR3 data.

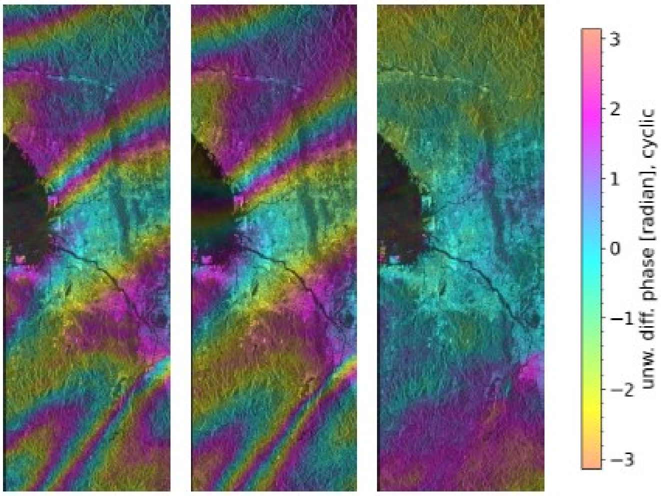
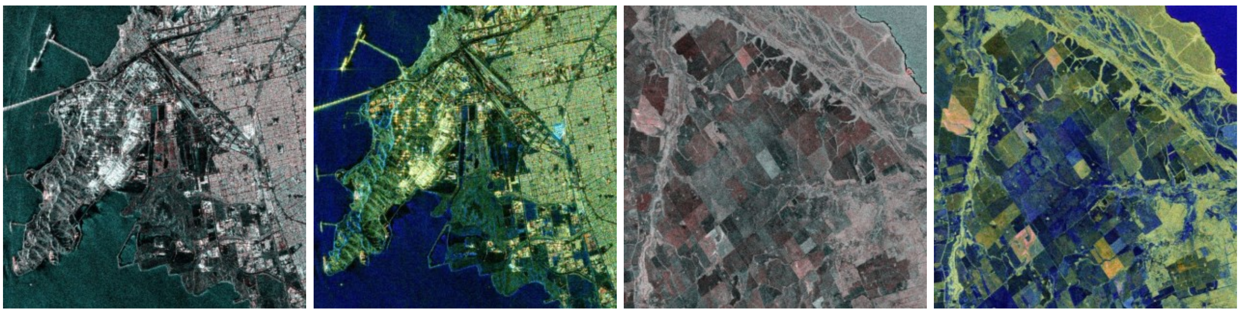



Figure 1 Differential interferogram phase for the main 28 MHz PALSAR3 band (left), estimated dispersive path delay phase (center), and non-dispersive or ionosphere-corrected phase (right).

***Compact polarimetry decompositions***

Support for compact polarimetry decompositions was complemented with the new Python program ***compact_pol_decomposition.py***, which permits conducting several polarimetric decompositions in a single step, starting from the compact or dual polarization SLC. The intention is to strongly facilitate the access to this functionality. The new program makes it easy, for example, to apply several decompositions and to compare the results. To facilitate the intercomparison, we harmonized the scaling used (now primarily intensities and not magnitudes). In addition, the related demo has been updated ("Gamma_Polarimetry_demo", see below), so that it also demonstrates both the individual commands and the use of ***compact_pol_decomposition.py***. Some of the demo results are shown in Figure 2.



a) PALSAR-2, RH-RV-RV b) PALSAR-2, m-alpha c) SAOCOM, RH-RV-RV d) SAOCOM, m-chi

Figure 2 For quad.-pol. PALSAR-2 and SAOCOM data right-circular pol. transmit / horizontal pol. receive (RH) and right-circular pol. transmit / vertical pol. receive (RV) channels were generated and used to test the compact polarimetry decompositions. (a) and (c) show for a small image section RGB composites of the compact pol. input channels. (b) and (d) show one of the compact pol. decompositions. The results for the three decompositions m-alpha, m-chi and m-delta look very similar.

***Co-registration of ScanSAR and TOPS burst SLC data***

New options have been added to the script *ScanSAR_coreg.py* for the co-registration of ScanSAR and TOPS burst SLC data and to its relatives *ScanSAR_coreg_pol.py* and *ScanSAR_coreg_stack.py*.

Options --*r_only* and --*az_only* permit applying the calculated intensity matching polynomial only in range and azimuth directions, respectively.

When using precise orbit data, we now highly recommend performing both intensity and spectral diversity matching procedure and applying the calculated intensity matching polynomial in range direction only.

