HYDROCOASTAL

SAR/SARin Radar Altimetry for Coastal Zone and Inland Water Level

Algorithm Theoretical Basis Document
Deliverable D1.3

Sentinel-3 and Cryosat SAR/SARin Radar Altimetry for Coastal Zone and Inland Water

ESA Contract 4000129872/20/I-DT

Project reference: HYDROCOASTAL_ESA_ATBD_D1.3
Issue: 1.1

08/10/2020

This page has been intentionally left blank

Change Record

Date	Issue	Section	Page	Comment
25/06/2020	1.0	all	all	1st version
08/10/20	1.1	various	various	Revisions following ESA review. Addition of new chapter to account for L2 official products variable processing.

Control Document

Process	Name	Date
Written by:	Albert Garcia-Mondéjar, Ferran Gibert, Ester Vendrell	08/10/2020
Checked by	David Cotton	
Approved by:		

Subject	Radar Altimetry for Coastal Zone and Inland Water Level	Project	HYDROCOASTAL

Author	Organisation	Internal references
Albert Garcia-Mondéjar, Ferran Gibert, Ester Vendrell	isardSAT	HYDROCOASTAL_ESA_ATBD_D1.3
Michele Scagliola	Aresys	
Ole Andersen, Karina Nielsen, Heidi Ranndal	DTU Space	
Pierre Fabry	Along Track	
Luciana Fenoglio-Marc	U Bonn	
Marcello Passaro	TUM	
Christine Gommenginger	NOC	
Nicolas Bercher	AltHydroLab.fr	
Angelica Tarpanelli	CNR-IRPI	
Elena Zakharova	NUIM	

	Signature	Date
For HYDROCOASTAL team		
For ESA		

Table of Contents

1. Introduction

1.1. The HYDROCOASTAL Project

The HYDROCOASTAL project is a project funded under the ESA EO Science for Society Programme, and aims to maximise the exploitation of SAR and SARin altimeter measurements in the coastal zone and inland waters, by evaluating and implementing new approaches to process SAR and SARin data from CryoSat-2, and SAR altimeter data from Sentinel-3A and Sentinel-3B.

One of the key objectives is to link together and better understand the interactions processes between river discharge and coastal sea level. Key outputs are global coastal zone and river discharge data sets, and assessments of these products in terms of their scientific impact.

1.2. Scope of this Report

This is the Algorithm Theoretical Basis Document (ATBD) report for HYDROCOASTAL and represents D1.3 of the project together with the TN *Dry and Wet Tropospheric Corrections for Coastal Zones and Inland Waters.*

The purpose of this document is to describe the processing schemes that will be applied in HYDROCOASTAL to generate the test data sets in WP2000 and the Global Products in WP3000.

1.3. Applicable Documents

AD-01 Sentinel-3 and CryoSat SAR/SARin Radar Altimetry for COASTAL ZONE and INLAND WATER - Statement of Work, V1.0 10/01/2019 Ref: EOP-SD-SOW-2018-089

1.4. Reference Documents

RD-01 HYDROCOASTAL Technical Proposal. V1.1 28/11/2019, SatOC and HYDROCOASTAL team.
RD-02 HYDROCOASTAL Implementation Proposal. V1.1 28/11/2019, SatOC and HYDROCOASTAL team.
RD-03 HYDROCOASTAL Management Proposal. V1.3 26/11/2019, SatOC and HYDROCOASTAL team
RD-04 HYDROCOASTAL Financial Proposal. V1.2 28/11/2019, SatOC and HYDROCOASTAL team
RD-05 HYDROCOASTAL Contractual Proposal. V 1.2 26/11/2019, SatOC and HYDROCOASTAL team
RD-06 HYDROCOASTAL Deliverable 2.1 IODD (Input Output Data Definitions). V1.1 08/10/2020, isardSAT and HYDROCOASTAL team.
RD-07 HYDROCOASTAL Deliverable 2.3 PSD (Product Specification Document). V1.1 08/10/2020, isardSAT and HYDROCOASTAL team.

1.5. Document Organisation

After this introductory section, section 2 provides an overview of the processing approach, and then subsequent sections provide the individual algorithm descriptions.

2. Overview of Algorithms

2.1. To L1B, L1B(S)

The algorithm to generate L1B/L1B(S) data levels after extraction from Copernicus Hub follows a Delay-Doppler Processing algorithm developed by isardSAT.

2.2. Stack Processing and Re-Tracking Algorithms to L2

The following L2 processing algorithms are considered:

- Two Step Analytical, developed by isardSAT
- Specialised SARin, developed by Aresys
- MWaPP, developed by DTU Space
- ICC-ER Empirical Retracker, developed by Along-Track
- STARS Type, developed by UBonn
- Adaptation of ALES+ for SAR, developed by TUM
- Specialised Coastal SAR, developed by NOC
- L2 official products variable ingestion.

2.3. L2 to L3 (River Level Time Series)

Two L3 processing options are considered:

- L3 River Level, developed by AltiHydroLab.fr
- L3 River/Lake Level, developed by DTU Space

2.4. L3 to L4 (River Discharge)

The following Table summarises the L4 algorithms considered in this document.

Table 2.1: Suggested discharge estimation algorithms

Empirical group of algorithms					
Algorithm Name	**RS Input parameters**	**Calibrated parameters**	**Auxiliary data**	**Spatial scale**	**Time scale**
1. Rating curves (NUIM)	Altimetric H	a,b,c (Eq. 14.1)	Simultaneous Qinsitu	Virtual station	Sub-monthly, best: (2-3)-monthly after Q_{VSi} combination
2. Bjerklie equation (NUIM)	Altimetric H,S, Optic dynamic Width (W)	Initial depth(D0) (Eq. 14.2)	Qinsitu	River reach 100-300 km	Daily
3. Merging approach (CNR-IRPI)	Altimetric H, Reflectance Ratio C/M	*K*, *b* and *f* (Eq 15.7)	Qinsitu	Box centered in the virtual station 20x20 km	Daily
Physical group of algorithms					
Algorithm Name	**RS Input parameters**	**Calibrated parameters**	**Auxiliary data**	**Spatial scale**	**Time scale**
1. Manning equation (NUIM)	Altimetric H, S; Optic dynamic HighRes Width	Depth (D0), roughness(n) in eq (3); both can be a guess	Depth, Qinsitu (for accuracy increase), SRTM	River reach 100-300 km	Daily

3. Algorithm Description: Altimeter raw data to L1B / L1B(S) (isardSAT)

This section describes the algorithm used to process raw data to L1B/L1B(S) products for both Sentinel-3 and CryoSat data. The algorithm description includes the following subsections: theoretical background, the processing steps of the algorithm, the development choices and trade-offs of the project, and a detailed description of the data flow.

3.1. Theoretical Description, physics of the problem

The Delay-Doppler altimeter uses the power backscattered from the scene more efficiently than does the conventional altimeter, since the whole beam-limited along-track signal is exploited, instead of the pulse-limited area typically considered by conventional altimeters, as schematically sketched in Figure 3.1. This is achieved thanks to the proper slant range (or delay) variation compensation. The extra delay observed from each Doppler bin in which the along-track beam is partitioned is removed, aligning all the Doppler beams to the same delay or range, known as range migration correction (RMC, see Figure 3.1).

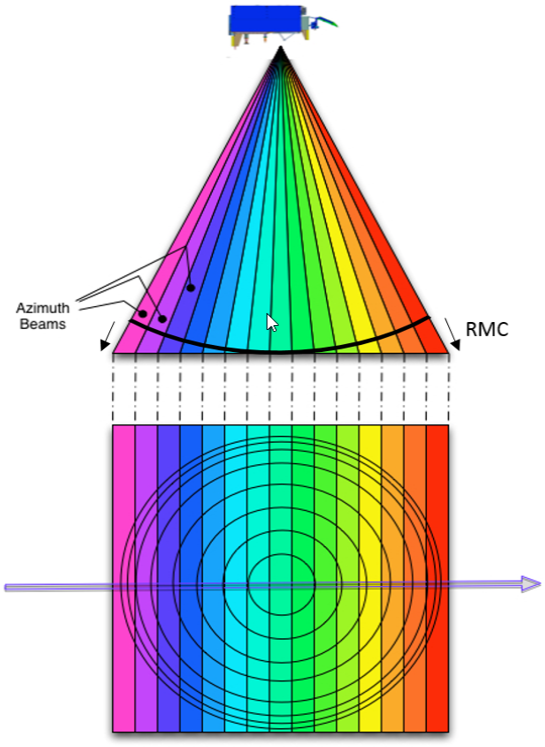


Figure 3.1: Delay-Doppler altimeter's illumination geometry side (top) and footprint (bottom) plan views. The along-track beam is partitioned in several Doppler beams with improved resolution. An extra delay or range per beam needs to be compensated, by introducing the range migration correction-RMC (credit: ESA).

The additional selectivity in the Doppler domain, which confers an additional degree of freedom, allows increasing the along-track resolution (i.e., reducing the along-track footprint), such that the impact of terrain variability on the imaged footprint can be minimised. Such improved resolution can be appropriately exploited for coastal altimetry, providing improved performance mainly due to the reduced land contamination as stated in Gommenginger et al. 2013.

This selectivity in the Doppler dimension can be also exploited to perform a specific focusing to a given defined location. This requires to perform additional processing in the along-track direction, which mainly consists of beam steering (to the desired surface position) and Fourier transformation.

In this manner, several looks are made available for a specific surface position, i.e., different Doppler beams from different bursts are pointed towards it, forming the Doppler stack, as exemplary shown in Figure 3.2. Therefore, the final signal-to-noise ratio (SNR) can be improved once the different range-compressed power waveforms are incoherently accumulated; such processing is known as multi-looking.

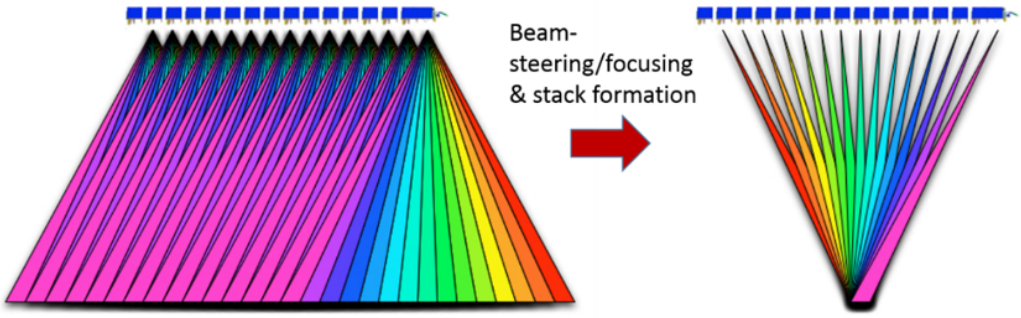


Figure 3.2: Conceptual representation of beam-steering processing and stack formation for a given surface (credit: ESA).

Hence, the intrinsic 2-D (range/Doppler) nature of the Delay-Doppler altimetric signals requires, as theoretically described above, to perform accordingly a 2-D processing, properly exploiting the potential capabilities conferred by this relatively new operational mode.

Standard Delay Doppler altimeters such as S3 produce waveforms with no information about the cross-track elevation angle of the point of closest approach. Such a parameter can only be measured by systems provided with two antennas in the across-track direction such as CS2. In this kind of systems, the cross-track elevation angle is retrieved by comparing the phase difference between the echoes received at both antennas.

3.2. Algorithm Definition: Processing Steps and Mathematical Description

In the following lines, the proposed Delay-Doppler processor baseline to be implemented within the HYDROCOASTAL project is presented, defining the different algorithms and processing steps included, as well as their rational mathematical description. The potential improvements are also defined in this description.

The considered SAR (aka Delay-Doppler) processor is based on the experience gained by isardSAT in the study and implementation of the Ground Prototype Processor (GPP) within the Sentinel-6 project.

The data chain proposed here is applicable to both CS2 and S3 data sets. While for CS2 the input data is called FBR, for S3 it is L1A. In addition to that, the SARIn step is only applicable to CS2 when operating in SARIn mode.

The main processing stages of the Doppler-Delay processor are:

1. Surface locations, Final burst datation and Window delay computation
2. Beam angles computation
3. Azimuth processing (Delay-Doppler processing + Stacking)
4. Geometry corrections
5. Range compression
6. Multi-looking
7. Scaling factor computation (sigma0 extraction)

For the SARin specific case, the interferometric processing will be implemented in the Multi-looking step.

The corresponding flow chart of the proposed Doppler-Delay processor is represented in Figure 3.3:

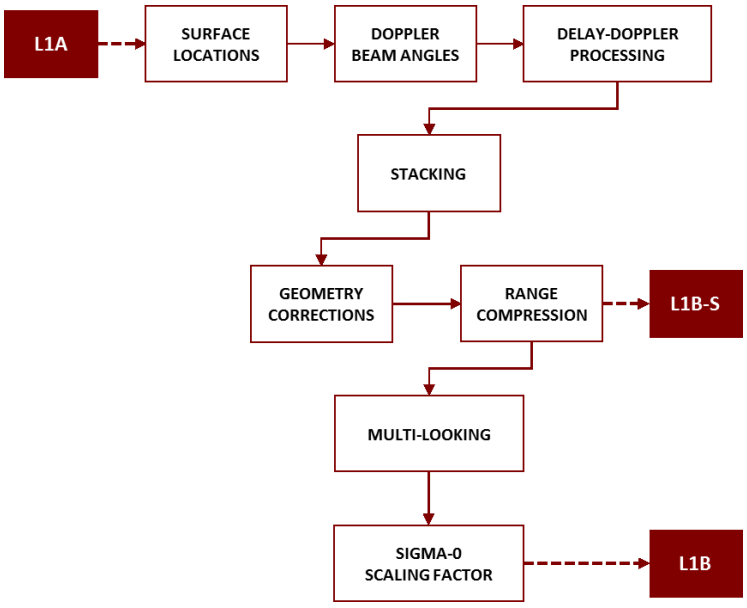


Figure 3.3: Level-1A/Level1-B SAR processing chain (credit: isardSAT). For details on the input/output data levels description, please refer to Section 3.4. For CS2, the input data is labelled FBR instead of L1A.

3.2.1. Surface locations, surface datation and window delay Computation

3.2.1.1. Purpose and scope

The aim of this algorithm is to compute the surface locations (and their corresponding datation and orbit parameters) defined by the intersection of the Doppler beams and the estimated surface positions along the satellite track. The block diagram of the processing flow for this algorithm is depicted in Figure 3.4.

3.2.1.2. Mathematical description

The first surface location is determined by the window delay associated with the first burst of the tracking cycle. Then, an iterative process starts and lasts until the end of the orbit data is reached. This process goes through the following steps:

- *Computation of angular Doppler resolution:* This is obtained at the current satellite position given the Doppler frequency expression (Cumming and Wong, 2005):

$$f_d = \frac{2|\vec{v_S}| \sin\theta}{\lambda} \qquad \text{Eq. 3.1}$$

where $|\vec{v_s}|$ is the satellite velocity vector and λ the carrier wavelength. As the azimuth processing will give a Doppler frequency sampling given by the inverse of the burst duration τ_B, the angular azimuth beam resolution is calculated as:

$$\theta = \arcsin\left(\frac{\lambda}{2|\vec{v_S}| \sin\theta}\right) \qquad \text{Eq. 3.2}$$

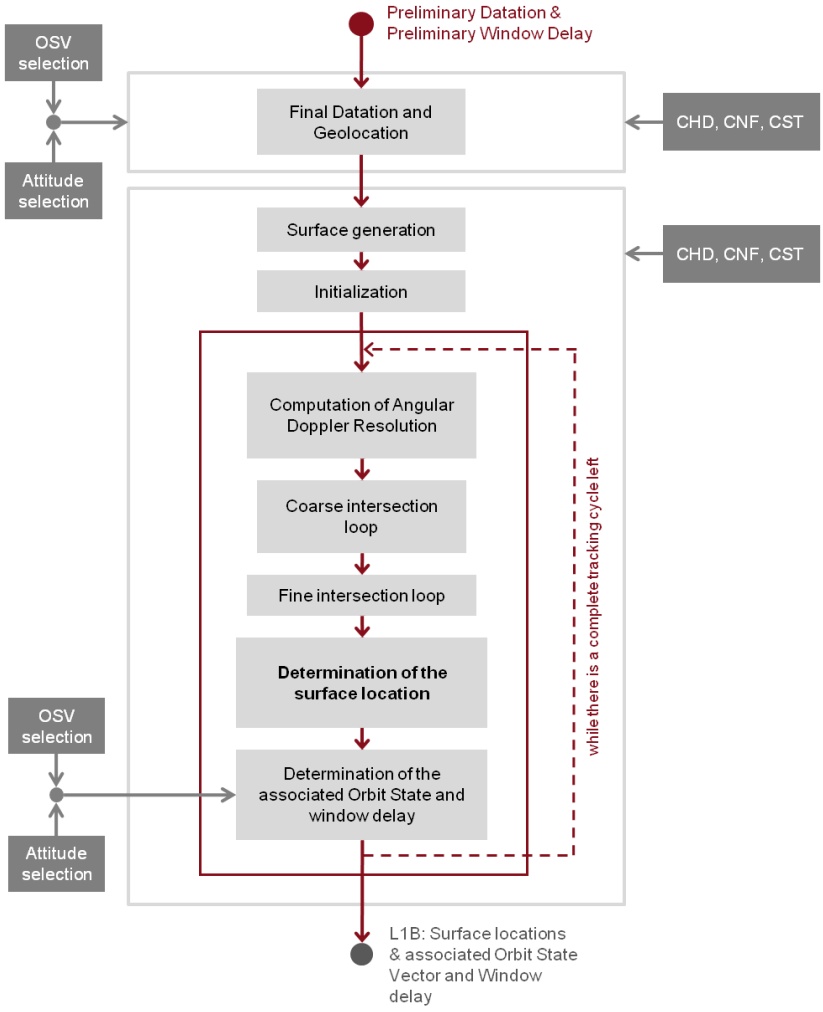


Figure 3.4: Surface Locations algorithm's flow chart (credit: isardSAT): OSV stands for (orbit state vector), CHD, CNF and CST refer to the characterization, configuration and constants files, respectively (please refer to Section 3.4).

- *Coarse and fine intersection loops*: Determine the intersection between the direction defined by the angle θ_j (angular azimuth beam resolution) with respect to the nadir and each surface location. This process is performed by iterating through the surface positions until the angle of sight α_i is bigger than the angular azimuth beam resolution θ_j (j being the current surface index), see Figure 3.5 on the left. Then an interpolation is performed between the last angle of sight and the previous one. After that, a second iteration process starts (the fine intersection loop) and finishes when the angle of sight coincides with the angular azimuth beam resolution (Figure 3.5, right).

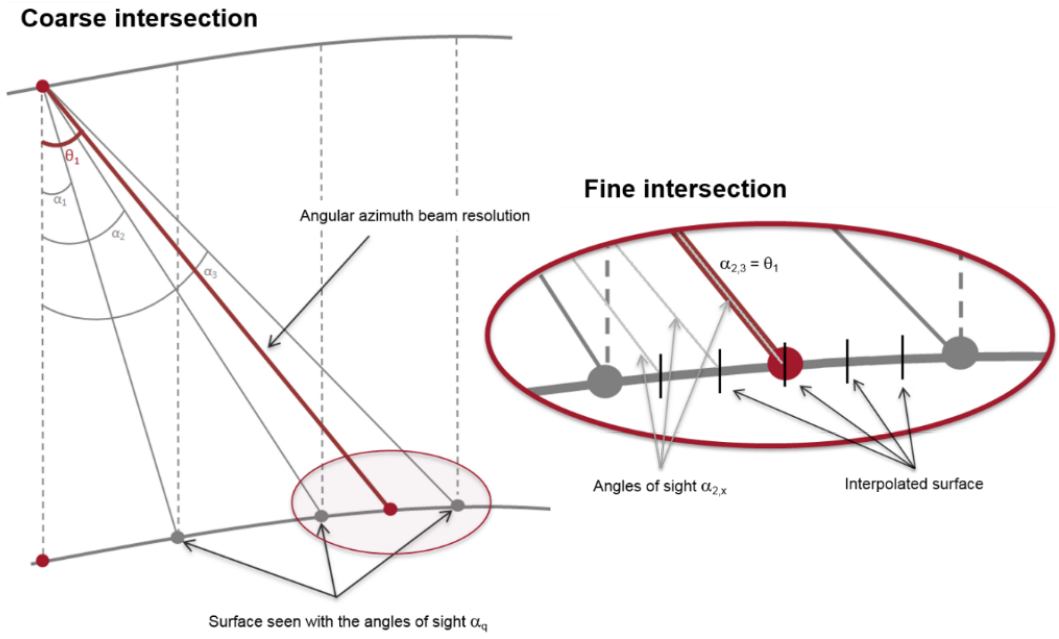


Figure 3.5: Coarse (left) and fine (right) intersection step of the surface locations algorithm (credit: isardSAT): α_i corresponds to the angle of sight (between the nadir direction and the vector from satellite position to each burst surface position) and θ_j is the angular/beam Doppler resolution for the j-th surface.

- *Determination of the associated orbit state and window delay:* The associated orbit state can be retrieved using orbit interpolators or libraries. If not available, the orbit can be manually interpolated. Then, the new surface location is also located on the orbit (this would be the action of going from the surface to the orbit and it is represented in Figure 3.6 with the red lines going from the surface locations to the orbit). In addition, the window delay of the new surface location is calculated. Then, the obtained state vector is stated as the new one and the iteration process starts again (in Figure 3.6 each red point on the orbit is a new point of start).

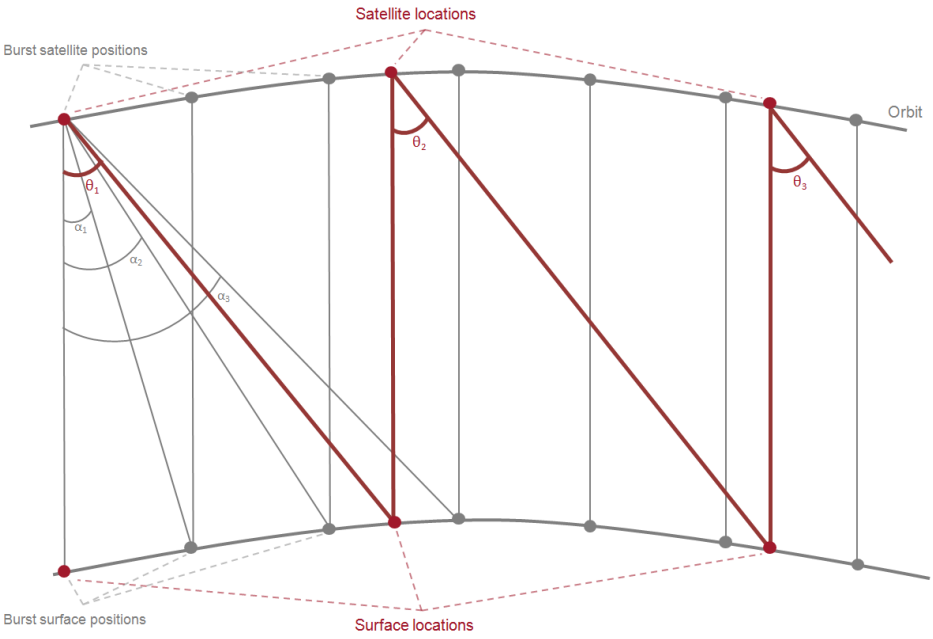


Figure 3.6: Determination of the associated orbit state and window delay (credit: isardSAT)

3.2.2. Beam angles

3.2.2.1. Purpose and scope

This algorithm computes, for every burst, the angles between the satellite velocity vector and the directions defined by the satellite location and the computed surface locations under the satellite's boresight. The block diagram of this processing algorithm is sketched in Figure 3.7.

3.2.2.2. Mathematical description

The algorithm calculates the angles between each satellite velocity vector and the vector connecting every surface location that is "observed" by the satellite at the current satellite burst position and the location itself. These angles are then used by the Azimuth Processing algorithm to steer the beams to the desired surface locations.

The process starts by iterating through the bursts. Then, for each burst, a few steps are followed:

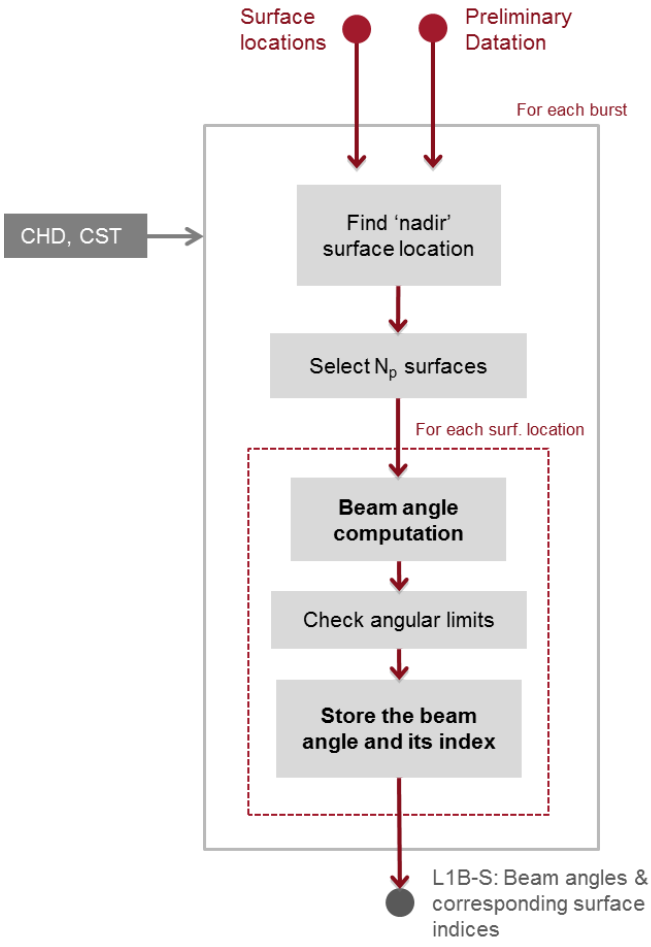


Figure 3.7: Flow chart of the beam angles algorithm (credit: isardSAT). CHD and CST refer to the characterization and constants files (see Section 3.4).

- Find the *surface location closest to the nadir direction* and store its index.
- Select N_p surface locations (N_p being the number of pulses per burst): $N_p/2$ forward and $\frac{N_p}{2}$ backwards, (see Figure 3.8).
- Finally, *store the number of selected surface locations and their indices*. These indices will be used later to perform the stack.

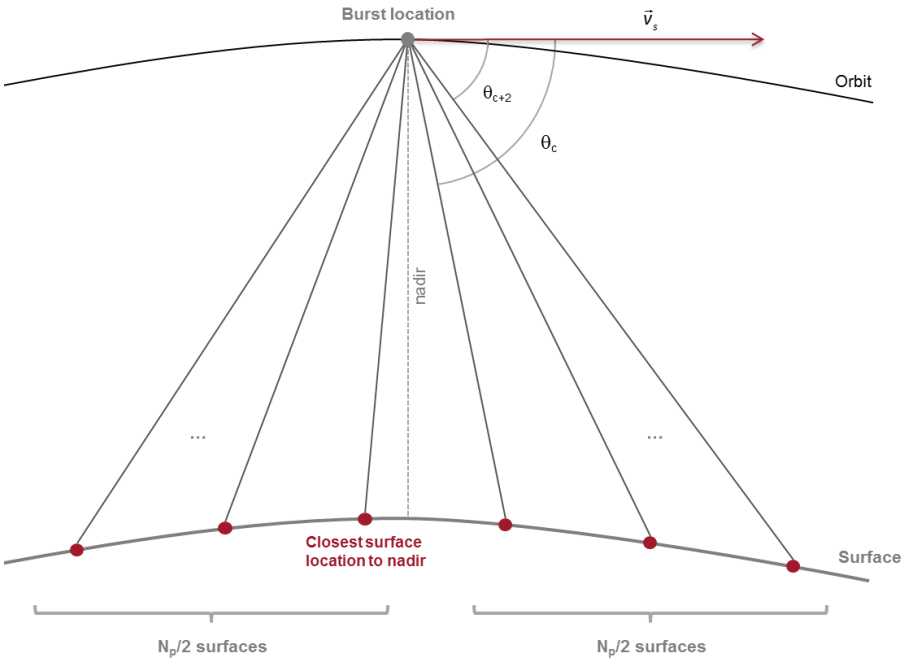


Figure 3.8: Geometry of the Beam Angles algorithm (credit: isardSAT). θ_c refers to the beam angle between the satellite velocity vector and the surface vector from the satellite's position to the specific surface location

Then, for each surface location the following processing steps are applied:

- Compute the angle between the satellite velocity vector and the satellite to surface direction. This angle is named ***beam angle***:

$$\theta_c = \arccos\left(\frac{\vec{v}_s \cdot \vec{w}_{sat\to surf}}{|\vec{v}_s| \cdot |\vec{w}_{sat\to surf}|}\right) \qquad \text{Eq. 3.3}$$

computed as the dot product or scalar product between the satellite's velocity vector $|\vec{v}_s|$ and the vector from the burst satellite's position to the specific surface location $\left|\vec{w}_{sat\to surf}\right|$. Note that the angular Doppler resolution defined by the angle θ, as shown in Figure 3.5 and Figure 3.6, corresponds to the angle from nadir direction to the surface direction vector; whereas, the beam angles θ_c, here considered and shown in Figure 3.8, are defined between the satellite's velocity vector and the surface direction vector.

Surface focusing

As an optional configuration, specific geographical locations, towards which the beam steering or focusing (over the satellite's track) should be performed, can be input to the Level-1B processor. This is a very interesting approach when operating in coastal regions and it can be also exploited for

colocation with other types of data as RDSAR or LRM. In this case, the location of the closest surface locations (to the given desired geographical locations) are accordingly updated over the track of the satellite as:

1. Transformation from geographical (geodetic) coordinates to Cartesian coordinates over an Earth-Centred Earth-Fixed (ECEF) system.

$$[lon_{int},\ lat_{int}, alt_{int}] \rightarrow [x_{int}, y_{int},\ z_{int}] \qquad \text{Eq. 3.4}$$

2. Identification of the surface with the minimum distance to the input desired position (minimum norm over the set of difference vectors)

$$i_{surf}^{min} = i_{surf},\ \| [x_{int}, y_{int},\ z_{int}] - [x(i_{surf}), y(i_{surf}),\ z(i_{surf})] \| \qquad \text{Eq. 3.5}$$

3. Projection of the desired location over the satellite's track (projection of the vector joining the previous surface to the one with minimum distance)
4. Updating the surface closest to the desired geographical location

3.2.3. Azimuth processing and stacking

3.2.3.1. Purpose and scope

The purposes of the azimuth processing and stacking algorithm are to steer the beams to the different surface locations and to generate the stacks. The flow chart of the steps corresponding to azimuth processing and stacking is shown in Figure 3.9, assuming the exact method for the azimuth processing is used. In this case, there is an additional loop running over each surface when compared to the approximate method.

3.2.3.2. Mathematical description

Azimuth processing:

In order to create Doppler beams, a specific process has to be performed in the along-track direction. This process consists in applying a different phase value $\theta_{beam}(b,p)$ as defined in Eq 3.6 to the different pulses in order to steer the beams towards the surface locations computed in Section 3.2.1, as originally proposed by Raney (1998).

In order to do so, and as an improvement processing option in this baseline, two different approaches are considered: the exact method (Figure 3.10), and the approximate one (Figure 3.11), which is an simplification of the former one.

The exact method uses all the beam angles computed in Section 3.2.2 to steer the beams to the surfaces. This implies that there will be an FFT process for each one of the surface locations. On the other hand, the approximate method simply uses the beam angle that is closer to the nadir to

spread the other beams and steer them to the other surfaces. This means that the approximate method only goes through one FFT process.

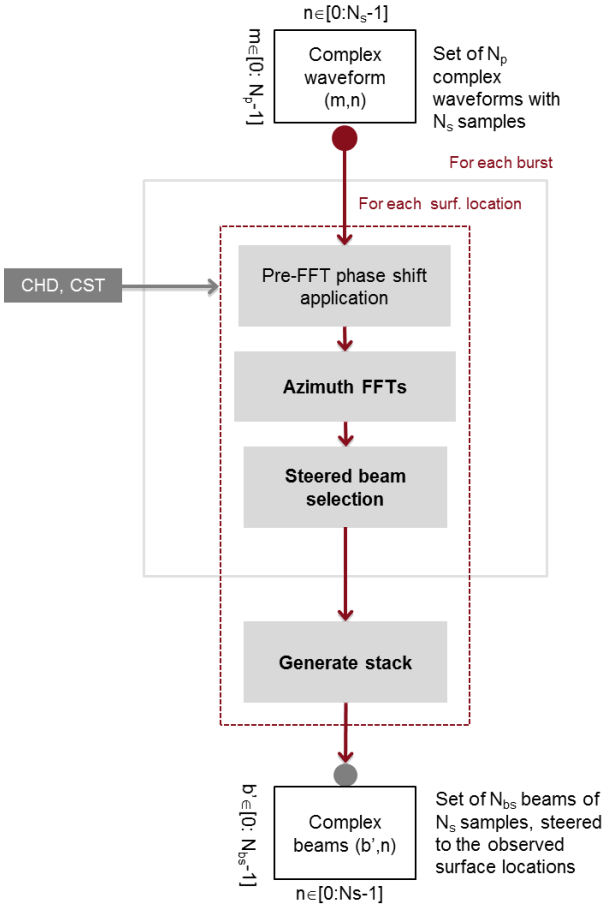


Figure 3.9: Block diagram of the azimuth processing and stacking algorithm, considering the exact method (credit: isardSAT). CHD and CST refer to the characterization and constants files (see Section 3.4).

Note that the FFT processes come from the angles (or phases) that are applied to the pulses, $\theta_{beam}(b,p)$. These angles have two components:

$$\theta_{beam}(b,p) = \theta_c(b) + \delta\theta(p) = \theta_c(b) + \arcsin\left(\frac{\lambda \cdot p}{2|\vec{v}_s| \cdot N_p \cdot PRI}\right) \quad [rad] \qquad \text{Eq. 3.6}$$

being b the beam index within a burst $b \in [0, N_b - 1]$; p the pulse index $p \in \left[-\frac{N_p}{2}, \frac{N_p}{2} - 1\right]$; $\theta_c(b)$ the beam angles computed in Section 3.2.2; and $\delta\theta(p)$ is the variable part, which leads intrinsically to

the FFT in the along-track (or azimuth) dimension. This second part of $\theta_{beam}(b,p)$ is the one that spreads all the beams along the surface locations, being the azimuth angular beam resolution.

From the mathematical point of view, the beamforming operation can be expressed as follows:

$$\Psi_b(k,n) = \frac{1}{\sqrt{N_p}} \cdot \sum_{p=0}^{N_p-1} \Psi_w(p,n) \cdot e^{-2j\cdot\left(\frac{2\pi}{\lambda}\cdot PRI\cdot|\vec{v}_s|\cdot\cos(\theta_c(b))+\frac{\pi\cdot k}{N_p}\right)\cdot p} \qquad \text{Eq. 3.7}$$

where *k* refers to the indexation at beam level. The phase shift applied over the burst $\Psi_w(p,n)$ (weighted in azimuth by a specific window) is hence performing the specific beamforming to the surface of interest. A normalization by number of pulses has been performed to ensure energy is kept constant between domains, pulse and Doppler/beam domains, so Parseval theorem applies. This equation defines the general case of implementing the exact method, so each beam is steered based on the corresponding $\theta_c(b)$, keeping only the central beam for each FFT; while the approximate method exploits the same formulation but a single FFT is performed with the angle closer to nadir, so all the other beams are accordingly steered.

