Airborne SAR efficient signal processing for very high resolution.

Hubert M.J. Cantalloube, and Carole E. Nahum

(Invited Paper)

***Abstract*—Frequency domain SAR image formation algorithms are of lower computation cost (both in number of elementary operations and in required memory storage) than direct time-domain integration, and do not make the narrow band (monochromatic) assumption. Both the advantages are critical to very high resolution imaging because a lower complexity yields a drastic computation time decrease as cross-range resolution increases and the narrow-band assumption is more and more a concern as range resolution (hence bandwidth) increases. Though an exact formulation exists (omega-k algorithm) for a perfect linear uniform acquisition trajectory, in real-life airborne case, the unavoidable trajectory deviation from a straight line needs to be compensated. This motion compensation (MoComp) operation is much more complicated in the case of frequency domain processing. An efficient technique for this purpose is presented. This method keeps the parallel processing aspect, and has been programmed both for multi-thread on multi-core/symmetrical multi-processor CPUs and for graphic processors units (GPU).**

***Index Terms*—Airborne radar, Synthetic aperture radar, Motion compensation, Focusing.**

I. Introduction

A. Overview of SAR image formation algorithms

IMAGE formation for synthetic aperture radar (SAR) is a computationally demanding application. Early SAR systems used optical image formation [2] because digital processing was beyond the capabilities of available computers, even though the radar resolution and range were far less demanding than today. As computer performance increased, digital processing replaced optical processing in the late 80s. SAR evolved from "Doppler beam sharpening" (DBS) which was merely narrow Doppler-band filtering, to "focused SAR" using correlation with a Doppler frequency chirp along slow-time (thus de-convolving the signal along azimuth columns versus the translation invariant phase history of points at a given range). The latter algorithm was limited to cases where range during synthetic antenna (integration time) remains within a range resolution cell (low range resolution and non squinted viewing). Increasing resolution led to develop techniques derived from the seismic prospection approach.

Namely, two broad families of processing emerged: First, time-domain processing in which, for each image pixel, pulse propagation time is equalised (range migration) before pulse signals are added. This time-domain processing is called backprojection algorithm (BPA). Frequency-domain processing is an alternative based on the observation that range migration within integration is nearly the same –with a slow-time translation– for the pixels at the same distance and depends only on the observation angle during integration. Hence the idea of compensating the range migration in the Doppler domain (under narrow-band assumption observation angle and Doppler frequency are one-to-one related) which led to the range-Doppler algorithm (RDA).

The clear advantage of the frequency-domain processing is that the interpolation in range needs only to be done once per Doppler & range, while in time-domain it should be done once per aperture sample, range & azimuth, thus the frequency domain computation for the whole image is equivalent to the time-domain computation of a single row of the image. The two extra Fourier transforms needed to switch between native (time) domain and frequency-domain are equivalent to a few time-domain image row computation, the frequency domain computation is clearly faster: For a given image area at resolution r, time-domain complexity is $O(r^3)$ while it is $O(r^2 \log(r))$ for frequency-domain.

However, the above sketched RDA fails to produce well focused images at very high resolution because range migration depends on the imaging angle (squint) δ within the aperture and not on the Doppler frequency, thus RDA introduces a "chromatic aberration" when the relative bandwidth increases. Post-compensation of the chromatic aberration of RDA, the "secondary range compression" (SRC) was proposed, but a more efficient solution came from a seismic imaging technique [1]: It uses a 2D Fourier transform (i.e. the processing is done in full frequency domain, unlike the RDA where only slow-time –or azimuth– is in the frequency domain), and can be adapted to SAR imaging [3] as the ω-k algorithm (also known as "range migration algorithm" or RMA) that will be further described sections II and below.

However, the fact that processing is done in the frequency-domain means that the space variations of processing across the image for motion compensation of ω-k algorithm are difficult. This is opposite to time-domain BPA which can be applied as-is for an arbitrary trajectory and for image points on an arbitrary ground surface. Early ω-k implementations [7] suggested the use of a "corrective autofocus post-processing" in the airborne case even when trajectory was accurately measured. However, the motion compensation can be computed (not estimated from image blur) and applied as presented in section IV below. An alternative, a simplified approximate (but faster) frequency-domain algorithm (called

H. Cantalloube is with the Département d'Électromagnétisme et Radar, Office National d'Études et Recherches Aérospatiales, Chemin de la Hunière 91120 Palaiseau, France tél +33 1 80 38 62 12, fax +33 1 80 38 62 69, http://www.normalesup.org/~cantallo/index.html .

C. Nahum is with the Direction Générale de l'Armement, 5 rue des Mathurins 92221 Bagneux, France

This work was funded and supported by the Délégation Générale de l'Armement (the French MoD procurement directorate)

"chirp scaling algorithm" or CSA) was proposed, and its motion compensation is easier[4].

Seismic imaging also brought a solution to the higher complexity of BPA: the idea is to combine smaller (less directive) apertures (starting from the real antenna aperture) into progressively wider (highly directive) but more numerous apertures (ending with the full synthetic aperture for each image row). This algorithm called fast backprojection (FBPA) [6] yields less accurate apertures (and antenna pattern compensation) than plain BPA but has a complexity in $O(r^2 \log(r))$ similar to that of frequency-domain processing, while still remaining quite easy to motion compensate.

As a fairly good compromise CSA is widely used today – especially for spaceborne SAR–, but for the higher resolution, high relative bandwidth and/or wide integration angle SAR imaging, one requires FBPA [11], BPA on supercomputers/ clusters [8], or ω-k approaches.

Current work on SAR processors are mainly in two directions: The first is on parallelizing the processing for increasing its speed. This started in the 90s with mainframe supercomputers [5] and small computer clusters, but recently regained interest with the availability of cheap and small multi-core CPUs and GPUs [13] [15] [17].

The second currently active domain is bistatic SAR (SAR with the transmitting radar and receiving sensor on two different aircraft/satellites) for which the same dilemma between easy focusing in time-domain [18] and fast synthesis in frequency domain [14] occurs.

B. ONERA motion compensated $\omega - k$ algorithm

The frequency domain (ω-k) algorithm motion compensation described below fulfills the challenge of simultaneously providing $O(r^2 \log(r))$ complexity, motion compensation capable of focusing high resolution and/or wide aperture airborne acquired images without introducing non-deterministic geometric radiometric and phase distortions.

The latter is all but a minor concern: State-of-the-art frequency domain processing for airborne SAR resorts to autofocus for motion compensation thus introducing unpredictable geometric distortion. This is not important for some applications (such as the imaging mode of the front end radar of a versatile combat aircraft), but it ruins other such as repeat-pass interferometry or coherent change detection (CCD). CCD is highly relevant since it allows detection of improvised explosive devices (IED) that account for most of the military casualties in current conflicts. That is why Gotcha SAR real-time processor uses BPA notwithstanding the drastic increase in weight, onboard power consumption and cost it implies[16] [19].

The algorithm presented here is routinely used for synthesizing airborne SAR imagery from the RAMSES/Sethi systems which are, at X-band, very similar to Gotcha SAR, while keeping a geometrical and radiometric accuracy compatible with CCD, repeat-pass interferometry or polarimetric interferometry (cf [20] for a review of civilian science research using Sethi images from this processor). Implementation of the algorithm on existing operational systems such as Gotcha SAR would allow decreasing weight, power consumption & cost of on-board computer system by at least one order of magnitude (which is of operational significance).

Images obtained from this processor may be combined with images obtained from the three other alternative processors used at ONERA, namely plain BPA, 2-stage factored backprojection algorithm (related to FBPA, but with higher $O(r^{2.5})$ complexity), and polar format algorithm (PFA). This is why the absolute radiometric gain of the $\omega - k$ algorithm is computed here (2b) since it is mandatory in order to have a radar calibration valid for all four algorithms.

C. Paper organisation

Section II recalls the principles of $\omega - k$ algorithm in the nominal (non motion compensated) case. Unlike the classical presentation of the algorithm [12], not only the phase, but also the amplitude of the nominal processing is addressed. A geometrical representation of the algorithm principle is introduced which will allow a basic description of the motion compensation.
Section III addresses computer implementation issues for the nominal case algorithm.
Section IV describes the motion compensation principles and computer implementation in the ONERA $\omega - k$ processor.
Last section VI sketches the implementation details for algorithm parallelisation which is instrumental in achieving SAR image synthesis with computation time of the same order of magnitude than the signal acquisition time.