When not using precise orbit data, we recommend performing both intensity and spectral diversity matching procedure and applying the calculated intensity matching polynomial in both directions.

The new option *--poly_sd* permits applying either a constant (as previously) or a linear (azimuth-dependent) correction based on spectral diversity method. This may be useful in case of long acquisitions, e.g., when performing co-registration of concatenated frames.

***WGS84 heights***

Several programs in the Gamma Software use height maps or point data heights as input. In many cases, e.g., when simulating a topographic phase, the height needs to be relative to the WGS 84 ellipsoid. **The usage of the programs that use height maps or point data heights as input will indicate if the heights must be relative to the WGS 84 ellipsoid or not.**

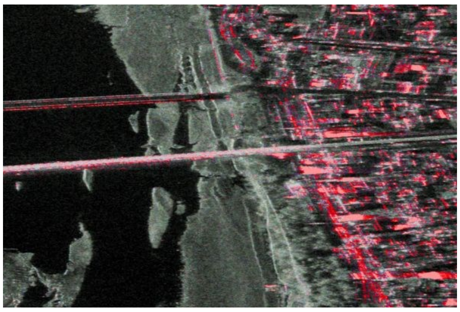
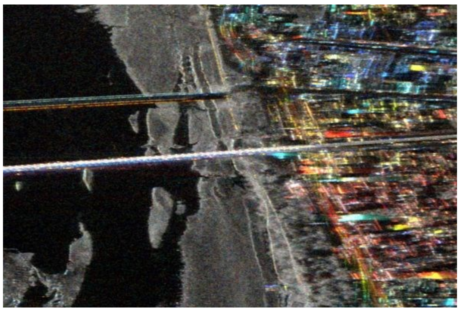
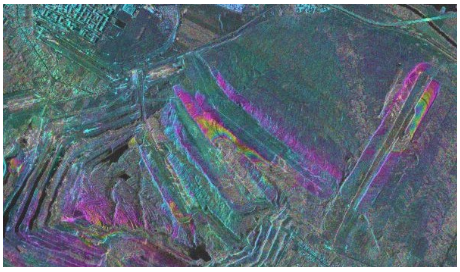
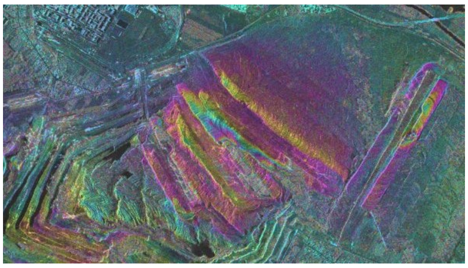
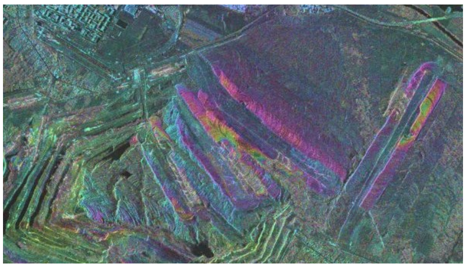
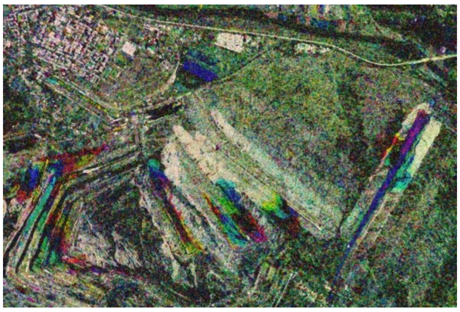
In this release, we added some tools to simplify conversion of heights relative to a local height reference into heights relative to the WGS 84 ellipsoid.

The program *geocode_hgt* has been added to convert a DEM into a height map in RDC geometry; an option permits selecting the height reference of the output height map. Using the new option *hgt_wgs84*, *pixel_area* now permits generating an output height map containing heights relative to the WGS 84 ellipsoid. The script *geocoding.py* now generates 2 height maps in RDC geometry, with heights relative to the input height reference and relative to the WGS 84 ellipsoid.