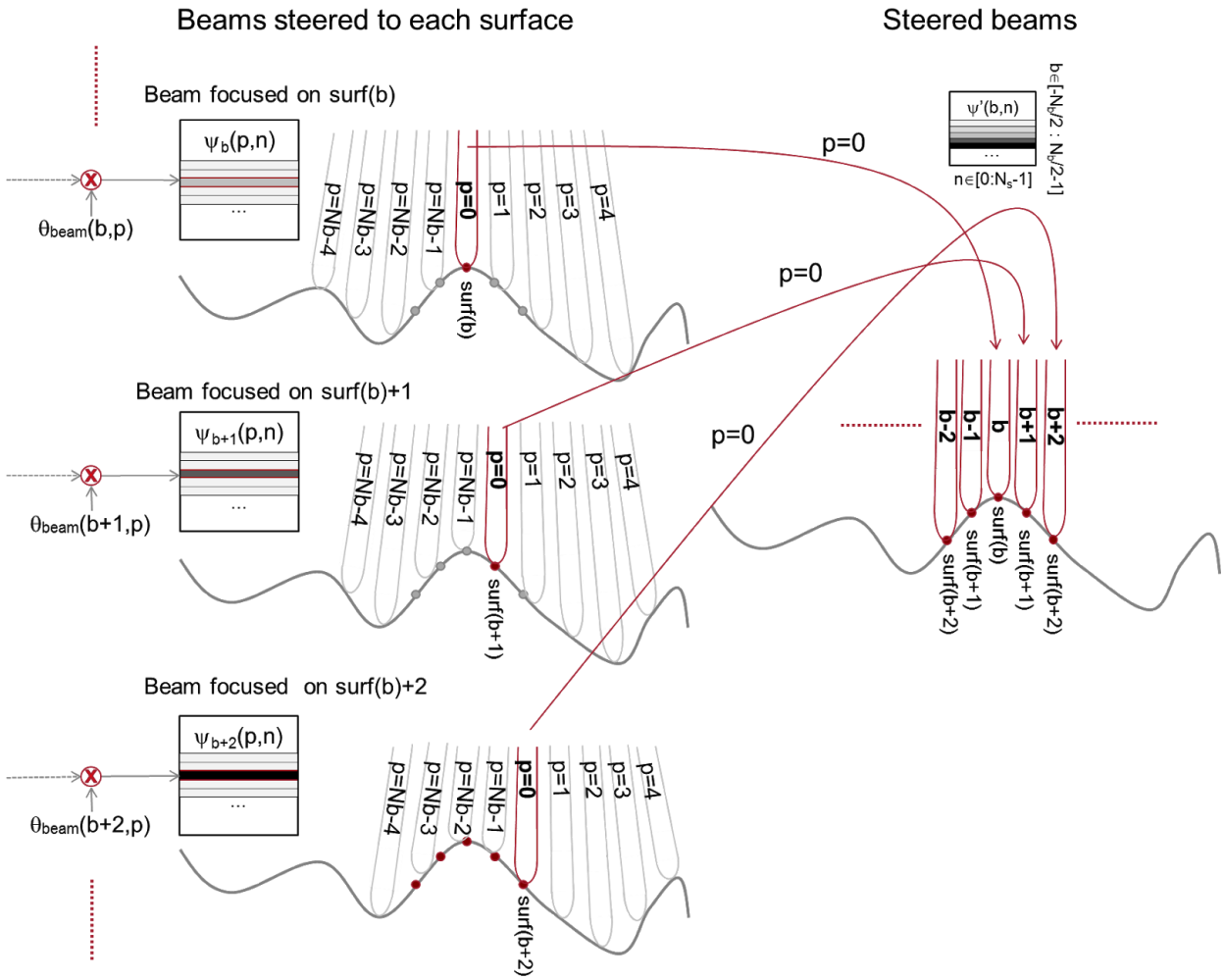


Figure 3.10: Exact beam-forming (or steering) geometry; each branch represents the process of focusing the central beam to a specific surface: $\theta_{beam}(b,p)$ refers to the phase used to steer the beams to the specific surface locations (b and p being the beam and pulse indexes, respectively); $\Psi_b(p,n)$ [n corresponds to the sample index] and $\psi'(b,n)$ refer to the set of waveforms before and after the beam steering, respectively (credit: isardSAT)

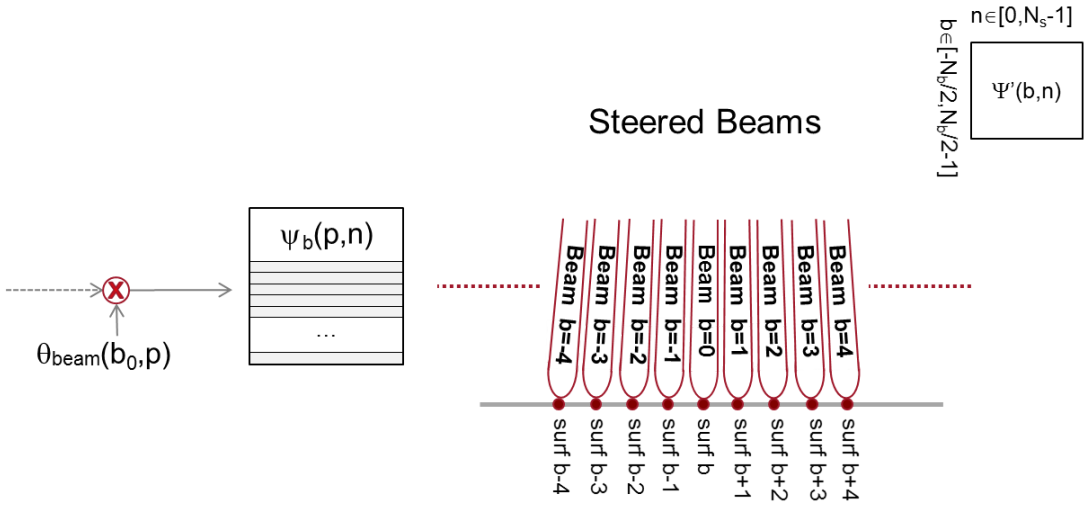


Figure 3.11: Approximate beam-forming (steering) geometry (credit: isardSAT). Only the central beam angle b_0 is used and the other beams are equally spaced.

We should note that a specific weighting or window $W(p)$ (Hamming or Hanning) can be optionally included (defined in the configuration file) at burst level before the beam forming procedure (phase ramp + along-track FFT) takes place:

$$\Psi_w(p,n) = \Psi(p,n) \cdot W(p),\ p \in \lceil 0, N_p - 1 \rceil,\ n \in [0, N_s - 1] \qquad \text{Eq. 3.8}$$

where $\Psi(p,n)$ refers to the complex-valued signal in the azimuth time and range-time domains (p and n being the pulse and sample index, respectively).

Such weighting can be used to minimize the impact of side-lobe effects in the Doppler/azimuth PTR (Point Target Response) and, so, the related Doppler ambiguities at the edges of the spectrum (edge beams). This is especially important when operating close to the coast as high reflectivity land scattering can contaminate the signal of interest. We must take into account that this weighting leads to a degradation on the along-track resolution or footprint.

Stack generation:

The stacking consists of regrouping the beams that have illuminated each surface location. This means going from the satellite to the surface point of view as schematically represented in Figure

3.12. For each surface all the beams (from the different bursts) pointing to that surface are grouped into a single stack.[1]

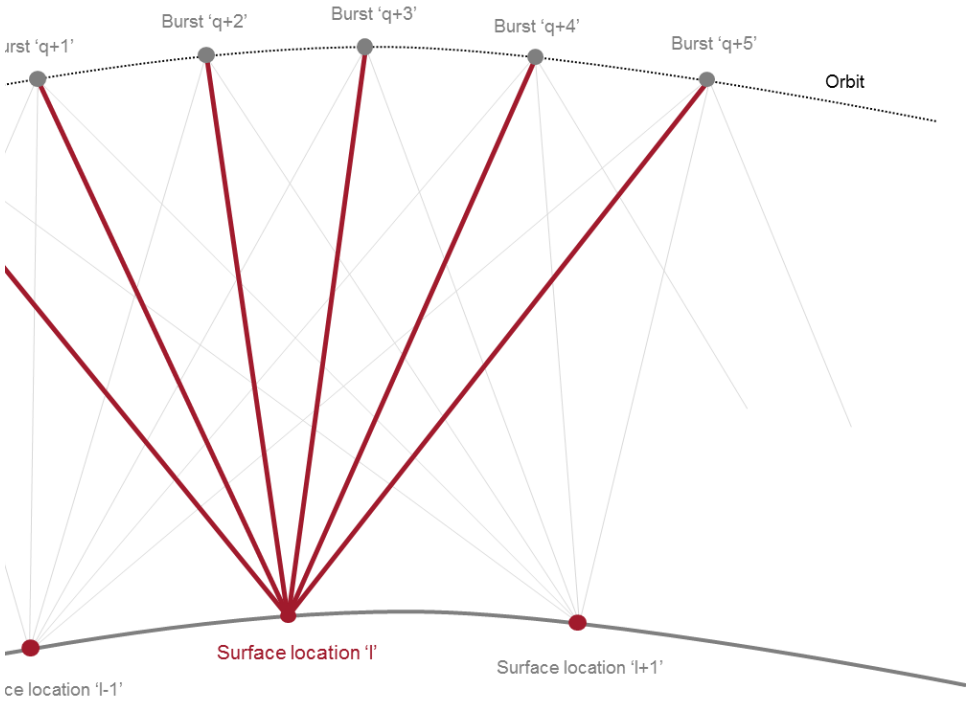


Figure 3.12 Stack formation for surface location 'l'. Red lines represent beams illuminating the surface location 'l', hence, the stack. Grey lines are other beams that have been steered to other surface locations (credit: isardSAT).

3.2.4. Geometry corrections

3.2.4.1. Purpose and scope

This algorithm computes and applies all the corrections associated with the geometry of the scenario. These are the Doppler, slant range and window delay misalignments corrections. As the stack has already been generated, these compensations are performed for each stack. In the case

[1] It must be noted that the objective of the azimuth processing is to steer for each burst the Np pulses (which lead to Np beams steered to different locations on the earth) and so this ends up in the so called 'b' index, relating to the different beams pointing to a specific surface; while the index j in Section 3.2.1.2 refers to the sweeping of the different projections of the burst locations on ground and it is used to look for the intersection to generate the new surface location. The index 'l' in Figure 3.12 sweeps the total number of surfaces (computed in the surface locations algorithm Section 3.2.1) over the track being processed, and so for each one of these locations the stacking is in charge of grouping the focused/steered beams from the different bursts (selecting the proper burst and the proper beam 'b' of that burst).

of the window delay misalignment, we can apply different methods depending on the observed or desired surface.

The block diagram of this processing algorithm is sketched in Figure 3.13.

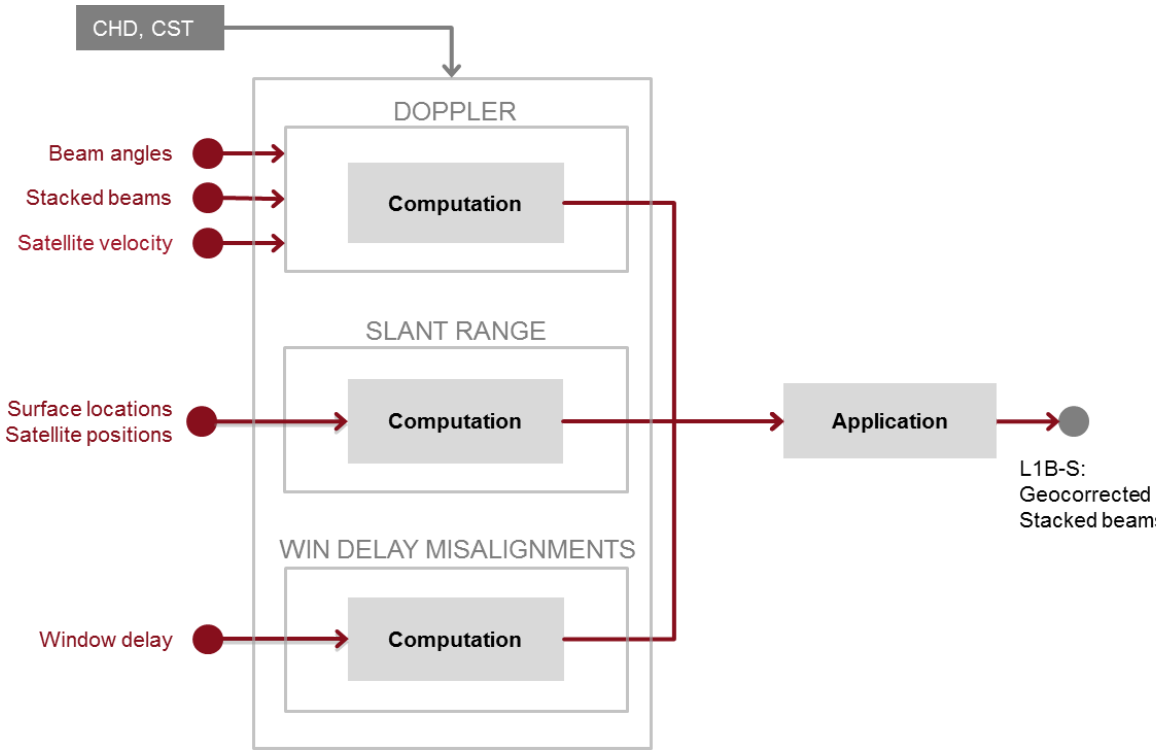


Figure 3.13 Block diagram of the geometry corrections algorithm (credit: isardSAT). CHD and CST refer to the characterization and constants files (see Section 3.4).

3.2.4.2. Mathematical description

Doppler Correction

The Doppler correction is needed to remove the echoes' frequency shifts due to the sensor-target velocity (compensate for the Doppler offset induced by movement of the platform while transmitting/receiving the pulse, see Prats-Iraola et al. (2014)). The correction is applied to the echoes in the time domain, before the FFT step of the range compression.

The frequency shift for a given Doppler direction, in meters, is computed as[2] (see Figure 3.14):

[2] The physics behind this correction and its derivation are clearly stated in the work of Prats-Iraola et al. 2014. The movement of the platform during transmission/reception produces a net Doppler centroid, which translates into a shift in the range-time. Such Doppler centroid can be computed from the instantaneous Doppler frequency for each beam/burst

$$\Delta r_D(b') = \frac{c}{2}\left[\frac{\tau}{B}\left(-\frac{2}{\lambda}|\overrightarrow{v_S}(b')|\cos\theta_c(b')\right)\right] \quad [m] \quad , b' \in [0, N_{bs}-1] \qquad \text{Eq. 3.9}$$

where $\left|\vec{v}_s\left(b'\right)\right|$ refers to the modulus of the satellite's velocity at the corresponding beam (burst) position, c to the speed of light, τ to the pulse duration, B to the transmitted bandwidth and N_{bs} to the number of beams pointing to that specific surface.

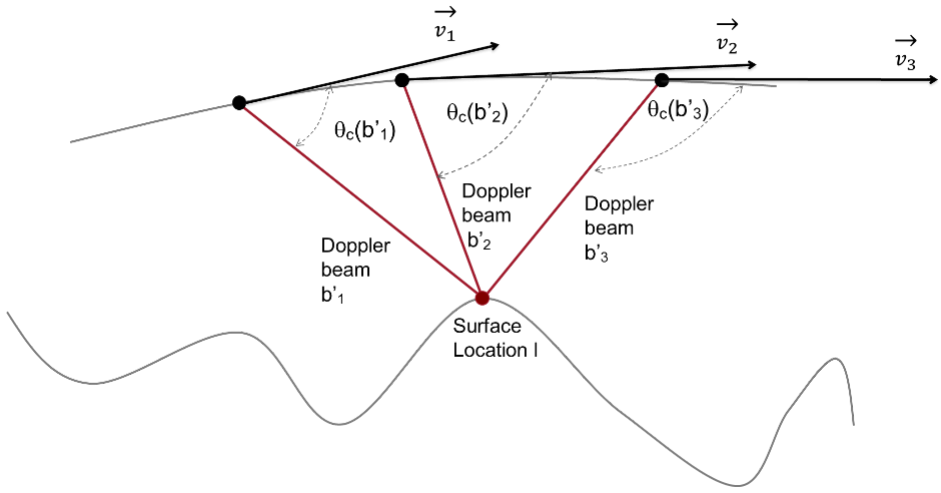


Figure 3.14: Geometry of Doppler shift effect (credit: isardSAT). $\theta_c(b')$ refers to the beam angle for each beam/burst focused to that surface; $\vec{v}$ is the satellite velocity vector.

Slant range correction

This correction compensates the range migration produced by the motion of the sensor along the orbit with respect to each surface location. In Figure 3.15, the ranges of the surface location 'l', $\left|\vec{r}(b')\right|$ are different from the real height $\left|\vec{h}(l)\right|$. This difference $\Delta r(b')$ can be computed in range as:

$$\Delta r\left(b'\right) = \left|\vec{r}(b')\right| - \left|\vec{h}(l)\right| \quad [m] \qquad \text{Eq. 3.10}$$

pointing to that surface, which in turn is converted to range delay through the normalization by the chirp rate and then to delay.

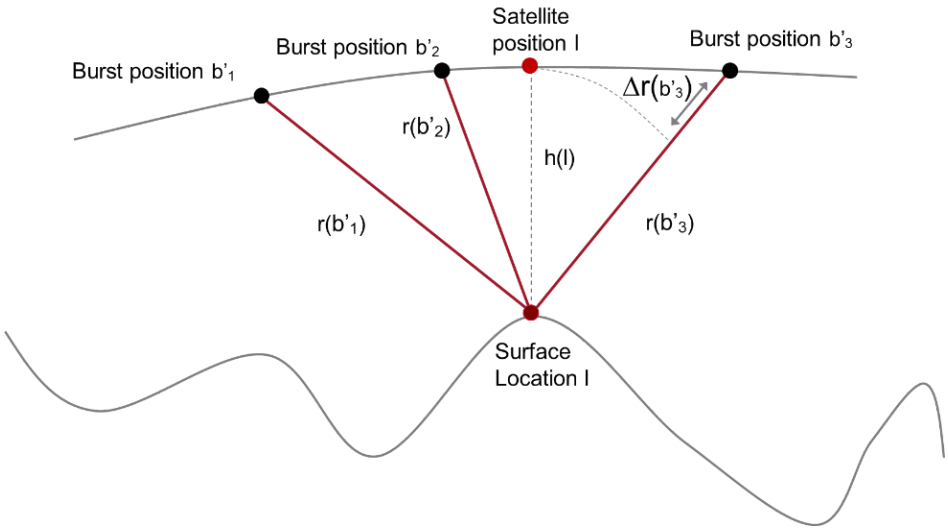


Figure 3.15: Geometry of the slant range correction (credit: isardSAT). $r(b')$ refers to the range to the surface location for each beam pointed to that surface and $h(l)$ to the real surface height.

Window delay misalignments

The beams of each stack come from different bursts, with different window delays. These misalignments have to be corrected. In order to do it, the window delay associated with the surface (and computed in Section 3.2.1) is taken as a reference and the differences with all the window delays of the bursts that have built the stack are computed:

$$\Delta s_{wd}\left(b'\right) = \frac{\tau_{wdref} - \tau_{wd}(b')}{T_0}\ [samples] \qquad \text{Eq. 3.11}$$

With τ_{wdref} being the reference window delay within the stack (closest approach or with higher power return), $\tau_{wd}(b')$ the window delay associated with the b' beam and T_0 the clock period.

In the case of being on coastal areas, the window delay taken as reference can be chosen as the one of the first ocean surface ahead, rather than the one associated to burst right above the surface that it is still tagged as land. In this way, we ensure that the alignment of the beams of the stack is done such that the ocean waveform is recovered (and not the one over land).

All these three corrections, not being an integer number of samples, are applied as a frequency shift by multiplying the beam waveforms in time by an exponential (Cumming and Wong, 2005; Raney, 1998):

$$\exp\left\{j\frac{2\pi}{N_s}\cdot\Delta s\left(b'\right)\cdot n\right\} \qquad \text{Eq. 3.12}$$

where $\Delta s(b')$ corresponds to the total correction in samples (N_s being the total number of range samples).

Finally, the window delay associated with each surface location has to be replaced by the reference window delay within the stack.

3.2.5. Range compression

3.2.5.1. Purpose and scope

This algorithm performs the range compression of the input bursts and then generates the power waveforms. The block diagram of this processing algorithm is sketched in Figure 3.16.

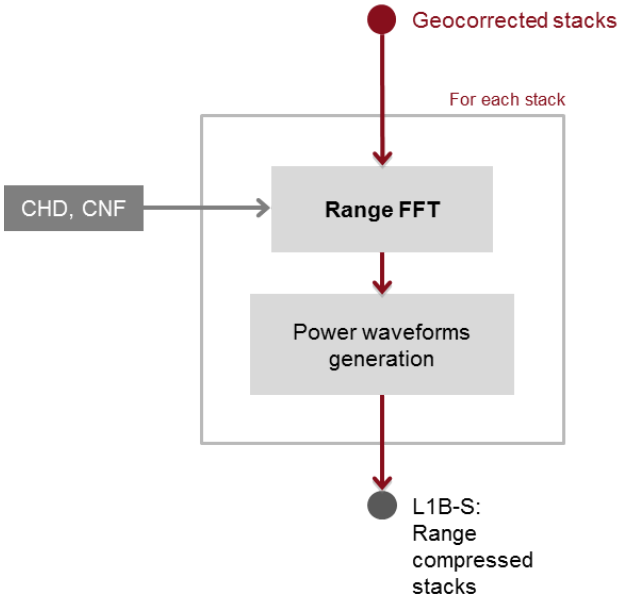


Figure 3.16: Block diagram of the range compression algorithm (credit: isardSAT). CHD and CNF refer to the characterization and configuration files, respectively, see Section 3.4

3.2.5.2. Mathematical description

The time domain waveforms are converted into beams (frequency domain) by use of a Fast Fourier Transform (FFT).

The on-board pulse compression used commonly in radar altimeters samples the radar echo at a frequency equal to the pulse bandwidth, which for the power echo results in an undersampling factor of 2 as described in Jensen (1999) and Smith & Scharroo (2014). A simple zero-padding (on the complex signal) before performing the FFT, can alleviate the related aliasing. In this way a better sampling of the power waveform leading edge is obtained, especially important for low SWH or very

specular echoes (e.g., leads or sea ice), in a way that both the range and SWH estimations are improved as stated in Smith and Scharroo (2014) for CryoSat-2 data.

We should note, as pointed out in Smith and Scharroo work (2014), that zero-padding can provide improved retrievals simply because more samples are available for the geophysical fitting of the model. This means that the leading edge can be properly sampled (this is especially interesting for very specular returns with low SWH) and that more samples can be available to perform a better estimation of the noise floor to be included in the retracking model. In this line, and for very small SWH, the resolution of the instrument (defined by the transmitted bandwidth) is not able to discriminate the leading edge, and so the inclusion or not of zero-padding will not help at all. On the other hand, and for high SWH, the inclusion of zero-padding does not provide any additional improvement, as the contribution of the power waveform spectrum outside the Nyquist bandwidth criterion is very low (differently from Jensen's claim, which ensures that deramped altimetric power waveforms are inevitably aliased). Therefore, and contrary to what is stated in Jensen (1999), zero-padding does not need to be necessarily applied regardless of the type of scenario being imaged (as it comes out from the analysis of the work of Smith and Scharroo 2014).

A zero-padding can be optionally considered in the time-domain, such that after FFT operation, an interpolated version of the range-compressed signal is obtained (Cumming and Wong, 2005). The impact of this operation is a finer range step (higher sampling frequency). Note that this process increases the number of samples of the output variable. After that, the power waveforms are computed (detection step):

$$\Psi_{stack}\left(b',n'\right) = \left|FFT_{zp}\left(\psi_{gm\_stack}(b',n)\right)\right|^2,\ b' \in \left[0, N_{bs}-1\right],\ n' \in [0, N_s \cdot ZP - 1],\ n \in [0, N_s - 1]$$

Eq. 3.13

where $\psi_{gm\_stack}(b',n)$ corresponds to the stack after applying the geometry corrections and *ZP* refers to the zero-padding factor or oversampling factor used in the range-compression via FFT.

Moreover, since the geometry corrections have been applied through an exponential (that is equivalent to a circular shift in the other domain as finite-length signals are considered), some samples may have suffered a wrapping. This has to be solved through a mask and force these samples to zero. This mask is computed through the sum of all the three geometry corrections together and it is applied the following way:

$$\Psi'_{stack}\left(b',n'\right) = \Psi_{stack}\left(b',n'\right) \cdot H_{mask}(b',n'),\ b' \in \left[0, N_{bs}-1\right],\ n' \in [0, N_s \cdot ZP - 1] \quad \text{Eq. 3.14}$$

3.2.6. Multi-looking

3.2.6.1. Purpose and scope

The objective of this algorithm is performing an average (incoherent integration) of all the waveforms that form each stack.

3.2.6.2. Mathematical description

The steps to multi-look the echoes in a stack, into a single waveform, are described in the following lines. Each step corresponds to each stage in the block diagram of Figure 3.17.

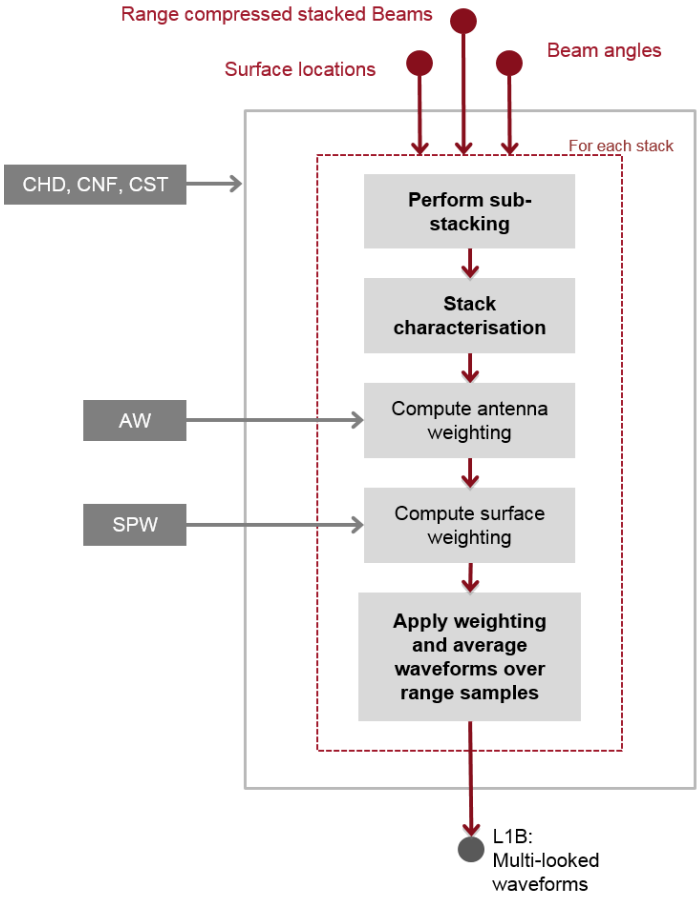


Figure 3.17: Block diagram of the multi-looking algorithm (credit: isardSAT). CHD, CNF and CST refer to the characterization, configuration and constants files, respectively; and AW and SPW are the antenna and surface weighting files (see Section 3.4).

For each stack, a few steps must be followed:

- Sub-stacking: In order to compute the stack characteristic parameters (3 dB width, skewness and kurtosis of a Gaussian power waveform fitting), a sub-stacking is performed. This means grouping the total power of the beams in small groups so as to reduce the noise.
- Retrieve stack characteristics: This reduction of the noise, leads the Gaussian fitting to a better performance, and then the related 3 dB width, skewness and kurtosis are retrieved.
- Antenna weightings computation: Before averaging the stacks, it might be of interest to remove or compensate for the antenna weighting (different beams are differently affected by the antenna pattern). These weightings, which are optionally activated (in the configuration file), can be both pre-set and/or provided by the user.
- Antenna weighting and averaging: After that, the weighting(s) are optionally applied and the waveforms are averaged over range samples:

$$\overline{\Psi}_{stack}(n') = \frac{1}{\sum_{b'=0}^{N_{bs}-1} W(b')} \cdot \sum_{b=0}^{N_{bs}-1} W\left(b'\right) \cdot \Psi'_{stack}\left(b', n'\right), \quad option\ \Psi'_{stack}\left(b', n'\right) \neq 0 \qquad \text{Eq. 3.15}$$

This operation (depending on a configuration flag) can be done taking into account all the samples or only the non-zero ones ($\Psi'_{stack}\left(b', n'\right) \neq 0$). Typically, all the range samples (even the zero-valued ones) are used when multi-looking the stack, which reduces the mean power on those samples, producing a distortion of the waveform. This is an issue of present discussion and so the re-trackers should be accordingly implemented taking into account the effect of such mask.

The Doppler- or beam-dependent weighting optionally included in the processing baseline may lead to an improved SNR, and hence to an improved SWH retrieval.

The SARin processing case requires and additional steps at this point: the coherence and the interferometric phase difference are computed over the cross product and as noted by Wingham et al 2006.

3.2.7. Sigma0 scaling factor

3.2.7.1. Purpose and scope

This algorithm is in charge of computing the scaling factor that allows to convert the power of the multi-looked waveform into σ^0 values (normalized radar cross section values). The flow chart of this algorithm is shown in Figure 3.18.

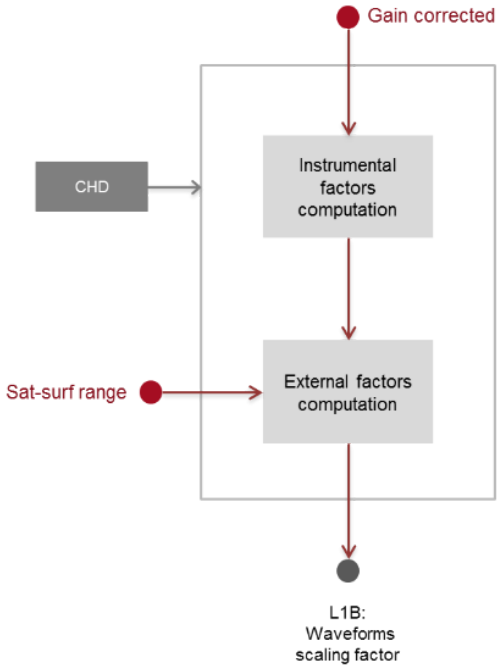


Figure 3.18: Block diagram of the sigma0 scaling factor algorithm (credit: isardSAT). CHD refers to the characterization file (see Section 3.4).

3.2.7.2. Mathematical description

Based on the classical radar equation of the received power (Curlander and McDonough, 1991), the scaling factor per each beam pointing a specific surface can be obtained as:

$$k_\sigma\left(b'\right) = \frac{4\pi}{P_{TX}\cdot G_0^2\cdot\lambda^2}\cdot\frac{\left(4\pi\cdot r^2(b')\right)^2}{A_{surf}(b')},\ \left[W^{-1}\right] \qquad \text{Eq. 3.16}$$

where the first fractional term is the so-called *instrumental scaling factor* and the second one represents the *external effects*.

P_{TX} is the transmitted average power; G_0 refers to the antenna gain at boresight (assuming the beam-dependent antenna pattern impact is compensated for in the antenna weighting step of the multi-looking algorithm Section 3.2.6 or in the retracking model); $r(b')$ is the slant range distance for the specific beam pointing the surface and $A_{surf}(b')$ corresponds to the on-ground and beam-dependent, resolved surface area.

The user can optionally provide the normalized sigma0 along-track pattern (as a function of the look angle) to obtain the scaling factor per beam.

Then, the sigma0 scaling factor to be applied over the multi-looked power waveform is obtained as the average value over the different beam-dependent scaling factors:

$$k_{\sigma^0} = \frac{1}{N_{bs}} \sum_{b'=0}^{N_{bs}-1} k_{\sigma^0}\left(b'\right), \quad [W] \qquad \text{Eq. 3.17}$$

3.3. Development Choices and Trade Offs

A set of new algorithms is being considered in the implemented Level 1-B processing as already described within the different processing steps in Section 3.2. The processing stages, which might potentially improve the performance, can be summarized as:

1. Burst azimuth weighting
2. Azimuth processing method: exact or approximate
3. Stack alignment method
4. Multi-looking with zeroes method
5. Zero-padding in across-track (range oversampling factor)

3.4. Data Flow

For details on the input/output description of the products and the format specification of the output products for the Delay-Doppler HYDROCOASTAL processor please refer to HYDROCOASTAL deliverables D2.1 IODD [RD-06] and D2.3 PSD [RD-07].

Input data

The input data for the HYDROCOASTAL Delay-Doppler processor consist of L1A calibrated data for Sentinel-3 and of calibrated FBR CryoSat-2 baseline-D products[3].

Ancillary information

The ancillary data can include apart from the auxiliary files (static and dynamic) additional information such as DEM, in-situ measurements from buoys, radiometers, and so on.

- **Static auxiliary files**:
  - **Characterisation file (CHD)**: includes system on-ground characterization (general, time pattern, platform, antenna, calibration…)
  - **Configuration file (CNF)**: Contains all the Delay-Doppler or SAR processor switches and processing options that can be accordingly activated/deactivated.
  - **Constants File (CST)**: includes the main physical constants used in the Level-1A to Level-1B processor.
  - Characterization arrays[4]:
    - **CAL2** masks corrections

[3] Calibration values are reported in the document "CryoSat characterisation for FBR users" available at https://wiki.services.eoportal.org/tiki-index.php?page=CryoSat+Technical+Notes).

[4] CAL2 mask corrections and CAL1 intraburst corrections are required to calibrate the FBR baseline-D from CryoSat-2. CAL4 is not considered here, therefore in SARin we are not starting from FBR but from L1B.

        - **CAL1** intraburst corrections
        - **Antenna weighting file (AW)**: includes the antenna pattern as a function of the antenna pointing angle, which is compensated before the multi-looking takes place.
    - Delay-Doppler **processor weighting** files:
        - **Azimuth weighting file**: includes the weighting applied at burst level prior to azimuth processing.
        - **Surface weighting file (SPW)**: includes the specific weighting for a given surface model as a function of look angle (applied before multi-looking).

- **Dynamic auxiliary files:**
    - **Orbit files:** include predicted orbit file, DORIS preliminary/precise orbit files, DORIS navigator file, GNSS orbit file.
    - **Attitude files:** include attitude restituted file, star tracker L0 files, star tracker configuration file, AOCS attitude file.
    - **USO drift file:** includes measured drift of the USO from its nominal clock value.
    - **CoM file:** includes variations on the satellite's centre of mass during the mission.
- **Land-sea mask (or water body mask file)**

Output

The science output files of the Delay-Doppler processor are: Level-1B-S (L1B-S) and L1B:

- **L1B-S** is an intermediate and optional output product of the SAR processor. It contains fully SAR-processed and calibrated SAR complex echoes arranged in stacks for each surface after slant range correction and before the multi-looking takes place.
- **L1B** is the final science product at the output of the SAR processor. It contains geo-located and fully calibrated multi-looked SAR power echoes.

3.5. References

[Cumming & Wong 2005] Cumming, I.G.; Wong, F.H., (2005) *Digital Processing of Synthetic Aperture Radar Data*. *Algorithms and Implementation*. Boston, MA: Artech House.

[Curlander and McDonough 1991] Curlander, John C., and Robert N. McDonough. (1991) *Synthetic aperture radar*. New York, NY, USA: John Wiley & Sons, 1991.

[Gommenginger et al. 2013] Gommenginger, C., C Martin-Puig, I Amarouche and R.K. Raney, (2013), Review of State of Knowledge for SAR altimetry over ocean, EUMETSAT Study Report EUM/RSP/REP/14/749304, V. 2.2, 21 Nov 2013

[Jensen 1999] Jensen, J. R. (1999), Radar altimeter gate tracking: Theory and extension, IEEE Trans. Geosci. Rem. Sens., 37(2), 651–658.