II. Omega-k algorithm principle

Frequency domain processing is motivated by the fact that given a perfectly linear uniform SAR acquisition, a point in the two dimensional signal spectrum corresponds exactly to a point in the spectrum for an image in cylindrical coordinates around the acquisition trajectory. Fig. 1 illustrates this in a simple way (it can be formally derived from the stationary phase principle) by replacing the forth & back propagation at radiowave velocity c by one-way propagation from sensor to "image" area at half velocity $\frac{c}{2}$. (Fig. 1, 3, 4 to 7 & 9 all use this representation of the radiowaves round-trip as a single-way propagation at half velocity.) A given point in the signal two dimensional spectrum correspond to the transmission of a monochromatic signal (fast time frequency) with a constant phase shift from one pulse to the next (corresponding to the slow-time, or Doppler, frequency). Each pulse transmitted radiates as a spherical wave around the sensor position, but the large number of successive pulses sums up to conical waves (In practice, the sensor antenna is directive, hence the elementary waves are spherical portions and the resulting waves are conical portions). Given the invariance through rotation around the trajectory axis, the canonical image plane is radial from the trajectory axis and, in this plane, the conical waves correspond to a pure space frequency.

Hence the basic ω-k algorithm architecture (fig. 2) for the ideal case: Fourier transform an acquired signal block, then select an integration angle (a single look) by trimming a Doppler range width proportional to frequency. Next, map

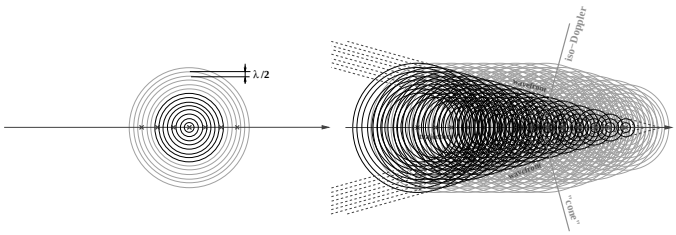

Fig. 1. Principle of the ω-k algorithm: spherical monochromatic waves (left) of linearly uniformly spaced sensor positions combine to conical waves (right) when their phase varies linearly along sensor trajectory (i.e. for a fixed Doppler frequency).

the signal frequencies to the corresponding image frequencies (Stolt resampling) and multiply by the appropriate ("nominal processing") complex factor. Last, inverse Fourier transform the result to a single-look complex image block.

The Stolt resampling from signal wavenumbers (k, k_u) to image wavenumbers (k_r, k_z) (1) and the nominal processing phase ϕ_{nom} (2a) can be graphically derived (fig. 3). The extra $\pi/4$ phase term in (2a) due to the curvature of the elementary wave fronts with respect to the resulting cone, can be derived, as well as the nominal processing amplitude A_{nom} (2b), from the principle of stationary phase (see Appendix A for details).

$$\begin{aligned} k_r &= \sqrt{4k^2 - k_u^2} \\ k_z &= k_u \end{aligned} \tag{1}$$

$$\phi_{nom}(f, \delta) = \frac{-4\pi f}{c}(1 - \cos(\delta - \delta_c))R_c + \frac{\pi}{4} \tag{2a}$$

$$A_{nom}(f, \delta) = \sqrt{\frac{f}{f_c}}\frac{\sin(\delta)}{\sin(\delta_c)} \tag{2b}$$

where f and f_c are the frequency of the sample and the center frequency of the band respectively, δ and δ_c are the sample and image squint angles respectively and R_c is the middle range to which signal array is centered. Physically, the principle of stationary phase states that the sign of the extra phase term with respect to the envelop of the elementary waves depends of the sign of the curvature of the elementary wave phase on the wave front, and the modulus depends on the second derivative. Here the curvature of the sphere with respect to the cone is one dimensional, hence the square root in (2b), and phase varies as the product of the optical path & the frequency, hence the f in (2b).

In fact, the final image is not given in plain cylindrical coordinates, but in squinted cylindrical coordinates (slant range is measured along a cone of angle δ_c instead of radially from trajectory axis) as δ_c may significantly differ from $\pi/2$ due to large possible drift angles in airborne case and/or wide antenna pattern allowing computation of multiple looks within the main Doppler lobe. This image shearing from plain to squinted coordinates (which is a range-proportional translation in azimuth of the image) may be encompassed in the Stolt resampling (because a time-domain shear parallel to the azimuth z axis is equivalent to a frequency-domain shear parallel to the range wavenumber k_r axis) which remains a one-dimensional resampling:

$$k_r = cos(\delta_c)k_u + \sqrt{4k^2 - k_u^2} \tag{3}$$

A thoroughgoing dissertation on the ω-k algorithm for satellite SAR processing in given in [12] (and a short comparison with other frequency domain algorithms in [9]). These references do not consider the nominal phase constant & modulus (needed to calibrate images from different synthesis algorithms *cf* section I-B).

III. Practical issues for the perfect case

The description in section II above is only conceptual. First, the input signal in (fig. 2) is range profiles (pulse responses from each sensor positions); while, in order to reduce peak vs average power ratio, real-life radars spread the pulse in time, and recompress the received echo. Though, at first approximation, the recompression is a deconvolution of the raw echo by the transmitted spread pulse, (which can be efficiently embedded in the scheme of fig. 2 by dividing the "nominal processing" factor, along the k axis, with the Fourier transform of the transmitted pulse) the motion of the sensor alters the transmitted spread pulse with Doppler effect. In the perfect case, this Doppler alteration of range compression can be solved by appropriately scaling the transmitted pulse depending on the δ angle before applying the "nominal processing" factor in the ω-k approach.

Second, pixel size of the resulting image may not match the sampling rate of the acquired signal, hence a "resampling" (or pre-summing) must be done prior to the first Fourier transform (this is why the width of the signal is narrower in frequency domain than in time-domain in fig. 2). As will be shown in subsection IV-A below, such a "resampling" will also be required for motion compensation.

Third, due to both limitations of available computer memory and proportionally increasing cost of Fourier transform with size, the signal is processed by blocks in slow time. The blocks should contain the full integration interval (interval of δ angle corresponding to the synthetic aperture yielding the desired azimuth resolution) for the produced row block of the image, hence successive blocks overlap duration must be the maximum (far range) integration (it is similar to the overlap-discard method for convolution). There is an optimal block size as smaller block are disadvantaged by proportionally bigger overlaps per produced image line while larger blocks are hindered by Fourier transform increasing cost per produced line. Computer architecture and limitations may also restrict the usable block sizes.

Last, the real antenna does not radiate a spherical wave (which would pose a severe left-right ambiguity problem) but a directional antenna pattern. This antenna pattern could be compensated during the first Fourier transform, by first doing the azimuth transform (hence to the range×Doppler domain). Then using the heading to Doppler and elevation to range relations (assuming flat terrain & antenna pattern linear shrinking with frequency, in this perfect case) divide by the antenna pattern. Eventually, a Fourier transform along range