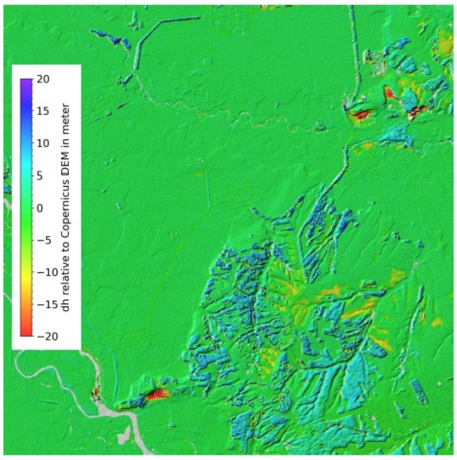
As a remainder (no change here), the IPTA program *pt2geo* permits converting heights relative to a local height reference into heights relative to the WGS 84 ellipsoid.

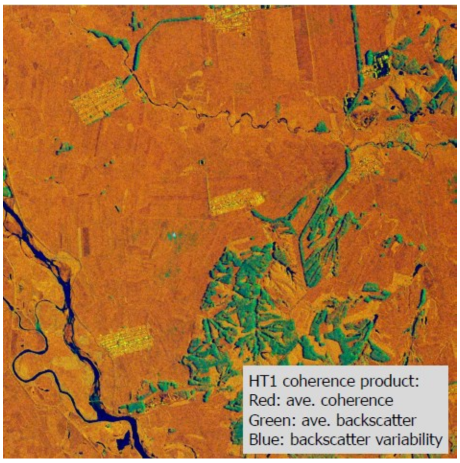
***New and updated SAR data readers***

| New / updated SAR data reader | Short description |
|---|---|
| ***par_NISAR_RSLC*** | The program supports reading NISAR RSLC data in HDF5 format. It produces one or multiple SLC parameter file(s) and SLC data file(s).<br>The user can either read all data provided in the HDF5 file or select which data should be read, by selecting the radar band (L/S), the frequency in case of split beam imaging (A/B) and/or the polarization (HH, HV, RH, RV, VH, or VV). The user can also select to only write the parameter files.<br>The following radiometric calibrations are available: none (output in as DN values), beta0, sigma0 (ellipsoid), and gamma0 (ellipsoid).<br>The output format is FCOMPLEX.<br>Note that the program was tested using simulated products (based on PALSAR 1/2 data). |

| | |
|---|---|
| ***par_ICEYE_SLC, par_ICEYE_GRD***<br><br>ICEYE SLED HSI composite showing high azimuth subband average to azimuth subband median ratios in red color (→ indicating narrow directional scattering).<br><br>ICEYE SLED RGB composite of azimuth subband contributions. | ICEYE SLED-DWELL data readers: Reading ICEYE extended Spotlight mode SLCs is supported by *par_ICEYE_SLC.*<br>*par_ICEYE_GRD* was updated to also support reading Colorized Sub-aperture Images (CSI) and SAR Video (VID) products. In these cases, a single common parameter file is generated, and one image for each layer included in the GeoTIFF, e.g., R, G, and B images for CSI data. Some additional information about the different layers such as UTC time of the start and end of the acquisition, frame time and duration, and frame colorization, is provided in the standard output.<br>In the ICEYE SLED-DWELL mode the SAR sensor keeps looking at the same location on the ground for a very long section (or time) along the orbit, which results in a very long synthetic aperture and consequently a very high azimuth resolution. Over the synthetic aperture the squint angle changes. It changes by more than the angular width of a typical narrow directional scattering effect (e.g. from a building or a powerline). Consequently, different azimuth subbands are affected by different directional scattering directions. The median over the azimuth subband backscatter images, on the other hand, is not much affected by directional scattering (present for each scatter direction only in a minority of subbands).<br>The different subband images also correspond to acquisition times differing by some seconds. Consequently, moving targets may in principle be identified. However, this is not straight forward, as moving targets may be poorly focused and spatially misplaced. |
| ***par_Fucheng_SLC***<br><br>DInSAR 1.10.-12.10.2023, B⊥ -41m.<br><br>DInSAR 12.10.-23.10.2023, B⊥ 84m.<br><br>DInSAR 23.10-3.11.2023, B⊥ 19m. | Chinese Fucheng-1 C-band interferometric SAR<br>The program supports reading SLC data of the Chinese C-band SAR Fucheng-1. Fucheng-1 operates in a 11-day repeat cycle with a narrow orbital tube.<br>The 3 differential interferograms shown to the left are for subsequent 11-day intervals over the Rovinari mining area in Romania. For the topographic phase simulation the Copernicus 1” DEM was used, which may be outdated for the mining area. Consequently, the phase is expected to correspond partly to residual topographic phase and partly to displacement phase. Ongoing activity is confirmed by the multi-temporal coherence RGB composite shown below (for the same 3 intervals with red corresponding to the first, green to the second and blue to the third interval).<br><br>Multi-temporal Fucheng-1 coherence |