[Prats-Iraola et al 2014] Prats-Iraola, P.; Scheiber, R.; Rodriguez-Cassola, M.; Mittermayer, J.; Wollstadt, S.; De Zan, F.; Brautigam, B.; Schwerdt, M.; Reigber, A.; Moreira, A., (2014) "On the Processing of Very

High Resolution Spaceborne SAR Data," in *IEEE Transactions on Geoscience and Remote Sensing*, vol.52, no.10, pp.6003-6016, Oct. 2014

[Raney 1998] Raney, R. K. (1998) "The delay/Doppler radar altimeter," in *IEEE Transactions on Geoscience and Remote Sensing*, vol.36, no.5, pp.1578-1588, Sep 1998

[Smith & Scharro 2014] Smith, W. H. F., and R. Scharroo (2014), Waveform aliasing in satellite radar altimetry, IEEE Trans. Geosci. Rem. Sens., 53 (4), doi:10.1109/TGRS.2014.2331193.

[Wingham et al 2006] D.J. Wingham, C.R. Francis, S. Baker, et al.CryoSat: a mission to determine the fluctuations in Earth's land and marine ice fields; Adv. Space. Res., 37 (2006), pp. 841-871

4. Algorithm Description: Two Step Analytical (isardSAT)

4.1. Theoretical Description, physics of the problem

The Two Step Analytical retracker is a physical-based retracker implemented by isardSAT, based on the model originally proposed by [Ray et al. 2015.] The implemented version is presented in [Makhoul et al. 2018], and though it was derived specifically for Delay-Doppler waveforms it is also used in Fully-Focused data. It is implemented in a two-step fitting procedure to operate over inland waters, providing robust surface height estimation with a minimal modification to the SAR ocean retracker model [Gao et al 2019]. The waveforms reflected from inland water bodies are actually treated as ocean waveforms with peaky shapes due to low sea state rugosity. It has been used in multiple projects with isardSAT participation such as SHAPE [SHAPE 2016 / Garcia-Mondéjar et al. 2018] and S3 performance assessments over inland waters [Gao et al 2019].

4.2. Algorithm Definition: Processing Steps and Mathematical Description

The block diagram of the Level-2 processing based on the analytical retracker is depicted in Figure 4.1.

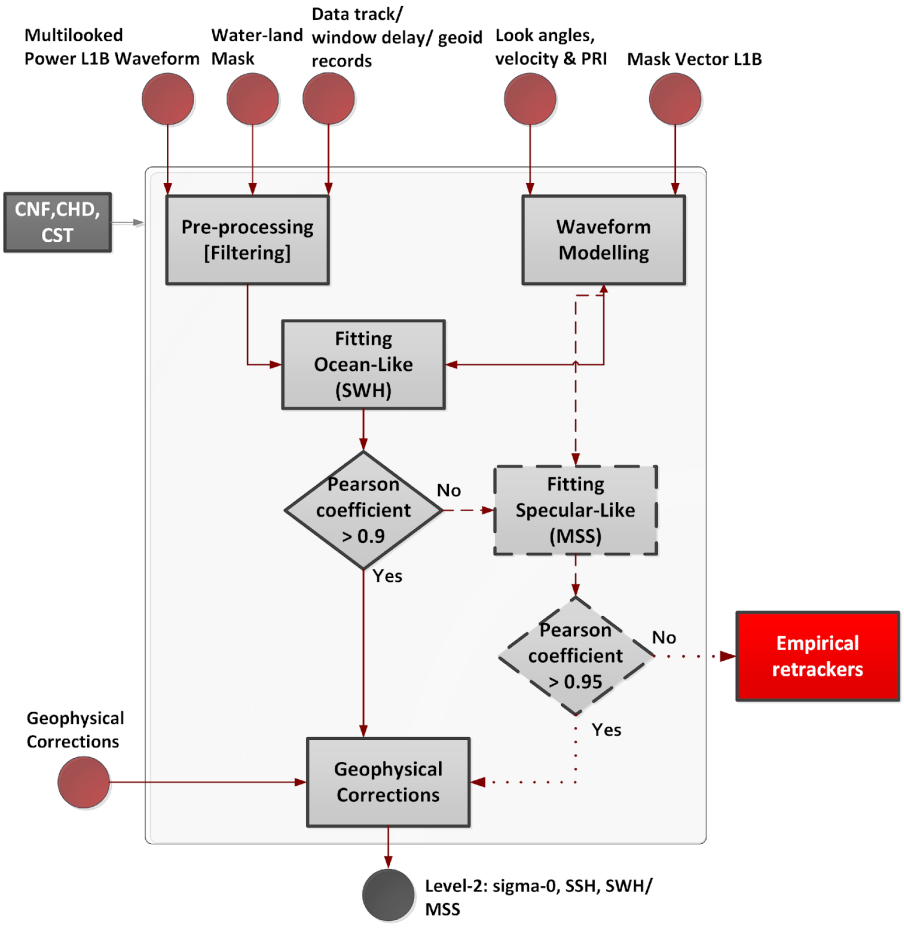


Figure 4.1: Analytical retracker block diagram. From now on CNF, CHD and CST refer to the configuration, characterization and constants files provided as inputs to the L2 processor (Credit: isardSAT).

The main steps included in this processing chain are:

1. Pre-processing (or waveform filtering).
2. Waveform modelling.
3. Fitting procedure.
4. Geophysical corrections.

Once the input multi-looked waveform is filtered out (land contamination and multiple peaks cropping), the fitting procedure is performed adjusting the multi-looked model waveform (obtained from the corresponding stack modelling) in a least square error minimization procedure.

Taking into account that the fully-analytical SAR retracker proposed by Ray et al. 2015 was originally derived for ocean-like surfaces, the implemented retracker tries, in a first iteration, to

perform an ocean-like fitting including Pu, epoch and SWH. In case the Pearson coefficient at the output is above 0.9, the waveform is classified as Ocean waveform; otherwise, a more specular- or leads-like scattering is expected, and so a new fitting procedure is carried out adjusting the MSS (mean-square slope) or roughness, while keeping SWH to a constant value (close to 0). Then, if the Pearson coefficient for this second fitting iteration is below 0.95, the waveforms are unfitted with SAMOSA analytical retracker and they might be better fitted with an empirical retracker (see section ICC-ER Empirical retracker).

4.2.1. Pre-processing

4.2.1.1. Purpose and Scope

This stage can be understood as a two-step filtering process (the logic of the pre-processing stage is shown in the flow chart of figure 4.3):

1. First, those **surfaces (or records) that fall outside the water body limits** provided by the water mask are filtered out and they are not considered in the retracking process at L2.
2. Secondly and for each one of the surfaces located within the limits of the water body, the **samples around** the one corresponding to **the nadir return (water body being imaged) need to be selected** as a multiple peak scenario can be expected due to land contamination.

In the following lines a detailed description of the second filtering stage is presented.

This second filtering will take place whenever a **multi-peak scenario** has been identified. Therefore, this requires to use a specific algorithm to identify the different peaks and depending on the separation of the identified peaks compared to a threshold value (configurable by the user), the subsequent filtering is carried out.

The block diagram of this second filtering stage is shown in figure 4.5. To minimize the impact of land contamination in the waveform retracking, a similar approach as the one proposed by P. García *et al.* in the CP4O project [isardSAT 2015] was in the SCOOP project as well. As shown in [isardSAT 2015], the **window delay information** can be used to properly extract a cut of the waveform corresponding to the nadir water body return:

- Within the limits of the water body of interest a smooth behaviour of the window delay (retracker range) should be expected, except for possible sudden changes related to land target contamination.
- Computing such **window delay step**, it is possible to locate the response corresponding to the nadir water body's return (unless this window delay jump is such that moves the corresponding nadir return outside of the tracking window).
- This window delay correction is used to generate the **seed (range bin location)** passed to the retracker along with the **number of samples at the right and left side of the seed**:
  - Retracker should be able to **dynamically adjust** the **region** of the waveform to be **fitted**.

- To compute the seed, **differences between the window delay** and its **smoothed version** is used:
  - The smoothed window delay can be obtained as a **polynomial fitting** to the original window delay (order of the polynomial is a configurable parameter by the user).
  - The previous approach (window delay fitting) may not work in case the on-board tracker is locked to the position of a strong off-nadir target, and if the water body section is short, then, the window delay is not representative of the nadir water body (which might be the case of small lakes or narrow rivers). In order to overcome such limitation, the window delay provided by **Geoid information** could be exploited as proposed by [García et al. 2018].
  - Water-land mask should be used to identify the sections of water body along the records, where the reference window delay should be extracted.

4.2.1.2. Data block and Diagram

Figures 4.2 and 4.3 show the block diagram for both preprocessing and waveform filtering.

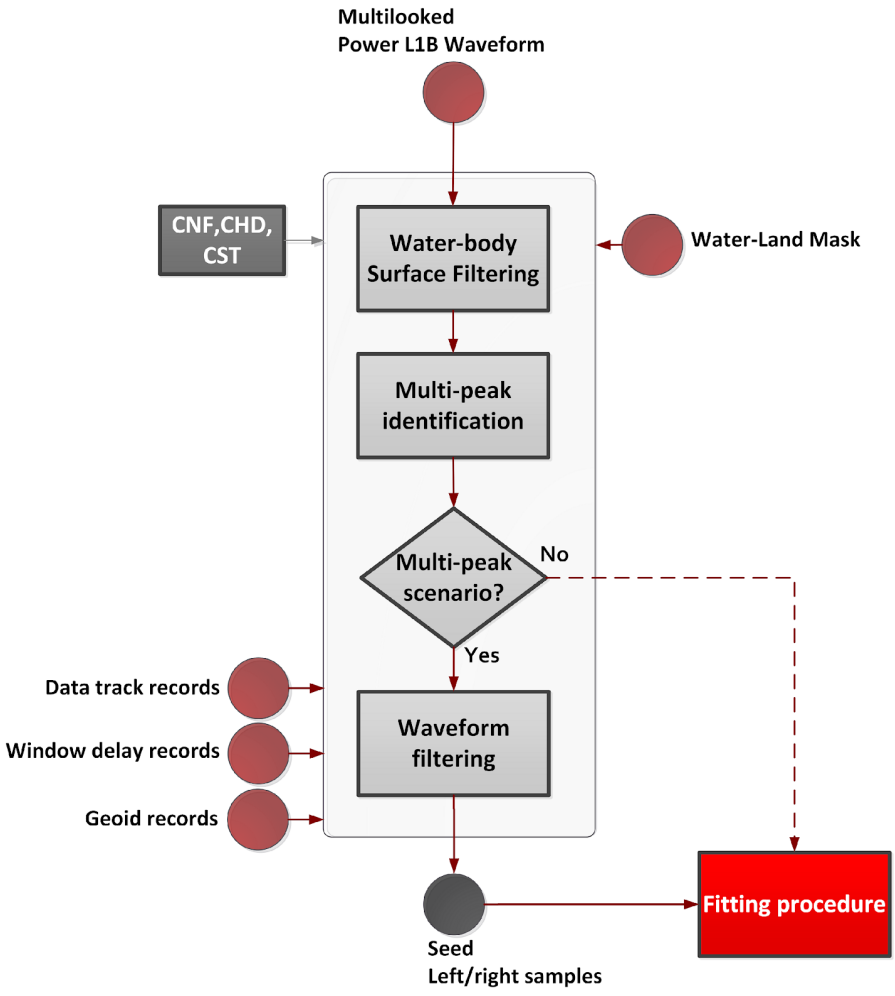


Figure 4.2 Pre-processing stage's block diagram. (Credit: isardSAT)

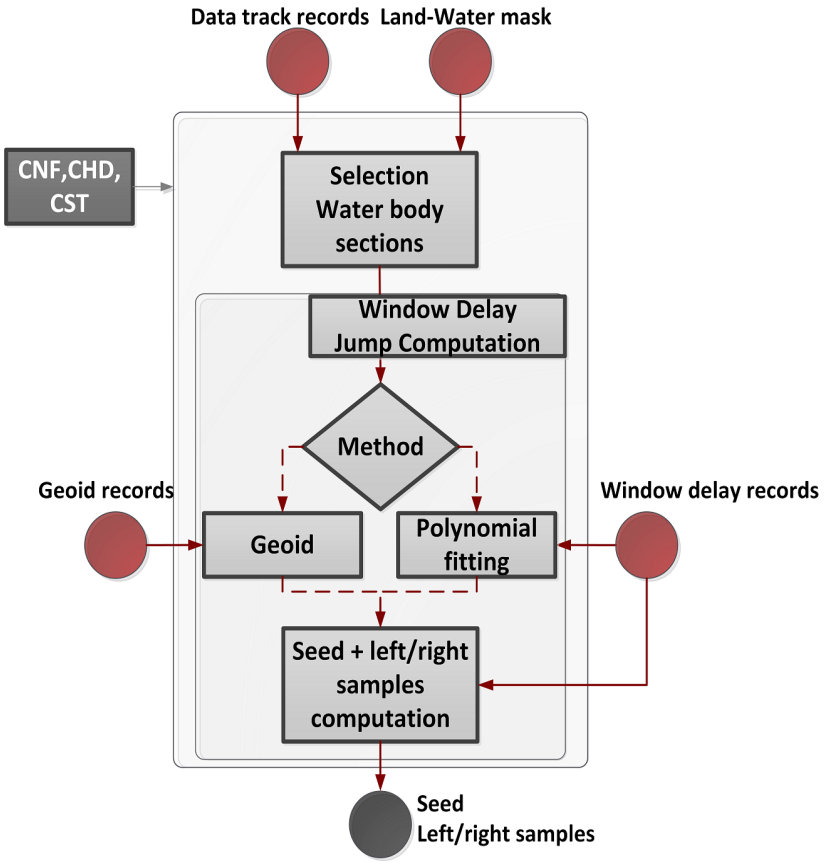


Figure 4.3: Waveform filtering block diagram. (Credit: isardSAT)

4.2.2. Waveform modelling

4.2.2.1. Purpose and Scope

This processing module is in charge of generating the theoretical model of the multi-looked SAR waveform used within the fitting procedure in order to infer the different geophysical estimates (including the retracked range correction).

The related block diagram showing the different stages involved in the stack modelling is represented in Figure 4.4.

The key steps required to generate the multi-looked SAR waveform are:

1. Noise Floor Estimation
2. Stack generation
3. Stack masking
4. Multi-looking

4.2.2.2. Data block and Diagram

Figure 4.4 shows the waveform modelling block diagram.

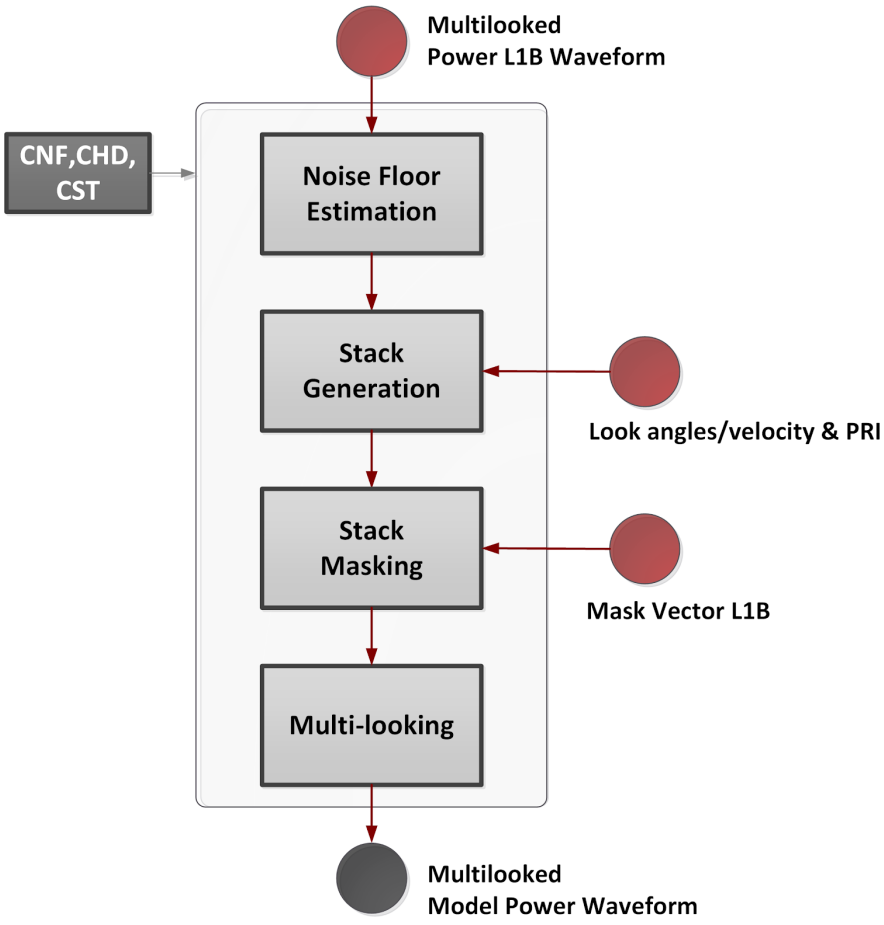


Figure 4.4: Waveform modelling block diagram. (Credit: isardSAT)

4.2.2.3. Mathematical Description

4.2.2.3.1. Noise Floor estimation

In order to ensure a realistic theoretical modelling of the SAR waveform the impact of both thermal noise and impulse response function (IRF)[5] side-lobe effects needs to be accounted for. In this way the theoretical single-look waveform can be described as shown in equation 4.1:

[5]The IRF is generally referred also as point target response (PTR). The 2-D SAR IRF has been well-approximated as a 2D-sinc function.

$$S_{k,l} = P_{k,l} + \sigma^2_{n,l} \quad \text{Eq. 4.1}$$

where $P_{k,l}$ refers to the power echo waveform backscattered by the surface being modelled (at range bin k and look index l) and $\sigma^2_{n,l}$ to the noise floor power for that given look index (including both thermal noise + side-lobe effects).

A simple estimation of the noise floor can be performed using a specific window, which should be located at the beginning of the observation window, right before the leading edge and sufficiently close to it in order to incorporate the impact of the secondary lobes, as shown in equation 4.2:

$$\sigma^2_{n,l} = \frac{1}{N_{ns}} \sum_{k=init_{ns}}^{end_{ns}=init_{ns}+N_{ns}-1} y_{k,l} \quad \text{Eq. 4.2}$$

where N_{ns} samples ($init_{ns}$ and end_{ns} the first and last samples) of the *l*-th beam power waveform $y_{k,l}$ are used to estimate the noise floor.

This option would require to have the stack information also available at the output of the Level-1B processor. Then, for each beam or look index the noise can be accordingly estimated and its value given to the single-look waveform modelling to build up the corresponding theoretical power waveform.

In case the stack is not annotated in the Level-1B product, the noise floor can be inverted from the multi-looked SAR waveform, and the same constant term is used in the generation of each single-look power waveform $\sigma^2_{n,l} = \sigma^2_{n,ML}, \forall l$. Taking into account that the thermal noise power is independent from look to look and equally distributed, both approaches should be equivalent except for any variation of side-lobes as a function of the beam pointing to that surface.

4.2.2.3.2. Stack generation

In order to obtain the multi-looked SAR waveform, the corresponding model stack should be build up from the single-look closed-form waveform solution of the SAMOSA retracker proposed in the technical note [SHAPE TN1 2018] (originally developed in [Ray et al 2015]).

From the block diagram in Figure 4.5 three main algorithms can be identified within the stack generation procedure:

1. Look or Doppler index generation
2. Single-look waveform modelling
3. Noise floor addition

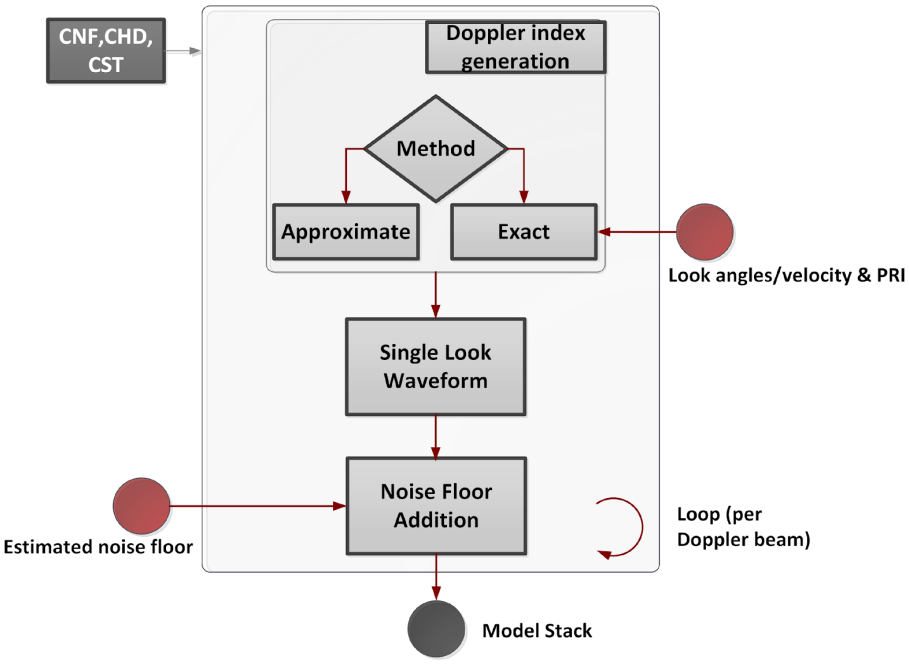


Figure 4.5: Stack generation block diagram. (Credit: isardSAT)

4.2.2.3.2.1. Look index generation

The look or Doppler index l associated to each look in the stack being modelled [see Eq. 4.3] should be properly initialised exploiting the look angle information $\theta_{look,b'}$ associated to each (b') of the contributing beams (in the stack) that point to that specific surface (see figure 4.6[6]). Such indexation information can be computed in the Level-2 processor, as shown in equation 4.3:

$$l = \frac{\theta_{look,b'}}{\delta\theta_{look,b'}} = \frac{\theta_{look,b'}}{\left(\frac{\lambda \cdot PRF_{b'}}{2 \cdot \left|\vec{v}_{s,b'}\right| \cdot Np}\right)} \qquad \text{Eq. 4.3}$$

where $\theta_{look,b'}$ refers to the look angle for the *b'* contributing beam of the stack for that surface; λ corresponds to the carrier wavelength; $PRF_{b'}$ is the PRF associated to the *b'* beam (linked to a given burst[7]); $|\vec{v}_{s,b'}|$ is the norm of the satellite's velocity vector for the *b'* beam or look (which

[6]For the flat earth geometry presented in figure 4.6, the look angle defined from the satellite point of view (measured from the nadir to the vector joining the satellite and the surface) is the same as the angle submitted from the surface point of view (normal of the surface to the vector joining the satellite and the surface).

[7]It corresponds to a generic formulation assuming that the contributing beams to the stack could have potential different PRFs as they are coming from different bursts, which can have different PRF as in the case of the future mission Sentinel-6.

corresponds to the satellite's velocity vector for the burst related to that beam within the stack); N_p is the number of pulses.

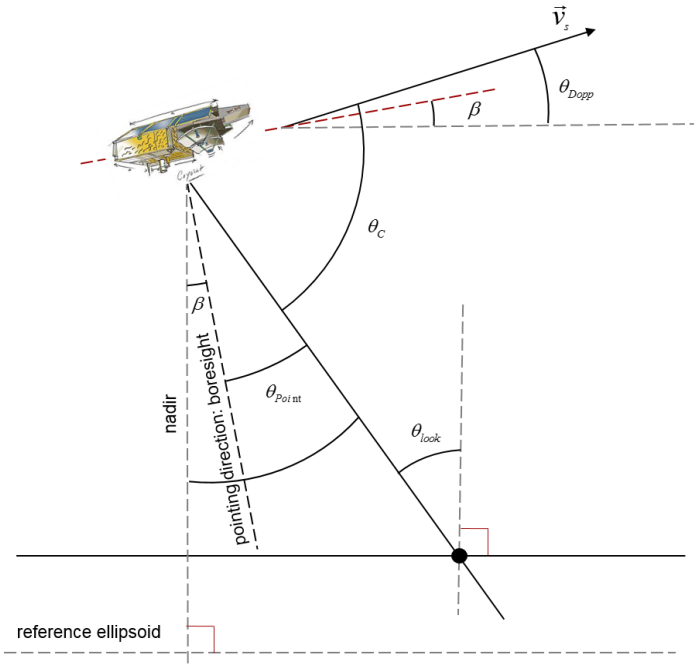


Figure 4.6: Along-track geometry: relationships between the different involved angles for a flat earth geometry. θ_{look} is the look angle defined by the angle from nadir to the vector joining the satellite position and the surface of interest; θ_{point} is the pointing angle defined by the angle from instrument boresight to the vector joining the satellite position and the surface of interest; θ_c corresponds to the beam angle defined between the satellite velocity vector and the vector from the satellite to that surface; θ_{Dopp} is the Doppler angle defined between the satellite's velocity vector and the vector perpendicular to the nadir; β refers to the pitch angle. (Credit: isardSAT.)

Such an approach, from here on referred as *exact indexation method*, would increase the amount of data volume to be included at the L1B product since for each surface and each beam conforming the corresponding stack, the look angle, the satellite's velocity vector and the PRF should be annotated in the L1B product. In this sense, an *approximate* solution would exploit the available information at Level-1B that can be analogously used to compute the indexation vector:

- From the maximum and minimum values of the look angles (θ_{look}) per stack (and the total number of contributing beams per stack), the corresponding linear vector information of such angle can be constructed (each element corresponds to a given $\theta_{look,b'}$)
- Using the satellite's velocity $|\vec{v}_{s,m}|$ and the PRF PRF_m right above the *m*-th surface, the indexation vector can be accordingly computed as described in equation 4.3, exploiting the uniformly sampled vector of look angles.

4.2.2.3.2.2. Single-look waveform modelling

As stated in [SHAPE TN1 2018], a wide variety of shapes for the received waveforms are expected over in-land waters (apart from the multi-peak response tackled in the pre-processing stage); i.e., in some cases a more ocean-like forms are feasible, while in others, a much narrower responses are obtained (very smooth surfaces acting as highly specular reflector).

Keeping such considerations in mind, the original SAMOSA single-look model developed by Ray *et al.* 2015 is being implemented in this case (assuming Gaussian ocean statistics). This model is adaptively tuned to consider also specular returns, characterizing the in-land water bodies (such as lakes), by means of mean-square slope (MSS) or roughness fitting, instead of significant wave height (SWH) fitting as in the oceanic-like scenario.

Then, this algorithm is in charge of implementing the following closed-form solution of the single-look power waveform, as shown in equation 4.4:

$$P_{k,l}(P_u, epoch, \tfrac{SWH}{MSS}) = P_u \cdot B_{k,l}(MSS) \cdot \sqrt{g_l(SWH)} \cdot f_0(g_l(SWH) \cdot k) \qquad \text{Eq. 4.4}$$

showing explicitly the dependency of the 3/4 parameters (P_u, $epoch$, $\frac{SWH}{MSS}$)[8] used in the fitting procedure. The *epoch* provides an estimation of the position of the leading edge, which is used to correct the measured range (window delay) in order to provide a refined estimation of the sea surface height (SSH). It must be noted that the first fitting parameter P_u allows the retrieval of the radar backscattering coefficient or normalized radar cross section (RCS), σ^0, once the appropriate scaling factor (computed at Level-1B) has been properly applied to the input waveform to be fitted.

As already mentioned, the index l denotes the look index and is related to the submitted look angle as described in the previous step. k refers to the range bin or index and the related vector of values can be obtained, as shown in equation 4.5:

$$\vec{k} = [1, \ldots, N_s] - epoch \qquad \text{Eq. 4.5}$$

From equation 4.4 three main components in the model can be identified:

- The term $B_{k,l}$ incorporates the information of the **antenna pattern, antenna pointing as well as the surface scattering model** being assumed. As noted in [Samosa 2015], this component corresponds to the constant term of the linear approximation of $\Gamma_{k,l}(z)$ which encloses the two-way antenna pattern $G^2(x = L_x \cdot l, y = L_y \cdot \sqrt{k + \frac{z}{L_z}})$ and the normalized RCS $\sigma^0(x = L_x \cdot l, y = L_y \cdot \sqrt{k + \frac{z}{L_z}})$ as shown in equation 4.6:

$$\Gamma_{k,l}(z) = G^2(x = L_x \cdot l, y = L_y \cdot \sqrt{k + \tfrac{z}{L_z}}) \cdot \sigma^0(x = L_x \cdot l, y = L_y \cdot \sqrt{k + \tfrac{z}{L_z}}) + G^2(x = L_x \cdot l, y = -L_y \cdot \sqrt{} \qquad \text{Eq. 4.6}$$

[8] Depending on the type of waveform being fitted, either the SWH (ocean-like scenarios) or the MSS (leads or in-land water bodies) is being considered in the fitting procedure.

where the along-track L_x and across-track L_y step distances (projected on-ground) are defined as (including possible across-track zero-padding factor ZP[9]) shown in equations 4.7 and 4.8 respectively.

$$L_x = \frac{c_0 H_{orb} PRF}{2 \cdot |v_s| f_c N_p} [m] \quad \text{Eq. 4.7}$$

$$L_y = \sqrt{\frac{c_0 H_{orb}}{\alpha ZP s \tau_u}} [m] \quad \text{Eq. 4.8}$$

with c_0 as the speed of light; H_{orb} the orbital altitude (right above the surface); f_c as the carrier frequency; $|\vec{v_s}|$ the norm of the satellite's velocity; N_p as the number of pulses per burst; α the orbital factor; s pulse chirp rate or slope; and τ_u as the pulse duration. All these parameters are accordingly initialised using the information available in the configuration and constant files. For each one of the filtered surfaces (or waveforms), the corresponding L_x, L_y and α should be computed based on the orbital height and satellite's velocity (when applies) at the surface location being analysed (this information is available as annotated variables in the Level-1B products). The computation of these variables is performed before the fitting procedure and saved in adequate structures to be accessible any time by the different functions.

Taking into account the antenna pattern (used for each specific instrument) and the underlying surface scattering models, an analytical closed-form expression of $B_{k,l}$ can be obtained as the constant term of the Taylor linear approximation of Eq. 4.6 around z=0.

The two other components in Eq. 4.4 are

- The **dilation term** g_l, in equation 4.9, takes into account the configuration of the instrument, the processing options (zero padding factor, along-track and across-track weighting) and the significant wave height ($SWH = H_s = 4\sigma_z$) as defined in Ray et al 2015.

$$g_l = \left(\sigma_{ac}^2 + (2\sigma_{al} \frac{L_x^2}{L_y^2})^2 + (\frac{\sigma_z}{L_z})^2 \right)^{-1/2} \quad \text{Eq. 4.9}$$

where σ_{ac} and σ_{al} correspond to the widths (standard deviations) of the Gaussian functions that approximate the PTR or IRF in the across- and along-track dimensions, respectively (any weighting is also included within the final values of σ_{ac} and σ_{al}); and σ_z represents the standard deviation of the Gaussian height probability density function (PDF).

- The **range-dependent functions** $f_n(g_l k)$ are modulated by the dilation term (including the beam or Doppler dependency) and can be obtained as shown in equation 4.10:

[9]Depending on the Level-1B processing baseline definition (like in CryoSat-2 baselines B, C and D), a zero-padding is applied in the Fourier domain (before applying the FFT) to effectively obtain an (across-track) interpolated version of the waveform, i.e., sampling rate in range is increased by the ZP factor.

$$f_n(\xi) = \int_0^{\infty} (v^2 - \xi)^n \cdot e^{-\frac{(v^2-\xi)^2}{2}} dv \qquad \text{Eq. 4.10}$$

which, for order n=0 and 1, can be solved using Bessel integral functions (exploiting the combination of the modified Bessel function of the first $I_p(\eta)$ and second $K_q(\chi)$ kinds[10]) as noted in Ray et al 2015.

Then, for each look index or Doppler beam (initialised accordingly in the previous step), the corresponding single-look signal power waveform is generated as defined in equation 4.4.

4.2.2.3.2.3. Noise floor addition

Once the single-look power waveform has been generated $P_{k,l}$, the estimated noise floor is added as shown in equation 4.11:

$$S_{k,l}(P_u, epoch, \tfrac{SWH}{MSS}) = P_{k,l}(P_u, epoch, \tfrac{SWH}{MSS}) + \sigma^2_{n,l} \qquad \text{Eq. 4.11}$$

When the different look indexes have been swept, the whole modelled stack is generated.

4.2.2.3.3. Stack masking

To be in line with the Level-1B processing, specific masking of the modelled stack should be performed in order to mask those samples per each beam, being affected by interference, ambiguities, aliasing and/or land contamination.

The same single mask (incorporating, among others, the ambiguities mask and the geometry corrections mask) used in the Level-1B processing before multilooking is used. For each (filtered) surface a vector mask is passed to the Level-2 processor, where for each beam the first non-valid range sample or bin is indicated.

Each beam of the masked stack can be modelled as shown in equation 4.12:

$$\tilde{S}_{k,l}(P_u, epoch, \tfrac{SWH}{MSS}) = S_{k,l} \cdot W_{k,l}$$
$$\text{where } W_{k,l} = \{1, k < k_{mask,l}\ 0, k \geq k_{mask,l} \qquad \text{Eq. 4.12}$$

with $k_{mask,l}$ being the first range bin for the l-th beam to be masked out. It must be noted that those samples forced to zero, can be alternatively set to a NaN (Not a defined number) such that they are omitted in the averaging procedure along the different beams. This is an option that can be integrated in the Level-2 processor to be aligned with the Level-1B case.

[10]The different mathematical symbols included in the Bessel functions notation is used to stress that different orders and arguments for the two types of Bessel functions might be involved in the combination to obtain a closed form expression for fn(glk), for further details please refer to [SAMOSA 2015].

4.2.2.3.4. Multilooking

After the stack has been formed, including the adequate masking, the stack is incoherently integrated (power averaging). This leads to the theoretical multilooked waveform fed to the fitting procedure.

The multilooking or averaging per range sample or bin *k* can be simply described as shown in equation 4.13:

$$S_{k,ML} = \frac{1}{N_{noNaN,k}} \cdot \sum_{l_{noNaN}} \tilde{S}_{k,l=l_{noNaN}},\ l_{noNaN} \in l \mid \tilde{S}_{k,l} \neq NaN \qquad \text{Eq. 4.13}$$

where in case the zero samples should not be considered in the integration, they are set to NaN values and so not included in the averaging (i.e., for each range bin only those samples different from NaN values are considered $N_{noNAN,k}$).

4.2.3. Fitting Procedure

4.2.3.1. Purpose and Scope

Based on the input waveform and the modelled stack, the fitting procedure tries to converge to a solution that minimizes the error between both on a LSE basis by iteratively updating the stack model.

4.2.3.2. Data block and Diagram

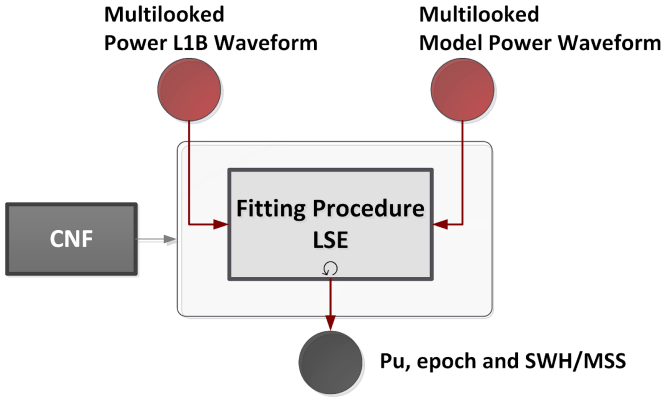


Figure 4.7 Fitting procedure block diagram. (Credit: isardSAT)

4.2.3.3. Mathematical Description

As already mentioned, different types of waveforms shapes are expected over in-land waters: more ocean-like type or more specular (narrower shape) like over leads.

Then, if the first kind of backscattering is expected, the SWH is being used as a fitting parameter, while keeping roughness (MSS) to a constant value. On the other hand, if a more specular return is observed, then the SWH is assumed constant and with a very small value, while the roughness is being incorporated in the fitting procedure. The least-square minimization problem can be implemented either using the “Levenberg-Marquardt” method or “trust-region-reflective” algorithm.

For the latter method, the non-linear system of equations involved in the minimization should not be undetermined, while for the “Levenberg-Marquardt" algorithm there are no bound constraints. In fact, the “trust-region-reflective” methods are an evolution of the classical “Levenberg-Marquardt” method, some discussion on this and more specifically optimization problems can be found in [Berghen 2004].