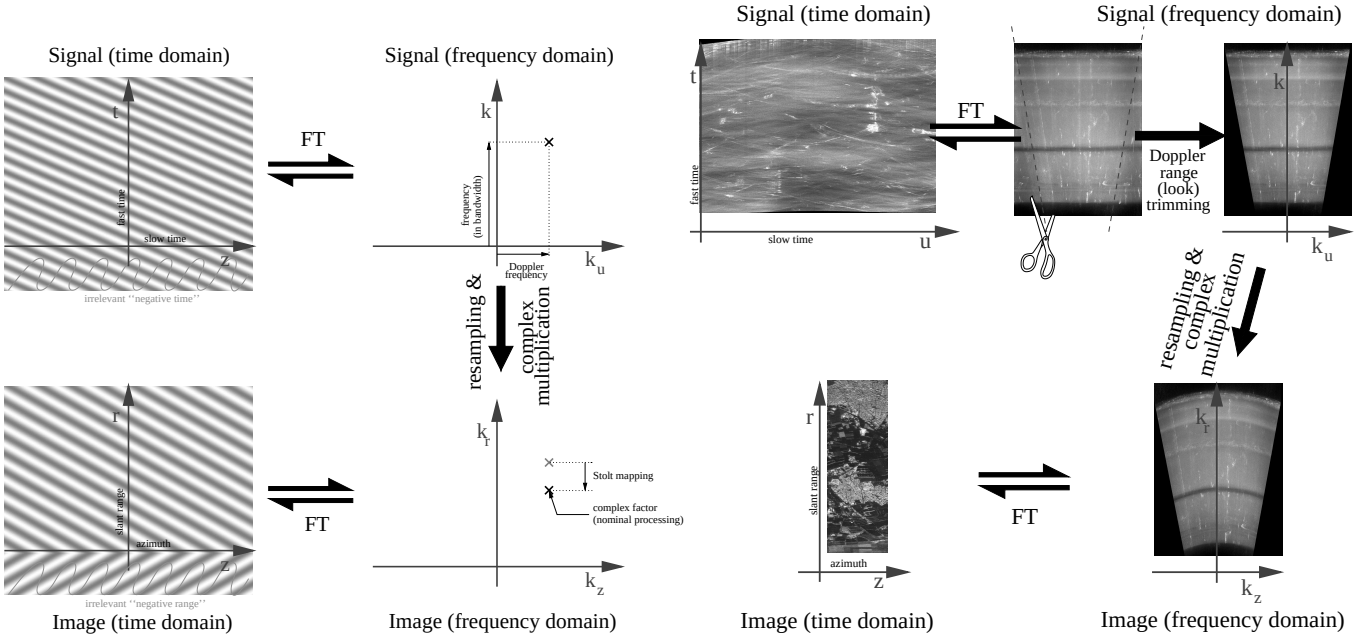

Fig. 2. Basic ω-k algorithm in case of perfect linear uniform trajectory: How a single frequency in signal is mapped into a single frequency in image (left), and how a range in signal Doppler frequency is processed into a single look image (right). UHF/VHF radar signal on the right is from a real acquisition hence the dark line in frequency-domain corresponding to a protected emergency frequency. The strong trajectory curvature –due to a late lining on acquisition axis– induces the top-left and bottom missing signal the time-domain because mocomp was applied in order to focus the final image.

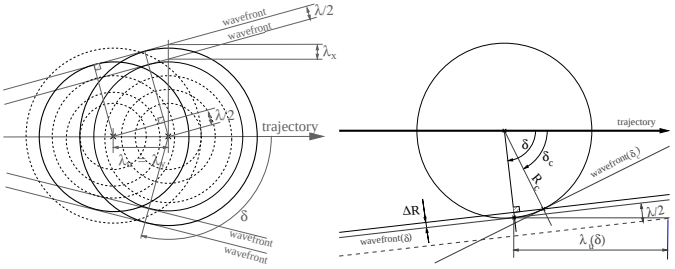

Fig. 3. Graphic derivations of the Stolt mapping (left): let δ be the wavefront angle, consider two transmissions separated by exactly one Doppler period λ_u then $\lambda_r \sin(\delta) = \lambda/2$ and also $\lambda_u \cos(\delta) = \lambda/2 \ldots$ Graphic derivation of the nominal processing phase (right): Let R_c & δ_c be the central range & squint angle for processing, the optical path difference ΔR for a wavefront at a given angle δ is $R_c - R_c \cos(\delta - \delta_c) \ldots$

axis yields the signal in frequency domain compensated for the antenna pattern.

IV. Motion compensation

From the simplified description above, the airborne SAR processing is complicated because air turbulence deviates the carrier aircraft from the planned linear uniform trajectory, and corrective manoeuvres further induce velocity/attitude fluctuations. As high azimuth resolution requires long synthetic antenna (long integration time), real acquisition trajectory can not be approximated by a linear uniform one during integration. Furthermore, the trajectory deviations from the straight line break the axial symmetry of the imaging geometry, thus inducing an elevation dependency in the focusing.

The main idea behind motion compensation for ω-k algorithm is to resample the signal prior to first Fourier transform in such a way that for the wavefront angle δ_c central to the integration interval, resampled signal pure frequency maps to image pure space frequencies, and that the correction of wavefronts with other angles within the integration interval remains small. This latter compensation is obtained by applying a "quadratic phase" (so called because its first approximation is a quadratic function of the Doppler similar to the effect of a longitudinal velocity error) varying across the Doppler frequencies. This quadratic phase, however, varies with range and azimuth, hence the block/sub-block organisation described in subsection IV-B below.

Since the axial symmetry is broken, the motion compensation will only concentrate to a surface in 3D space corresponding to the image focusing points (fig. 4). This surface is computed, before image synthesis, by projecting the imaged area digital elevation model (DEM) in the cylindrical coordinates with the nominal trajectory as axis. This projection into the (downsampled) image geometry is computed by a z-buffer algorithm, initialised with the altitude of the points at angle δ_c behind the nominal trajectory (Nadir imaging). In case more than one DEM cell map to the same point in the result (overlay) there is an option in selecting the top, bottom or average altitude, but in any case the elevation will be discontinuous at the far end, near end or both of the overlay region (as a matter of fact, because the focusing is elevation dependent, it cannot focus simultaneously all layers of an overlaid area). In case no DEM cell maps to a point, the initial value force the focusing on the vertical plane below the nominal trajectory, in which case there is a slope discontinuity in the elevation at the transition between DEM surface and the vertical plane. For further use (block-global quadratic phase strategy described in subsection IV-B) projected elevation points are classified into 3 classes (normal, overlay and vertical

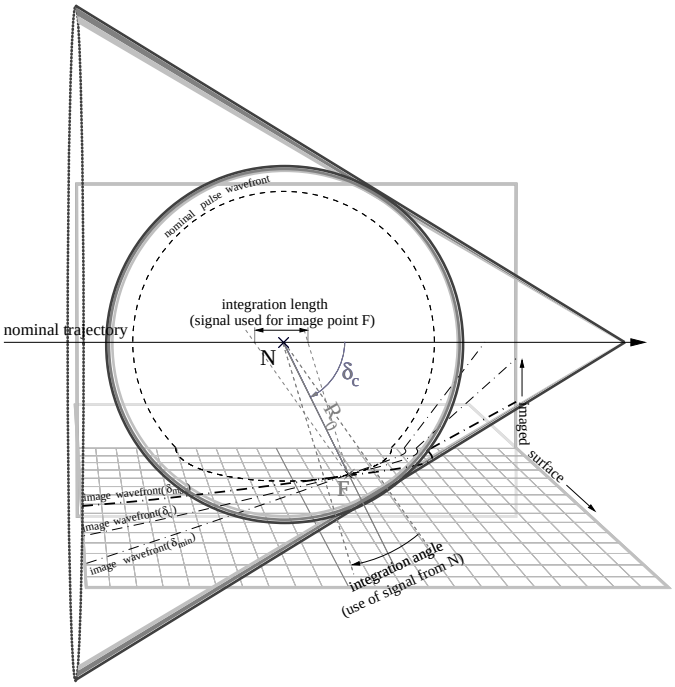

Fig. 4. Imaged surface is the terrain surface (altitude given by DEM) on the observing side and the vertical plane between nadir and trajectory axis. Since the motion compensation is elevation dependent, only the intersections of the 3D wavefronts with this surface are relevant. E.g. for the spherical wavefront for pulse transmitted from N at range R_0 (fat dashed line). For the conical image wavefront of angle δ_{max} that N contributes to at end of integration (fat mixed line), it is half an hyperbola branch on ground and a straight line on the vertical plane since for simplification terrain is represented flat here. In real life, terrain is modelled as a bilinear mesh and the image wavefront as a broken line.

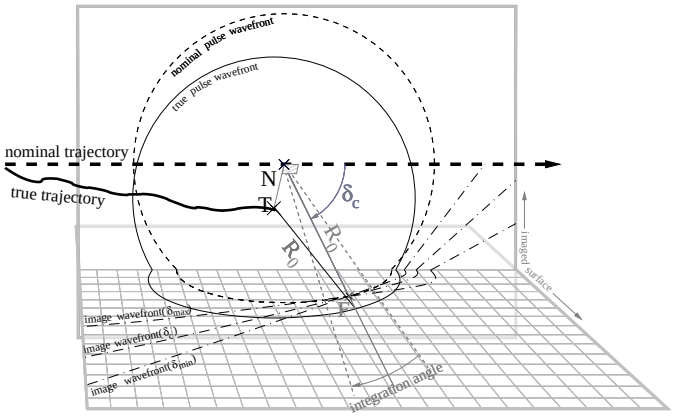

Fig. 5. Before range & azimuth migrations, true signal elementary wavefront for a given acquisition azimuth N and range R_0 (solid fat line) does not match the image wavefronts (mixed lines) that are tangent to the nominal pulse wavefront (dashed line). Range & azimuth migration will try to match both nominal and true pulse wavefront around the focus point F at range R_0 of N under the center squint angle δ_c.