***par_HT1_SLC***




Chinese Hongtu-1 X-band multi-static constellation

The HT1 reader program *par_HT1_SLC* supports reading mono- and bistatic SLCs of multi-static L-band Hongtu-1 acquisitions. A multi-static acquisition includes one monostatic SLC (HT1A is the active sensor) and 3 bi-static scenes (transmitted by HT1A, received by HT1B, HT1C, and HT1D). With a multi-static acquisition six differential interferograms can be calculated. In the data we had available the perpendicular baselines were between 60m and 506m.

Using the phase of the multi-static differential interferograms we determined a height correction relative to the Copernicus 1-arc-second DEM. Difference between about -20m and +20m were observed corresponding to changes related to the finer spatial resolution used, forest height, changes in the forest canopy, and terrain changes (e.g. landfill). Also interesting is the coherence of the multi-static differential interferograms. Decorrelation relates mainly to volume scattering effects (geometric decorrelation) and signal noise (in low backscatter areas as open water surfaces and radar shadow). The coherence over forest decreases with increasing perpendicular baseline.

Geocoded multi-static DInSAR Hongtu-1 (HT1) DEM update (top) and coherence product (bottom).

## *Gamma Software Demo examples*

The access to the Gamma Software Demo examples is limited to Gamma Software users with a valid license. The access information is provided with the software delivery. A list of the Demo examples is available here:

*https://gamma-rs.ch/uploads/media/README_Gamma_Software_demo.html*.

| New / modified demo example: | Contents |
|---|---|
| Gamma_Polarimetry_demo. | The PALSAR-2 and SAOCOM parts of the polarimetry demo were updated to also demonstrate the updated compact polarimetry decomposition tools. |
| PALSAR3_iono_demo | This new demo uses simulated PALSAR3 data with SLCs for two separate spectral bands, the main 28MHz bandwidth band and an additional 10MHz band available for the ionosphere mitigation.<br>For the demo an interferometric pair over Osaka, Japan is used. One of the two acquisitions of the pair is affected by significant ionospheric effects. Methods to identify the presence of ionospheric effects for the individual SLC and the interferometric pair are shown. To generate a useful differential interferogram with this pair the second image (which is affected by ionospheric effects) is co-registered to the first (reference) scene using a refinement step with an offset field. Then the dispersive and non-dispersive phase components are determined using the split-spectrum method described in Wegmüller, U.; Werner, C.; Frey, O.; Magnard, C. Estimation and Compensation of the Ionospheric Path Delay Phase in PALSAR-3 and NISAR-L Interferograms. Atmosphere 2024, 15, 632. https://doi.org/10.3390/atmos15060632. |

| | |
|---|---|
| Gamma_demo_ICEYE_SLED | In this demo we show how ICEYE SLED SLC azimuth sub-bands can be read, generated and analyzed. |
| Gamma_demo_ArcGIS | This demo example shows how to use the tools provided in the Gamma Plugin for ArcGIS. The example can be run either using the graphical interface of the tools or by running the tools a as Python functions in a Jupyter Notebook. A model prepared using ArcGIS ModelBuilder is also provided to perform one of the processing steps.<br>The demo example uses a set of Sentinel-1 images acquired over Switzerland and includes the following processing steps: reading the data, geocoding, co-registration, estimation of the interferometric coherence, multi-temporal filtering, calculation of temporal average and variability, generation of composite images, radiometric calibration, and change detection. |
| IPTA_demo_S1_Aletsch_from_orig | The demo example can now also be carried out through a Jupyter Notebook, and has been updated to take advantage of the latest version of the Gamma Software. |

### *MSP*

*az_proc:* Added vu_flg to permit updating the effective velocity along-track using velocity parameters (constant and rate) determined by *af*. Each patch is then processed using a velocity parameter that is updated. This is particularly useful for processing of data with ionospheric phase gradients.

### *ISP*

*S1_BURST_tab_from_zipfile.py*: New option --out_dir to specify the output directory. Now also supports directories as inputs in addition to the Sentinel-1 ZIP archives. The directories must be decompressed Sentinel-1 ZIP archives with their original names.