Such fitting problem can be mathematically formulated as shown in equation 4.14:

$$[P_u, epoch, \frac{SWH}{MSS}] = min\|S_{ML}(k; P_u, epoch, \frac{SWH}{MSS}) - y_{ML}(k)\|^2 \qquad \text{Eq. 4.14}$$

being $S_{ML}(k; P_u, epoch, \frac{SWH}{MSS})$ the multilooked model waveform and $y_{ML}(k)$ the input multilooked waveform from Level-1B product (once properly filtered in the pre-processing stage).

4.2.4. Geophysical Corrections

4.2.4.1. Purpose and Scope

Level 2 (L2) products contain the time of measurement and the geolocated height of the surface above the ellipsoid. To obtain accurate L2 product information (surface height/water body levels), geophysical corrections (see details in the Technical Note *Dry and Wet Tropospheric Corrections for Coastal Zones and Inland Waters*[11]) are necessary to correct the impact of instrumental- and environmental-dependent effects on the altimeter measurements.

[11] This Technical Note is provided appended to this ATBD.

4.2.4.2. Data block and Diagram

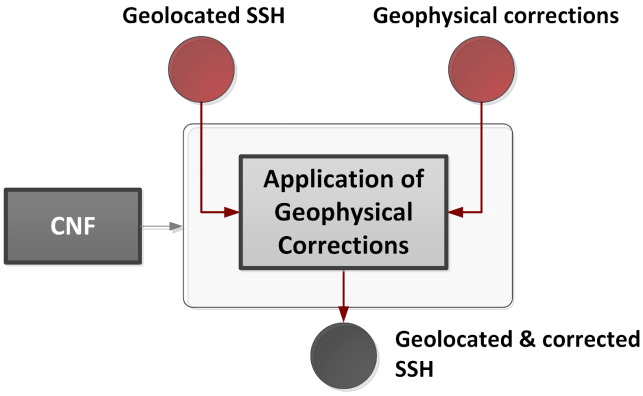


Figure 4.8 Geophysical corrections block diagram. (Credit: isardSAT)

4.2.4.3. Mathematical Description

In order to obtain the height information of the surface above the reference ellipsoid (SSH, equation 41), the satellite altitude H_{orb} and the measured range R to the surface of interest should be estimated:

$$SSH = H_{orb} - (R + \Delta R_{GEOcorr}) \qquad \text{Eq. 4.15}$$

where the different geophysical corrections $\Delta R_{GEOcorr}$ (Dry tropospheric correction, Wet tropospheric correction, Ionospheric correction, Ocean loading tide, Solid Earth tide, Geocentric polar tide) are properly applied. The range is obtained from the measured window delay τ_{wd} after considering the retracker correction $\Delta\tau_{retrack}$ (by means of the fitted epoch information) as shown in equation 4.16:

$$R = c_o/2 \cdot (\tau_{wd} + \Delta\tau_{retrack}) \qquad \text{Eq. 4.16}$$

where c_0 is the speed of light.

4.2.5. List of Symbols

$B_{k,l}$	Constant term of the Taylor approximation (around *z* = 0) of the antenna pattern and surface radiation patterns' product for the *l*-th beam and *k*-th range bin
c_0	Speed of light
end_{ns}	Last sample of the noise estimation window
$epoch_{init}$	Initial epoch provided to the fitting procedure

f_c	Carrier frequency
$f_n(\xi)$	Family of integral functions used to partially model the range-dependence of the single-look waveforms depending as a function of the dilation term g_l
f_s	Sampling frequency
$G(x,y)$	Antenna gain (same for transmission and reception) as a function of the along-track (x) and across-track (y) coordinates
G_0	Antenna gain at boresight (maximum gain)
g_l	Dilation term in the analytical SAMOSA model
H_{orb}	Satellite orbital height w.r.t reference ellipsoid
$init_{ns}$	First sample of the noise estimation window
$I_p(\eta)$	Modified Bessel function of the first kind and order *p*
k	Range bin or sample
k_{ns}	Noise range bin or sample
k_{offset}	Range bin offset due to the account for differences between sea height mean and the electromagnetic height bias
$K_q(\chi)$	Modified Bessel function of the second kind and order *q*
l	Look, beam or Doppler index
L_x	On-ground along-track sampling
L_y	On-ground across-track sampling
L_z	Vertical/height sampling
MSS	Mean-square slope
$N_{k,noNaN}$	Total number of beam samples not marked as NaN for a given range bin *k*
N_{ns}	Number of samples in the noise estimation window
N_p	Number of pulses per burst
$P_{k,l}$	Ideal noise-free modelled power waveform for range *k* and beam *l*
$PRF_{b'}$	Pulse repetition frequency for the *b'*-th (burst-related) beam pointing to the surface of interest
P_u	Fitted peak power
R	Range distance between satellite and surface
$S_{k,l}$	Modelled waveform for range *k* and beam *l*
$\tilde{S}_{k,l}$	Modelled waveform after application of the mask for range *k* and beam *l*
$S_{k,ML}$	Multi-looked modelled waveform
SWH	Significant wave height
$T_{k,l}$	Related to the linear term of the Taylor approximation (around *z* = 0) of the antenna pattern and surface radiation patterns' product for the *l*-th beam and *k*-th range bin
$\vec{v}_s$	Satellite's velocity vector over the surface of interest
$\vec{v}_{s,b'}$	Satellite's velocity vector for the *b'*-th (burst-related) beam pointing to the surface of interest

$W_{k,l}$	Stack mask for range *k* and beam *l* constructed from the stack mask vector contained in the L1B
x	Along-track coordinate
x_p	Ground projection of the pitch angle
y	Across-track coordinate
$y_{k,l}$	Measured and focused power waveform for the *l*-th look or beam within the L1B stack
$y_{ML}(k)$	Multi-looked power waveform to be fitted
y_p	Ground projection of the roll angle
z	Elevation (height) coordinate
ZP	Zero-padding factor in range
α_R	Orbital factor taking into account the earth curvature
β	Pitch angle from nadir
$\delta\theta_{look}$	Doppler resolution for the *b'*-th contributing beam pointing to the surface of interest
δR	Range sampling including potential zero-padding
$\delta R_{GEOcorr}$	Geophysical correction to be applied to the retracked range
a	Two-way antenna and surface radiation patterns' product
κ	Modified range bin index by the k_{offset} term
θ_c	Beam angle between vector from satellite to surface and the satellite's vector
θ_{Dopp}	Doppler angle between the satellite's vector and the vector perpendicular to the nadir vector
θ_{look}	Look angle between nadir and vector from satellite to surface
θ_{point}	Pointing angle between antenna boresight and vector from satellite to surface
$\sigma^2_{n,l}$	Noise power for the *l*-th look or beam
σ_s	Normalized (by the vertical sampling L_z) standard deviation of the Gaussian sea height probability density function
σ_z	Standard deviation of the Gaussian sea height probability density function
τ_{wd}	Measured window delay

4.3. Development Choices and Trade Offs

Available options:

- Hamming or Hanning windows.
- Waveform portion selection.

4.4. Data Flow

For details on the input/output description of the products and the format specification of the output products for the HYDROCOASTAL 2-step analytical retracker please refer to HYDROCOASTAL deliverables D2.1 IODD [RD-06] and D2.3 PSD [RD-07].

Input data

The input data for the HYDROCOASTAL 2-step analytical retracker consist of L1B calibrated data from Sentinel-3 or Cryosat-2.

Output

The science output files of the Delay-Doppler processor are: L2 data files, being its main output variables:

- **retracked_range:** Surface height wrt the ellipsoid in [m]
- **swh**: Significant wave height in [m]
- **retracked_sig0:** Backscattered coefficient σ_0 in [dB].
- **retracked_epoch:** Estimated epoch in seconds w.r.t center of the window
- **retracked_Pu: retrieved power**
- **flags:** Flag indicating successful or failed retracking
- **misfit_analytical:** Misfit between the real waveform and the fitted one as percentage

4.5. References

[Berghen 2004] Berghen, F. Vanden. "CONDOR: a constrained, non-linear, derivative-free parallel optimizer for continuous, high computing load, noisy objective functions." These de doctorat, Université Libre de Bruxelles, Belgium (2004).
https://dipot.ulb.ac.be/dspace/bitstream/2013/211177/1/96f72e79-d6ec-477c-823e-db05dfadf93e.txt

[Ray et al. 2015] C. Ray, C. Martin-Puig, M. Clarizia, G. Ruffini, S. Dinardo, C. Gommenginger, and J. Benveniste, “SAR Altimeter Backscattered Waveform Model,” IEEE Transactions on Geoscience and Remote Sensing, vol. 53, no. 2, pp. 911–919, 2015

[Makhoul et al. 2018] Makhoul, E., Roca, M., Ray, C., Escolà, R., & Garcia-Mondéjar, A. (2018). Evaluation of the precision of different Delay-Doppler Processor (DDP) algorithms using CryoSat-2 data over open ocean. *Advances in Space Research*, *62*(6), 1464-1478.

[Gao et al. 2019] Gao, Q.; Makhoul, E.; Escorihuela, M.J.; Zribi, M.; Quintana Seguí, P.; García, P.; Roca, M. Analysis of Retrackers’ Performances and Water Level Retrieval over the Ebro River Basin Using Sentinel-3. Remote Sen. 2019, 11, 718.

[Garcia-Mondéjar et al. 2018] Garcia-Mondéjar A., Bercher N., Fabry P., Roca M., Makhoul E., Fernandes J., Lazaro C., Vieira T., Gustafsson D., Restano M., Ambrozio A., Benveniste J. (2018). Evaluation of the Sentinel-3 Hydrologic Altimetry Processor prototypE (SHAPE) methods.

[García et al. 2018] Pablo García, Cristina Martin-Puig, Mònica Roca, "SARin mode, and a Window Delay approach, for Coastal Altimetry", Advances in Space Research, March 2018. DOI: 10.1016/j.asr.2018.03.015

[isardSAT 2015] isardSAT, "CP4O: SARin (and beyond) for Coastal Ocean", technical note ref. ISARD_ESA_CP40_TN_320, issue 1.c, December 2015.

[SAMOSA 2015] SAMOSA, "*Detailed Processing Model of the Sentinel-3 SRAL SAR altimeter ocean waveform retracker*", SAMOSA3-WP2300, ESRIN Contract No. 20698/07/I-LG, September 2015.

[SHAPE 2016] SHAPE- Sentinel-3 Hydrologic Altimetry Processor prototype – State Of the Art Technical Note – ATK_ESA_SHAPE_TN01_2015_040, 11/04/2016.

[SHAPE TN1 2018] ATK_ESA_SHAPE_TN01_2015_040, Technical Note 1: State of the Art, Issue 2.0, 2018-12-08.

5. Algorithm Description: Specialised SARin (Aresys)

5.1. Theoretical Description, physics of the problem

The Specialised SARin retracker is a physical-based retracker implemented by Aresys starting from the waveform model for interferometric SAR altimeter acquisition originally proposed in [Wingham et al., 2004]. A retracker for delay/Doppler multi-looked power waveform was already developed and exploited by Aresys [Scagliola et al. 2019] starting from a semi-analytical implementation waveform model for interferometric SAR altimeter acquisition [Recchia et al., 2017]. This retracker has been used in different projects with Aresys participation such as Sentinel-6 L1b Simulator and CryoSat Expert Support Laboratory.

Starting from the retracker for delay/Doppler multi-looked power waveform, it was identified that a step further could be done by exploiting in the retracking not just the power waveform but also the coherence and the phase difference waveforms. The Specialised SARin retracker is based on the interferometric waveform model in order to verify if it is possible to increase the accuracy and/or the precision of the retrieved geophysical parameters by a joint retracking of the power, coherence and phase difference waveforms from CryoSat SARin L1b products

The key component of the retracking tool is the waveform model. The theoretical single look power waveform model from a rough scattering surface as a function of the delay τ and of look angle ξ is obtained by the triple convolution $W(\tau,\xi) = p_t(\tau) * p_z(\tau) * X(\tau,\xi)$ where X() is the surface impulse response, $p_t()$ is the system point target response and $p_z()$ is the sea surface height probability density function. $X(\tau,\xi)$ represents a model of the stack of single look echoes as function of the look angle ξ, that is here defined as the angle at which the surface sample itself is seen with respect to the nadir direction. In the formulation of the adopted waveform model according to [Wingham et al., 2018], we have that:

$$X(\tau,\xi) = \frac{\lambda^2 G_0^2 D_0 c}{32\pi^2 h^3 \eta} H\left(\tau + \frac{\eta h \xi^2}{c}\right) \int_0^{2\pi} d\vartheta \cdot exp\left[j\frac{2\pi}{\lambda} B(\rho_k sin\vartheta - \chi)\right] \cdot d(\rho_k cos\vartheta - \xi) \cdot exp\left[-2\left(\frac{(\rho_k cos\vartheta - \mu)^2}{\gamma_1^2} + \frac{(\rho_k sin\vartheta - \chi)^2}{\gamma_2^2}\right)\right] \quad \text{Eq. 5.1}$$

The notation is consistent with [Recchia et al., 2017]. In this equation the last term is a circular integration along an iso-range line as function of the parameter ρ_k. The integrand is the product of the synthetic beam pattern and the elliptical antenna pattern:

$$G(\rho,\vartheta) = exp\left[-2\left(\frac{(\rho_k cos\vartheta - \mu - \zeta/\eta)^2}{\gamma_1^2} + \frac{(\rho_k sin\vartheta - \chi - \beta/\eta)^2}{\gamma_2^2}\right)\right] \quad \text{Eq. 5.2}$$

which is assumed to be separable in the along- and across-track direction and whose illumination width in the along- and across-track directions is determined by γ_1^2 and γ_2^2, respectively. The antenna pattern is centred around the pitch mispointing angle μ and the roll mispointing angle χ, that combine with the surface slopes in both the along and across track directions. The parameter $\eta = 1 + h/R$, where R is the Earth radius, is a geometric factor for spherical surface, while ζ and β denote the surface slopes in the along and across track direction, respectively.
The synthetic beam pattern is expressed as:

$$d(\phi) = \left(\sum_{n=0}^{N_a - 1} w_n exp\left[j2 \frac{2\pi v PRI}{\lambda} \left(n - \frac{N_a - 1}{2} \right) \phi \right] \right)^2 \qquad \text{Eq. 5.3}$$

where w_n are the coefficients of the window function applied in Level 1 processing, that determines the shape of the along-track impulse response.

Starting from the modelled stack of single look echoes $X(\tau, \xi)$, the impact of any type of masking at stack level in the Level 1 processing can be evaluated. Finally, the multi-looked waveform $\Psi(\tau)$ is obtained by incoherent averaging of the stack. It is worth noticing that the underlying assumption in the model is that the range migration is totally corrected, including the so-called range walk [Scagliola et al. 2019].

According to the waveform model above, we denote as $W(\tau, \xi)$ the cross-product impulse responses of individual beams and as $\Psi(\tau)$ the multi-looked cross-product waveform. We recall that the multi-looked cross-product waveform is the multiplication of three terms: $\Psi(\tau) = \Psi^p(\tau)\Psi^c(\tau)exp\left(j\Psi^d(\tau)\right)$, where $\Psi^p(\tau)$ denotes the power waveform, $\Psi^c(\tau)$ denotes the coherence waveform, and $\Psi^d(\tau)$ the phase difference waveform.

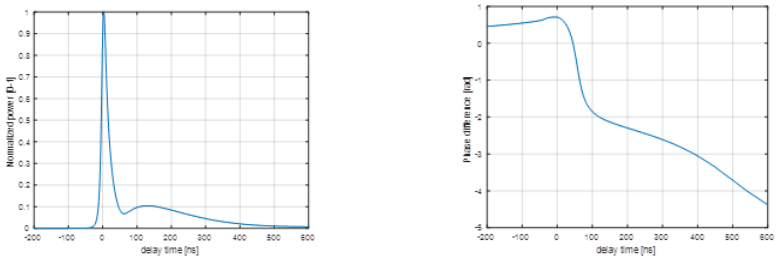


Figure 5.1: Modelled waveform: absolute value (left) and phase (right).

In Figure 5.1 above the modelled waveform in case of CryoSat, SWH = 2 m, pitch mis-pointing angle $\mu = -0.055\ deg$ and the roll mis-pointing angle $\chi = -0.125\ deg$ is shown. The model waveform was already validated in [Recchia et al., 2017]

5.2. Algorithm Definition: Processing Steps and Mathematical Description

The block diagram of the Specialised SARin retracker is depicted in Figure 5.2. The following main processing blocks are there included

1. Pre-processing: the waveforms plus the needed ancillary information are extracted from the L1b file and the subset of waveforms to be processed is selected
2. Waveform modelling: the semi-analytical model is called to compute the theoretical multi-looked SARin cross-product and phase difference waveforms, as function of the ancillary information associated to the current L1b waveform.
3. Fitting procedure: by iteratively updating the output of the waveform modelling, a least-square minimization approach is used to obtain an estimate of the geophysical parameters as the set of values that allow to reduce the distance between the L1b waveforms and the output of the waveform modelling.

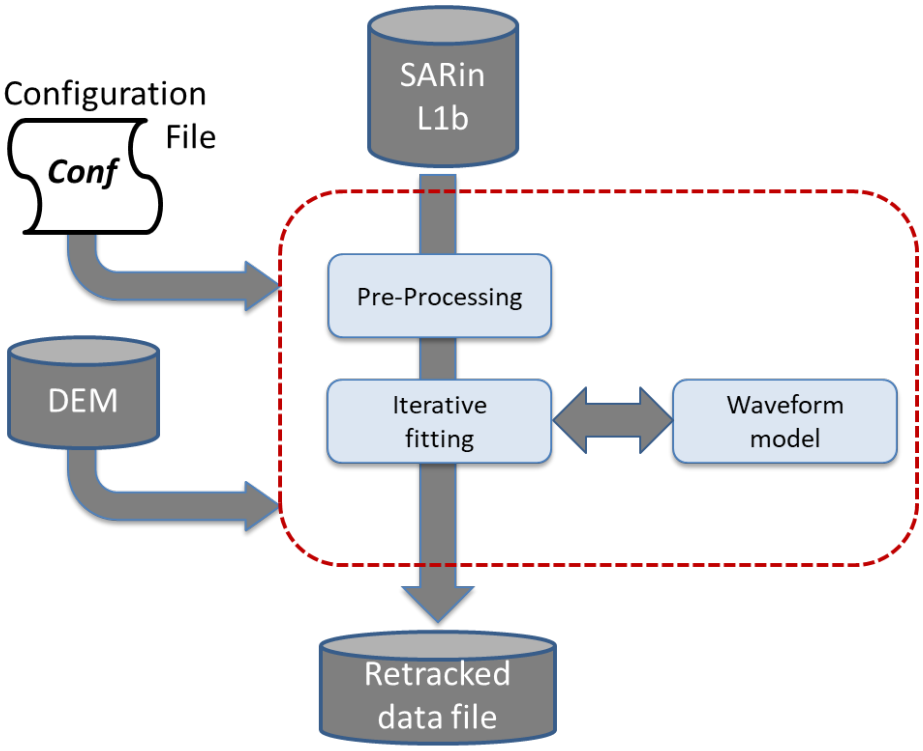


Figure 5.2: Specialised SARin retracker: block scheme.

5.2.1. Pre-Processing

This processing block is aimed at

- extracting from the L1b file the waveforms plus the needed ancillary information: the input L1b file is read and the following information are put in an internal data structure for each 20Hz record
    - timestamp
    - Latitude/Longitude/Altitude
    - Mispointing angles (yaw, pitch, roll)
    - Measurement confidence flags
    - Power waveform and Coherence waveform, that are multiplied sample by sample to obtain the cross-product waveform
    - Phase difference waveform
    - Tracking range or window delay
    - Look angle start/stop
    - Number of contributing beams in the stack
    - Surface characterization type flag
- selecting the subset of waveforms to be processed: a subset of waveform to be processed is selected according to the following conditions
    - the record is valid by inspection of the Measurement confidence flags
    - the record is flagged as ocean by looking at the Surface characterization type flag
    - if the Number of contributing beams in the stack are below a given threshold, the record is processed but flagged as incomplete stack
- computing the across-track surface slope, according to the user configuration: if in the user configuration it is selected to apply the slope correction, the across-track surface slope is to be added to the antenna bench roll angle. The across-track slope is computed from the EGM08 geoid following the approach described in Appendix C in [Galin et al. 2013].

5.2.2. Waveform Model

This processing block is in charge of computing the theoretical multi-looked SARin waveforms (cross-product and phase difference) that are then used in the iterative fitting procedure.
The block scheme of the waveform model is shown in the following figure.

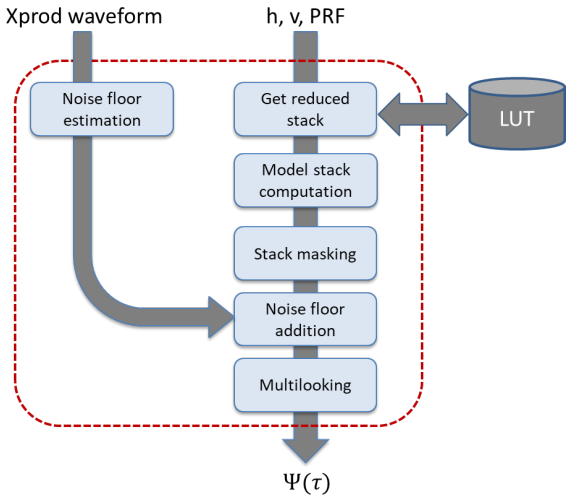


Figure 5.3: Waveform model: block scheme.

The waveform model is composed by the following elementary processing steps:

1. Noise floor estimation: it is performed on L1B cross-product waveform by averaging N power samples located at the beginning of the observation window, right before the leading edge

$$\sigma_n^2 = \frac{1}{N} \sum_{k=LE-N}^{LE} P_k \qquad \text{Eq. 5.4}$$

where P_k is the L1B cross-product waveform as function of the sample index *k* and LE is is the leading edge sample that is roughly estimated by an OCOG retracker and N is equal to N = round(LE/4).

2. Get Reduced Stack: according to the semi-analytical model in [Recchia et al. 2017], the surface impulse response $X(\tau,\xi)$ can be computed as a linear combination of 7 terms that are obtained by Maclaurin expansion of the integral to be computed in case of fully numerical model. From a practical point of view, look-up table can be precomputed of the 7 different terms, namely $I_n \; with \; n = 1,...,7$ for different values of the altitude h, the velocity v and the instrument PRF. This processing step is just in charge of loading from the available look-up table file, the set of I_n terms that were computed with the same PRF and with the closer values of h and v with respect to the current waveform. It is worth noticing that the I_n terms are already convolved with the system point target response. The reduced stack is then computed for k=1,...,64 look angles as

$$X(\tau,k) = H\left(\tau + \frac{\eta h \xi(k)^2}{c}\right) exp\left\{-2\left(\frac{(\mu)^2}{\gamma_1^2} + \frac{(\chi)^2}{\gamma_2^2}\right)\right\} [I_1(\tau,k) + (-\widehat{\mu})\frac{4\rho_k}{\gamma_1^2} I_2(\tau,k) + (\mu - \widehat{\mu})\frac{4\rho_k}{\gamma_1^2}(\mu \qquad \text{Eq. 5.5}$$

$$(\theta - \widehat{\theta})\frac{4\rho_k}{\gamma_2^2}\ (\theta - \widehat{\theta})\frac{4\rho_k}{\gamma_2^2} I_4(\tau,k) +$$
$$(\theta - \widehat{\theta})\frac{4\rho_k}{\gamma_2^2} I_5(\tau,k) + (\mu - \widehat{\mu})(\theta - \widehat{\theta})\frac{16{\rho_k}^2}{\gamma_1^2\gamma_2^2} I_6(\tau,k) + (h - \widehat{h}) I_7(\tau,k)$$

where H() is the Heaviside function, $\eta = 1 + h/R$, h is the satellite altitude w.r.t the ellipsoid, c is the speed of light, γ_1 and γ_2 are parameters related to the illumination width in the along- and across-track directions of the physical antenna [C2-TN-ARS-GS-5179], μ and θ are the pitch and the roll mispointing angles, $\widehat{\mu}$ and $\widehat{\theta}$ are the pitch and the roll with respect to the Maclaurin expansion was computed and $\widehat{h}$ is the satellite altitude used in the expansion. In case that in the user configuration it is selected to apply the slope correction, the across-track surface slope is combined with the roll, i.e. to compute $X(\tau,k)$ it is used $\theta + \beta/\eta$ instead of θ. Moreover we have that the k-th look angle $\xi(k)$ results in $\xi(k) = k\frac{\pi}{N_p k_o v * PRI}$, $k = -31,-30,...,32$, with N_p the number of pulses in a burst and k_o the carrier wavenumber, v the spacecraft velocity and PRI the instrument pulse repetition interval. Finally the term $\rho_k = \sqrt{c\tau/(\eta h) + \xi^2(k)}$ is function of both the delay and the look angle and accounts for the range migration correction.

3. Model stack computation: starting from the reduced stack $X(\tau,k)$, the actual theoretical model stack of single look echoe is computed by
    a. convolving with the sea surface height probability density function $p_z(\tau)$, modelled as a gaussian function with standard deviation equal to SWH/(2*c);
    b. the current delay axis is computed as function of the current epoch and then each single look in the reduced stack is interpolated on the so computed delay axis;
    c. interpolation on the look angle axis: starting from the 64 single looks in the reduced stack, that are related to a discrete set of look angles $\xi(k)$, the reduced stack is interpolated on the look angle axis composed by a number of samples equal to the number of contributing beams in the stack and equally spaced between the look angle start and the look angle stop that are read from the L1b file.
    d. The actual theoretical stack $X(\tau,\xi) = X^{pc}(\tau,\xi)exp\left(jX^d(\tau,\xi)\right)$ is thus obtained, that can be considered as the combination of a cross-product single look waveform stack $X^{pc}(\tau,\xi)$ and of a phase difference single look waveforms stack $X^d(\tau,\xi)$
4. Stack masking: the stack can be masked to be in line with the Level-1b processing. In the case of current CryoSat L1b processing the effect of the range migration alignment only is required to be masked.
5. Noise floor addition: the noise floor previously computed is combined with the single look stack

$$X(\tau,\xi) = (X^{pc}(\tau,\xi) + \sigma_n^2)exp\left(jX^d(\tau,\xi)\right) \qquad \text{Eq. 5.6}$$

6. Multilooking: After the stack has been formed, including the adequate masking, the stack is averaged in the look angle direction to obtain the theoretical multi-looked waveform

$$\Psi(\tau) = \Psi^{pc}(\tau) exp\left(j\Psi^{d}(\tau)\right) = \sum_{\xi} X(\tau, \xi) \qquad \text{Eq. 5.7}$$

5.2.3. Iterative fitting

This processing block is in charge of estimating the geophysical unknowns exploiting an iterative fitting procedure based on non-linear least square algorithm by minimization of the error between the cross-product and phase difference waveform from L1b product and those computed by the waveform model block.

The block scheme of iterative fitting is shown in the following figure.

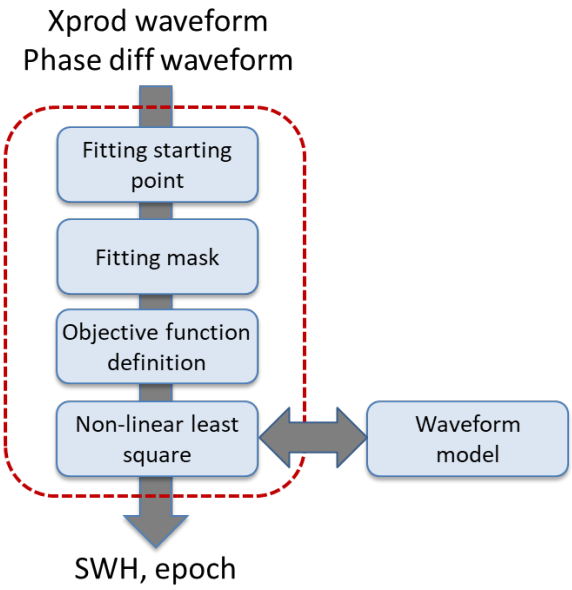


Figure 5.4: Waveform model: block scheme.

The iterative fitting model is composed by the following elementary processing steps:

1. Fitting starting point: the starting values for epoch and SWH are defined. The starting value for SWH is fixed to SWH = 2 m for the first N waveforms, then it is computed as the average of the already estimated SWH for the previous waveforms. The starting value for the epoch is computed using an empirical retractor on the power waveform (i.e. OCOG retracker).
2. fitting mask: it is possible to limit the fitting of the waveforms in a given interval of samples in the delay direction according to
    a. a distance in samples before and after the starting value for the epoch
    b. a coherence threshold, in order to fit only the samples where the coherence is higher than the given threshold

3. Objective function definition: the objective function is defined as the weighted sum of the absolute difference of the two waveforms

$$g(x) = \alpha \cdot \left| \Psi^{pc,L1b} - \Psi^{pc,mod}(x) \right| + \beta \cdot \left| \Psi^{d,L1b} - \Psi^{d,mod}(x) \right| \qquad \text{Eq. 5.8}$$

where x is the vector of the unknown, i.e. SWH and epoch.

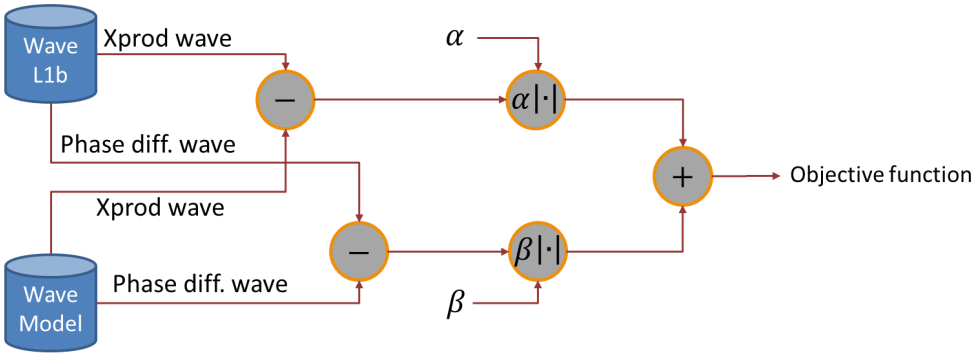


Figure 5.5: Objective function definition.

4. Non-linear least square fitting: the actual fitting is performed using iterative Levenberg Marquadt to obtain an estimate of the epoch and of the SWH. The problem is basically to look for the values of the unknowns that allow to minimize the objective function

$$x = [epoch, SWH] = min \|g(x)\|^2 \qquad \text{Eq. 5.9}$$

5.3. Development Choices and Trade Offs

The performance assessment in terms of accuracy and/or the precision of the retrieved geophysical parameters is still to be done. A fine tuning of the following parameters will be needed:

- fitting mask parameters
- weights in the objective function

A development choice is to allow also for independent retracking of the cross-product and phase difference waveform. The advantage of this choice is that it will be possible to retrack not only using a joint objective function but also in a sequence, i.e. obtaining a first estimate of the epoch and of the SWH from the cross-product waveform to be then refined by combining with the second estimate of the epoch and of the SWH obtained from the phase difference.

5.4. Data Flow

For details on the input/output description of the products and the format specification of the output products for the HYDROCOASTAL Specialised SARin retracker please refer to HYDROCOASTAL deliverables D2.1 IODD [RD-06] and D2.3 PSD [RD-07].

Input data

The input data for the HYDROCOASTAL Specialised SARin retracker consists of CryoSat SARin L1b products.

Output

The output file of the HYDROCOASTAL Specialised SARin retracker is a geophysical retracked data file, being its main output variables:

- **retracked_range:** Surface height wrt the ellipsoid in [m]
- **swh**: Significant wave height in [m]
- **retracked_epoch:** Estimated epoch in seconds w.r.t center of the window
- **flags:** Flag indicating successful or failed retracking
- **misfit_analytical:** Misfit between the real waveform and the fitted one computed according to the following formula:
  sqrt(sum((L1B_wave-fitted_wave).^2))/sqrt(sum((L1B_wave).^2))

5.5. References

[Galin et al., 2013] N. Galin, D. J. Wingham, R. Cullen, M. Fornari, W. H. F. Smith and S. Abdalla, "Calibration of the CryoSat-2 Interferometer and Measurement of Across-Track Ocean Slope," in IEEE Transactions on Geoscience and Remote Sensing, vol. 51, no. 1, pp. 57-72, Jan. 2013, doi: 10.1109/TGRS.2012.2200298.

[Recchia et al., 2017] Recchia, L., Scagliola, M., Giudici, D. and Kuschnerus, M. (2017). An Accurate Semianalytical Waveform Model for Mispointed SAR Interferometric Altimeters. IEEE Geoscience and Remote Sensing Letters, 14(9), pp.1537- 1541.

[Scagliola et al. 2019] Michele Scagliola, Lisa Recchia, Luca Maestri, Davide Giudici, Evaluating the impact of range walk compensation in delay/Doppler processing over open ocean, Advances in Space Research, 2019, ISSN 0273-1177, https://doi.org/10.1016/j.asr.2019.11.032.

[Wingham et al., 2004] D. J. Wingham, L. Phalippou, C. Mavrocordatos and D. Wallis, "The mean echo and echo cross product from a beamforming interferometric altimeter and their application to elevation measurement," in *IEEE Transactions on Geoscience and Remote Sensing*, vol. 42, no. 10, pp. 2305-2323, Oct. 2004, doi: 10.1109/TGRS.2004.834352.

[Wingham et al., 2018] D. J. Wingham, K. A. Giles, N. Galin, R. Cullen, T. W. K. Armitage and W. H. F. Smith, "A Semianalytical Model of the Synthetic Aperture, Interferometric Radar Altimeter Mean Echo, and

Echo Cross-Product and Its Statistical Fluctuations," in *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 5, pp. 2539-2553, May 2018.

[C2-TN-ARS-GS-5179] CryoSat Characterization for FBR users, issue 2.0, C2-TN-ARS-GS-5179, 13 June 2016

6. Algorithm Description: MWaPP (DTU Space)

The Multiple Waveform Persistant Peak (MWaPP) retracker was introduced in [Villadsen et al. (2019] and has since been implemented by others for studies of inland water altimetry [Xue et al. (2018), Jiang et al. (2020)]. The retracker is completely empirical, and did originally not rely on auxiliary data. However, to improve the performance of the retracker, a DEM and a water mask has been included in the modified processing for this project.

6.1. Theoretical Description, physics of the problem

The motivation for the MWaPP retracker is the issue of snagging from multipeak inland water waveforms. Snagging and hooking occurs when the on-board tracking system is dominated by specular surfaces that are located off-nadir, which leads to incorrect height estimates. Specular surfaces are most often shallow or calm water bodies, which reflect the signal from the altimeter without the scattering caused by rougher surfaces. The presence of these calm waters is often seen near the shore of lakes, or in areas prone to flooding.

The MWaPP retracker looks at adjacent waveforms in order to determine the best sub- waveform for retracking. In this way it is possible to identify persistent peaks, which are expected to represent the underlying water body of interest. Looking at neighbouring waveforms can help alleviate snagging issues, where a waveform is dominated by reflections from points off-nadir. The method presented here does not average waveforms, but simply tries to determine the bins in the waveform where the reflection from the water surface at nadir is most likely found. Since the range to the water body at nadir should be the same in all waveforms, off-nadir echoes will not dominate the averaged waveform.

6.2. Algorithm Definition: Processing Steps and Mathematical Description

For each waveform acquired above the water body, the heights corresponding to all bins are determined as defined in Eq. 6.1 below. This yields $N_w \times N$ height estimates, where N_w is the number of waveforms in the track and N is the number of bins in each waveform. Thus, a height is estimated for each $k = 1{:}128$ and $p = 1{:}N_w$ for all SAR mode waveforms.

$$H^{all}(p,k) = H_{alt}(p) - \tfrac{c}{2}WD + w_b(k_0 - k) - H_{geo}(p) - N_{geoid}(p) \quad \text{Eq. 6.1}$$

where H_{alt} is the satellite altitude, c is the speed of light, WD is the window delay, w_b is the bin width, k_0 is the nominal range bin number, H_{geo} is the sum of the applied geophysical and atmospheric corrections, and N_{geoid} is the geoid correction.