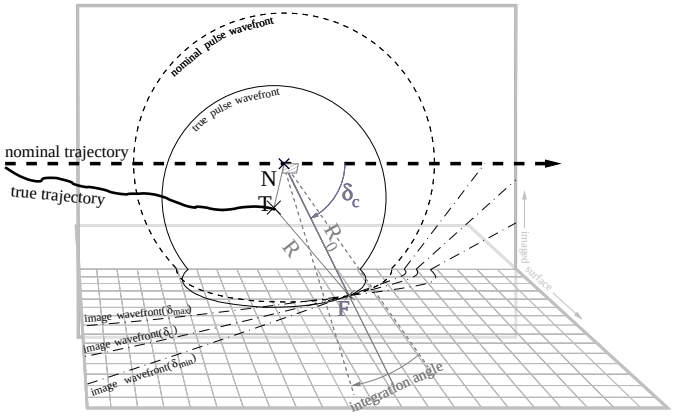

Fig. 6. After mere range migration $R - R_0$ to match the range at the center squint angle δ_c, the true signal elementary wavefront (fat solid line) is secant but not tangent to the image wavefronts (mixed lines), hence would require further strong δ-dependent range corrections.

plane).

A. Range & azimuth migrations

1) Motivation: Fig. 5 to 7 gives the motivation for range and azimuth migrations: The elementary wave corresponding to nominal trajectory point N is a sphere centered at N (as in Fig. 1). For minimising compensation along integration, the elementary wave from the true trajectory point T (a sphere centered at T) must remain as close as possible to the nominal sphere. Resampling (range resolved) signal both in range and azimuth allows it: For any azimuth z and range R_0, we compute the position of the focus point F on the imaging surface, then find the true trajectory point T minimising the angular distance between the $\overrightarrow{FN}$ and $\overrightarrow{FT}$ vectors. With respect to the point on the nominal trajectory of azimuth N, the signal will be sampled at range $R = \|FT\|$ from the range profile acquired at a trajectory point offset by Δz in azimuth (see Appendix B for details on its computation).

2) Explanation: In fact, as only the intersection of both spheres and the imaged surface are required to match (Fig. 4), one idea could be to adjust Δz to have tangent spheres at point F (solution initially implemented [10]). Alternatively one can sample the intersections along integration angle and adjust Δz for best fitting the samples between the spheres. However, because of possible edges and/or discontinuities on imaged surface[1], this may lead to discontinuous azimuth migration, hence it is preferable to use an approximation that does not depend on the imaging surface as described above.

The range & azimuth-migrated signal point at N for range R_0 will have a contribution in the synthesised image for all image wavefronts with δ in the integration interval $[\delta_{min}..\delta_{max}]$, hence on the corresponding sector on the intersection between the nominal elementary (spherical) wavefront at N and the imaged surface. As visible on Fig. 7 deviation of the true trajectory induces a change in curvature of the true pulse wavefront with respect to the nominal one on the imaged surface. This curvature difference implies that a further range correction depending on δ should be applied for this sample. This is the function of the quadratic phase compensation described in subsection IV-B below.

[1]DEM is linearly interpolated in the current implementation for easing its slant range projection. Edges between cells could be suppressed by smoother interpolation such as bicubic or splines. However, in mountain areas, terrain overlay would cause discontinuities in azimuth migration that ruin image focusing in the neighbourhood.

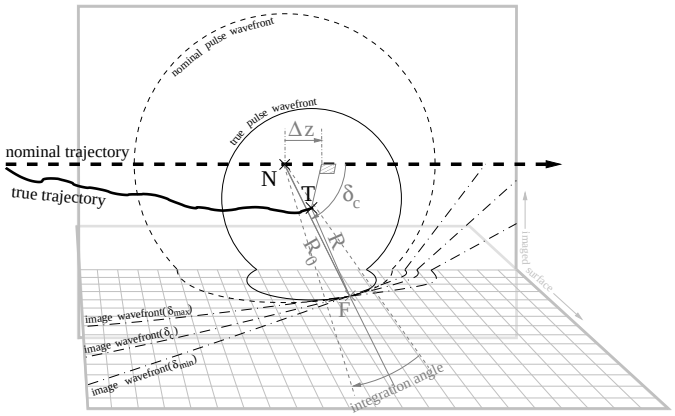

Fig. 7. After both azimuth (slow time) migration Δz and range migration $R-R_0$ of the signal, elementary true and nominal wavefronts are close along the integration interval. The remaining curvature difference is later compensated as "quadratic phase".

Note that minimising the angular distance between N and T from F instead of having the sphere tangent at F means that there is actually a residual linear component in the so called "quadratic phase", but it is not much an issue as long as the phase deviation remains moderate in the integration angle.

3) Implementation details: Range migration is evaluated for each pulse but only for focus points F computed on the imaged surface for a subset of the range gates and the migration is interpolated (on the 3D space) between adjacent computed points. Azimuth migration values, which vary smoothly from pulse to pulse, are only computed for a mesh in range & azimuth and interpolated. To implement the range & azimuth signal migration, a first buffer is required with a fine range sampling and a width larger than the equivalent in slow time of the maximum azimuth migration Δz. Azimuth migration and minimum along-track velocity are computed prior to processing in the "off-line" processing case, hence the buffer size can be safely bound. In the "real-time" processing case, future maximum amplitudes of the azimuth migration and the minimum along-track velocities can not be guessed, hence a sufficient prior bound on the buffer size should be provided to the program (from observed trajectory nonlinearities in similar meteorological conditions).

The fine range sampling, which allows a coherent arbitrary resampling through linear interpolation, is obtained by zero-padding before the last Fourier transform of the range-compression. Though range-compression varies with Doppler, it is done at this stage for the center integration angle δ_c. For the azimuth migration, three interpolation methods are available: Lanczos resampling, pre-summing (time-domain) or Fourier resampling (frequency domain) plus linear resampling.

Lanczos resampling does not change the Doppler spectrum width, and is therefore only used in case the initial slow time sampling is close to (typically up to twice) the desired azimuth resolution, because it otherwise requires the resampled signal to be oversampled with respect to the desired azimuth resolution.

Pre-summing is a low-pass time-domain filter, which can be optimised to minimise the periodic fluctuations with respect to the time-domain full integration across the integration interval (such an optimisation yields a nearly triangular window), due to the window's finite duration however, it imposes an oversampling in azimuth (typically a factor of two) which allows suppression of the first side-lobes when clipping the integration Doppler interval (see Fig. 2).

Fourier (zero-padding & zeroing out the unused Doppler range) interpolation, by typically a factor of 4, followed by a linear interpolation yields an azimuth resampled signal which does not require azimuth oversampling, thus saving both memory used for processing block storage and required computing power for subsequent processing. It has the further advantage that the Doppler dependency of the range processing can be elegantly corrected by doing the forward azimuth Fourier transform before the range compression final Fourier transform and introducing a Doppler varying adapted replica in the range processing. However, this method increases significantly the computation load of the resampling compared to the other two methods. This method is, therefore, mostly used when memory requirements are more constraining than available computing power (namely on GPU, because of the limitation on available graphic memory).

B. Quadratic phase

As visible on Fig. 7 (right), after both range & azimuth signal migration, the elementary spherical wavefronts corresponding to a sample (range/fast-time & azimuth along trajectory) do not present the same curvature on the imaged surface. This curvature implies that there is a mismatch in range that "increases as δ deviates from δ_c" in a more or less quadric manner. As it depends on the relative distance of the nominal N & true "migrated" T trajectory points to the focus point F on the image corresponding to the given signal sample, the curvature varies both with range and with azimuth. (Note that, on Fig. 5 to 7, the deviation of the true trajectory is exaggerated for visibility.) One point critical for the algorithm implementation is that significant curvature changes in azimuth are much slower than significant range fluctuations. Indeed, a range fluctuation larger than, say, $\frac{1}{10}$ of the wavelength may easily occur from pulse to pulse (and are corrected by the range migration). Curvature change requires a significant change of the trajectory deviation with respect to the observing range. Therefore the main idea of the quadratic phase compensation in the ω-k algorithm is to compensate, on a wide band assumption, the average quadratic phase for a processing block in the frequency domain. The difference between the local and average quadratic phase are compensated for azimuth sub-blocks and range gates in a mixed range$\times$Doppler time-frequency-domain (the processing sub-block allows slow azimuth variations of the quadratic phase). This latter compensation being done on a range resolved domain, it is applied as a phase, hence with narrow-band assumption. An important point is that the sub-block size depends on the transversal velocity error independently of the azimuth resolution, hence the algorithm remains with $O(r^2 \log(r))$ complexity.