*read_S1_TOPS_SLC.py*: New script to read, select and concatenate S1 TOPS SLC data, replaces *S1_import_SLC_from_zipfiles*. Supports reading Sentinel-1 data from a ZIP file, a directory, a list of ZIP files, or a list of directories.

Bursts can be selected using a reference burst_number_table file, a reference SLC_tab, a BURST_tab file or a KML file containing georeferenced polygon corner coordinates.

Options permit replacing the state vectors by precise state vectors, selecting the subswaths to read, the polarization, the output format, the root name of the output files, and the output directory. Option permits generating one KML per subswath showing the coverage of the selected bursts. For a quick examination of a Sentinel-1 data take, an option permits extracting the metadata without writing binary files.

Also supports reading Sentinel-1 data produced using the new "burst2safe" tool of ASF (still in development). That tool allows to download only the bursts you're interested in. It also permits downloading a whole time series. (see https://github.com/forrestfwilliams/burst2safe)

*SLC_copy_ScanSAR*: Can now also be used without input SLC binary data: in that case, only parameter files will be generated.

*SLC_cat_ScanSAR*: New option [bin_flag]: when set to 0, no binary files are generated, only the parameter files are concatenated.

*LT1_precision_orbit.py*: New program to extract LT1 precision state vectors and write these to an SLC/MLI parameter file.

*par_NISAR_RSLC*: New program to generate SLC parameter and image files for NISAR Level-1 RSLC data.

*par_Fucheng_SLC*: New program to generate SLC parameter and image files for Spacety Fucheng SLC data.

*par_HT1_SLC*: New program to generate SLC parameter and image files for HT1 / Hongtu-1 / PIESAT-1 SLC data. The program supports reading both the mono- and bi-static SLCs.

*par_ICEYE_GRD*: Now also supports reading Colorized Sub-aperture Images (CSI) and SAR Video (VID) products. In these cases, a single common parameter file is generated, and one image will be generated for each layer included in the GeoTIFF, e.g., R, G, and B images for CSI data. Some additional information about the different layers is provided in the standard output: UTC time of the start and end of the acquisition, frame time and duration, frame colorization.

*SLC_interp_lt_ScanSAR*: When option burst_check is set to 1, the first / last valid sample / line parameters as well as the extended burst window parameters in the output TOPS parameter file will be checked and updated according to the actual data: the program will inspect the data and determine the largest rectangle containing valid data. This may be useful for some ScanSAR data with large burst overlaps.

*run_all, make_tab*: Placeholders for column numbers >= 10 can now be used, in that case, the column number must be entered between curly braces, e.g., ${17}.

*ionosphere_check*: New option [cleaning] to keep or delete intermediate data.

*ionosphere_mitigation_factors.py*: New program to calculate scale factors used in ionosphere mitigation methods (Wegmüller, U.; Werner, C.; Frey, O.; Magnard, C. Estimation and Compensation of the Ionospheric Path Delay Phase in PALSAR-3 and NISAR-L Interferograms. Atmosphere 2024, 15, 632. https://doi.org/10.3390/atmos15060632.)

## *DIFF&GEO*

*pixel_area*: Now shows the estimated memory usage and allows setting a different number of threads using the environment variable "PIXEL_AREA_NUM_THREADS".

New output height map option [hgt_wgs84] with heights relative to the WGS-84 ellipsoid.

*geocode_hgt*: New program to convert a DEM from map to radar coordinates using a forward geocoding transformation. Output heights can be either relative to the WGS-84 ellipsoid, or relative to the local ellipsoid as defined in the input DEM parameter file.

*geocoding.py*: Output height maps in RDC geometry are now provided both with heights relative to the local ellipsoid (as defined in the input DEM parameter file) and with heights relative to the WGS-84 ellipsoid.

*CARD4SAR.py*: New script to produce CEOS ARD-SAR XML files.

*SLC_interp_ScanSAR*: When option burst_check is set to 1, the first / last valid sample / line parameters as well as the extended burst window parameters in the output TOPS parameter file will be checked and updated according to the actual data: the program will inspect the data and determine the largest rectangle containing valid data. This may be useful for some ScanSAR data with large burst overlaps.

*ScanSAR_coreg.py, ScanSAR_coreg_pol.py, ScanSAR_coreg_stack.py*: New options --r_only and --az_only to apply intensity matching polynomial only in range and azimuth directions, respectively.