The surface height span of all waveforms within each river or lake crossing is determined as $[min(H^{all}){:}max(H^{all})]$ and the waveforms are oversampled to 1 cm height intervals using linear interpolation to derive the interpolated waveforms W_p^{int}. This allows for aligning the waveforms with respect to the obtained surface height instead of bin number.

The average of all W_p^{int} waveforms obtained over the water surface is then calculated. Since the height of the water body at nadir should be the same for all observations, an average of waveforms should not be dominated by off-nadir echoes and can be used to determine the subwaveform, which holds the nadir reflection. The waveforms will be different from each other due to off-nadir contamination caused by varying surface cover or topography.

For each of these averaged waveforms, the first peak that exceeds 20% of the maximum power is flagged. This is assumed to represent the water level common to all waveforms. In the original L1 waveforms, the peak closest to the flagged peak from the averaged waveform is found, and a subwaveform consisting of the three previous and following bins around this peak is extracted. The off-centre-of-gravity (OCOG) amplitude [Vignudelli et al. (2010)], *A*, is then calculated for the extracted subwaveform, which consists of *N* bins of which all but seven are zero. The point where the subwaveform exceeds 80% of *A* is marked as the retracking point.

6.3. Development Choices and Trade Offs

For this project, we have decided to make an adjustment to the original version of MWaPP presented in [Villadsen et al. (2016)]. For each track, all waveforms obtained over a water body (according to the Global Surface Water Explorer occurrence data [Pekel et al. (2016)]) are summed in order to achieve the average waveform and determine the most plausible retracking points in the original waveforms.

6.4. Data Flow

The required input data consists of L1b+L2 data from CryoSat-2 or enhanced data files from Sentinel-3A/B. The required input parameters are longitude, latitude, altitude, window delay, atmospheric corrections (ionosphere, dry+wet troposphere), geophysical corrections (solid earth tide, pole tide, loading tide) and the geoid height. Since the algorithm is empirical and developed solely for inland water purposes, the only output is the water level.

6.5. References

Jiang, L., Nielsen, K., Dinardo, S., Andersen, O.B., Bauer-Gottwein, P. (2020): Evaluation of Sentinel-3 SRAL SAR altimetry over Chinese rivers, Remote Sensing of Environment, *237*, 111546, https://doi.org/10.1016/j.rse.2019.111546.

Vignudelli, S., Kostianoy, A., Cipollini, P., Benveniste, J., 2010. Coastal Altimetry. Springer, Berlin, Heidelberg, https://books.google.com/books?id=M_2ljwEACAAJ

Villadsen, H., Deng, X., Andersen, O.B., Stenseng, L., Nielsen, K. & Knudsen, P. (2016): Improved inland water levels from SAR altimetry using novel empirical and physical retrackers, *Journal of Hydrology, 537*, https://doi.org/10.1016/j.jhydrol.2016.03.051

Pekel, J., Cottam, A., Gorelick, N. *et al.* (2016). High-resolution mapping of global surface water and its long-term changes. *Nature* 540, 418–422. https://doi.org/10.1038/nature20584

Xue, H., Liao, J., & Zhao, L. (2018). A Modified Empirical Retracker for Lake Level Estimation Using Cryosat-2 SARin Data. *Water*, *10*(11), 1584. https://doi.org/10.3390/w10111584

7. Algorithm Description: ICC-ER Empirical Retracker (ATK)

7.1. Theoretical Description, physics of the problem

The ICC-ER (Isolate, Cleanse, Classify - Empirical Retracker) is a software suite developed by ALONG-TRACK to address the non Brownian radar altimeter echoes. This suite is dedicated to multi-peak L1B-S data acquired in continental hydrology when the water network is dense and over sea ice regions where the multiple strong reflectors are leads, polynas and young ice.

Peaks are defined as consistent groups of consecutive range gates associated to powerful echo returns. Multi-peak WFs and Stack beams are obtained when a Radar Altimeter illuminates an heterogeneous surface that is made of a large number of targets of different types, roughnesses, sizes, shapes and locations within the footprint. Such WFs are very hard to model and the model fitting based retrackers have difficulties with Multi-peak WF. The situation is worsened by the fact that ghost signals may pollute the WF from the side-lobes of the antenna pattern. A common solution to this problem is to reject all of the “mixed” class WF, causing the loss of too many measurements.

The ICC-ER addresses the issue of noisy echoes potentially corrupted by spurious (ghost) peaks. The strategy is to cleanse the Stack prior to the multi-looking so as to deliver better waveforms to the sub-waveform retracking scheme. The ICC-ER has initially been designed for SARINM with the aim to measure epochs at several peaks per WF jointly with the possibility to determine one class per peak. In SARM the ICC-ER simply addresses the first major peak, assuming that it is at nadir. Multiple measurements can be exploited in SARINM after correcting for the slant range geometry thanks to the cross-track angle that is itself obtained from the phase difference.
The main steps included in the algorithm of the “Isolate, Cleanse, Classify - Empirical Retracker” are:

For each Doppler beam of the Stack (current record):

- **Isolate** the M major peaks, in a valley-to-valley definition of the peaks that account for local noise. This is reached in three major steps:
    - all ‘raw’ peaks and valleys are detected : each peak is naturally delimited by a fore and an aft valley,
    - all peaks are sorted in descending order with two options for the sorting method: sorting_method: {'peakvalue', 'v2p+p2v'} which is either the peak value or the sum of the two amplitudes around the peak (from fore valley to peak and from peak to aft valley),
    - the vicinity of each of the M highest peaks ($1 < m \leq M$) is iteratively browsed, first in the fore direction and then in the aft direction, in order to expand the peak's limits (the fore and aft valleys) by progressively integrating the small peaks (‘noise’) that are beyond these limits. The process ends in one direction as soon as a large enough peak is found in that direction. At each iteration the “browsed peak” with index m_B shall pass a “noise” test that involves a Noise-to-Peak ratio K_{NP} that is applied to the highest peak (m=1) to define an absolute acceptable noise threshold.

- **Cleanse** the Stack by removing the peaks that are poping up at some beams and are absent at most others. Rather than producing a unique “clean” waveform this step produces up to M “clean” pseudo-waveforms (PWF) in SARINM and up to one PWF in SARM. Let’s describe the SARINM case which is the most generic one. In order to build the pseudo-waveforms associated to the m-th peak, we build a new Stack by simply masking the initial Stack and keeping the samples that are within the valley-to-valley limits of this peak at all beams ; the rest of the Stack is set to *NaN*. At any range gate, whenever a peak is not “seen” in more than a given ratio of Contributing Beams, K_{CB}, then it is ignored and the whole range gate line is set to *NaN*. At the end of the process the most powerful contiguous block of range gates is the selected extent for the m-th peak. All other range gates of the Stack are set to *NaN*. If the peak extent does not reach a minimum size of I_{PEmin} range gates then the peak is discarded, otherwise the PWF corresponding to the m-th peak is obtained through multi-looking. We denote M_{PWF} the number of output PWF e.g. the number of peaks among M that have passed this step. In SARM we select the first PWF e.g. the one that has its non *NaN* values starting at the lowest range gate index.

Process all PWF of all records as follows:

- **Classify** the pseudo-waveform. There are two different versions of the classifier :
  - SARM classifier : a simple Pulse Peakiness (PP) test is performed on the PWF with two possible classes : {Water, Non Water} and a single PP threshold KPP,
  - SARINM classifier : WIP.
- **Retrack** the selected peak(s). The retracked point is located at $K_R \in]0,1[$ of the highest peak’s value on the leading edge side of the PWF (same threshold as the one used with ICE1 for SARM CS2 WF). The value of K_R depends on the *mission_id* ∈ {‘cryosat2’, ‘sentinel3a’, ‘sentinel3b’} and the *altimeter_mode* ∈ {'LRM', 'SARM', 'SARINM'}. A linear interpolation is used to provide the epoch value as a floating point number of bins.

7.2. Algorithm Definition: Processing Steps and Mathematical Description

7.2.1. Isolate()

This section describes the function *valley_to_valley_peaks()* at a single record.

7.2.1.1. Inputs

y : *y* : array of radar echoes which can be one of these two:

- $A(i,n)$: $I \times N$ array : linear scale power stack (at left antenna in SARM ; as the mean of both antennas in SARINM), indexed by the range gate number i and the beam index n,
- $w(i)$: $I \times 1$ array : linear scale power waveform indexed by the range gate number i.

M : *max_num_peaks* : scalar : $M \in [1, floor[I/2]]$ is the maximum number of major peaks to look for and I is the number of range gates in the input stack or WF (in SARM $M > 1$ is

accepted since this controls an intermediate step of the algorithm).

S_{SM} : *sorting_method* : string of characters : {'peakvalue', 'v2p+p2v'} : either the peak value itself or the sum of the two amplitudes around the peak.

K_{NP} : *max_noise_to_peak_ratio* : scalar with value in $]0,1[$.

7.2.1.2. Outputs

i_{start} : *ind_v2v_start* : $M \times N$ (stack case) or $M \times 1$ (waveform case) array : array of range gate indexes, in ascending order, indicating the beginning of each master peak for each of N beams or for the waveform. To ensure inter-record consistency of the outputs, the first dimension is set to M by default even if the array is partially or totally empty.

i_{stop} : *ind_v2v_stop* : $M \times N$ (stack case) or $M \times 1$ (waveform case) array : array of range gate indexes, in ascending order, indicating the end of each master peak for each of N beams or for the waveform. To ensure inter-record consistency of the outputs, the first dimension is set to M by default even if the array is partially or totally empty.

7.2.1.3. Algorithm

This function can process either a power stack or a power waveform.

The **steps** which are described below are applied to the linear scaled power waveform $w(i)$ but the code can handle the power stack $A(i,n)$ thanks to an additional loop (not given here for simplicity) on the beam indexes n and the outputs are then stored in 2D arrays instead of 1D arrays. The general principle of the algorithm is to start from the "highest peak" and then iterate on the peaks before and after. If these are small they can be considered as "noise" or "acceptable roughness" in order to integrate them to the leading edge or tail of the peak (consolidation of the limits of the peak). Two parameters are needed : M the maximum number of searched peaks ; K_{NP} the maximum acceptable **noise to peak ratio** that once applied to the maximum peak of the WF provides an absolute noise threshold K_{abs} that is used to test if the surrounding peaks are noise (are part of the current peak) or if we shall stop the expansion of the current peak.

1. STEP-1 **initialization** of variables.
2. STEP-2 determine the **indexes of all peaks** (also called **raw peaks**) $i_{peak}(p), p \in [1,P]$ (red crosses in figure [fig:example-step-3]) by detecting a change of sign in the waveform's first derivative (slope sign change test).
3. STEP-3 determine the **indexes of the pairs of valleys** $\left[i_{val\_bfr}(p), i_{val\_aft}(p)\right], p \in [1,P]$ around the peaks (black crosses in figure [fig:example-step-3]) by detecting a change of sign in the waveform's first derivative (slope sign change test). These valleys are also called the **raw valleys**.
4. STEP-4 ensure there is **one raw valley before and after each raw peak**, by removing peaks at the extremities.

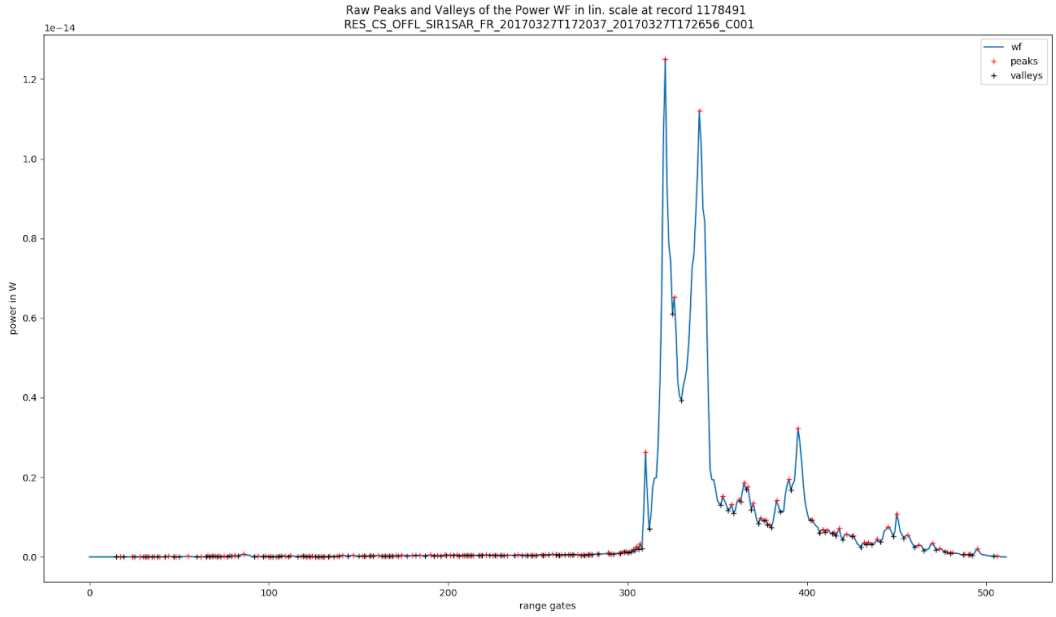


Figure 7.1: Example of the intermediate result at step 3 of the valley-to-valley peaks detection technique.

5. STEP-5 find the **left** and **right closest valleys for each peak** and compute the **valley-to-peak amplitude** $A_{v2p}(p)$ and **peak-to-valley amplitude** $A_{p2v}(p)$ for $p \in [1, P]$; these values will serve later to test if a peak is part of the "noisy peaks" or not.
6. STEP-6 **sort the raw peaks indexes by decreasing value** into i_{sorted_peak} and apply the **same sorting to the raw valleys** into $i_{sorted\_valley\_bfr}$ and $i_{sorted\_valley\_aft}$ other related tables (A_{sorted_v2p} , A_{sorted_v2p}).
7. STEP-7 **loop on the major peaks with index** $i_{sorted\_peak}(m), m \in [1, M]$ to consolidate them by integrating noisy peaks into their aft and fore limits (valleys) as follows:
   1. **iterate on** the decreasing values of i , the index of **the peaks before** $i_{sorted\_peak}(m)$, until one of the $M-1$ other major peaks is reached or until $w(i)$ does not pass the "noise test" defined in (c). When the exit condition is reached, update $i_{sorted\_valley\_bfr}(m)$,
   2. **iterate on** the increasing values of i , the index of **the peaks after** $i_{sorted\_peak}(m)$, until one of the $M-1$ other major peaks is reached or until $w(i)$ does not pass the "noise test" defined in (c). When the exit condition is reached, update $i_{sorted\_valley\_aft}(m)$,
   3. the "noise test" is:

$$A_{sorted\_v2p}(i) + A_{sorted\_v2p}(i) < K_{NP} \cdot \left(A_{sorted\_v2p}(0) + A_{sorted\_p2v}(0) \right)$$

8. STEP-8 **trade-off related to overlaps** (not yet implemented like this): as the tail consolidation is independent from one major peak to the other, the tails of two consecutive

consolidated peaks may overlap. In such a case the separation index should be the one of the lowest valley inside the intersection zone.

9. STEP-9 **valley-to-valley Integrated Power (VVIP)** associated to the consolidated peaks $E_w(m), m \in [1, M]$.

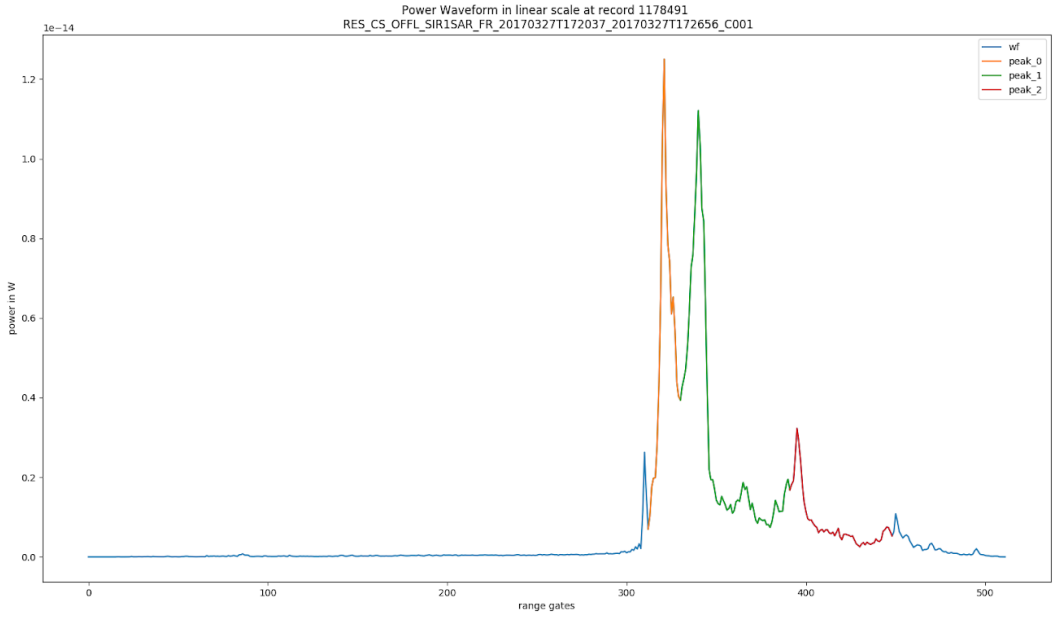


Figure 7.2: Example of the valley-to-valley peaks detection technique output with $M = 3$ peaks and $K_{npr} = 0.2$.

7.2.2. Cleanse()

This section describes the inputs, outputs and major steps of the function *clean_spurious_peaks_in_stack()* at a single record.

7.2.2.1. Inputs

y : *y* : array of radar echoes which can be one of these two:

- $A: I \times N$ array : linear scale power stack indexed by the range gate number i and the beam index n : $A(i, n)$,
- $w : I \times 1$ array : linear scale power waveform at current record indexed by the range gate number i : $w?(i)$.

i_{start} : *ind_v2v_start* : $M \times N$ (Stack case) or $M \times 1$ (waveform case) array : is an output of Isolate().

i_{stop} : *ind_v2v_stop* : $M \times N$ (Stack case) or $M \times 1$ (waveform case) array : is an output of Isolate().

K_{CB} : *min_contrib_beams_ratio* : scalar : the minimum proportion (of the total number of beams in the stack) that shall contribute to the same peak for any range gate within the valley-to-valley extent of the peak (it is recommended that this parameter is set greater than or equal to 0.33).

I_{PEmin} : *min_peak_extent_rg* : scalar : minimum extent of a WF / PWF peak in counts of range gates. This parameter cannot be lower than 3 and it is recommended to set it within 3 to 6.

S_{AM} : *altimeter_mode* : string of characters taking any value in {'LRM', 'SARM', 'SARINM'}.

7.2.2.2. Outputs

Please note that when *altimeter_mode is in {'LRM','SARM'}* then $M = 1$ in the definition of the output parameters, whatever the implicit value of M in the input parameters.

w_{pseudo} : *pwf_v2v* : $I \times M$ array : up to M pseudo-waveforms derived from either cleansing+multi-looking the power stack or cropping the input power waveform. In any case it is checked that the peak extent is greater than or equal to I_{PEmin} otherwise no PWF will be produced and this peak will be ignored.

$i_{pkstart}$: *ind_peak_start* : $M \times 1$ ('SARINM') or 1×1 (*'LRM','SARM'*) : array of range gate indexes, in ascending order, indicating the beginning of each master peak. In SARINM the first dimension is set to M by default even if the array is partially or totally empty (only the first M_{PWF} values are non Nan ; the non empy items being grouped at the low indexes).

i_{pkstop} : *ind_peak_stop* : $M \times 1$ ('SARINM') or 1×1 (*'LRM','SARM'*) : array of range gate indexes, in ascending order, indicating the end of each master peak. In SARINM the first dimension is set to M by default even if the array is partially or totally empty (only the first M_{PWF} values are non Nan ; the non empy items being grouped at the low indexes).

M_{PWF} : *num_peaks* : scalar : the number of processed major peaks in the waveform (the number of non Nan items in the first dimension of all other outputs).

7.2.2.3. Algorithm

$M_{PWF} = 0$

Initialise the output array of PWFs w_{pseudo} as an $I \times M$ array of NaN values

$I = y.shape\,[0]$

if $y.ndim = 2$: # if the input is a Stack

- $N = y.shape\ [1]$
- for $m \in [0, M-1]$:
  - Initialise a temporary mask $mask$ as an $I \times N$ boolean array filled with $False$ values
  - # create a mask to select the bins that lie between the start and stop limits defined for each beam :

$mask\left[i_{start}[m,n]:i_{stop}[m,n]+1,n\right]=True,\ n\in[1,N]$ # Note that +1 in the end index of the first dimension is a peculiarity of python when slicing arrays

- # duplicate (copy.deepcopy) the initial stack array in order to select only its m-th peak samples (the masking is performed by transforming A_{pseudo} into a numpy masked array)

  $A_{pseudo}[mask]=y[mask]$

- for $i\in[0,I-1]$: check all range bins and keep only those with a sufficient number of contributing beams (CB)

  if $\sum mask[i,n]>K_{CB}\cdot N$: # note that $True$ values account for 1 and the $False$ values for 0.

  - $w_{pseudo}[i,m]=\frac{1}{\sum mask[i,n]}\sum A_{pseudo}[i,n]$
  - # note that in theory different bins (indexed by i) will not necessarily have the same (nor the same number of) contributing beams, but in practice most of the beams will carry a very similar power distribution along the range bins and the Isolate() step designed to identify the most powerful consistent peaks. Mainly the beams at the edges of the Stack may loose consistency with the others which, in this case, will be a good reason for discarding their contribution into the pseudo-waveforms. If needed the less contributing beams may entirely be removed in order to improve consistency of the contributing beams).

  else:

  - $w_{pseudo}[i,m]=NaN$

  # identify contiguous blocks of $not(NaN)$ values in the m-th pseudo-waveform $w_{pseudo}[:,m]$

  - $condition=not\left(numpy.isnan\left(w_{pseudo}[:,m]\right)\right)$
  - i_c = *contiguous_regions*($condition$)
  - # i_c is a 2D array where the 1st column is the start index and the 2nd column is the end index of a contiguous blocks. We nominally expect only one interval but it might have been chopped into several intervals due to the beam to beam variability in the stack. In this eventuality we therefore have to select the interval that is the most representative of the peak all over the stack (the most powerful one).

  Among the contiguous blocks of range bins $\left[i_{cstart},i_{cstop}\right]\in i_c$ with a sufficient extent ($i_{cstop}+1-i_{cstart}\geq I_{PEmin}$) we select the one that maximises the peak power $numpy.sum\left(w_{pseudo}\left[i_{cstart}:i_{cstop}+1,m\right]\right)$ and we set the rest as NaN values:

- $M_{PWF} = M_{PWF} + 1$ # a PWF will be created for current peak (m)
- $w_{pseudo}\left[0 : i_{cstart}, m\right] = NaN$ # remove all other contiguous blocks before
- $w_{pseudo}\left[i_{cstop} + 1 :, m\right] = NaN$ # remove all other contiguous blocks after
- $i_{pkstart}[m] = i_{cstart}$; $i_{pkstop}[m] = i_{cstop}$ # store final peak's limits in the output arrays

else:

- # cancel this peak i.e. create a NaN values filled PWF for current peak (m)
- $w_{pseudo}[:, m] = NaN$; $i_{pkstart}[m] = NaN$; $i_{pkstop}[m] = NaN$

else: # the input is a waveform

- for $m \in [1, M]$:
  - $i_{cstart} = i_{start}[m, n]$; $i_{cstop} = i_{stop}[m, n]$
  - if $i_{cstop} + 1 - i_{cstart} \geq I_{PEmin}$:

    $M_{PWF} = M_{PWF} + 1$

    $w_{pseudo}\left[i_{cstart} : i_{cstop}, m\right] = y\left[i_{cstart} : i_{cstop} + 1\right]$

    $i_{pkstart}[m] = i_{cstart}$; $i_{pkstop}[m] = i_{cstop}$
  - else:

    $w_{pseudo}[:, m] = NaN$; $i_{pkstart}[m] = NaN$; $i_{pkstop}[m] = NaN$

if S_{AM} is not 'SARINM':

- $M_{PWF} = 1$
- # option 1: select the most powerful PWF as the one carrying the most powerful peak
  - $m_{final} = numpy.argmax\left(\sum w_{pseudo}[i, m]\right)$
  - if $i_{pkstart}\left[m_{final}\right] \neq min\left(i_{pkstart}\right)$ then raise a flag "possibly off nadir measurement"
- # option2 (the default one for now): select the very first peak
  - $m_{final} = argmin\left(i_{pkstart}\right)$
- # store the selection in the output arrays
  - $w_{pseudo} = w_{pseudo}\left[:, m_{final}\right]$; $i_{pkstart} = i_{pkstart}\left[m_{final}\right]$; $i_{pkstop} = i_{pkstop}\left[m_{final}\right]$

else:

- reshuffle the output arrays in a consistent way so that all non NaN PWF appear first.

7.2.3. Classify()

This section describes the inputs, outputs and major steps of the function *multi_class_per_waveform()* at a single record.

Please note that, as from this stage of the processing, when *altimeter_mode* is in {'LRM','SARM'} then $M = 1$ in the dimensions of the input parameters.

7.2.3.1. Inputs

w_{pseudo} : *pwf_v2v* : $I \times M$ array : is an output of the Cleanse() step.

K_{PP} : *thresh_pp* : scalar with value in $]0, 1]$: absolute Pulse Peakiness threshold (below : class : "Non Water", equal or above : "Water").

7.2.3.2. Outputs

Please note that when *altimeter_mode is in {'LRM','SARM'}* then $M = 1$ in the definition of the output parameters.

c : *surf_type* : $M \times 1$ array : among the possible values which depend on the application:

- hydrology: {0:'unknown/mix', 1:'water', 2:'non water'}
- sea ice: {0:'unknown/mix', 1:'open ocean', 2:'sea ice', 3:'lead'}

7.2.3.3. Algorithm

Please note that $numpy.nansum(x, axis = 0)$ is used here to compute the sum over index i.

for $m \in [1, M]$:

- if this PWF is made of NaN values only: if $\sum w_{pseudo}[i, m]$ =0:
  - $c[m] = 0$
- else:
  - compute the Pulse Peakiness, using $numpy.nanmean(w_{pseudo}[i, m])$ which mathematically can write $\frac{1}{I - \sum \delta(w(i), NaN)} \cdot \sum w_{pseudo}[i, m]$ to obtain the properly weighted mean in the presence of NaN values.

    $PP = \frac{m_i(w_{pseudo}[i,m])}{\frac{1}{I - \sum \delta(w(i),NaN)} \cdot \sum w_{pseudo}[i,m]}$, $\delta(x, y) = \{1, x = y \; 0, x \neq y$ #the Kronecker function $\delta()$

    is for mathematical modelling only
  - if $PP < K_{PP}$:

    $c[m] = 2$
  - else:

    $c[m] = 1$

7.2.4. Retrack()

This section describes the inputs, outputs and major steps of the function *multi_target_retracker()* at a single record.

7.2.4.1. Inputs

w_{pseudo} : ***pwf_v2v*** : $I \times M$ array of *float64* (do) : is an output of Cleanse().

w : ***wf*** : $I \times 1$ array of *float64* (do).

$r_{tracker}$: ***range*** : scalar of *float64* (do) : tracker range (range at the reference gate) in m.

r_{bin} : ***bin_width*** : scalar of *float32* (fl) : range bin width in m.

$i_{pkstart}$: ***ind_pk_start*** : $M \times 1$ ('SARINM') or 1×1 (*'LRM','SARM'*) array of *uint16* (us).

i_{pkstop} : ***ind_pk_stop*** : $M \times 1$ ('SARINM') or 1×1 (*'LRM','SARM'*) array of *uint16* (us).

S_{AM} : ***altimeter_mode*** : *string* of characters (str) : any value in {'LRM', 'SARM', 'SARINM'}.

S_{MI} : ***mission_id*** : one value in {'cryosat2', sentinel3a, sentinel3b}

K_R : ***thresh_rtk*** : float32 (fl) : table of retracker threshold as a function of the *mission_id* and *altimeter_mode*. In theory the threshold may also depend on some processing options (such as antenna pattern compensation, exact vs approximate beam steering, ...) but we ignore this for now. The values are stored in a python dict of dict: *thresh_rtk[mission_id][altimeter_mode]*. The following table provides the values of the retracker threshold which are currently used at ATK, but the table may evolve over time and this is the reason why it is passed to the function :

$K_R\,[S_{MI}, S_{AM}]$	'cryosat2'	'sentinel3a'	'sentinel3b'
'LRM'	0.3	0.3	0.3
'SARM'	0.87	0.87	0.87
'SARINM'	0.87	0.87	0.87

K_{SF} : ***scale_factor*** : dB : Scaling factor provided in the L1B product in order to retrieve sigma-0. It includes antenna gains and geometry satellite - surface. It is not applied to waveforms. For information, the computation of the sigma-0 scaling factor is based on the radar equation which indicates the power relationship between the echo transmitted and received considering a single beam. When the scaling_factor is given in dB it shall be added to the

7.2.4.2. Outputs

e_{rtk} : ***retracked_epoch*** : $M \times 1$ ('SARINM') or 1×1 (*'LRM','SARM'*) array of *float32* (fl): estimated epoch in meters w.r.t center of the window (tracker range is given to the center of the window) using the ICC-ER retracker.

r_{rtk} : ***retracked_range*** : $M \times 1$ ('SARINM') or 1×1 (*'LRM','SARM'*) array of *float32* (fl): range resulting from the sum of the ICC-ER epoch and the retracker offset (reference range which includes already the USO frequency drift and the internal/instrument corrections).

p_{rtk} : ***retracked_Pu*** : scalar of signed long (sl) in dBW : Retrieved power using the ICC-ER retracker.

σ_{rtk} : ***retracked_sig0*** : scalar of signed long (sl) in dBW : Backscattering coefficient computed from the retracked power once corrected by the sigma0 scaling factor.

flags

7.2.4.3. Algorithm

$I = y.shape[0]$

$M = y.shape[1]$

for $m \in [1, M]$:

- The retracked point of the PWF is located on the first leading edge before the peak extremum at $K_R \in]0,1[$ times $m_i\left(w_{pseudo}[i,m]\right)$. The value of K_R is taken from the input table. A linear interpolation is used to provide the retracked point abscissa i_{rtk} as a floating point number of bins.
  - get the bin number of the maximum value of the m-th PWF: $i_{max} = am_i\left(\left\{w_{pseudo}[i,m],\ i \in \left[i_{pkstart}, i_{pkstop}\right]\right\}\right)$
  - get the retracked point value in the m-th PWF: $w_{rtk} = K_R\left[S_{AM}, S_{MI}\right] \cdot w_{pseudo}[i_{max}, m]$
  - get I_a as the set of indexes of all points above the retracked point: $i \in [0, I[,\ w_{pseudo}[i,m] > w_{rtk}$
  - get I_b as the set of indexes of all points below the retracked point: $i \in [0, I[,\ w_{pseudo}[i,m] < w_{rtk}$
  - select i_a such that $i_a < i_{max}$ and $i_a = am_i(\{|i - i_{max}|, i \in I_a\})$
  - select i_b such that $i_b < i_{max}$ and $i_b = am_i(\{|i - i_{max}|, i \in I_b\})$
  - $i_{rtk} = i_b + \frac{w_{rtk} - w_{pseudo}[i_b,m]}{w_{pseudo}[i_a,m] - w_{pseudo}[i_b,m]}$
- Therefore, the ***retracked_epoch*** e_{rtk}, in m, is computed with respect to the reference gate located at $i = \frac{I}{2}$ in a zero based indexing system, as a decimal number of *bin_width:*
  - $e_{rtk}[m] = r_{bin} \cdot \left(i_{rtk} - \frac{I}{2}\right)$
- The ***retracked_range*** r_{rtk} , in *m,* is obtained from the *retracked_epoch* and the tracker range ; no geophysical correction is applied:
  - if $S_{AM} \neq 'SARINM'$:

$$r_{rtk}[m] = e_{rtk}[m] + r_{tracker}$$

- else:

  phase difference + phase coherence based retrieval of $r_{rtk}\,[m]$ to avoid errors from tracker hooking off nadir.

- The *retracked_Pu* p_{rtk}, in dBW, is the retracked waveform power estimate simply corresponding to the value of the waveform at the *retracked_epoch* (no integration, single value). It doesn't matter if the retracker is empirical, physical or just a random point.
  - $p_{rtk}\,[m] = 10 \cdot l_{10}\,(w_{rtk})$
- The *retracked_sig0* σ_{rtk}, is just the sum of the *retracked_Pu* and the *scaling_factor*, when they are both expressed in dB:
  - $\sigma_{rtk}\,[m] = p_{rtk}\,[m] + C_{SF}$

7.3. Development Choices and Trade Offs

Development choices and Trade Offs are inserted as comments in the algorithm description.

7.4. Data Flow

The data flow is very simple and is described in the 4 steps and 2 loops of section "Theoretical Description, physics of the problem".

7.5. References

None.

8. Statistical Retracker STARS Type (U Bonn)

8.1. Theoretical Description, physics of the problem

The Spatio-Temporal Altimeter Retracker for SAR altimetry (STARS) is an enhancement of the STAR retracker originally developed for low resolution mode (LRM, Roscher et al., 2017) and uses the functional waveform model Signal model Involving Numerical Convolution for SAR (SINCS, Buchhaupt et al., 2018) to retrack the Delay Doppler (DD) waveforms by estimating the three parameters epoch (Δt), amplitude (A) and significant wave height (SWH).

The STARS methodology consists of three steps: (1) partitioning the altimeter waveform into individual sub-waveforms; (2) retracking individual sub-waveforms; (3) analysing the point-cloud and deriving final estimates at each along-track position (see Figure 8.1).

STAR was applied to conventional altimetry (Roscher et al., 2017) and pseudo resolution mode (PLRM, Fenoglio et al., 2019).

SINCS is the SAR waveform model applied in the TuDaBo processor on ESA G-POD (https://gpod.eo.esa.int). Its main benefits compared to other SAR models as the SAMOSA model (Ray et al., 2015) are its flexibility, the possibility to use the real point target response (PTR) or more complex representations of the height probability density function (PDF) of scattering sea surface elements.

A sparse representation (SR) scheme is applied to partition each complete altimeter waveform in sub-waveforms. The return power of each waveform x_l, with $l = 1, ..., L$ and L the number of consecutive waveforms along a cycle, is represented in Eq. 8.1 by a linear combination of synthetic SINCS waveforms collected in a dictionary D,

$$x_l = D\alpha_l + \epsilon \qquad \text{Eq. 8.1}$$

with ϵ the reconstruction error and α_l the activation vector. Due to the sparse representation approach, most of the elements in α_l are zero. As neighbouring measurements along the track and on neighbouring waveforms are not independent from each other, STARS utilizes spatial and temporal information to analyse the return signal. Each range gate g (dependent on the input data) of the measured altimeter return waveform x_l is represented by a windowed waveform (centered at $\xi_{l,g}$), i.e. the range gate itself and neighbouring range gates. By using a Conditional Random Field (CRF, e.g. Lafferty et al., 2001; Halimi et al., 2016; Roscher et al., 2018), sub-waveforms are detected integrating the information about neighbouring range gates. Each windowed waveform is assigned to the best-fitting model $y_{l,g}$.

Within the CRF-framework, the energy functional (Roscher et al., 2017)

$$E(Y) = \sum_{l,g} U\left(\xi_{l,g}, y_{l,g}\right) - w \sum_{l,g,q \in Q} B\left(\xi_{l,g}, \xi_{l,q}, y_{l,g}, y_{l,q}\right) \qquad \text{Eq. 8.2}$$

with the sparse representation model or so-called non-zero activation indices $Y = \left[y_{l,g} \right]$, the unary term U and the binary term B, weighted with the hyperparameters w, and the set of direct neighbours (see Fig. 8.2) of each range gate $q \in Q$, needs to be minimised. The unary term describes how well the measured windowed waveform agrees with the sparse representation model. In order to influence the similarity of neighbouring range gates, the binary term includes prior knowledge between spatially adjacent range gates within a waveform and between temporally adjacent waveforms along the track. Those neighbouring range gates that are assigned to the same dictionary elements with the same activation vector are associated to one sub-waveform.