Range$\times$Doppler quadratic phase compensation is done on (optionally overlapping) sub-blocks with the result being zero-

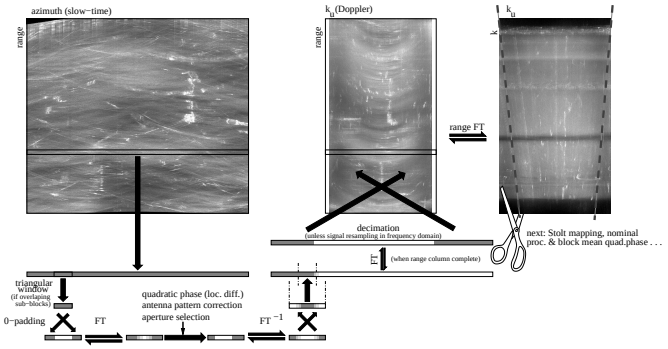

Fig. 8. Signal pre-compensation for second order motion compensation: range columns are processed "sequentially". A range & azimuth varying difference in quadratic phase is applied on overlapping azimuth sub-blocks. Note the sub-block padding because of the azimuth smearing beyond its bounds induced by quadratic phase.

padded (generally by a factor of 2, but larger interpolations may be needed in case of strong motion compensation) to avoid azimuth wrapping of the compensated signal. Indeed, the quadratic phase (difference) applied to a sub-block has an azimuth spread effect very similar to the effect an unmodelled velocity bias has on a SAR image, hence the compensated signal azimuth slice (sub-block) is wider than the original signal slice. Fig. 8 shows how the local difference in quadratic phase is compensated (the block-wise average quadratic phase is simply added to the "nominal processing" phase).

Note that the "quadratic phase" is defined as an arbitrary function of the Doppler frequency, and not as a quadratic function nor as a polynomial (as was the case in early implementation [10]). It is computed from the sensor trajectory and the DEM (see Appendix C for details on its computation). As it is applied on slow-time signal sub-blocks, the sub-block limits appear in the frequency domain of the final image and thus do not cause any geometric, radiometric nor phase blocking artefacts on the image (which is very important for interferometric or polarimetric applications). Moreover, at high resolution, integration spans several sub-blocks and their limits are not parallel in the image spectrum, which smears the side lobes induced by the phase step between adjacent sub-blocks in range below significant level. The azimuth antenna pattern (and also the average elevation antenna pattern on the azimuth sub-block for this range gate) as well as the blanking out of the image Doppler band (if not done in the range & azimuth migration stage) are also applied at this stage (see section V below). Note also that depending on the computer architecture (multi-core or GPU) several range columns may be processed "in parallel" instead of sequentially for performance enhancement (see section VI below).

As the azimuth migration, the quadratic phase is computed for a mesh in azimuth and range, but also for a subset $\{\delta_0 \dots \delta_{n-1}\}$ of the integration interval. For a given resampled range, azimuth & squint angle δ_i, the conical image wavefront of angle δ_i at the given range of the nominal point N is computed on the imaged surface (due to the imaged surface mesh, it is a broken line). The range difference $d\Delta R$ is computed by comparing with the range to true trajectory point

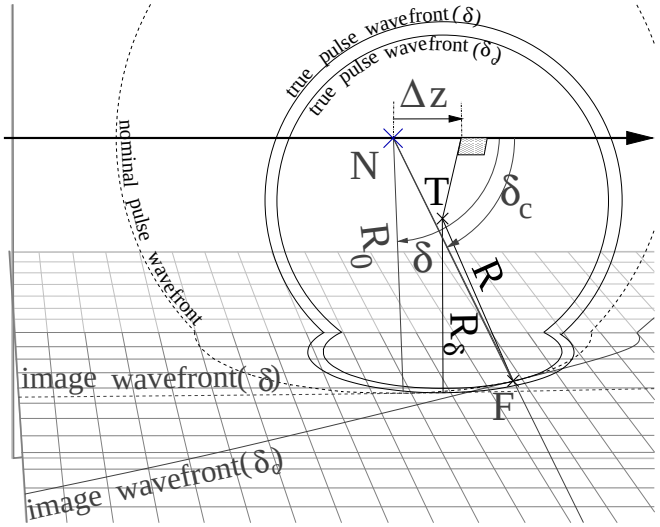

Fig. 9. Computation of the "quadratic phase" for the azimuth of N and the range R_0: The range from the image (conical) wavefront for an angle δ to the true trajectory point T –corresponding to the range & azimuth migrated signal– (i.e. the radii of the "true pulse wavefront" solid lines) varies with δ unlike the range to the nominal trajectory point N (i.e. the radius of the dashed line). Due to the DEM mesh, the image wavefronts are broken lines, computing the distance from it to point T is basic geometry which yields the δ-varying range correction to nominal processing $d\Delta R = R_\delta - R$.

T of the conical wavefront corrected by the range migration (which was computed at the integration center angle δ_c) see Fig. 9.

For simplification of the computations, the "quadratic phase" is computed for a grid with $\sin(\delta_i)$ evenly spaced and aligned with the sampling of the azimuth sub-block in frequency domain (only the samples within the aperture of course). Since the quadratic phase is applied at the azimuth sub-block level when the signal is range time-domain, it is converted to a phase at the middle-frequency of the aperture for each Doppler frequency (Fig. 10).

This conversion of a range error to a phase makes a narrow band assumption not appropriate for high resolution, hence we minimise this correction by applying the "common part" of the range correction in full frequency domain for the whole processing block (while applying the "nominal processing" phase). At this point the conversion of a range difference varying with δ to phase is done using the exact frequency for each sample, but conversely, there is no space variation of the correction (it is constant on the whole block). Only the smaller difference between the local (space varying) and global (constant on the processing block) corrections is applied at the azimuth sub-block level (thus reducing the amplitude of the correction applied with narrow band assumption).

There are several options for determining the processing-block "global quadratic phase", from simply the value at middle-range & the middle of the azimuth block to more complicated ways as taking the average on the block for

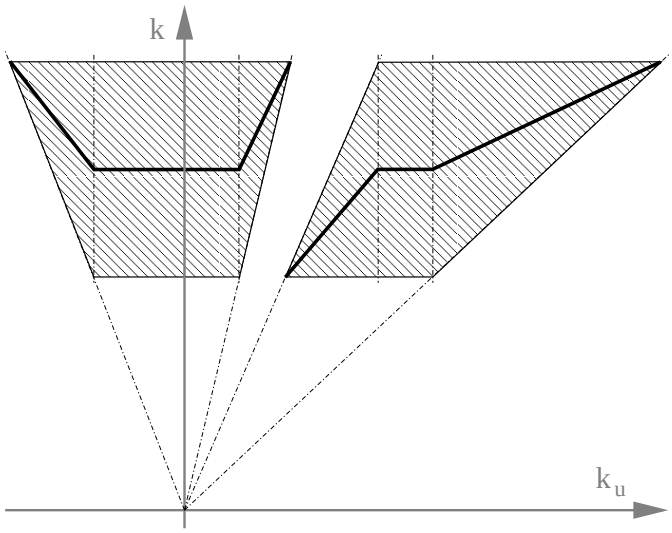

Fig. 10. Signal (fast-time) frequency k (fat line) used for converting "quadratic phase" range error $R_\delta - R_{\delta_c}$ to phase while in the range×Doppler domain. It corresponds to the average wave-number k in the aperture along the constant Doppler frequency (or constant k_u) vertical lines. It is represented for an aperture containing the 0-Doppler (left) and strongly squinted (right).

points on non-overlaid area of the DEM (i.e. averaging with discarding the points on the vertical plane and the points on overlaid parts of the image). Note also that the "global quadratic phase" differs from a processing block to the next, this means that the signal pre-compensation for second order of Fig. 8 needs to be done twice for the overlapping signal between two consecutive blocks. As the range compression and the range & azimuth migrations are not affected by the processing block change, we need to save the overlapping preprocessed signal. Either use an array for saving the overlap, which is the solution implemented on CPU, or use two buffers for before and after second order compensation, which is the solution implemented on GPU. When overlap is less than 50% of the block, the buffer space freed by the decimated part of the spectrum may be used for overlap storage.

C. The range aperture migration issue

In fact, the "difference between the local and global corrections" is slightly more complicated in practice than presented in subsection IV-B above: One important point is that the radar frequency within the band (the aperture in range) is mapped to a space varying interval in k once in full frequency domain after the motion pre-compensations. Indeed, as the range to true trajectory is locally compressed with a rate $\frac{\partial R}{\partial R_0}$ by the range & azimuth migration, it is scaled by the reciprocal in the frequency domain, and this yields an offset dk_r (4) for the center of the signal band after pre-compensation. However, the global quadratic phase is applied (with the nominal processing phase) around the bandwidth center for the middle of the processing block (an offset on k of dk_{r_0}), while the local aperture is mapped to the spectrum with a range & azimuth varying offset dk_r.

$$dk_r = k_r(\frac{1}{\frac{\partial R}{\partial R_0}} - 1) \tag{4}$$

This range aperture offset is very similar to the range band offset that would limit the possibility to obtain an interferogram between an image acquired from the nominal trajectory and another image acquired from the true trajectory: It is a consequence of the baseline between the nominal and true trajectories, hence in airborne case, it varies both in range (due to the incident angle change) and in azimuth (due to true trajectory nonlinearities).