New option --poly_sd to apply either a constant (as previously) or a linear (azimuth-dependent) correction based on spectral diversity method.

When option --burst_check is selected, the first / last valid sample / line parameters as well as the extended burst window parameters in the output TOPS parameter file will be checked and updated according to the actual data: the program will inspect the data and determine the largest rectangle containing valid data. This may be useful for some ScanSAR data with large burst overlaps.

*par_EORC_PALSAR_geo, par_JERS_geo*: Now support geocoded data in polar stereographic coordinates.

*typedef_DIFF.h*: Addition of EPSG keyword to DEM parameter files: either EPSG number of the map projection or EPSG number of the geographical coordinate system in case of EQA coordinates.

### *LAT*

*m_alpha, m_chi, m_delta.c, ras_m_chi*: The programs *m-alpha, m-chi, m-delta*, and *ras_m-chi* were renamed to *m_alpha, m_chi, m_delta*, and *ras_m_chi* for compatibility with Python and Matlab.

*compact_pol_decomposition.py*: New program to calculate m-chi, m-alpha, and m-delta decomposition from compact-pol or dual-pol SLC data. The script generates the 3 output elements of the selected decomposition as well as an RGB composite (as bmp file).

### *DISP*

*get_data_values*: Now supports various separators between the coordinates in the positions text file.

### *IPTA*

*phase_sim_pt*: Parallelized using OpenMP.

*mcf_pt, expand_data_inpaint_pt*: Parallel computation of data records is now possible when multiple records are being processed. The allocated memory is proportional to the number of threads. In case of excessive memory usage, setting the new option [parallel_flg] to 0 will disable this feature. Alternatively, the number of threads can be changed using the environment variable OMP_NUM_THREADS.

*def_mod, multi_def*: Option [def] was renamed to [def_rate] to avoid conflicts with function definition in Python.

### *TDBP (Time-Domain Back-Projection)*

The TDBP programs are available for Ubuntu 22.04 Linux and Windows 11 WSL2 with Ubuntu 22.04 Linux (and legacy 20.04 Ubuntu Linux and Windows 11 WSL2 versions).

The time-domain back-projection focusing program *az_proc_tdbp_gpu* implemented in C/CUDA requires an NVIDIA GPU for parallelized TDBP imaging.

The TDBP module supports image focusing of SAR data from airborne platforms and mobile-mapping platforms such as UAVs and cars, or trains, with high-quality 3-D geometry/motion compensation. Subsequent interferometric/tomographic processing and value-adding is supported. See also our ground motion/slope stability examples acquired with the car-borne Gamma L-band SAR system and processed with the TDBP module: https://www.gamma-rs.ch/L-band-specific-Publications.

***Python wrapper***

The support for Gamma program calls using keyword parameters has been further improved.

***Matlab wrapper***

-

***GIS***

With the end-of-2023 distribution the additional module *GIS* has been included. All the users with a valid license and running maintenance for the *GEO* or a combination of modules that includes the *ISP/DIFF&GEO* obtain the new additional module *GIS*. Notice that many of the programs of the *GIS* module require access to the *LAT*. Users without the *LAT* will not see the tools requiring *LAT* programs.

The *GIS* programs compose the "***Gamma Plugin for ArcGIS Pro***". This plugin supports running SAR data specific tools in the ArcGIS Pro environment (Windows only). The GIS module consists of several text files (README file, Change log file) and Python programs. The text files are in the main GIS directory, the Python programs and the related documentation (xml files that can be accessed within ArcGIS Pro) are located in the subdirectory scripts. The text file README_GIS is a listing of the contents of the GIS module.

*arc_pol_decomposition.py*: New tool added to the Gamma Plugin for ArcGIS to calculate polarimetric decompositions for quad-pol and compact-pol data. The following decompositions are available: Pauli, H/A/alpha, H/A/alpha eigenvalues, Cloude, Freeman-Durden, Krogager, m-chi, m-alpha, and m-delta.

*arc_read_data.py*: The same data selection options are now available for Sentinel-1 TOPS SLC data from zip file(s) and from directory(ies): selection and concatenation of multiple consecutive acquisitions as well as subswath and burst selection using various methods.

The following additional sensors are now supported: Fucheng, HT1/Hongtu-1/PIESAT-1, NISAR. Furthermore, support for LuTan-1 precision orbit data has been added.

***All packages***

*HTML-based documentation*: The HTML-based documentation has been updated to improve its consistency, for easier maintenance and for a nicer look.