Each sub-waveform is retracked by fitting the SINCS model, providing that the sub-waveform contains a sufficient number of observations to allow the fitting algorithm to converge. The functional waveform model SINCS describes the backscattered power of the SAR signal P_D (Martin-Puig & Ruffini, 2009) as

$$P_D = FSSR_D(\tau, x_D) ** \left[PTR_\tau(\tau) \cdot PTR_x(x_D) \right] * PDF(\tau) \qquad \text{Eq. 8.3}$$

with the along-track coordinate of a Doppler beam x_D, range time τ, flat sea surface impulse response $FSSR_D(\tau, x_D)$, PTR in range direction $PTR_\tau(\tau)$, PTR in azimuth direction $PTR_x(x_D)$ and the probability density function (PDF) of point scatter on the sea surface $PDF(\tau)$. The one-dimensional convolution operator is represented by $*$, the two-dimensional convolutional operator by $**$, the simple multiplication by $\cdot$. The point target response (PTR) is not approximated.

Applying Fourier transforms to Eq. 8.3, we obtain P_D in the frequency/slow-time domain:

$$\overline{\overline{P}}_D(f, \eta) = \overline{\overline{FSSR}}_D(f, \eta) ** \left[\overline{PTR}_\tau(f) \cdot \overline{PTR}_x(\eta) \right] * \overline{PDF}(\tau) \qquad \text{Eq. 8.4}$$

with frequency f and slow-time η. The double bar represents a double Fourier transform. Retracking all candidate sub-waveforms results in a point-cloud for the three parameters sea surface height (SSH), SWH and backscatter coefficient (σ^0) each.

The point-cloud is further analysed to obtain a final solution for each 20 Hz measurement position. The algorithms used in STARS (STAR V2.5) differ from the approach in Roscher et al. (2017), where a simple Dijkstra algorithm was applied to SSHs to obtain the shortest path under the assumption that neighbouring measurement positions prefer similar solutions. Instead, we (1) replace SSH with SLA to remove bathymetry influences and (2) use a modified simple shortest path algorithm which allows for more dynamic edge weighting and for the incorporation of prior information, such as distance to coast and a first coarse retracking estimate. This rough retracking prior information is combined with an estimated straight line to remove large outliers from the point cloud and reduce the search space for the shortest path algorithm. For estimating the straight line, the DBSCAN algorithm (Density-Based Clustering of Applications with Noise; Ester et al., 1996) is applied to cluster the points and the RANSAC algorithm (Random Sample Consensus; Fischler and Bolles, 1981) is applied to the median of the clustered points to remove noisiness and to find the best fitting line within a window moving along the track. The final points are selected by a modified shortest path algorithm in combination with the prior information (RANSAC line).

8.2. Algorithm Definition: Processing Steps and Mathematical Description

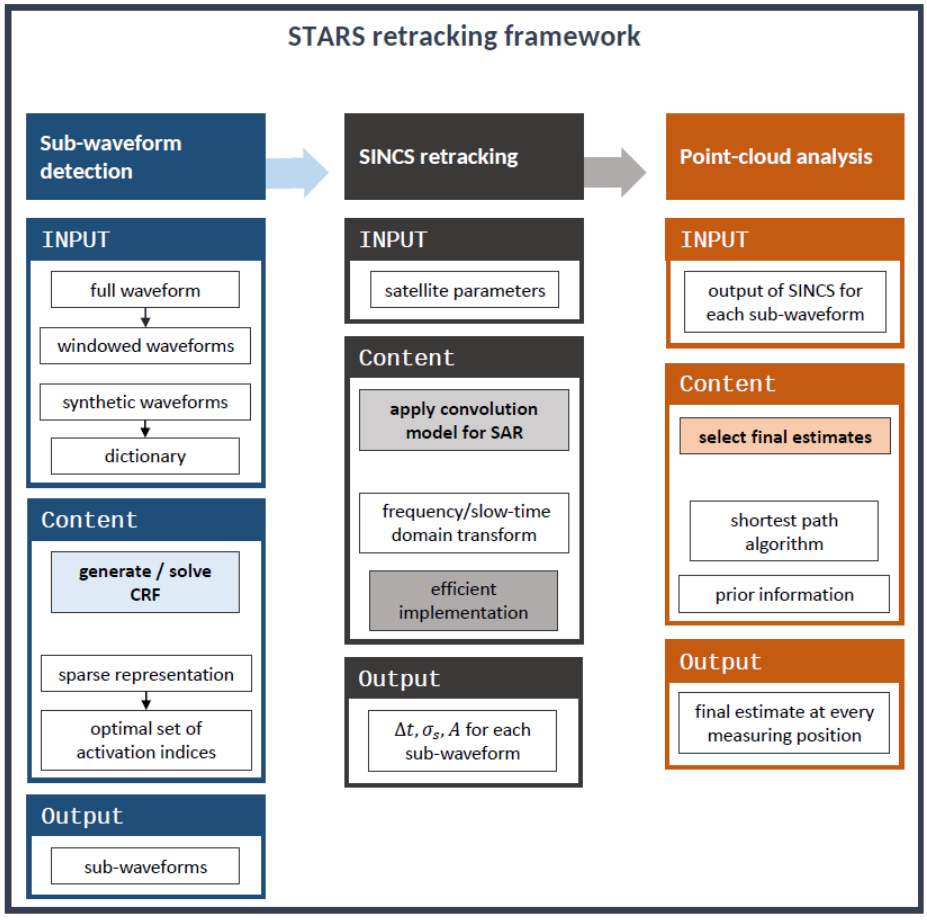


Figure 8.1: Diagram of the STARS Algorithm.

The STARS V1.0 retracker combines STAR V2.5 and SINCS V1.7. It comprises the following processing steps (see Figure 8.1):

1. partitioning the waveform into individual sub-waveforms
    a. generating dictionary elements
    b. constructing the CRF
    c. selecting sub-waveforms based on the CRF-solution
2. retracking all individual sub-waveforms using SINCS
3. selecting final estimates for each 20 Hz position
    a. pre-processing of the point-cloud
    b. applying DBSCAN and RANSAC for line detection within point-cloud

c. selecting final estimates

8.2.1. Partitioning the waveform into individual sub-waveforms

Input for the STARS algorithm are L waveforms along the altimetry ground-track, each containing G range gates, arranged in a $G \times L$ matrix, x_l, $l = 1, ..., L$. A waveblock matrix is defined with dimension $G \times K$ with usually $K = 20$, based on the 20 Hz measurements.

To get a first approximation of the waveform parameters to generate the dictionary, we apply an empirical approach based on the Off-Centre of Gravity (OCOG) / threshold method (Wingham et al., 1986). To get a broader basis for the dictionary generation, we chose four thresholds (1%, 10%, 30%, 50%).

Generating dictionary elements

For each waveblock, a dictionary is formed. Synthetic waves are created by generating 1,000 waveforms from the SINCS model. The waveform parameters are randomly picked based on the derived OCOG / threshold outputs and combined with noise. The dictionary D is set up including 15 elements, where only those waveforms are kept, which are most distinctive from each other (measured with e.g. cross-correlation coefficients).

Constructing the CRF

The next step is to compute the adjacency matrix, which connects the spatially and temporally neighbouring range gates represented by the binary term (Eq. 8.2). The adjacency matrix $A_{RR} = \left[a_{i,j}\right]$ is symmetric ($n_R \times n_R$) with n_R the number of range gates per signal. If $a_{i,j} = 1$, there is an edge between two vertices i,j: this is the case between direct neighbour and measuring point. If $a_{i,j} = 0$, there is no connection between two vertices. The diagonal elements of A_{RR} are zero.

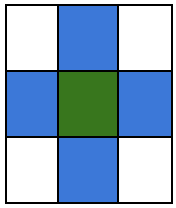


Fig. 8.2: Example for direct neighbours (blue) of measure point (green), in our case number of neighbours is $n_N = 1$.

The distance to each connected neighbour is calculated and registered in the adjacency matrix. Finally, we obtain the adjacency matrix A_{RJ}, with n_J the number of signals, by including the distance information of spatial and temporal neighbours for the whole waveblock. Note that the size of the neighbourhood for each windowed waveform can be different. Here, we set the number of neighbours in a windowed waveform to $n_\xi = 5$. The hyperparameter w is chosen to be

$w \in W = \{5, 10, 25, 100\}$ in order to provide meaningful partitioning of the total waveform based on the employed SINCS model and due to different influences such as land impact or sea state conditions. Consequently, Eq. 8.2 is solved four times, resulting in four sub-waveform partitionings.

For solving the cost function $E(Y)$ in Eq. 8.2, we use the GCoptimization[12] software (Boykov et al., 2001; Kolmogorov & Zabih, 2004; Boykov & Kolmogorov, 2004). In addition to the adjacency matrix, the unary term in Eq. 8.2 is used as input to solve the CRF. Therefore, solving Eq. 8.1 by finding the optimal solution of activation vectors $\widehat{\alpha}_{l,g}$ is necessary. The optimal can be formulated as $\widehat{\alpha}_{l,g} = argmin||D_g \alpha_{l,g} - \xi_{l,g}||_2$ where the number of non-zero elements is set to 2, i.e. the number of basis elements in the dictionary used to represent a range. Based on the optimal activation, the reconstruction error is $r_{l,g} = ||D_g \widehat{\alpha}_{l,g} - \xi_{l,g}||$ and $r^*_{l,g} = [r_{l,g}]$ the corresponding vector for all possible sets of dictionary elements. The unary terms lead to

$$U\left(\xi_{l,g}, y_{l,g}\right) = \frac{1}{\sigma_*} r^*_{l,g} + \frac{1}{\sigma_{**}} abs\left(1 - \sum_v \widehat{\alpha}_{v,l,g}\right) \qquad \text{Eq. 8.5}$$

where both terms are normalised with their standard deviation: σ_* is the standard deviation of subsequent variable $r^*_{l,g}$ and σ_{**} the standard deviation of, correspondingly, $abs\left(1 - \sum_v \widehat{\alpha}_{v,l,g}\right)$. As mentioned above, this part of the energy functional describes how well data and a specific sparse representation model agree with each other. It has to be prevented that dictionary elements are chosen more than once.

Selecting sub-waveforms based on the CRF-solution

We receive for each waveform a vector of 2 combinations from the dictionary element containing the selected elements from the dictionary, which are found to best represent the individual range gates. Neighbouring range gates represented by the same dictionary elements are defined as sub-waveform. Sub-waveforms that are too small due to less than four observations are removed, since the following parameter estimation requires at least three observations in order to fit three parameters.

8.2.2. SINCS retracking

Each sub-waveform is retracked by fitting the SINCS model to each sub-waveform. It is assumed that the noise of the sub-waveform follows a normal distribution and the following objective function is minimized:

$$X \in R^n \rightarrow min_X \sum_{k=1}^{N_S} (y_k - s_k)^2 \qquad \text{Eq. 8.6}$$

[12] https://github.com/nsubtil/gco-v3.0/

with the number of free parameters n and the number of considered gates N_s of the sub-waveform. The measured sub-waveform is denoted by y_k and the modelled sub-waveform by s_k. The parameters epoch, amplitude and SWH are estimated, so $n = 3$. In order to solve [Eq. 8.5] efficiently, the design matrix is defined as

$$J_{k,i} = -\frac{\partial s_k}{\partial X_i}. \qquad \text{Eq. 8.7}$$

Therefore, the flat sea surface response has to be estimated in the frequency/slow-time (f/η) domain (see Eq. 29 in Buchhaupt et al., 2018) and is evaluated at discrete samples depending, amongst others, on the number of pulses per burst and at discrete frequencies f_n. In this domain, both PTR functions are multiplied to the flat sea surface response (see Eqs. 51/52 in Buchhaupt et al. 2018). In the next step, this product needs to be transformed back into the range time/Doppler frequency (f/x_D) domain by using a discrete Fourier transform in order to cope with the not equidistant samples of the Doppler frequencies of the measured beams. Then, the range cell migration is applied by shifting the leading edges of all beams to the same range bin (see Eqs. 54/55 in Buchhaupt et al., 2018). We obtain a matrix containing the product of the flat sea surface response and the two point target response functions, $\widehat{\bar{q}}_{n,\bar{l}}$, with $\bar{l}$ the number of waveforms within a stack.

For each Doppler beam iteration, the PDF of the sea surface displacement in the frequency domain at discrete frequencies is computed ($\widehat{pdf}_n$, see Eq. 56 in Buchhaupt et al., 2018) as well as the stack mask ($M_{k,\bar{l}}$, see Eqs. 57-59 in Buchhaupt et al., 2018). The latter is one/zero and concentrates on the power in the receiving window, while the rest of the waveform is masked.

Thermal noise is estimated by averaging over the first 10 bins which are the same in CA: for oversampled waveform signals, which contain 256 range gates instead of 128, samples considered for waveform fitting are from 25 to 232 in order to avoid the wraparound area. Thermal noise is thus estimated via $\widehat{T}_N = \frac{1}{10}\sum_{k=25}^{34} y_k$. In order to remove the thermal noise signal, we use the stack mask as basis in $m_k = \sum_{\bar{l}=1}^{\bar{L}} M_{k,\bar{l}}$ and $m_{ave} = \sum_{k=25}^{34} m_k$ and subtract $\frac{m_k}{m_{ave}}\widehat{T}_N$ from the measured waveform. These limits as well as information about stack mask and thermal noise are adjusted to the objective function [Eq. 8.6]. For the adjusted design matrix [Eq. 8.7] we obtain:

$$\bar{J}_{k,i} = J_{k,i} - \frac{1}{10}\frac{m_k}{m_{ave}}\sum_{k=25}^{35} J_{k,i} \qquad \text{Eq. 8.8}$$

with the corresponding elements described in Table 8.1.

Table 8.1: Columns of the Jacobian matrix.

derivative wrt. A	derivative wrt. Δt	derivative wrt. σ_s
$\widehat{s}_{n,\bar{l}} = -\widehat{pdf}_n \cdot \widehat{\bar{q}}_{n,\bar{l}}$	$\widehat{s}_{n,\bar{l}} = 2K\pi i f_n \cdot \widehat{pdf}_n \cdot \widehat{\bar{q}}_{n,\bar{l}}$	$\widehat{s}_{n,\bar{l}} = 4K\lvert\sigma_s\rvert\pi^2 f_n^2 \cdot \widehat{pdf}_n \cdot \widehat{\bar{q}}_{n,\bar{l}}$

$J_{k,1} = \sum_{l=1}^{\bar{L}} s_{k,l} \cdot M_{k,l}$	$J_{k,2} = \sum_{l=1}^{\bar{L}} s_{k,l} \cdot M_{k,l}$	$J_{k,3} = \sum_{l=1}^{\bar{L}} s_{k,l} \cdot M_{k,l}$

In Table 8.1, $\widehat{s}_{n,l}$ is the input for the inverse discrete fast Fourier transform with output $s_{k,l}$. The variable K depends, amongst others, on the antenna gain, two-way propagation loss, pitch angle etc.

If the retracking of a sub-waveform does not converge, the sub-waveform is removed from further processing. The collection of estimates from each sub-waveform creates a point-cloud of epoch, leading edge and amplitude, which are further processed to estimate one set for each waveform.

8.2.3. Selecting final estimates for each 20 Hz position

Pre-processing of the point-cloud

Epoch, leading edge width and amplitude from the previous retracking step are converted to range, wave height and backscatter coefficient. The range is then converted to corrected sea level height above the reference ellipsoid (SSH) by applying all the environmental and geophysical corrections. Finally, the sea level anomaly (SLA) is derived by accounting for the mean sea surface (MSS). This is beneficial as (1) very large outliers within the SLA point-cloud are easily removed for values exceeding ± 10 m and (2) strong bathymetry signals, which are visible in SSH, are removed.

Prior information from retracking the total waveform is used and temporally filtered to avoid outliers and short-term variability. We remove this prior information from SLA, $\Delta SLA = SLA - SLA_{prior}$, and correspondingly for ΔSWH and $\Delta\sigma^0$. This is beneficial as it allows to compute anomalies for SWH and $\Delta\sigma^0$ and also perform a rough outlier detection.

Note that prior retracking information for SWH and $\Delta\sigma^0$ are used over the open ocean and coastal areas whereas prior retracking information for SLA are used only over the ocean with a 10 km or more distance to coast. The retracking prior information has been introduced in order to counter very strong deviations of several meters from the MSS on small spatial scales, which generally only occur during strong storm events or in case of significantly wrong MSS. During “normal” conditions, this prior information will not affect the final results of STARS.

Applying DBSCAN and RANSAC for line detection with the point-cloud

Due to the use of anomalies, the desired points for the final selection tend to cluster around a straight line along the track. This feature can be exploited as prior information in order to limit the search space for the final point selection. For the first step of the line detection, all potential points along-track at each 20 Hz measuring position are clustered in order to reduce the number of points and thus the noise for the following RANSAC algorithm. We use all potential points of ΔSLA, ΔSWH, $\Delta\sigma^0$ along-track given a neighbouring length $\varepsilon_N = 0.05$ and a minimum number of points building a cluster ($N_c = 5$). With this slightly modified version of DBSCAN, we obtain cluster indices

for each point at each 20 Hz measuring position and the corresponding number of points within each cluster. The next step is to calculate the median for each cluster at each measuring position. With this computation, small noise clusters are eliminated. This is beneficial for the application of the RANSAC algorithm in order to fit a line through the cluster points. Two points along track are randomly selected. Between these points, a current line model is estimated. Given a distance threshold, we find those cluster median points in agreement with the line model. We receive a final line model via a least square estimate using the set of points which showed largest agreement with a randomly generated line. The RANSAC algorithm is applied for a moving window of 9 seconds width, where the RANSAC line output is kept for the central second within the window. The window is then shifted by one second.

Selecting final estimates

From the last step, we receive a RANSAC line for SLA, SWH and σ^0. Selecting final estimates from the point-cloud is based on the shortest-path approach. This prefers similar estimates at neighbouring measurement positions if possible, by implicitly assuming that the surface conditions will not change drastically over 300 m along the track, thus, avoiding large jumps in estimates if possible. Large jumps (e.g. > 0.5 m SLA) over 300 m are not desired and thus marked as outliers and replaced by NaN in the final output file.

At each position, the differences between all points in the point-cloud and the RANSAC line are calculated. Differences above a pre-defined threshold (e.g. > 0.5 m) are removed in order to reduce the noise and avoid single large jumps. This threshold should be carefully selected as it tunes the noise level and thus affects the final results. In the next step, the differences (for SLA, SWH, σ^0) of the remaining points between two successive positions are determined and weighted with the cluster size. By adding these two differences, we receive edge weights (in terms of the shortest-path algorithm). At each position, the final solution is the point that has the lowest edge weight value. This procedure is repeated for each 20 Hz position along the track. We obtain an index list with which the final solutions Δt, σ_s and A are selected from the point-cloud. This results in final estimates for each 20 Hz measurement position that can be further processed depending on the desired output.

Note that the final estimates are still based solely on the original retracking output from individual sub-waveforms and no filtering is applied.

8.3. Development Choices and Trade Offs

STARS V1.0 will be further developed regarding its performance and computational speed.

8.4. Data Flow

Input data

For SINCS, the following input is needed:

roll; pitch; yaw; Doppler-frequencies (to calculate Doppler beams); along-track velocity; slant-ranges used to vertically align the Doppler beams (is necessary to calculate the stack mask) including slant ranges from bursts to surface, Doppler-range shift for each Doppler-beam, tracker range differences, fine range adjustments; satellite height; altitude height; tracker gate; tracker range; bandwidth [GHz]; number of pulses per burst; 20 Hz waveform; RIP; speed of light; central frequency [GHz]; pulse repetition frequency; along-track beam width; across-track beam width; Ku-frequency; pulse duration; range gate spacing (standard 3.125 ns);

For STAR, the following additional input is needed:

longitude; latitude; time; environment and geophysical corrections (dry troposphere, wet troposphere, ionosphere, solid earth tide, pole tide, ocean tide, loading tide, inverse barometric); MSS; distance to coast;

Output data

The output of the STARS retracker is uncorrected range, significant wave height and backscatter coefficient including the corresponding spatial (longitude/latitude) and temporal information. The output is saved in NetCDF files.

8.5. References

Boykov et al. (2001): Efficient Approximate Energy Minimization via Graph Cuts. In *IEEE TPAMI*, 20(12), 1222-1239.

Boykov & Kolomogorov (2004): An Experimental Comparison of Min-Cut/Max-Flow Algorithms for Energy Minimization in Vision. In *IEEE TPAMI*, 26(9), 1124-1137.

Buchhaupt et al. (2018): A fast convolutional based waveform model for conventional and unfocused SAR altimetry. *ASR*, 62(6), 1445-1463, https://doi.org/10.1016/j.asr.2017.11.039.

Ester et al. (1996): A Density-Based Algorithm for Discovering Clusters a Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining,* KDD'96. Portland, Oregon: AAAI Press, pp. 226–231.

Fenoglio et al. (2019): Calibrating CryoSat-2 and Sentinel-3A sea surface heights along the German coast, In *International Association of Geodesy Symposia*. Springer, Berlin, Heidelberg, doi.org/10.1007/1345_2019_73.

Fischler & Bolles (1981): Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. In *Commun. ACM,* 381-395, http://doi.org/10.1145/358669.358692.

Halimi et al. (2016): Bayesian Estimation of Smooth Altimetric Parameters: Application to Conventional and Delay/Doppler Altimetry. In *IEEE Trans. Geosci. Remote Sens.*, 54(4), 2207-2219, https://doi.org/10.1109/TGRS.2015.2497583.

Kolmogorov & Zabih (2004). What Energy Functions can be Minimized via Graph Cuts? In *IEEE TPAMI*, 26(2), 147-159.

Lafferty et al. (2001). Conditional Random Fields: Probabilistic Models for Segmenting and Labelling Sequence Data. In *Proceedings of the Eigtheenth International Conference on Machine Learning*, San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., pp 282-289, ISBN 978-1-55860-778-1

Ray et al. (2015):. SAR altimeter backscattered waveform model. In *IEEE Trans. GeoSci. Remote Sens.*, 53 (2), 911-919. https://doi.org/10.1109/TGRS.2014.2330423.

Roscher et al. (2017): STAR: Spatio-temporal altimeter waveform retracking using sparse representation and conditional random fields. *RSE*, 201, 148-164, https://doi.org/10.1016/j.rse.2017.07.024.

Wingham et al. (1986): New Techniques in Satellite Altimeter Tracking Systems. In *Digest - IGARSS*, 1339-1344.

LIST OF SYMBOLS

Symbol	Definition
Δt_0	epoch
σ^0	backscatter coefficient
σ_s	leading edge width
A	amplitude

9. Adaptation of ALES+ for SAR (TUM)

9.1. Theoretical Description, physics of the problem

ALES+ SAR is based on an empirical application of the Brown-Hayne functional form that models the radar returns from the ocean to the satellite. The Brown-Hayne theoretical ocean model [Brown (1977), Hayne (1980)] is the standard model for the open ocean retrackers in Low Resolution Mode altimetry and describes the average return power of a rough scattering surface (i.e. what we simply call waveform). The return power V_m is modelled as follows (equations reported in Passaro et al., 2014):

$$V_m(t) = a_\xi P_u \frac{|1+erf(u)|}{2} \exp(-v) + T_n \qquad \text{Eq. 9.1}$$

where

$$a_\xi = \exp\left(\frac{-4\sin^2\xi}{\gamma}\right) \qquad \text{Eq 9.1b}$$

$$\gamma = \sin^2(\theta_0)\ \frac{1}{2\ln(2)} \qquad \text{Eq. 9.1c}$$

$$u = \frac{t-\tau-c_\xi\sigma_c^2}{\sqrt{2}\sigma_c} \qquad \text{Eq. 9.1d}$$

$$v = c_\xi\left(t-\tau-\frac{c_\xi\sigma_c^2}{2}\right) \qquad \text{Eq. 9.1e}$$

$$\sigma_s = \frac{SWH}{2c} \qquad \text{Eq. 9.1f}$$

$$\sigma_c^2 = \sigma_p^2 + \sigma_s^2 \qquad \text{Eq. 9.1g}$$

$$c_\xi = b_\xi a \qquad \text{Eq 9.1h}$$

$$a = 4c\left[\gamma h\left(1+\frac{h}{R_e}\right)\right]^{-1} \qquad \text{Eq. 9.1i}$$

$$b_\xi = \cos 2\xi - \frac{\sin^2(2\xi)}{\gamma} \qquad \text{Eq. 9.1j}$$

where c is the speed of light, c_ξ the satellite altitude, R_e the Earth radius, ξ the off-nadir mispointing angle, θ_0 the antenna beam width, τ the Epoch with respect to the nominal tracking

reference point, σ_c the rise time of the leading edge (depending on a term σ_s linked to SWH and on the width of the radar point target response σ_p), P_u the amplitude of the signal and T_n the thermal noise level.
In practice, the model in equation 9.1 is a raised sigmoid $\frac{|1+erf(u)|}{2}$ describing the increasing power in the waveform leading edge and the subsequent plateau, multiplied by a negative exponential which models the reduction of power in the waveform tail (decay), plus thermal (additive) noise T_n. The amplitude of the signal P_u is attenuated by a term a_ξ dependent on mispointing. P_u can be converted into a measurement of the backscatter coefficient σ_0 on the basis of the instrument calibration.

In the case of the DD waveforms, ALES+ adopts a simplified version of the Brown-Hayne functional form as an empirical retracker to track the leading edge of the waveform. While the rising time of the leading edge still has a strict relationship to the significant wave height, the equation 9.1f does not hold anymore. Moreover, since as explained subsequently a fixed decay of the trailing edge is chosen, the equations 9.1g-j are not considered. This empirical application of the Brown-Hayne model implies that ALES+ cannot estimate a physical value of SWH and of σ_0. Nevertheless, the retracker is fully able to track the mid-point of the leading edge. To summarise, the simplified version of the Brown-Hayne functional form used to retrack DD waveforms is:

$$V_m(t) = P_u \frac{|1+erf(u)|}{2} \exp(-v) + T_n \qquad \text{Eq. 9.2}$$

where

$$u = \frac{t-\tau-c_\xi \sigma_c^2}{\sqrt{2}\sigma_c} \qquad \text{Eq. 9.2a}$$

$$v = c_\xi \left(t-\tau-\frac{c_\xi \sigma_c^2}{2}\right) \qquad \text{Eq. 9.2b}$$

9.2. Algorithm Definition: Processing Steps and Mathematical Description

Leading edge detection

Since ALES+ SAR is based on the selection of a subwaveform, it is essential that the leading edge, containing the information on the range between satellite and reflecting surface, is correctly detected in all cases. Lead waveforms and ocean/coastal waveforms are characterised in this respect in two different ways: in the first case, the lead return (if at nadir) clearly dominates any other return, but the decay of the trailing edge is extremely quick; in the latter, the leading edge is better characterised, but spurious peaky returns can precede (if from icebergs, ships, or targets at a

higher height than the water level) or follow (if from areas of the footprint characterised by different backscatter characteristics) the main leading edge, whose trailing edge decreases very slowly.

For the reason above, in ALES+ SAR the leading edge detection for peaky waveforms is different than for oceanic waveforms. To distinguish between the two cases, a Pulse Peakiness (PP) index is computed following the formula in Peacock and Laxon (2004). Waveforms with PP<1 are sent to the oceanic leading edge detection (OLED) procedure, the others are sent to the peaky leading edge detection procedure (PLED). This is not a physical classification aimed at detecting leads, but only a way to aid the correct detection of the leading edge; moreover, the retracking remains the same in both cases. Peaky waveforms are in our case not only the leads, but any waveform whose trailing edge decay is more pronounced than in the standard ocean return. The aim is therefore different from Peacock and Laxon (2004), in which a strict classification is needed in order to send each kind of waveform to a different retracker and to avoid the detection of false leads, which would determine inconsistencies in the sea level retrieval.

For DD waveforms, the OLED threshold is defined at PP<3. Once this is done, the leading edge is found in a similar way as to LRM, The steps followed by PLED are the following:

1. The waveform is normalised with normalisation factor N, where N = 1.3 * median(waveform)
2. The leading edge starts when the normalised waveform has a rise of 0.01 units compared to the previous gate (startgate)
3. At this point, the leading edge is considered valid if, for at least four gates after startgate, it does not decrease below 0.2 units (20% of the normalised power).
4. The end of the leading edge (stopgate) is fixed at the first gate in which the derivative changes sign (i.e. the signal start decreasing and the trailing edge begins), if the change of sign is kept for the following 3 gates

The steps followed by OLED are the following:

1. The waveform is normalised with normalisation factor N, where N = max(waveform)
2. The stopgate is the maximum value of the normalised waveform
3. Going backwards from stopgate, the startgate is the first gate in which the derivative is lower than 0.01 units

Choice of trailing edge slope

The choice of the parameters defining the trailing edge slope depends on the PP of the waveforms. The following cases are found:

1. DD altimeter and standard ocean waveform: here the slope of the trailing edge cannot be physically defined by the full Brown-Hayne functional form. Nevertheless, the trailing edge decay does not influence the fit of the leading edge for a subwaveform retracker such as ALES+, as long as a predefined realistic value is used. In this development phase of ALES+ SAR, the used value is c_ξ =0.04. This value is purely empirical.
2. DD altimeter and non-standard ocean waveform: The non-standard ocean waveforms undergo a further preliminary step: c_ξ is estimated externally. In the external estimation, the

full waveform is fitted using the simplified Brown-Hayne functional form, having 4 unknowns: τ, σ_c, P_u, c_ξ. From this result, only c_ξ is kept and used as an input in the remaining steps of the ALES+ algorithm.

Subwaveform retracking

The ALES+ SAR concept aims at fitting waveforms whose trailing edge is perturbed by areas of the footprint with different backscatter conditions, such as patches of calm waters, land or ice, while guaranteeing a comparable accuracy in typical open ocean conditions. Defining startgate and stopgate the first and last gate of the subwaveform of choice, in effect the issue is one of defining an appropriate stopgate.

This retracking step therefore consists on a single pass on a subwaveform defined as:

Cryosat-2, Sentinel3a, Sentinel3b: Stopgate = $\text{Stopgate}_{\text{LE}}$ + 20

where $\text{Stopgate}_{\text{LE}}$ is the last gate of the leading edge.

Ses State bias correction

Firstly, it is important to underline that no retracking algorithm can be defined as complete, if the appropriate Sea State Bias correction is not provided. In the original products of DD altimetry, the Sea State Bias correction is either missing (Cryosat-2) or computed using the Jason model. In this study instead, a first model is computed specifically for the ALES+ SAR retracker. As a reference parameter on which the model is built, we take the rising time of the leading edge, which can be used as a proxy for the significant wave height, as shown in figure 9.1.

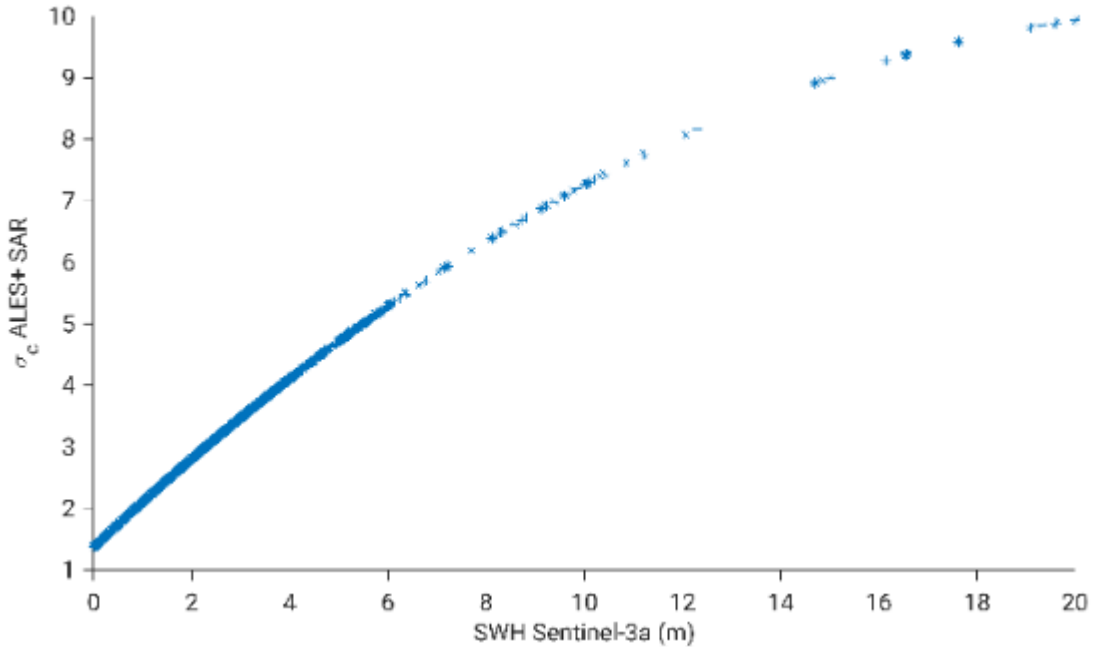


Figure 9.1: polynomial interpolation of the rising time of the leading edge estimated by ALES+ SAR and the corresponding significant wave height estimated by SAMOSA2 in the original Sentinel-3A product.

We derive the corrections by observing the sea level residuals (with no correction applied) at the crossover points. We use a region covering the North Sea and the Mediterranean Sea in order to have different different sea state characteristics. The residuals are modelled w.r.t. the variables influencing the sea state (here the rising time of the leading edge) in a parametric formulation.

$$SSB = \alpha + \sigma_c$$

The equations needed to compute the Sea State Bias model are built using the high-frequency sea level anomalies at each crossover m:

$$\Delta SLA = \alpha\sigma_{c_o} - \alpha\sigma_{c_e} + \varepsilon$$

where o and e stand for odd and even tracks (indicating ascending and descending tracks respectively), ε accounts for residual errors that do not depend on the Sea State Bias correction.

We have therefore a set of m linear equations, which will be solved in a least square sense. The chosen α is the one that maximises the variance explained at the crossovers, i.e. the difference between the variance of the crossover difference before and after correcting the sea level anomaly for the sea state bias using the computed model.

In the table below, the variance at the crossover before and after the application of the sea state bias correction is reported, together with the values reported by Gaspar et al., 1994, who estimated the coefficients of Fu-Glazman model (a representation that depends on significant wave height and wind) on a global scale. We also report the results of a high-rate sea state bias correction derived for the standard product of Jason-1 mission in the North Sea by Passaro et al., 2018b. The variance explained by the sea state bias correction in ALES+ SAR is at the same level of the one explained by the high-rate sea state bias correction of Jason-1 and more than the one explained by Gaspar et al., 1994. This is expected, since Passaro et al., 2018b demonstrated that the application of the SSB at high-rate is one way to reduce the intra-1Hz correlation between the retracked parameters. Notably, the crossover variance from ALES+ SAR is lower than in Jason-1, which signals the higher precision of SAR altimetry and of the ALES+ SAR retracking.

Dataset	XO var before SSB (cm2)	XO var after SSB (cm2)	Variance explained
Gaspar et al. (1994)	127.7	120.4	6%
SGDR Jason-1 Mediterranean Sea	135.6	108.4	20%
ALES+ SAR Sentinel-3A	106.0	84.9	20%

9.3. Development Choices and Trade Offs

In the current form of the algorithm for this project, the subwaveform is indeed not adaptive, but fixed. In fact, the use of the Montecarlo simulation as in LRM case (see Passaro et al., 2018) is not possible for the empirical application of ALES+ on DD waveforms, since the Brown-Hayne model, even with an adapted c_ξ, cannot be considered as a DD simulator.