This means that the "global quadratic phase" deduced from the local quadratic phase for a given range gate & azimuth sub-block must be computed from the δ-dependent range error with the signal middle frequency shifted by the $dk_r - dk_{r_0}$ offset. Furthermore, the nominal processing must be applied to a wider interval of k than the mere (dk_{r_0}-shifted) radar bandwidth, but on the interval scanned by the bandwidth for the minimum and maximum "relevant" dk_r. Here again, several strategies are available to select the relevant dk_r offsets, such as the average on the full block or the average excluding point on the vertical and/or overlaid areas...

V. Illumination compensation & remodulation

Unlike the spaceborne SAR, airborne SAR illumination pattern is not a constant function of range as sensor irregular attitude motions during acquisition change the antenna pattern projection on the ground. Aside from the induced radar cross sections (RCS) calibration errors on the final image, uncompensated illumination changes during integration would apply a unwanted (and space-varying) azimuth weighting, thus impacting azimuth resolution and other image quality factors such as peak side lobe & spurious side lobe ratios (PSLR & SSLR).

The illumination compensation is done for each range gate & each azimuth sub-block in the range×Doppler domain. For each sample, the average illumination is computed for the given range & the direction δ_i and compensated. The average (with triangular weighting) illumination during the sub-block acquisition is evaluated from the attitude, the antenna pattern and optional on-board pre-summing of the signal before recording. Note that the antenna pattern may contain a phase component(and not only amplitude) which is sometimes necessary for wide integration angles as the "phase center" approximation for an antenna may not apply (e.g. the UHF/VHF signal used in Fig. 2 & 8 is acquired from a low directivity antenna suffering multi-paths via the aircraft wings that cannot be approximated by a point-like phase center)

Because of this subblockwise compensation, the compensation is not perfect along the integration, hence the final synthesised image is post-corrected in amplitude by comparing at each point the average illumination during its integration and the subblocks compensation applied. The average illumination is also output by the processor because it may be needed for further image processing (such as mosaicking or multi-looking).

In fact there is a last operation before the final image is output: In order to make resampling operations easier, the initial signal is offset in frequency (both within the radar bandwidth and in Doppler) yielding a zero average frequency in both

axes. This frequency offset is reinjected in the image (keeping track of the phase origin) because otherwise the frequency would make image comparison or combination difficult. For example, it would be difficult to make an interferogram or a coherent change detection between two acquisitions. It could also be difficult to combine images obtained by processing sub-apertures, while after offset cancellation, an image with the full aperture is simply the sum of the images from sub-apertures (splitting the radar bandwidth is not uncommon as it allows an antenna pattern compensation varying with frequency –useful for very wide bandwidth– and is often used to share an image synthesis between several GPUs).

VI. Algorithm parallelisation

This ω-k synthesis can be parallelised for increasing the processing speed, as it can be simply separated in stages where data is processed along rows (range) and stages where data is processed along columns (azimuth). In case several processing units are available, they can process simultaneously the data of separate rows during the along-row stages and the data of separate columns during along-column stages, thus increasing the processing speed.

In fact, SAR processing is also limited by data transfers (signal read and, to a lesser extent, image write), hence pipelining (parallelising successive stages of the processing) is also implemented to mask the disk read/write latencies. Namely, 1) signal input to range & azimuth migration, 2) second-order pre-compensation to azimuth forward FT, 3) range forward FT to backward azimuth FT & 4) optional range resampling, illumination post-compensation and image output are the four pipeline stages. In the multi-core/multiprocessor architecture (programmed with the native posix thread interface) true pipelining is done between the pair of stages 1 & 2 and the pair of stages 3 & 4, with a dynamic balancing of the number of threads allocated to stages 1 to 3 (4 is single thread) done after the first block (using measured disk read & processing times). Within each pair, the two stages are run in sequence.

In the GPU architecture (programmed with the OpenCL interface) true pipelining is not possible since a GPU can only run one given program (kernel) at a time (in a "block vector-like" way). It can, however, perform CPU to/from GPU transfers (and disk input/output) during processing: Typically, blocks of signal are input (or image blocks are output) in parallel with processing the previous signal block (respectively the next image block). Furthermore, during disk write (or disk read-modify-write in case of multi GPU sub-aperture processing) which is long compared to the last processing stage 4 (illumination post-compensation & reformatting from GPU complex interlacing to native CPU complex), signal vector-blocks for the next processing block are read and preprocessed up to range & azimuth migrations (a kind of "cycle steal" by stage 1 during stage 4). Of course, all this requires tedious synchronisations and double-buffering that are beyond the scope of this paper.

Fig. 11 is an example of high resolution image (3 dB azimuth & range resolutions are both 11 cm). It has been processed in 4 min 30 s on an Nvidia M2050 GPU (448 elementary processors) with the ω-k algorithm (the only one programmed to date for GPU) using 2843 Mb of video RAM (out of 3072 Mb available)[2]. The same processing takes 24 min on two Xeon X5650 2.67 GHz (12 processor cores) using 3692 Mb RAM (for 64 Gb available, NTP version of the program is less optimised for memory usage, hence the larger RAM footprint). Restricting to a single processor core, the processing takes 3 h 47 min using 3672 Mb RAM (this show the scalability is not perfect, probably because of the insufficient size of the L2 and L3 caches that are shared by the 6 CPU cores of each die).

For comparison, time-domain processing using 2-stage factored back-projection algorithm on 12 processor cores takes 1 h 59 min using 1599 Mb RAM, and unfactored (plain) back-projection algorithm on 12 processor cores takes 23 h 46 min using 2449 Mb RAM.

VII. Conclusion & perspectives

The motion-compensated ω-k algorithm described above allows focusing both wide band and wide aperture airborne SAR acquisition with a deterministic motion compensation which preserves image geometry, radiometry and phase, while keeping a $O(r^2 \log(r))$ complexity. This makes it appropriate for demanding SAR applications as repeat-pass interferometry, coherent change detection or polarimetric analysis.

It can be further enhanced: For example, it is possible to have a full polarimetric processing even when real antenna radiates imperfect polarization and aircraft attitude fluctuations rotate the polarization axes during integration. This could be performed by parallel processing of the four polarization channels with a recombination of the four channels for restoring the linear orthogonal basis for each range gate & azimuth sub-block during the local quadratic phase compensation.

It is also possible to increase the compensatable trajectory deviation (N to T baseline) by using a blockwise nominal trajectory, but this imposes an uneven resampling of the processing block image lines to map them in the final (global) image coordinates. This could be necessary for real-time processing, because the nominal trajectory would be the planned one and not a linear fit of the final true trajectory as in the "off-line" case. Such a resampling is also needed for bistatic SAR processing, as the image coordinates in which the signal frequencies correspond to image frequencies are no longer the cylindrical coordinates.

Adapting the algorithm to time-varying waveforms such as orthogonal frequencies modulation (OFDM) for using the transmitted signal as a downlink or for low probability of interception (LPI) still raises difficult problems in the coupling of range compression and Doppler effect along the wide integration angle.

Appendix A
nominal processing derivation

First, (2b) is derived: For a sample of the signal spectrum of given signal fast-time frequency of f and slow-time (*i.e.*

[2]note: mocomp parameters are processed in 9 s on one CPU core, and the 4 CPU threads used for pipelining, disk transfers & GPU control consume around 30% of one core capability and 40 Mb of CPU RAM.

Fig. 11. High resolution acquisition example: X-band acquisition of downtown Toulouse from a C160-Transall aircraft. Bandwidth is 1.224 GHz (5 successive frequency agilities covering 240 MHz each), duration is 120 s, signal size (single channel 8 bits) is 34.3 Gb, full resolution swath is 1200 m (slant), stripe length (for square resolution) is 9300 m.

Doppler) frequency $\frac{2f}{c}\cos(\delta)$, we need the corresponding illumination for a one-way trip at $\frac{c}{2}$ (which correspond to the imaging factor for the back-and-forth propagation at c).

We shall later correct for propagation and antenna pattern, hence we first derive the image *vs* signal spectrum amplitude due to SAR geometry alone. Note also that c is the radiowave celerity in the atmosphere during acquisition, which is derived from air pressure, temperature & dew point, information that is mandatory for the pilot. In our software, it is averaged between the flight altitude and the ground by integrating the 1976 standard atmospheric model.

Consider a given point F at range R in the direction squinted by angle δ from an acquisition point N taken as origin of phase here (we shall later change that while deriving (2a)). Image value at F is obtained from signal acquired from other acquisition points on the nominal trajectory around N (the integration interval). Let denote s the algebraic distance of the generic acquisition point from N. The phase (single trip illumination at $c/2$) from this generic acquisition point is:

$$\psi(s) = \frac{-4\pi f}{c}\sqrt{\left(R\cos(\delta)-s\right)^2+\left(R\sin(\delta)\right)^2} + \frac{-4\pi f}{c}\cos(\delta)s \quad (5)$$

The first term being the contribution from the difference in propagation and the second the contribution of slow-time (Doppler) frequency. Thus the cumulative illumination (or equivalently back-and-forth at c image contribution) can be expressed as:

$$\sigma \cong \frac{1}{\Delta z_s}\int_{\Sigma}\frac{p(s)}{(R\cos(\delta)-ds)^2+(R\sin(\delta))^2}e^{j\psi(s)}ds \quad (6)$$

where Δz_s is the signal slow time sampling along trajectory (for illumination is in fact a discrete sum, not an integral), Σ is the signal support (*i.e.* the processing block), $p(s)$ is the two-way antenna pattern factor and the R^{-2} factor the propagation gain.

Here $\psi'(0)=0$ (hence 0 is a stationary phase point) and $\psi''(0)=\frac{-4\pi f(\sin(\delta))^2}{cR}$ hence the illumination can be further approximated (thanks to the principle of stationary phase[7]) as:

$$\sigma \cong \frac{p(0)e^{j\psi(0)}}{\Delta z_s\sin(\delta)}\sqrt{\frac{cR}{2f}}e^{-j\frac{\pi}{4}} \quad (7)$$

Hence (2b) (if we assume processing gain compensated for the center of the processing block). The $-\pi/4$ extra phase term shows there is a constant phase offset for σ from the phase $\phi(0)$ (value of (5) at $s=0$) which correspond to the direct N to F propagation (and this justify the geometrical derivations from sections II & IV).

In practice, range compression is centered at a middle swath range R_c, the range frequency is cancelled for the middle frequency f_c and the Doppler frequency offset by $\frac{2f_c}{c}\cos(\delta_c)$ with origin at the center N_c of the processing block. This centering is important for easing the complex interpolations both in time and frequency domain (i.e. both image and spectrum are zero-centered). The range compression centering at range R_c changes the phase $\phi(0)$ at the processing block center N_c to $\frac{4\pi f}{c}R_c$. At this point the phase is that of the transmitted signal and does not depend on δ. The translation of the time domain origin of the processing bloc to range R_c in squint direction δ_c (or $(R_c\cos(\delta_c), R_c\sin(\delta_c))$ in Cartesian –unsquinted– (x,y) image coordinates) changes the phase to $\frac{4\pi f}{c}R_c(1-\cos(\delta_c)\cos(\delta)-\sin(\delta_c)\sin(\delta))$ since $k_x=\frac{4\pi f}{c}\cos(\delta)$ and $k_y=\frac{4\pi f}{c}\sin(\delta)$, hence (2a) when the $\pi/4$ constant phase offset is added.

APPENDIX B
AZIMUTH MIGRATION DETAILS

Prior to this computation, sensor trajectory and DEM are converted to Cartesian coordinates centered at an origin point (generally the zero altitude point below the acquisition start point) with axes corresponding to the East, North and zenith directions at the origin point. The nominal trajectory is defined as a straight line from the point at a given altitude above origin point with a given heading and slope (measured in the Cartesian coordinate, i.e. at the origin point). DEM is projected in the final image coordinates as described in section IV.

For any azimuth z (measured from origin along the nominal trajectory) and nominal range R, the ground altitude H of the mid-integration focusing point F is read in the projected DEM. Cartesian coordinates of F are given by:

$$\begin{aligned} x_F &= z\sin(\eta)\cos(\rho)+d_1\sin(\eta)\pm d_2\cos(\eta)\\ y_F &= z\cos(\eta)\cos(\rho)+d_1\cos(\eta)\mp d_2\sin(\eta)\\ z_F &= \begin{vmatrix} H-H_{org} & \text{if } d_2>0\\ z\sin(\rho)+R\sin(\rho-\delta_c) & \text{if } d_2=0\end{vmatrix}\end{aligned} \quad (8)$$

with H_{org}, η & ρ respectively the nominal trajectory origin altitude, heading & slope. and

$$\begin{aligned} d_1 &= \begin{vmatrix} R+z\tan(\rho) & \text{if } d_2>0\\ R\cos(\rho-\delta_c) & \text{if } d_2=0\end{vmatrix}\\ d_2 &= \begin{vmatrix} \sqrt{R^2-z^2-d_1^2} & \text{if } R^2-z^2-d_1^2>0\\ 0 & \text{if } R^2-z^2-d_1^2\le 0\end{vmatrix}\end{aligned} \quad (9)$$

Upper (resp. lower) signs of $\pm$ & $\mp$ in (9) are used when imaging to the right (resp. left) side. The lower condition in (10) corresponds to imaging below the track (above ground).

From F coordinates, the acquisition time τ for the migrated cell for range R and azimuth z is derived from an iterative search. Given two adjacent sample times τ_{n-1} and τ_n, the barycentric coordinate w of the nearest angle to F from N on the straight line $T_{\tau_{n-1}}T_{\tau_n}$ is given by the elementary algebraic expression:

$$w = \frac{\left(\widehat{FN} - \widehat{FT_{\tau_{n-1}}}\right)^T \left(\widehat{FT_{\tau_n}} - \widehat{FT_{\tau_{n-1}}}\right)}{\left\|\widehat{FT_{\tau_n}} - \widehat{FT_{\tau_{n-1}}}\right\|^2} \qquad (10)$$

with $\widehat{\mathbf{V}}$ denoting normalized vector $\frac{\mathbf{V}}{\|\mathbf{V}\|}$. When $0 \leq w \leq 1$ the acquisition time is $\tau = w\tau_n + (1-w)\tau_{n-1}$, when $w < 0$ (resp. $w > 1$) we consider the previous (resp. next) sampling time $\tau_{n+1} = \min(\tau_{n-1}, tau_n) - \delta\tau$ (resp. $\tau_{n+1} = \max(\tau_{n-1}, tau_n) + \delta\tau$) and reiterate except in the case $\tau_{n+1} = \tau_{n-1}$ in which case the acquisition time is $\tau = \tau_n$.

This algorithm seems quite rudimentary, however, because of the continuity of the azimuth migration, when initialized from the previous azimuth $z - \delta z$ at the same range R by the two sampling times around $\tau|_{z-\delta z} + \delta z/v$ (v being the sensor velocity along track at $\tau|_{z-\delta z}$) it typically converges in a couple of iterations.

This time of acquisition τ yields both range migration $\Delta R = \|FT_\tau\| - R$ and azimuth migration $\Delta z = z(T_\tau) - z$.

APPENDIX C
QUADRATIC PHASE DETAILS

Consider the sample for spectrum point i (at squint angle δ_i) of the quadratic phase for the sub-block j (of azimuth center z_j): The conical wavefront at squint angle δ_i for sample with range R and azimuth z_j is parametrised in cylindrical $(\mathfrak{z}, \mathfrak{r})$ coordinates as:

$$\mathfrak{r}(\mathfrak{z}) = -\tan(\delta_i)\mathfrak{z} + \left(\frac{R}{\cos(\delta_i)} + \tan(\delta_i) z_j\right) \qquad (11)$$

with $\mathfrak{z}$ the azimuth and $\mathfrak{r}$ the range to nominal trajectory. Those are non-squinted coordinates linked to final image (r, z) coordinates by:

$$\begin{aligned} r &= \frac{\mathfrak{r}}{\cos(\delta_c)} \\ z &= \mathfrak{z} - r\sin(\delta_c) \end{aligned} \qquad (12)$$

From (12) the focusing elevation is retrieved from the projected DEM, for each intersection of the wavefront with the rows of the projected DEM. (11) provides the other two coordinates and (9) allows to derive the 3D Cartesian coordinates of the intersections of the wavefront with the rows of the projected DEM (i.e. determines the vertices of the 3D broken line wavefront).

Wavefront point M closest to T_τ is first bracketed by stepping through the intersections with DEM rows (vertices) and comparing range to T_τ of intersections with adjacent rows. Then a Fibonacci search is applied between the two adjacent DEM rows yielding the closest point of the wavefront, and the corresponding quadratic phase point is given by $\|MT_\tau\| - \|FT_\tau\|$ in distance (it is later converted into a phase).