Thibaut et al. (2014) showed that also in SAR altimetry a reduced retracking window can be used without significant decrease of the performances. The optimization of the subwaveform to different levels of σ_c can be an interesting field of improvement if the validation finds that the current strategy guarantees a level of performance similar to the current baseline.

9.4. Data Flow

ALES+ SAR is written in Python 2.7 and saved in a GIT project. No external data are needed except for the original L1B data, or an equivalent version of the products containing the following parameters: latitude, longitude, time, on-board tracker output, multi-looked waveform. For every waveforms, the algorithm provides as output the range (derived from the epoch), the rising time of the leading edge (in seconds) and the amplitude of the received signal.

9.5. References

Brown, G, The average impulse response of a rough surface and its applications, IEEE Transactions on Antennas and Propagation, 25, 67-74, 1977

Gaspar, P., Ogor, F., Le Traon, P.Y. and Zanife, O.Z.: Estimating the sea state bias of the TOPEX and POSEIDON altimeters from crossover differences. Journal of Geophysical Research: Oceans, 99(C12), pp.24981-24994, 1994.

Hayne, G. Radar altimeter mean return waveforms from near-normal-incidence ocean surface scattering, IEEE Transactions on Antennas and Propagation, 28, 687-692, 1980

Passaro M., Cipollini P., Vignudelli S., Quartly G., Snaith H.: ALES: A multi-mission subwaveform retracker for coastal and open ocean altimetry. Remote Sensing of Environment 145, 173-189, 10.1016/j.rse.2014.02.008, 2014

Passaro M., Rose S.K., Andersen O.B., Boergens E., Calafat F.M., Dettmering D., Benveniste J.: ALES+: Adapting a homogenous ocean retracker for satellite altimetry to sea ice leads, coastal and inland waters. Remote Sensing of Environment, 2018

Passaro M., Zulfikar Adlan N., Quartly G.D.: Improving the precision of sea level data from satellite altimetry with high-frequency and regional sea state bias corrections. Remote Sensing of Environment, 245-254, 10.1016/j.rse.2018.09.007, 2018b

Peacock, N. R., and Laxon, S. W., Sea surface height determination in the Arctic Ocean from ERS altimetry, J. Geophys. Res., 109, C07001, doi:10.1029/2001JC001026., 2004

Thibaut P., Aublanc J., Moreau T., Boy F., Picot N.: Delay/Doppler waveform processing in the coastal zone. Presented at the 8th Coastal Altimetry Workshop, Lake Constance, Germany, 23-24 October 2014

10. Specialised Coastal SAR (NOC)

10.1. Theoretical Description, physics of the problem

To be finished.

10.2. Algorithm Definition: Processing Steps and Mathematical Description

To be finished.

10.3. Development Choices and Trade Offs

To be finished.

10.4. Data Flow

To be finished.

10.5. References

To be finished.

11. L2 official products variable ingestion (isardSAT)

Some variables from the L2 official ESA products from both S3 and CS2 are of interest for later stages of the processing and therefore need to be incorporated to the data chain. The only processing required is an interpolation from their original time grid to the global processing time grid defined in the L1B processing stage (Section 3).

11.1. Theoretical Description, physics of the problem

The interpolation required applied follows the criteria of “closest-value”, and since the sampling rates of both original and final time vectors are the same, no further issues are envisaged at this point.

11.2. Algorithm Definition: Processing Steps and Mathematical Description

The processing sequence of this block is as follows:

1. Download from ESA repositories the whole set of L2 variables of interest as defined in the HYDROCOASTAL IODD [RD-06], Table 3.3 (S3) and Table 3.4 (CS2).
2. Interpolate the variables to the L2 Master product time vector following the criteria of “closest value”. Notice that some input variables are initially sampled at 1Hz while other ones at 20Hz, so different original time vectors need to be considered.

Some particularities apply to the following variables:

- *retracked_Pu_ESA*:
  - For S3, this variable is defined as the product between *amplitude_ocean_20_ku* and *scale_factor_20_ku*.
  - Left empty for CS2.
- *retracked_epoch_ESA*:
  - Left empty for CS2.

11.3. Development Choices and Trade Offs

No development choices are considered for this algorithm.

11.4. Data Flow

Input data:

- ESA official L2 products for either S3 or CS2 (see RD-01).
- Time vector of the waveforms as obtained in the L1B processing stage.

Output data:

- Vectors of interpolated variables.

11.5. References

No specific references are considered in this Section.

12. L3 River Level (AHL)

The following algorithm allows the generation of L3 data of river water level, from L2 retracked data, in the form of[13]:

- **time series** of river water levels for **repeat orbit satellites**, overflying regularly the same locations over inland waters and eligible to the concept of Virtual Stations (VS).
- **space-time series** of river water levels for **non-repeat orbit satellites**, not overflying regularly the same locations over inland waters, but regularly the same waterbody/region in different locations, not eligible to the concept of Virtual Stations (VS).

All L3 data contain one and only one Water Level (WL) measurement per overflight of the satellite over the waterbody (cf. OPO Reduction).

Short Acronyms list & Glossary:

- **Measurement**: Surface height measurement over land or water, belongs to a data record.
- **OPO**: “One Per Overflight”.
- **Overflight**: Uninterrupted sequence of along-track altimetry records, all falling into any WM polygon.
- **Record, data record**: Data record that holds a measurement value along with all of the necessary variables (time, alt, lon, lat, range values, flags, etc.).
- **Time (variable)**: Is always supposed to be “high rate” time, the higher frequency along-track time of the L2 data holding measurement records (e.g., 20 Hz for L2 data from CryoSat-2 and the Sentinel-3 satellites).
- **VS**: Virtual Station, an idealised abstraction representing the crossing of one or two satellite track(s) over a waterbody, this concept assumes satellite orbit stability and a certain degree of temporal revisit (in practice not more than 35 days repeat period).
- **WM**: Water Mask, a collection of geographical polygons representing the boundaries of te water surface of waterbodies at a given date (or typical hydrological season).

12.1. Theoretical Description, physics of the problem

The algorithm proposes to **exploit along-track L2 altimetry data** acquired over land in order to produce L3 **water level data**, obviously limited to free surface waters. Particularly, it exploits subsets of L2 data falling into a water mask (WM). The algorithm is mainly about assembling well known routines into a **new data processing flow scalable for global data processing** and without any use of regional/specific datasets (such as river path, river profile, etc.). The quality of the WM is

[13] *Note that “time series” (variations from a fixed location through time) are a specific case of the more general case embodied by the “space-time series” (variations from various locations through time). This is true in the field of Alti-Hydrology since data coming from any satellite orbit can be processed as data coming from non-repeat orbit. Moreover, since the era of CryoSat-2 flying on a geodetic orbit, the instability of SARAL/AltiKa orbit and the pre-decommissioning orbit phases of the Jason satellites, the use of non-repeat orbit data tends to become mainstream and algorithms have to adapt to this in their design.*

an important driver of the quality of the final products, as well as the quality of various retracker outputs in L2 data.

The processor is able to process separately each retracker output in order to provide WL time or space-time series for each of them, sharing basically the same space/time sampling in the L3 products.

The main challenging steps of the algorithm are:

1. Recurrent issues w.r.t. L2 data product files: presence of duplicate and/or interleaved records, null time/lon/lat records. It is also necessary to account for missing records and to remove any extraneous or empty L2 files. Such errors are recurrent, even in official datasets from Agencies.
2. Automated detection of satellite track / waterbody crossings, there are several possible caveats to avoid when performing water masking and subsequent tasks:
    1. Unwanted over split of L2 along-track data in too many L2 subsets: this may arise when either altimetry data are missing within the WM or if the WM is incorrect also.
    2. Unwanted merge of L2 subsets from unrelated waterbodies: this may arise in case of very close waterbodies (as represented by the WM) triggering relative distance under thresholds.
    3. Since WM is driven by L2 records location on the ground, very thin waterbodies located in between two consecutive records (in-memory, and not necessarily in due time) will result in ignored waterbody in the subsequent following steps of the L3 Processor. This concerns only waterbodies narrower than ~300 m in the along-track direction.
3. Deal with different kinds of L2 space-time data organisation:
    1. For a repeat orbit satellite:
        1. Detect Virtual Stations automatically (this requires stable repeat orbit through cycles).
        2. Perform outlier rejection at the scale of VS (temporal filters).
    2. For a non-repeat orbit satellite or satellites with unstable orbit, for altimetric modes:
        1. SARM: Perform outlier rejection filters at the scale of the overflight (flat filter).
        2. SARINM: Estimate local river slope and perform outlier rejection filters at the scale of the overflight (slope filter).
4. Design a coherent, scalable, memory- & CPU-efficient algorithm able to process data at global scale. This basically implies to reduce as much as possible the use of computing memory as more L2 and WM files are involved.
5. Define a user-friendly L3 product file format compatible for both repeat and non-repeat orbit data:
    1. Essential L3 variables: `water_level` for (each retracker), `{time, lon, lat}`, `geoid`
    2. Add useful “add-on variables” such as record counts, basic statistical indicators and L2 quality flags
    3. Friendly data and files organisation, clear files naming convention.

12.2. Algorithm Definition: Processing Steps and Mathematical Description

The main processing steps of the algorithm are:

1. Read L2 product files & data normalisation
2. Water Masking of L2 products
3. Time series or space-time series
    1. Repeat orbit / SARM case => merge by Virtual Station => time series
    2. Non-repeat orbit / SARM or SARINM cases => directly use as space-time series
4. Outliers rejection
5. OPO Reduction
6. Tiles Grouping & Write data to L3 product files

Below are presented the mathematical descriptions of the main processing steps of the algorithm.

12.2.1. Read L2 product files & data normalisation

The tool used to read the L2 product files is a flexible code able to read L2 data from any netCDF format providing a simple descriptive configuration file (aka "*pdesc*", for "product descriptor").
A *pdesc* file describes all of informations (attributes) necessary to read (file patterns, netCDF dimensions, etc.) & process the data (mission_id, orbit parameters, altimeter mode, etc.) as well as the list of required netCDF variables to be read (e.g., time, alt, retracked_range, etc.). The tool is able to interpolate data on the fly, e.g., in the typical interpolation case of "1 Hz to 20 Hz" of geophysical L2 corrections.

After reading, the normalisation of the L2 data guarantees that all spatio-temporal and geophysical variables share the same standardised units & reference systems. For example, **`{alt,lon,lat}`** variables are always translated from any product-dependent ellipsoid system to WGS84. The **`longitude`** variable is converted to signed longitude **`[-180.0° ; 180.0°]`**. The **`time`** variable is internally converted to a common numerical system with a resolution of one microsecond or better.

Inputs:

L2 product files
L3 processor configuration file

Outputs:

In-memory read L2 data
New variables:

- **`{time, alt, lon, lat}`** as described above.

Synopsis:

1. For each L2 product file, read it
2. Create in-memory new variables from L2, the result are in-memory L2 data

12.2.2. Water masking of L2 products

The Water Masking step permits to isolate altimetric records of water surface height (acquired within the WM) from those acquired over land. WM data are sets of geographical polygons expressed in an ellipsoidal coordinate system (longitude, latitude). This implementation uses one of the most elegant and fast "point in polygon" algorithm known to date [Franklin1994]. It has been ported from C to the numpy library (Python) and among its important characteristics it "will locate each point into exactly one polygon" ("considers each polygon to be topologically a semi-open set"). The algorithm also supports WM polygons that might be *"a lake on an island within a river bed in land surrounded by oceans"* and so on.

Below is the C-code for the elementary point-in-polygon *"pnpoly"* algorithm (one polygon versus one point):

```
int pnpoly(int nvert, double *vertx, double *verty, double px, double py) {
  int i, j, c=0;
  for (i=0, j=nvert-1; i < nvert; j=i++) {
    if ( ((verty[i]>py) != (verty[j]>py)) &&
        (px < (vertx[j]-vertx[i])*(py-verty[i])/(verty[j]-verty[i]) + vertx[i]) )
       c=!c;
  }
  return c;
}
```

The Python-numpy version in use in this algorithm is a generalised implementation able to cope directly with **N** polygons and **M** points to be tested against.

Inputs:

In-memory read L2 data
Water Mask tile files
L3 processor configuration file

Outputs:

L2WM in-memory data:

- Variables inherited from L2: **time**, **lon**, **lat, geoid_height**
- New Variables:
  - boolean **water_mask_is_inside(time)** : True if the record falls into any WM polygon, False otherwise.
  - unsigned integer **water_mask_polygon_tile_uid(time)** : Unique identifier of the intersecting WM polygon. This identifier is guaranteed to be unique among all of the WM polygons.
  - unsigned integer **overflight_uid(time)** : Unique Identifier of the overflight to which a record belongs (**[1...N]** for all of the L2 data currently processed). This identifier is guaranteed to be unique among all of the L2/WM overflights to be processed.

Synopsis:

1. For each L2 product file, read intersecting WM tiles
2. Apply point-in-polygon algorithm with all polygons from all intersecting WM tiles

3. Create in-memory new variables, the result are in-memory L2WM data

12.2.3. Time series or space-time series

L2 altimetry data are naturally organised in pass (usually one product file contains an entire pole-to-pole pass occurrence of a track).
The aim of the merge of the L2WM data is to group data in time series or space-time series.
This is necessary to the subsequent processing steps that rely on the assumption of space and/or time neighbourhood of the altimetric measurements. Typically, this is a prerequisite to process data at the scale of a VS (repeat orbit) or on a small area (non-repeat orbit). This step produces new variables that store Unique Identifiers assigning records to groups of neighbour altimetric measurements in terms of space and/or time.

Inputs:
1. L2WM data (in-memory)
2. L3 processor configuration file

Outputs:
1. New Variables (added to L2WM data):
   - double **`water_level_<mission>_<band>_<mode>_<retracker>`**: Water Level w.r.t. Geoid height.
   - Repeat orbit satellites:
     unsigned integer **`vs_uid(time)`** : record's Unique Identifiers of overflights group.
   - Non-repeat orbit satellites:
     None (data will be processed considering its natural space-time sampling).

Synopsis:
1. Repeat orbit satellites:
   1. For each WM tile, get L2WM data
   2. Perform geographical linkage for L2WM records close to each other in space
   3. Set common **`vs_uid`** value to all overflight records belonging to the same overflights group
2. Non-repeat orbit satellites: *n.a.*

12.2.4. Outliers rejection

Rejecting outliers always constitute, to a certain degree, a challenging step. Actually, the quality of the data after outliers rejection is largely conditioned by the quality of the L2 products and in particular of the L2 retracker outputs.

As foreseen earlier, different approaches are implemented for the three different data cases handled by the project:
1. Repeat orbit in SARM,
2. Non-repeat orbit in SARM,
3. Non-repeat orbit in SARINM.

The non-repeat orbit cases (CryoSat-2) will induce space-time data which deserve specific routines in order to discard outliers. In contrast with past projects on CryoSat-2 & inland water (such as [SHAPE project]), and because of the global scale constraint, it is not possible to manipulate data in space to transform them to time series ("Migration" in SHAPE's project terminology [Bercher, 2012b, 2016b]). Alternative but simpler routines have been developed specifically for cases 2. and 3. Both of them are rejection filters applied at the scale of the overflight groups while case 1. is applied at the scale of VS.

Moreover, since the project supports several retracking approaches, each range variable from the L2 is processed separately, resulting in new variables for each of them. The same applies for altimeter band (e.g., Ku) and modes (e.g., SARM). The combinations of these parameters define the **altimetric context** of a variable outputted by the L3 processor. For more details about altimetric contexts, please refer to the PSD [RD-07], section 4.

Inputs:
L2WM data (in-memory)
L3 processor configuration file
Outputs:
New Variables (added to L2WM data):
- boolean **`flag_preserve_record_<mission>_<band>_<mode>_<retracker>`** : True if record has been marked as good by the outliers rejection routines, False otherwise.

Synopsis:
1. For each L2WM overflight group:
    1. For each altimetric **`<context>`** :
        1. Apply appropriate outlier rejection routine
        2. Set the result in boolean variable **`flag_preserve_record_<context>`**.

12.2.5. OPO Reduction

OPO stands for "One per Overflight" and means that only one (real or estimated) measurement shall remain as the representative of a waterbody overflight. It is an "N to 1" operation that reduces L2WM data down to L3 by applying appropriate operators onto L2 variables (e.g., **`time`**, **`lon`**, **`lat`**, **`water_level`**, etc.) in order to produce L3 data that contain only one record per L2 overflight group. Hence, L3's **`time`**, **`lon`** and **`lat`** variables are of different length and values than their L2 counterparts. Regarding the Water Level variables specifically, the reduction operator is the median of the L2 valid measurements.

Inputs:
1. L2WM data records (in-memory)
2. L3 processor configuration file

Outputs:
1. In-memory L3-structured data records, comprising new variables:
    1. double **`time`** : Time of L3 record, the mean at waterbody overflight.
    2. double **`lon`**, **`lat`** : Longitude & Latitude of L3 record, the mean at waterbody overflight.
    3. string **`mission_id`** : Identifier of the satellite mission providing this record.

4. unsigned int **operation_mode** : Code for the altimeter mode in operation.
5. double **water_level_<mission>_<band>_<mode>_<retracker>** : Estimated Water Level w.r.t. Geoid height.
6. unsigned int **nb_l2_meas_<mission>_<band>_<mode>_<retracker>** : Number of L2 measurements involved in the computation of this L3 Water Level.
7. double **std_l2_meas_<mission>_<band>_<mode>_<retracker>** : Standard Deviation of L2 measurements involved in the computation of this L3 Water Level.
8. double **geoid_height** : Geoid height, useful if users wants to use their own (and remove this value remove variable(s) **water_level_***).

Synopsis:

1. Create new (in-memory) L3 data structure
2. For each overflight group:
    1. For each L3 variable:
        1. For each altimetric **<context>** :
            1. Discard bad records with variable **flag_preserve_record_<context>** to records
            2. Perform reduction from L2WM to L3 (compute new variables).
            3. Set results into L3 data structure

12.2.6. Tiles Grouping & Write data to L3 product files

In order to ease the use of the data outputted by this algorithm, the data are organised by geographical tiles (aka, geo-tiles), replicating the tiled cartographic breakdown of the Water Mask. It is anticipated that these tiles will be of 1°×1° lon/lat as in the SRTM/SWBD WM dataset.
The merge as L3 files is done by grouping the altimetry on the basis of the WM tile name of the L2WM track-tile files from which data originate.

Ex: All L2WM data from the attached to the same WM tile, e.g., from pass files **L2WM_w061s11s_S3A_L2_*.nc** will populate the output L3 file **L3_w061s11s_S3A.nc**.

The output of this algorithm are sets of L3 product files in netCDF format (cf. PSD [RD-07] for more details about the L2 file format). This step manages the writing of L3 netCDF files to disk.

Inputs:

L3 data records (in-memory)
L3 processor configuration file

Outputs:

L3 netCDF files

Synopsis:

1. For each records belonging to each WM tile in L3 data records:
    1. Create L3 output netCDF filename w.r.t. the naming convention
    2. Write L3 records to netCDF file

12.3. Development Choices and Trade-offs

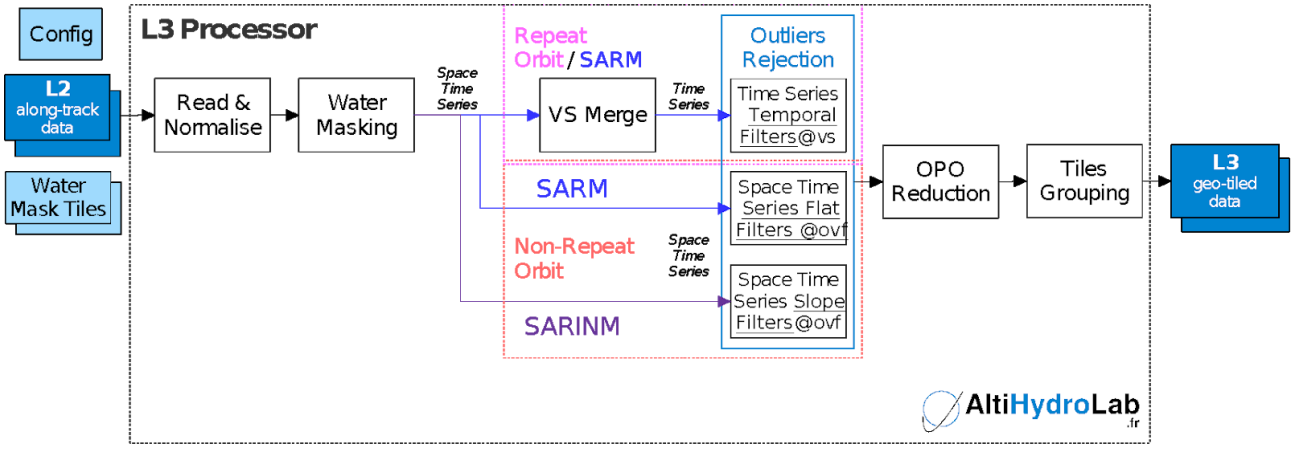
The algorithm is designed to deal with data at global scale while preserving computing resources as much as possible (RAM in particular) and reducing I/O operations as much as possible (because of the large file collections involved in L2 and WM in particular).

The algorithm is written in Python language which is very popular and free/open source software and remains fast & powerful when used adequately (e.g., with the numpy library).

While specific studies have addressed this issue, an important limitation of the algorithm for non-repeat orbit data is that it is not possible to apply the traditional temporal filters because there are no means to produce time series from measurements spread in space & time at global scale. The choices made here are to perform outliers rejection on a per-overflight rather than per time series. The algorithm will also serve to evaluate the retrievability of local river slope from SARINM data in the objective of improving outliers rejection.

12.4. Data flow

The diagram below summarises the practical organisation of the processing steps described in the sections above.



12.5. References

Bercher N., Calmant, Picot N., Fleury S. (2012b), “Evaluation of CryoSat-2 measurements for the monitoring of large river water levels”. In Proceedings of the Symposium on "20 years of progress in radar altimetry", 24-29 September, Venice, Italy. Poster and paper.

Bercher N., Fabry P., Roca M., Martinez B., Fernandes J., Lázaro C., Gustafsson D., Arheimer B., Ambrózio A, Restano M, Benveniste J. (2016b). "Validation of CryoSat-2 SAR and SARin modes over rivers for the SHAPE project". In « New era of altimetry, new challenges », Ocean Surface Topography Science Team meeting (OSTST), 31 Oct – 4 Nov 2016, La Rochelle, France. Poster.

W. Randolph Franklin, https://wrf.ecse.rpi.edu//Research/Short_Notes/pnpoly.html, 1994-2006.

SHAPE project web page, https://projects.along-track.com/shape/, 2020.

13. L3 River/Lake Level (DTU Space)

The following algorithm describes the generation of lake and river (virtual station) level time series generated from Level 2 retracked water levels.

13.1. Theoretical Description, physics of the problem

When deriving the L3 product Water level time series for lakes and rivers, the goal is to provide the best possible summary measure of the individual along-track measurements. Altimetry based water levels from lakes and rivers may be erroneous due to land contamination in the waveform. Here we assume that observations related to the nadir water surface and the erroneous observations originate from two different distributions. To describe all observations we use a mixture between Cauchy and Normal distributions. This will ensure a more robust estimate, which will reduce the influence of outliers. We also expect that measurements that are close to each other in time will be more alike compared to measurements that are far apart. Hence, ensuring that the temporal correlation is taken into account will also improve the robustness in the time series. Hence, to reconstruct the water level time series of lake and river crossings we set up a state-space model, which is described in the sections below.

13.2. Algorithm Definition: Processing Steps and Mathematical Description

The water level time series is reconstructed by a simple state-space model given in equations 13.1-13.3. Equation 13.1 describes the observation part of the model.

$$H^{obs}{}_{ij} = H^{true}{}_{j} + bias(sat_i) + \sigma(sat_i)\varepsilon_{ij} \quad , \qquad \text{Eq. 13.1}$$

where ϵ_{ij} follows the mixture distribution given in equation (13.3). The index i indicates the individual observation and j indicates the time of a given observation. It is assumed that measurements along a crossing of a lake/river have the same time stamp. If more than one mission is used a bias is added to the model. This is relevant for lakes in this study. σ is a scaling parameter. $H^{true}{}_{j}$ are the random effects expressing the underlying true water levels.

The process part is a simple random walk.

$$H^{true}{}_{j} = H_{j-1}{}^{true} + \sqrt{t_j - t_{j-1}}\sigma_{RW} z_j \text{ , where } z_j \sim N(0,1) \qquad \text{Eq. 13.2}$$

The random walk ensures that the temporal correlation between measurements is taken into account. Here the error term $\sigma_{RW} z_j$ is scaled by the square root of the time difference between the current and previous measurements. To ensure a robust reconstruction, we assume that the

measurement follows a mixture distribution between a Gaussian and a student t-distribution with one degree of freedom (Cauchy distribution).

$$f(x) = (1-p)\varphi(x) + pt_1(x) \qquad \text{Eq. 13.3}$$

Here $f(x)$ describes the mixture distribution, where p is a number between $[0,1]$, describing the fraction of the Cauchy distribution t_1 and φ is the density of a standard Gaussian distribution. This distribution is illustrated in Figure 13.1 and characterized with heavier tails making it more robust against erroneous heights.

The above model has the following parameter vector $\theta = (\sigma_{obs}(sat_1),\ \sigma_{obs}(sat_2),\ ...,\ \sigma_{Rw},\ bias(sat_1),\ bias(sat_2),\ ...)$ and the random effects $H^{true}{}_j$. To estimate the parameters and the unobserved random effects we construct the marginal likelihood function given by

$$L_M(H^{obs}, \theta) = \int L(H^{obs}, H^{true}, \theta) dH^{true} \qquad \text{Eq. 13.4}$$

Where L is the joint likelihood of the process and observation part. The log-likelihood functions cannot be minimized directly since H^{true} is unobserved. To solve the integral in Eq. 13.4 a Laplace approximation is used. The marginal log-likelihood functions can then be minimized as a fixed parameter problem. A more detailed description is found in Nielsen et al, (2015). A preliminary source code is available from GitHub https://github.com/cavios/tshydro . The model is implemented in “R” via the package “TMB” (Template Model Builder) (Kristensen et al., 2016).

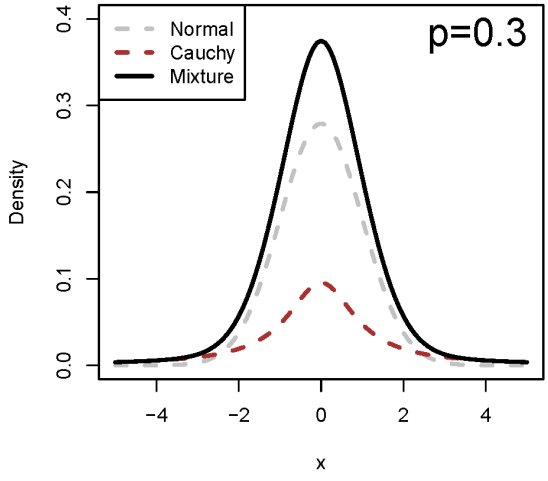


Figure 13.1: Illustration of the mixture distribution.

The processing steps to derive a water level time series for one virtual station is outlined in the flowchart below. In summary the processing steps can be described as follows:

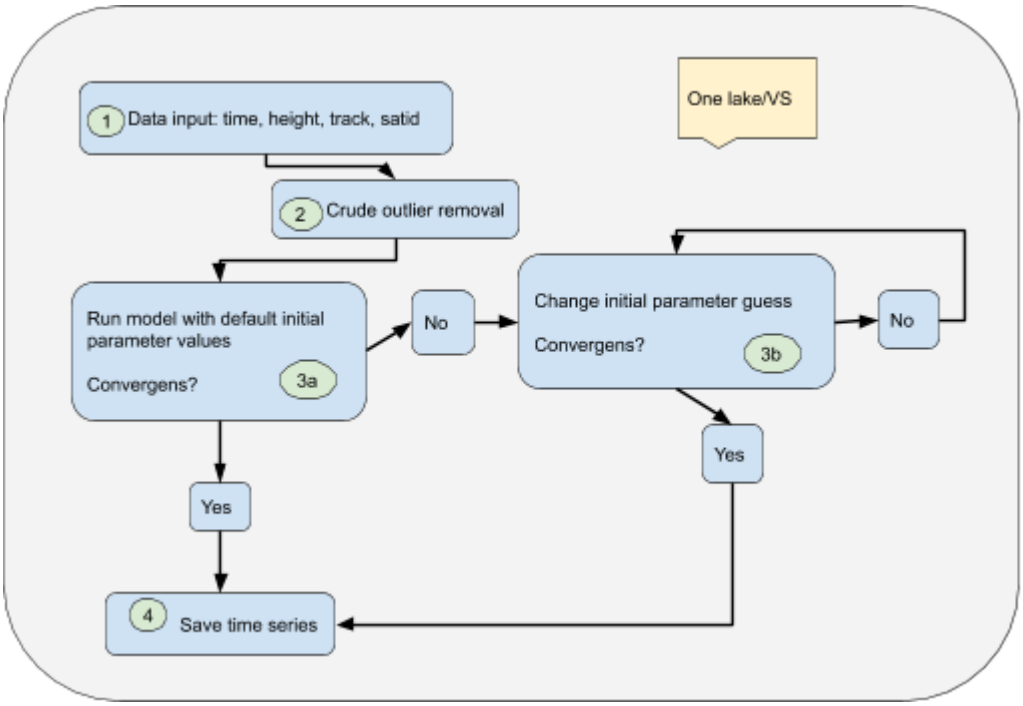


Figure 13.2: Flowchart illustrating the processing steps to derive water level time series via the method described in section 13.2.

1. Preparation of input data. The input data needs at least to contain the columns; time, height, track identifier, satellite identifier (if more than one mission is used).
2. Before running the time series model the surface water level is filtered to remove crude outliers.
3. Run time series model. If convergence is not reached the initial parameter values are changed.
4. Save output to file.

13.3. Development Choices and Trade Offs

The retracked surface water elevations may contain erroneous observations if the height is wrongly estimated in the retracking procedure. This can happen for noisy waveforms with several peaks. Hence, it is important to have a robust algorithm when reconstructing the surface water elevation time series. Here we use a robust error distribution in the observation part of the time series model, which in an objective manner downweight erroneous observations. In some cases outliers are grouped, this can happen if an off-nadir signal in the waveform is wrongfully retracked for several measurements. Another reason for erroneous measurements can be if the range window is incorrectly positioned. This will typically create outliers that are several meters off. To identify crude outliers and avoid removing a potential signal it is necessary to identify the amplitude of the signal.

To aid the process of outlier identification waveform parameters such as; leading edge position, OGOG width, backscatter, max power could be useful or simply a grade indicating the quality of the waveform. In this way the individual measurements can be weighted differently when reconstructing the time series.

Special attention must be paid when reconstructing the water level time series at river virtual stations, where the orientation of the ground track with respect to the river may be of importance. Hence, in the extreme case where the ground track is parallel to the river, the crossing may be a sloping surface. This could be accommodated by adding a slope parameter in the model (Eq. 13.1).

13.4. Data Flow

The data flow below is described in the bullet points below

- The level 2 surface water elevations must be extracted by water mask to collect observations related to the considered water body
- Prepare data input format
- Once the time series is constructed the parameters: time, modeled water elevation, and standard deviation of the modeled water elevation is saved to a file

13.5. References

Kristensen, Kasper, Nielsen, Anders, Berg, Casper Willestofte, Skaug, Hans J. Skaug, Bell, B. (2016). TMB: Automatic differentiation and laplace approximation Authors. Journal of Statistical Software, 70(5), 1–21.

Nielsen, K., Stenseng, L., Andersen, O. B., Villadsen, H., & Knudsen, P. (2015). Validation of CryoSat-2 SAR mode based lake levels. Remote Sensing of Environment, 171, 162–170. https://doi.org/10.1016/j.rse.2015.10.023

14. L4 River Discharge (NUIM)

14.1. Theoretical Description, physics of the problem

14.1.1. Empirical group of algos description (NUIM)

Rating curve method

Rating curves discharge estimation is a fundamental approach used to obtain daily discharges at gauging stations. Relations are established between simultaneously measured water heights and water discharges. The latter are calculated from instrumentally measured water area and velocity in a given river section. These measurements are then used for development of rating curves describing the H-Q relations. Daily discharges are calculated from the daily measurements of water level at gauge stations.

Bjerklie equation

Using thousands of field observations, Bjerklie et al. (2003, 2005) developed a simplified empirical equation based on hydraulic laws for parabolic river channels.The discharge is estimated as a function of width (B), depth (D) and water slope (S). The parameters of the equation are calibrated on the US rivers, nevertheless they have been used in many studies worldwide. A regional adjustment of the parameters can be beneficial for discharge accuracy.

14.1.2. Physical group of algos description (NUIM)

Manning method

A hydraulic equation known in Manning formulation underlines all physically-based recently developed satellite discharge estimation methods [Durand et al., 2016]. A particular interest of the use of the hydraulic equations consists in their potential application for ungauged rivers.

14.2. Algorithm Definition: Processing Steps and Mathematical Description

Rating curves

The rating curves have a power form and can be approximated by equation 14.1.

$$Q=a(H-d)^b \qquad \text{Eq. 14.1,}$$

where H is water height, a and b are parameters of the equation specific for a cross-section, d is the coefficient related to zero flow equivalent water height (m). The parameter a is controlled by section width, bottom slope and friction, while b is mostly related to the section shape - configuration of channel/banks/floodplain [Rantz et al., 1982]. During periods of low variability in the water level

especially on the rivers covered by seasonal ice, a polynomial function can produce a better fit between Q and H [Zakharova et al., 2019]. As water flow is not a stationary process, H-Q relations are rarely uniform, especially in the case of boreal or Arctic rivers with ice cover, or when rivers have large floodplains covered by high vegetation. Both ice and high vegetation significantly change the roughness conditions resulting in changes of water velocity and in point deviations from the main H-Q stationary line. In certain cases the use of a set of rating curves developed for specifique flow conditions could be advantageous [Zakharova et al., 2020]. As the rating curve is a river section specific relation, an application of ground station equations in altimetric practice is problematic. The altimetry-build rating curve (Halti - Qinsitu) can be used instead. For this the simultaneous data on daily in situ discharge better within 100 km distance from a point of retrieval of altimetric water level time series is required [Zakharova et al., 2006, Zakharova et al., 2020]. Of course, each pair of virtual/gauge stations requires an exploration of general hydraulic conditions affecting quality of fit (similarity of river morphology, effect of tributaries, bad slope change etc).

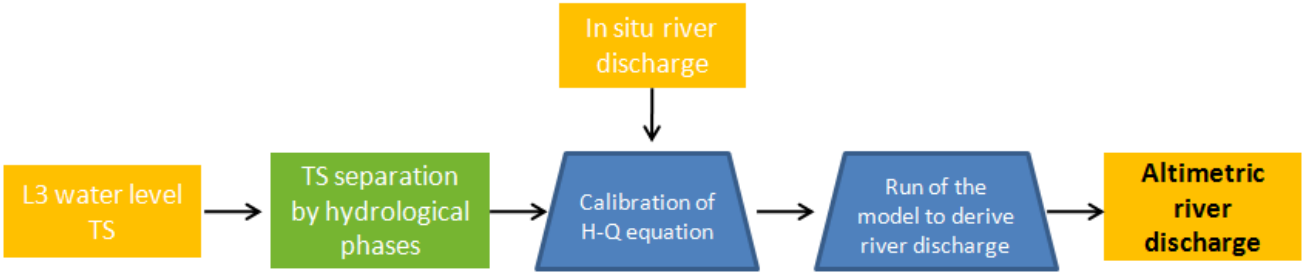


Figure 14.1. Flow chart for the river discharge retrievals using the rating curves.