As mentioned in subsection IV-B the squint angle δ_i for which the quadratic phase is evaluated are such that the δ_i span the integration interval and $\sin(\delta_i)$ is aligned with sub-block's frequency domain azimuth samples. Thinner grid may be evaluated in order to have a better estimate of the block average quadratic phase (which is applied on the full block in frequency domain, thus without narrow band condition). The local (i.e. at range R in the sub-block of center z) quadratic phases distance is converted to phase at a frequency as illustrated in Fig. 10 and with distance offset by the global (block average) quadratic phase distance itself converted with compensation of range aperture migration (subsection IV-C). The local quadratic phase difference is applied in range Doppler domain (and here under narrow band assumption), but the fact that the block average quadratic phase is applied under wide band conditions allows larger deviation of the sensor trajectory from the nominal one.

The latter point is critical when two SAR images are synthesized with a common nominal trajectory, thus allowing direct interferogram computation with the images. The range aperture migration issue translates directly into critical baseline issue in that case.

Note that it is not mandatory to synthesize both images with the same nominal trajectory for generating an interferogram because with a deterministic motion compensation we also have a deterministic determination of the space frequency envelop for each image, hence it is possible to coherently resample them (e.g. resample the slave image in the master image coordinate and then compute the interferogram).

The sub-block zero-padding in Fig. 8 is decisive in applying the "quadratic" phase correction: Indeed the effect of such a phase is to smear the sub-block in azimuth exactly as a velocity bias smears (unfocuses) the point echoes in a SAR images along azimuth axis. Hence, the sub-block must be padded with margin in which it could smear (instead of wrapping in azimuth, with disastrous consequences on the final image). The pre-compensated signal is eventually obtained by adding the overlapping padded sub-blocks.

The azimuth smearing of the sub-blocks has however a paradoxical positive impact: If the quadratic phase changes between sub-blocks, then the quadratic phase difference applied on each sub-block differ, but the phase step between sub-blocks is smoothed by the azimuth smear even when the sub-blocks are not overlapping. The 50% sub-block overlap is the default option of the software, but even if disabled, sub-blocking artefacts do not appear in the point echo image spectrum (aperture).

ACKNOWLEDGEMENT

Authors acknowledge Laurette Pastoré (Renault vehicles corp.) incitations to use the ω-k algorithm during her Ph.D thesis, Faustine Cantalloube (Institut d'Optique graduate school) for demonstrating the usefulness of GPU for radio (astronomical) signal processing, Jean-François Degurse (ONERA) for his kind help in GPU programming & Bernard Vaizan (ON-

ERA) for the intensive error checking and help in clarifying the dissertation.

REFERENCES

[1] R.H. Stolt, "Migration by Fourier transform," *Geophysics*, vol. 43, no. 1 pp. 23-48, Feb, 1978.

[2] E. Felstead, "Optical range-curvature correction for synthetic-aperture radar (SAR) signals," *Int. Optical Computing conf.*, Washington, DC, Apr, 1980, vol. 231 pp. 298-305

[3] A.S. Milman, "SAR imaging by omega-kappa migration," *Int. J. Remote Sensing*, vol. 14, no. 10 pp. 1965-1979, Jul, 1993.

[4] A. Moreira, Yonghong Huang, "Airborne SAR processing of highly squinted data using a chirp scaling approach with integrated motion compensation," *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 32, no. 5 pp.1029-1040, Sep, 1994.

[5] C. Yerkes and E. Webster, "Implementation of omega-k synthetic-aperture radar imaging algorithm on massively-parallel supercomputer," *proc. SPIE*, vol. 2230 pp. 171-178, 1994.

[6] J. McCorkle and M.Rofheart, "Order N^2 log(N) backprojector algorithm for focusing wide-angle wide-bandwidth arbitrary-motion synthetic aperture radar," *Proc. SPIE*, vol. 2747 pp. 25-36, 1996.

[7] R. Soumekh, *Synthetic Aperture Radar Processing*, R.Wiley-interscience, 1999.

[8] A. Brenner, "Distributed SAR processing in the time-domain," *EUSAR proc.*, Cologne, Germany, Jun, 2002, pp. 573-576.

[9] I. Cumming , Y.L. Neo and F. Wong,"Interpretation of the omega-K algorithm and comparisons with other algorithms," *Int. Geoscience and Remote Sensing Symp. (IGARSS)*, Toulouse, France, 2003, pp. 1455-1458.

[10] Cantalloube, H., Dubois-Fernandez, P., "Airborne X-band SAR imaging with 10 cm resolution - Technical challenge and preliminary results," *IEEE IGARSS*, Toulouse, France, vol: 1-7 p 185-1872, Jul, 2003.

[11] L. Ulander, H.Hellsten and G. Stenstrom, "Synthetic aperture radar processing using fast factorized back-projection," *IEEE trans. Aerosp. Electron. Syst.*, vol. 39, no. 3 pp. 760-776, Jul, 2003.

[12] I. Cumming and F. Wong, *Digital Processing of Synthetic Aperture Radar Data.* Boston MA: Artech House, ch. 8, pp. 323-367, 2005.

[13] M. Blom and P. Follo, "VHFSAR image formation implemented on a GPU," *Int. Geosc. & Remote Sensing Symp. (IGARSS)*, Seoul, Korea, Jul, 2005.

[14] I. Walterscheid, J. Ender, A. Brenner, and O. Loffeld, "Bistatic SAR processing and experiments," IEEE Transactions on Geoscience and Remote Sensing, vol. 44, no. 10, pp. 2710-2717, October 2006.

[15] C. Clemente, M. di Bisceglie, M. Di Santo, N. Ranaldo and M. Spinelli, "Processing of synthetic aperture radar data with GPGPU," *IEEE Workshop on Signal Processing Syst.*, Tampere, Finland, Oct, 2009.

[16] Backues, M., Majumder, U. "Embedded Gotcha Radar Exploitation with GPU-Based Backprojection Image Formation," AFRL/RYAS In-house Research, 2009.

[17] C. Nahum and H. Cantalloube, "SAR Image Synthesis Algorithm Improvement on Multi-processor/Multi-core Computers : vectoring on massively parallel processors," *Int. Remote Sensing conf*, Leipzig, Germany, Sep, 2011.

[18] M. Rodriguez-Cassola, P. Prats,G. Krieger et al., "Efficient time-domain image formation with precise topography accommodation for general bistatic SAR configurations," *IEEE trans. on aerosp. & electron. syst.*, vol. 47, no. 4, pp. 2949-2966, Oct, 2011.

[19] Linderman, R. "Architectural considerations for a 500 Tflops heterogeneous HPC," *IEEE Int. Parallel & Distributed Proc. Symp.*, Atlanta, Georgia, Apr, 2012.

[20] Dubois-Fernandez, P. at al. "The Sethi remote sensing airborne platform and the related science activities," Rev. fr. photogramm. télédétect., vol. 200, pp. 36-47, 2012.

Hubert M.J. Cantalloube was born in Bourges, France, in 1962. While student in mathematics from École Normale Supérieure of Paris in 1982-1987, he graduated in computer science and signal processing from the University of Paris VI, receiving the Lic.es Sci. and Mai.es Sci. degrees in 1983, the DEA degree (equivalent of M.S.) in 1984. He received the Ph.D. degree in electrical engineering from the McGill University of Montreal in 1989.

He served (interrupting his Ph.D studies) at the Embassy in Ottawa, then, at the end of Ph.D, joined the Office National d'Études Aérospatiales as a scientist in the image processing department, and was mostly involved with optical satellite remote sensing and infra-red missile navigation and homing systems. In parallel, he contributed to researches in computer science for biology. In 1997, he moved to the Electromagnetics and radar department in Palaiseau, France. His present interests are in the field of signal processing for synthetic aperture radar.

Carole E. Nahum was born in Carthage, Tunisia in 1960. She graduated from the ENSI in Caen France, receiving the Dipl.Ing. degree in 1983. She received the M.S. degree from the École Nationale des Télécommunications of Paris in 1984 with a major in signal processing. She received the Ph.D. degree in mathematics from McGill University of Montreal in 1989.

She joined the Office National d'Études Aérospatiales (the French Aerospace Laboratory) as a scientist in 1989 where her principal research first dealt with fighter aircraft flight control, air defense related radar tracking issues, and more recently synthetic aperture radar image synthesis and fusion. In 2001, she joined the Délégation Générale de l'Armement (the French MoD procurement directorate) in Bagneux as a Scientific Expert for electromagnetic detection and image processing. She is in charge of research orientation, proposal assessment, funding and scientific evaluation of exploratory programs. She also supervises several PhD thesis and is involved in several European research projects.