Bjerklie equation

The Bjerklie equation relates the river discharge to the river width (B), depth (D) and water slope (S) (Eq. 14.2). The equation is applicable for the bankfull discharge. The water width can be taken from the optical images or related to the altimetric river height via a simple power equation [Zakharova et al., 2020]. The water height (H) and the slope are reconstructed from the space-state altimetric height model (Eq. 14.1) with user-defined frequency. The river depth is the most problematic parameter, which is calculated from the altimetric H and the cross-sectional mean river depth (D0) at the lowest H. The D0 can be found via an optimisation, from regional relations with the width, from cross-sectional profiling or navigation maps or in the global database developed from historical discharge measurements [Andreadis et al., 2013].

$$Q=1.77*B^{1.02}*D^{1.74}*S^{0.35}, \qquad \text{Eq. 14.2,}$$

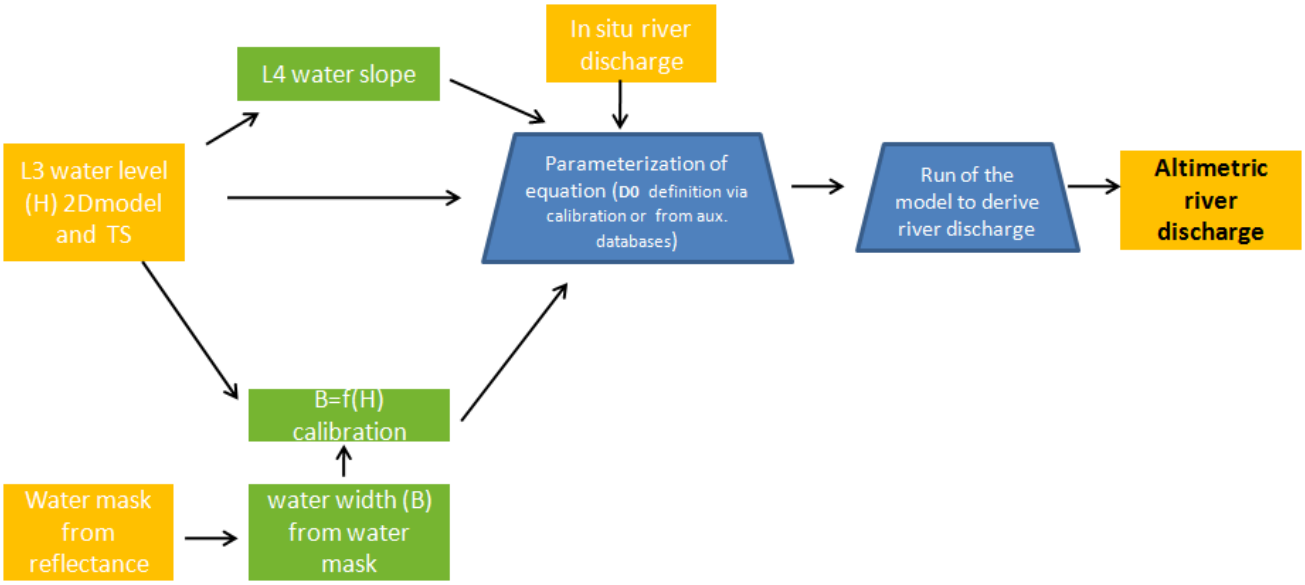


Figure 14.2. Flow chart for the river discharge retrievals using the Bjerklie equation.

Manning Method

The river discharge is calculated as a product of cross-section area and mean water velocity:

$$Q = A \times V, \qquad \text{Eq. 14.3}$$

A - flow contributing section area, m^2, V- water velocity m/s. For rivers, where the channel width is significantly larger than the depth, the area can be approximated by assuming a rectangular cross-section:

$$A = B \times h, \qquad \text{Eq. 14.4,}$$

where

$$h = D0 + \Delta H \qquad \text{Eq. 14.5,}$$

B - channel width, m; h - mean depth, m; D0 - initial depth at minimal water level, m; ΔH - water height correction at time t, m. The Manning water velocity has following formulation:

$$V = 1/n \times R^{2/3} \times S^{1/2} \qquad \text{Eq. 14.6,}$$

where

$$R = A/P, \qquad \text{Eq. 14.7,}$$

n - Manning's roughness coefficient, R - hydraulic radius, S - water surface slope m/m, P -wetted perimeter, m.

As the channel width (B) varies with the water level, a relation B-H can be established using the river width, derived from dynamic water mask, and the river height, retrieved from altimetry at the moment of acquisition of satellite image used for the mask.

$$B = a \times H_{alti}^{\ b} \qquad \text{Eq. 14.8,}$$

where B is the channel width, a and b - parameters. The a and b parameters are defined by the shape of the cross-channel section at a given location (rectangular, trapezoidal or arc). However, in practice they are usually calibrated for following reasons: 1) the cross-section shape is unknown; 2) the cross-section shape is complex enough for approximation by one of the cited forms, 3) the floodplain or sandbanks can be accounted for in one single equation. Similar to Bjerklie method, the water width and slope are derived from the satellite measurements.

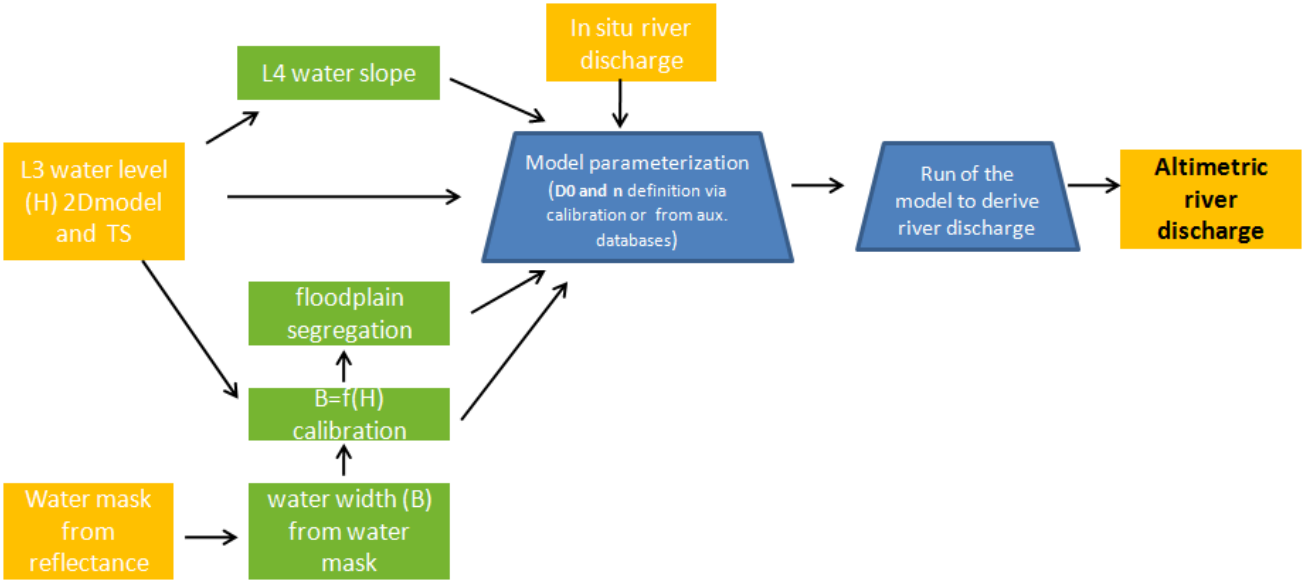


Figure 13.3. Flow chart for the river discharge retrievals using the Manning equation.

14.3. Development Choices and Trade Offs

The complexity of the algorithms increases from the rating curve to the Manning approach. The availability of the auxiliary information is beneficial for discharge retrieval accuracy. All algorithms involve the calibration step. The length of the training dataset can be critical for calibration results. The calibration period should include all hydrological phases. For altimetric observations of sub-monthly frequency, 2-3 years calibration period can be enough for RC establishing for discharge recession and low flow periods. However, in [Zakharova et al., 2020] it was shown that ENVISAT with 35-days observational frequency often missed the Ob R. spring flood and 3 years of observations were not sufficient for fitting the flood rise RC. The application of the Bjerklie and Manning algorithms can be constrained by the insufficient accuracy of the altimetric water slope retrievals or by the lack of information on river morphology. The Bjerklie and Manning algorithms are highly sensitive to the accuracy of the initial river depth. When simultaneous in situ river discharge is

available the initial river depth can be optimised along with the roughness coefficient. Alternatively to optimisation, the estimation of the roughness coefficient can be taken from manuals [Chow, 1959] or evaluated by equation (9) proposed in Bjerklie et al. (2003):

$$n = 0.22 \times S^{0.18} \qquad \text{Eq. 14.9}$$

Special correction for Manning coefficient for winter period for rivers covered by seasonal ice can be required. This correction is important for Arctic river reaches with hummocky ice cover. The correction can be found in [Bruner, 2016].

14.4. Data Flow

The algorithms are written in Matlab. The main algorithms' input are the altimetric water height and the water slope (for Bjerklie and Manning approaches). All algorithms require a set of external river reach specific parameters. These parameters will be derived during the calibration phase and presented within the final discharge product or as the georeferenced database.

14.5. References

Andreadis, K. M., G. J.-P. Schumann, and T. Pavelsky, 2013: A simple global river bankfull width and depth database. Water Resour. Res., 49, 7164–7168.

Bjerklie, D.M.; Lawrence Dingman, S.; Vorosmarty, C.J.; Bolster, C.H.; Congalton, R.G. Evaluating the potential for measuring river discharge from space. J. Hydrol. 2003, 278, 17–38.

Bjerklie, D.M.; Moller, D.; Smith, L.C.; Dingman, S.L. Estimating discharge in rivers using remotely sensed hydraulic information. J. Hydrol. 2005, 309, 191–209.

Bruner G.W., Hec-Ras, River Analysis System, Hydraulic Reference Manual, Ver.5, February, 2016. US Army Corps of Engineers, Hydrologic Engineering Center.

Chow V. T., Open-Channel Hydraulics, McGraw-Hill, New York, 1959.

Durand, M., Gleason, C.J., Garambois, P.A., Bjerklie, D., Smith, L.C., Roux, H., Rodriguez, E., Bates, P.D., Pavelsky, T.M., Monnier, J., Chen, X., Di Baldassarre, G., Fiset, J.-M.,Flipo, N., Frasson, R.P.D.M., Fulton, J., Goutal, N., Hossain, F., Humphries, E., Minear, J.T., Mukolwe, M.M., Neal, J.C., Ricci, S., Sanders, B.F., Schumann, G., Schubert, J.E., Vilmin, L., 2016. An intercomparison of remote sensing river discharge estimation algorithms from measurements of river height, width, and slope. Water Resour.Res. 52, 4527–4549.

Rantz S.E. et al., Measurement and computation of streamflow. Measurement of Stage and Discharge, US Geological Survey Water Supply Paper, 1982, vol. 1, p. 284.

Zakharova E., Nielsen K., Kamenev G., Kouraev A. River discharge estimation from radar altimetry: assessment of satellite performance, river scales and methods. J. of Hydrology, 2020.

Zakharova E.A., I.N. Krylenko, A.V. Kouraev, Use of non-polar orbiting satellite radar altimeters of the Jason series for estimation of river input to the Arctic Ocean, Journal of Hydrology, 2019, 568, 322-333.

15. L4 River Discharge (CNR-IRPI)

15.1. Theoretical Description, physics of the problem

The process to estimate river discharge from satellite remote sensing is based on the merging of data from two sensors: altimeter and multispectral. Based on the traditional definition of river discharge as the product of river flow area and velocity, the two satellite sensors are used to define the two quantities, respectively. Indeed, once known the survey of the cross-section geometry, the flow area is calculated as a function of the water level derived by satellite altimetry, whereas the flow velocity, traditionally measured through specific instruments installed in-situ (current meter, Acoustic doppler current profiler, velocimeter), is here a proxy coming from the reflectance measured by the Near Infrared signal of the multispectral sensor (*Tarpanelli et al., 2015*).

While for the river water level, the measurement is provided directly by the altimeter, for the velocity measurement, the process is more complex (*Tarpanelli et al., 2013*). In detail, this measurement depends on the physical process whereby the passive response of the reflectance signal coming from the soil is different from those coming from the water. This difference is the key parameter to identify a change in the land area nearby the river channel that is demonstrated to be strongly correlated with river discharge. The increase of the river discharge produces an increase of water surface width, and the area close to the river becomes wetter changing its reflectance response. For an area near the river that is not affected by water, the reflectance remains almost constant (except for changes in vegetation cover) and its ratio with the reflectance of the wetted area can more accurately determine the estimation of changes in hydrological forcing, than the wet area alone. Indeed, due to the variations of water volume during flood events, the reflectance of a wet pixel decreases, while the reflectance of a dry pixel remains fairly constant. Consequently, in case of flooding the reflectance ratio between the dry pixel (called calibration pixel, C) and the wet pixel (called measurements pixel, M) is sensitive to the increase of water in the wet pixel and, hence, is directly correlated to the increasing of river discharge (see *Figure 15.1*).

The advantage of the method is the medium spatial resolution and the high (almost daily) temporal resolution of the images. In the study of *Tarpanelli et al., (2013)*, where the main process is described, the reflectance ratio C/M has been extracted from a temporal series of seven years of almost daily images of MODIS over four stations along the Po River. The reflectance ratio C/M has been seen vary with the discharge, even if more affinity has been found with the river flow velocity, for which it was possible to derive a regional linear regression (between C/M and flow velocity). In a successive study (*Tarpanelli et al., 2015*), the same authors leveraged the regional law extracted for the Po river to estimate the flow velocity in another site along the river, and they combined this information with the water level derived by altimetry. Knowing the bottom of the river from previous surveys, through the entropic theory (Moramarco et al., 2011) it was possible to derive the geometry of the section and, hence, the flow area. If no other information is available to define the geometry of the cross-section, a revised version of the approach has been proposed in the RIDESAT project and in the next Section the description of the approach is given.

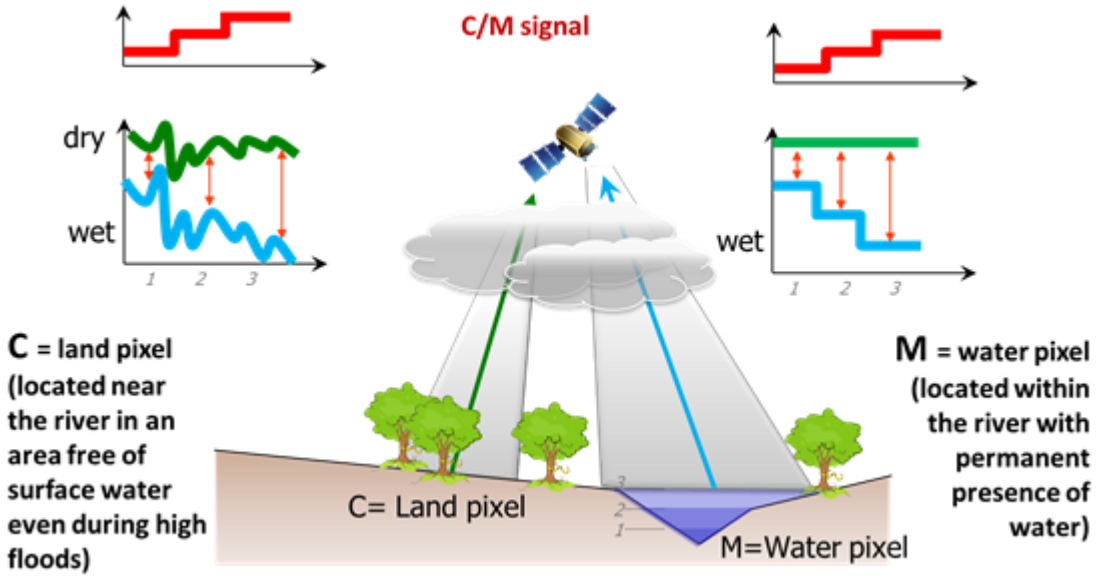


Figure 15.1: Approach for the identification of the water surface variation by optical images.

15.2. Algorithm Definition: Processing Steps and Mathematical Description

The main processing to derive the reflectance ratio C/M from Near Infrared images is described in *Tarpanelli et al. (2013)* and represented in *Figure 15.2*. From each image, a box of size $J \times K$ is selected. Generally, the box is centred on a gauged station measuring hydraulic variables of river discharge or water level in order to carry out a direct comparison between the signal ratio versus these variables. Pixels affected by cloud cover and/or snow are identified by using a fixed threshold on the reflectance (0.2), confirmed by visual inspection, and discarded. A resampling 3 by 3 pixels from the original grid is done to smooth the pixel variability of the reflectance *(Li et al., 2019).* For each pixel i (i = 1, 2, …, $J \times K$), the matrix X [N, $J \times K - 1$] of the C/M time series is calculated by assuming the pixel i as C and the remaining ones as M. N represents the number of available imagery. In particular, X is formed by $J \times K - 1$ columns, each one representing the C/M time series of length N. Finally, $(J \times K) \times (J \times K - 1)$ C/M time series of length N are obtained and compared with the time series of in-situ observations of river discharge and for every time series, the correlation coefficient is calculated. The maximum value of correlation identifies the location of M and C pixel more representative. Once the C/M time series are processed, they are smoothed through a low pass filter (averaging moving window). The resulting products represent the Level-3 product of reflectance, and will be identified with the name of the optical sensors originally used to derive the dataset (MODIS or OLCI).

The reflectances ratio C/M [-] is found correlated with the flow velocity, v, according a relationship assumed in the form:

$$v = m\left(\frac{C}{M}\right)^{f} \qquad \text{Eq. 15.1}$$

in which m [m/s] and f [-] are empirical parameters of the regression.

According to the base hydraulic definition, river discharge, Q [m^3/s] is given by the product:

$$Q = v \cdot A \qquad \text{Eq. 15.2}$$

in which A [m^2] is the cross-sectional area of flow that can be written as a function of water stage h, in the form:

$$A = ah^b = a(H - H_0)^b \qquad \text{Eq. 15.3}$$

where H [m] is the water level, H_0 [m] the null-discharge elevation (bottom of the cross-section), a [m^{2-b}] and b [-] are parameters related to the surface width and the shape of the section (*Neal et al., 2015*). Radar altimeter measures the water surface elevation H, and in case of low flow it can provide a measure of minimum water level, H_{min} rather than H_0. Indeed, H_0 is a rather difficult if not impossible variable to estimate by satellite. For this reason, it is assumed that the flow area A is given by the sum between the cross-sectional flow area beneath the lowest height measurement A_{min} and an incremental area δA (*Frasson et al., 2017*):

$$A = A_{min} + \delta A = A_{min} + a(H - H_{min})^b \qquad \text{Eq. 15.4}$$

It follows that the discharge is given by two components, one linked to the minimum quantity, Q_{min} flowing into the river (corresponding to the minimum water level observed) and another linked to the incremental discharge, δQ:

$$Q = Q_{min} + \delta Q = v\left(A_{min} + a(H - H_{min})^b\right) \qquad \text{Eq. 15.5}$$

Considering the analysis at daily scale, it is useful to write the discharge per unit area, dividing Q for the drainage basin upstream of point on the stream, A_d. This operation is necessary to normalize the parameters values and compare between a site and another. The discharge at Eq. (15.5) expressed in m^3/s, is represented by symbol q and it is expressed as m^3/km^2/day in Eq. (15.6). The constant c [-] is equal to 0.0864, if A_d is expressed in km^2.

$$q = c\frac{Q}{A_d} = c\ \frac{Q_{min}}{A_d} + c\frac{\delta Q}{A_d} \qquad \text{Eq 15.6}$$

Because the first term is smaller than the second term, the first term can be neglected. Substituting the Eq. 15.1 and Eq. 15.4 in Eq. 15.6:

$$q = c\frac{\delta Q}{A_d} = c\frac{v\delta A}{A_d} = \frac{K}{A_d}\left(\frac{C}{M}\right)^f (H - H_{min})^b \qquad \text{Eq. 15.7}$$

in which *K* is a parameter given by the product of the other two parameters *m* and *a* and the constant *c*.

The parameters *K*, *b* and *f* are estimated by the minimization of the Nash-Sutcliffe efficiency, *NS*, (*Nash and Sutcliffe, 1970*) between the simulated discharge and the ground observed discharge.

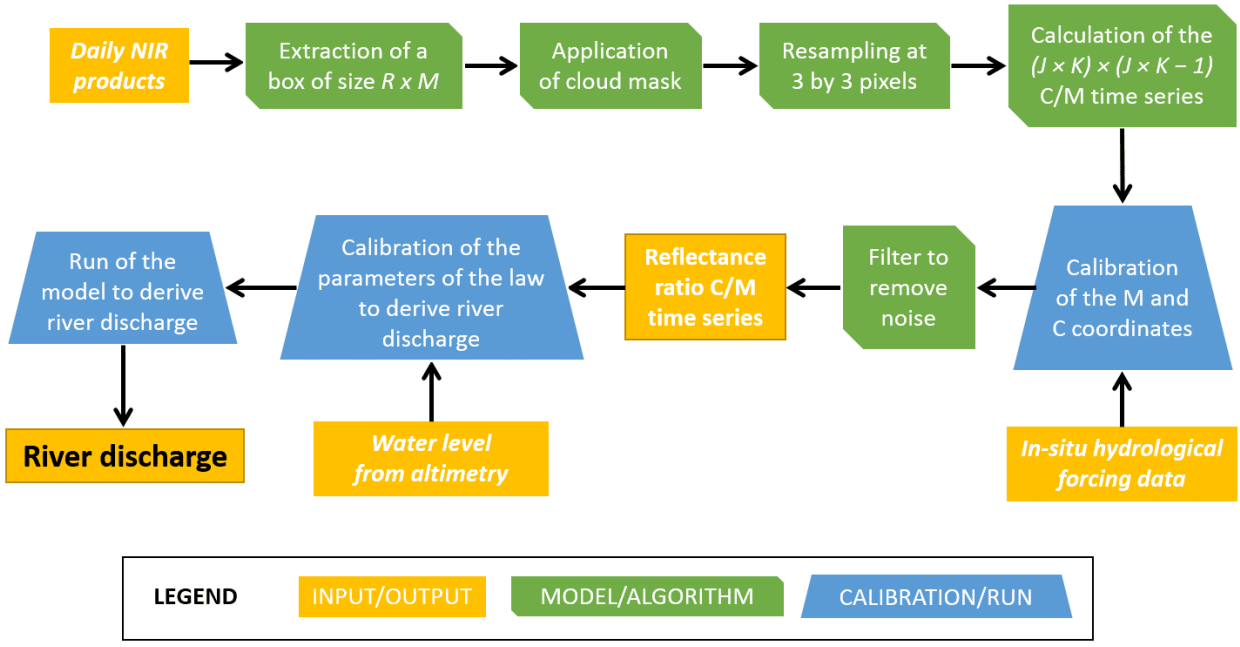


Figure 15.2: Flow chart of the procedure to derive river discharge from NIR images and altimetry derived water level.

15.3. Development Choices and Trade Offs

The algorithm to derive river discharge takes into account the water level and the reflectance ratio as a proxy of flow velocity. No other hydrological forcing is considered in the estimation of river discharge, despite it is well known the importance to have information such as the water width and the water depth (water level above the bottom of the river cross-section section), fundamental for describing the area of the cross-section and hence, to improve the assessment of the river discharge. Improvements in this sense are planned through the use of existing methods and

datasets *(Allen et Pavelsky, 2018)* in order to enhance the physical processes and refine the algorithm.

The proposed merging procedure was tested in RIDESAT project exclusively in calibration. No validation period (totally unconstrained) was tested for any sites. A further test will be carried out by dividing the datasets of river discharge in calibration and validation. This is possible thanks to the collection of longer time series (more than 4 years) and hence a more robust dataset.

In sites where a reasonable period of data is available to calibrate the parameters, the procedure can be extended to longer periods based on the availability of altimetry water level and NIR images, in order to have long and continuous time series of river discharge.

In sites where no ground data are available to calibrate the parameters, a set of parameters belonging to other sites characterized by hydrological similarity will be used to estimate river discharge.

15.4. Data Flow

The approach to extract reflectance ratio from NIR images is written in Matlab as well as the approach to derive river discharge. As input, the NIR images from multispectral sensors (MODIS, OLCI) and the water level derived by altimetry are required. The in situ data are required to calibrate the parameters of the approaches.

As output, the approach provides the coordinates, the date and the simulated river discharge for each site analysed.

15.5. References

Allen G. H., Pavelsky T. M. (2018). Global extent of rivers and streams, *Science* 361(6402), 585-588. https://doi.org/10.1126/science.aat0636

Frasson R. P. D. M., Wei R., Durand M., Minear J. T., Domeneghetti A., Schumann G., Williams B.A., Rogriguez E., Picamilh C., Lion C., Pavelsky T. Garambois P.-A. (2017). Automated river reach definition strategies: Applications for the surface water and ocean topography mission. *Water Resources Research*, *53*(10), 8164-8186.

Li H., Li H., Wang J., Hao X. (2019) Extending the Ability of Near-Infrared Images to Monitor Small River Discharge on the Northeastern Tibetan Plateau, *Water Resources Research*, 55(11), 8404-8421. doi: 10.1029/2018WR023808

T. Moramarco, G. Corato, F. Melone, and V. P. Singh, “An entropy-based method for determining the flow depth distribution in natural channels,” J. Hydrol., vol. 497, pp. 176–188, 2013.

Nash J.E., Sutcliffe J.V. (1970). River flow forecasting through conceptual models, part I—A discussion of principles. *Journal of hydrology*, *10*(3), 282-290.

Neal J.C., Odoni N.A., Trigg M.A., Freer J.E., Garcia-Pintado J., Mason D.C., Wood M., Bates P. D. (2015). Efficient incorporation of channel cross-section geometry uncertainty into regional and global scale flood inundation models. *Journal of Hydrology*, *529*, 169-183. Doi:10.1016/j.jhydrol.2015.07.026

Tarpanelli A., Brocca L., Lacava T., Melone F., Moramarco T., Faruolo M., Pergola N., Tramutoli V. (2013) Toward the estimation of river discharge variations using MODIS data in ungauged basins. *Remote Sensing of Environment*, 136, 47–55. Doi:10.1016/j.rse.2013.04.010

Tarpanelli A., Brocca L., Barbetta S., Faruolo M., Lacava T., Moramarco T. (2015) Coupling MODIS and radar altimetry data for discharge estimation in poorly gauged river basin. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 8(1), 141-148. doi:10.1109/JSTARS.2014.2320582.

16. List of Acronyms

ACE2 Altimeter Corrected Elevations (vers. 2)
AD Applicable Documents
AGC Automatic Gain Control
AH Alti-Hydro
AHP Alti-Hydro Product(s)
AI Action Item
AIM Action Item Management (tool)
AltiKa Altimeter in Ka band and bi-frequency radiometer instrument
AMSR-E Advanced Microwave Scanning Radiometer-Earth Observing System
ANA Agência Nacional de Águas (National Water Agency, Brazil)
AoA Angle of arrival
API Application Programming Interface
AR Acceptance Review
ASAP As Soon As Possible
ASCII American Standard Code for Information Interchange
ATBD Algorithm Technical Basis Document
ATK Along-Track S.A.S.
AVISO Archivage, Validation et Interprétation des données des Satellites Océanographiques
BIPR Background Intellectual Property Right
CASH Contribution de l'Altimetrie Spatiale à l'Hydrologie (Contribution of Space Altimetry to Hydrology)
CCN Contract Change Notice
CFI Customer Furnished Item
CLASS NOAA/Comprehensive Large Array-Data Stewardship System
CoG Centre of Gravity
CPP CryoSat-2 Processing Prototype (CNES)
CryoSat-2 Altimetry satellite for the measurement of the polar ice caps and the ice thickness
CRF Conditional Random Field
CRISTAL Copernicus polaR Ice and Snow Topography ALtimeter
CRUCIAL CRyosat-2 sUCcess over Inland wAter and Land
CSV Comma Separated Values
CTOH Centre de Topographie des Océans et de l'Hydrosphère (Centre of Topography of the Oceans and the Hydrosphere)
DAO Data Access Object
DARD Data Access Requirement Document
DBSCAN Density-Based Spatial Clustering of Applications with Noise
DD Delay-Doppler
DDM Delay-Doppler Map
DDP Delay-Doppler Processor
DEM Digital Elevation Model
DGC Doppler Ground Cell
DPM Detailed Processing Model
DPP Data Procurement Plan
DTC Dry Tropospheric Correction
DTU Danmarks Tekniske Universitet (Technical University of Denmark)
DVT Data Validation Table
ECMWF European Centre for Medium-Range Weather Forecasts
ECSS European Cooperation for Space Standardisation
EGM Earth Gravitational Model
ENVISAT ENVIronment SATellite
EO Earth Observation
EOEP Earth Observation Enveloppe Programme
EOLi Earth Observation Link
EOLi-SA EOLi-Stand Alone
EPN EUREF Permanent Network
ERA ECMWF ReAnalysis
ESA European Space Agency
EUREF IAG Reference Frame Sub-Commission for Europe
FBR Full Bit Rate
FFT Fast Fourier Transform
FR Final Review
FTP File Transfer Protocol
FCUP (from portuguese) *"Faculdade de Ciências da Universidade"*, Science faculty of the University of Porto
GDAL Geospatial Data Abstraction Library
GDR, [I-,S-] Geophysical Data Record, [Interim-, Scientific-]

GFZ	Deutsche GeoForschungsZentrum (German Research Centre for Geosciences)
GNSS	Global Navigation Satellite System
GOCE	Gravity field and steady-state Ocean Circulation Explorer
GPD	GNSS-derived Path Delay
G-POD	Grid Processing on Demand
GPT2	Global Pressure and Temperature model (vers. 2)
GPP	Ground Processing Processor
GPS	Global Positioning System
GRACE	Gravity Recovery And Climate Experiment
GRDC	Global Runoff Data Centre
GRGS	Groupe de Recherche de Géodésie Spatiale (Space Geodesy Research Group)
GRLM	Global Reservoir and Lake Monitor
GTN-L	Global Terrestrial Network - Lakes
HDF-EOS	Hierarchical Data Format - Earth Observing System
HGT	A SRTM file format
HWS	High Water Stage
HYCOS	Hycos Hydraulics & Control Systems
HYPE	Hydrological Predictions for the Environment model
IAG	International Association of Geodesy
IDAN	Intensity-Driven Adaptive-Neighbourhood
IE	Individual Echoes
IGS	International GNSS (Global Navigation Satellite Systems) Service
IM	Internal Meeting (e.g. not with the client)
IODD	Input Output Data Document
IPF	Integrated Processing Facility
ISD	isardSAT
ITRF	International Terrestrial Reference Frame
IRF	Impulse Response Function
Jason-1	Altimetry satellite, T/P follow-on
Jason-2	Altimetry satellite, also known as the « Ocean Surface Topography Mission » (OSTM), Jason-1 follow-on
Jason-3	Altimetry satellite, Jason-2 follow-on
Jason-CS	Jason Continuity of Service
KML	Keyhole Markup Language
KO	Kick Off
L1A	Level-1A
L1B	Level-1B
L1B-S, L1BS	Level-1B-S (aka, Stack data)
L2	Level-2
L3	Level-3
L4	Level-4
LAGEOS	Laser Geodynamics Satellite
LEGOS	(french acr.) Laboratoire d'Études en Géophysique et Océanographie Spatiale (Laboratory for Studies in Geophysics and Spatial Oceanography)
LOTUS	Preparing Land and Ocean Take Up from Sentinel-3
LPS	Living Planet Symposium
LRM	Low Resolution Mode
LSE	Least Square Estimator
LWL	Lake Water Level
LWS	Low Water Stage
MARS	Meteorological Archival and Retrieval System
MDL	Minimum Description Length
MMSE	Minimum Mean Square Error
MNDWI	Modification of Normalised Difference Water Index
MoM	Minutes of Meeting
MPC	Mission Performance Centre
MRC	Mekong River Commission
MTR	Mid Term Review
MSS	Mean Square Slope
MSS	Mean Sea Surface
MWR	Microwave Radiometer
NAVATT	Navigation and Attitude
NDVI	Normalised Difference Vegetation Index
NDWI	Normalised Difference Water Index
netCDF	Network Common Data Form
NOAA	National Oceanic and Atmospheric Administration
NR	New Requirement (w.r.t. the SoW)
NRT	Near Real-Time
NWM	Numerical Weather Model
OCOG	Offset Centre of Gravity
OPC	One per Crossing
OSTM	Ocean Surface Topography Mission (also known as Jason-2), is also the name of the satellites series T/P, Jason-1, Jason-2 and Jason-3
OVS	Orbit State Vector
PDF	Probability Density Function
PEACHI	Prototype for Expertise on AltiKa for Coastal, Hydrology and Ice
PEPS	Sentinel Product Exploitation Platform (CNES)

PISTACH (french acr.) Prototype Innovant de Système de Traitement pour les Applications Cotières et l'Hydrologie
PLRM Pseudo Low Resolution Mode
PMP Project Management Plan
POCCD Processing Options Configuration Control Document
PR Progress Report
PRF Pulse Repetition Frequency
PSD Product Specification Document
PTR Point Target Response
PVP Product Validation Plan
PVR Product Validation Report
PVS Pseudo Virtual Station(s)
PWF Pseudo Waveform
RADS Radar Altimeter Database System
RANSAC Random Sample Consensus
RB Requirements Baseline (document)
RCMC Range Cell Migration Curve
RCS Radar Cross Section
RD Reference Document
RDSAR Reduced SAR (also known as Pseudo-LRM)
RF Random Forest
RGB Red, Green, Blue
RID Review Item Discrepancy
RIDESAT RIver flow monitoring and Discharge Estimation by integrating multiple SATellite
RIP Range Integrated Power (of the MLD) sometimes referred as Angular Power Response (APR)
RMS Root Mean Square
ROI (geographical) Region(s) Of Interest
RP Report Period (a month that is being reported into a Progress Report)
RSS Remote Sensing Systems
RWD River Water Discharge
RWL River Water Level
SAMOSA SAR Altimetry MOde Studies and Applications
SAR Synthetic Aperture Radar
SARAL In Indian "simple", in english "SAtellite for ARgos and AltiKa.
SARIn SAR Interferometric (CryoSat-2/SIRAL mode)
SARM SAR Mode
SARINMSARIn Mode
SARvatore SAR Versatile Altimetric Toolkit for Ocean Research & Exploitation
SCOOP SAR Altimetry Coastal & Open Ocean Performance
SDP Software Development Plan
SEOM Scientific Exploitation of Operational Missions
SHAPE Sentinel-3 Hydrologic Altimetry PrototypE
SINC Signal model Involving Numerical Convolutional
SINCS SINC for SAR
SLA Sea Level Anomaly
SME Small and Medium-sized Enterprise
SMHI Swedish Meteorological and Hydrological Institute
SNAP SeNtinel Application Platform
SOA State Of the Art
SOW Statement Of Work
SPR Software Problem Reporting
SPS Sentinel-3 Surface Topography Mission System Performance Simulator
SR Sparse Representation
SRAL SAR Radar Altimeter
SRTM Shuttle Radar Topography Mission
SSB Sea State Bias
SSH Sea Surface Height
SSMI/IS Special Sensor Microwave Imager (SSM/I) Sounder
SSO Single Sign-On
Stack Matrix of stacked Doppler beams
STAR Spatio-Temporal Altimetry Retracker
STARS Spatio-Temporal Altimetry Retracker for SAR
STD Standard Deviation
STM Sentinel-3 Surface Topography Mission
SUM Software User Manual
SWBD SRTM Water Body Data
SWH Significant Wave Height
TAI Temps Atomique International (International Atomic Time)
TBC To Be Confirmed
TBD To Be Done
TCWV Total Column Water Vapour
TDS Test Data Set
TMI Tropical Rainfall Measuring Mission (TRMM) Microwave Imager
TN Technical Note
T/P Topex/Poseidon (altimetry satellite)
TR Technical Risk
UNESCO United Nations Educational, Scientific and

	Cultural Organization
URL	Uniform Resource Locator
USGS	United States Geological Survey
USO	Ultra Stable Oscillator
UTC	Coordinated Universal Time
UWM	Updated Water Mask
VS	Virtual Station(s)
VH	Vertical-Horizontal polarisation
VV	Vertical-Vertical polarisation
WBS	Work Breakdown Structure
WF	Waveform
WFR	Water Fraction Ratio
WMO	World Meteorological Organization
WP	Work Package(s)
w.r.t.	with respect to
WTC	Wet Tropospheric Correction
XML	eXtensible Markup Language
ZP	Zero Padding