Processing Fully-Focussed SAR with GPUs

Gorka Moyano[1], Enrique Monzó[1], Sergi Hernández[1], Albert Garcia-Mondéjar[1], Mònica Roca i Aparici[1], Marco Fornari[2], Rob Cullen[3]
[1]isardSAT, [2]RHEA/ESA-ESTEC, [3]ESA-ESTEC | Correspondence: Gorka.Moyano@isardSAT.cat

isardSAT
esa

ABSTRACT

The Fully-Focussed (FF-SAR) algorithm for nadir radar altimeters [1] allows to achieve along-track resolution close to the theoretical limits, which is about 0.5 m for typical nadir altimeters such as Cryosat-2, Sentinel-3 A/B, and Sentinel-6. Such improvement with respect to the roughly 300 m-scale resolution of classic Delay/Doppler (DDP) algorithms has enabled the development of new applications over a wide variety of scenarios, such as inland water level and water extent retrievals, sea ice, and ocean swell studies.

FF-SAR algorithms can be implemented following two approaches: time domain (backprojection, BP) and frequency domain. The back-projection approach is more accurate than the frequency one because it relies on less assumptions, but the computational time associated to its classic implementation is too high for envisaging an operational implementation. To improve the poor runtime performance, the processor code has been rewritten to work on GPU devices, improving this performance by a factor around 100 using commercial, powerful GPUs, when compared with the CPU-based processor.

This new implementation of the FF-SAR BP processor to work with GPUs is a code that mixes C++ and CUDA programming language (an extension of C language specially designed to work with GPU devices), unlike the CPU-based processor, which was totally implemented in C++. The new CUDA code is entirely tailored to follow the needs of the FF-SAR BP for Sentinel-6 MF, and will take care of all big matrix operations (the ones that take most of the time), so they are performed in the GPU in an efficient and optimal way.

With this new implementation, we achieve performances relative to the input data sensing time around a factor of 5 - depending on the configuration and taking into account other accelerations of the BP algorithm-, leaving it close to a near-real time performance, which could be then achieved using several GPU devices in parallel.

The new GPU processor has been verified to provide the same results as the CPU-based processor, in order to assess differences in the precision of the results.

FF-SAR BP PROCESSORS FOR SENTINEL-6

isardSAT is in charge of the design and implementation of the **Sentinel-6 Poseidon-4 Level-1 Ground Prototype Processor (S6 P4 L1 GPP)**, with ESA. Within this project, we developed the **Level-1 FF-SAR Back-Projection processor**, which provides

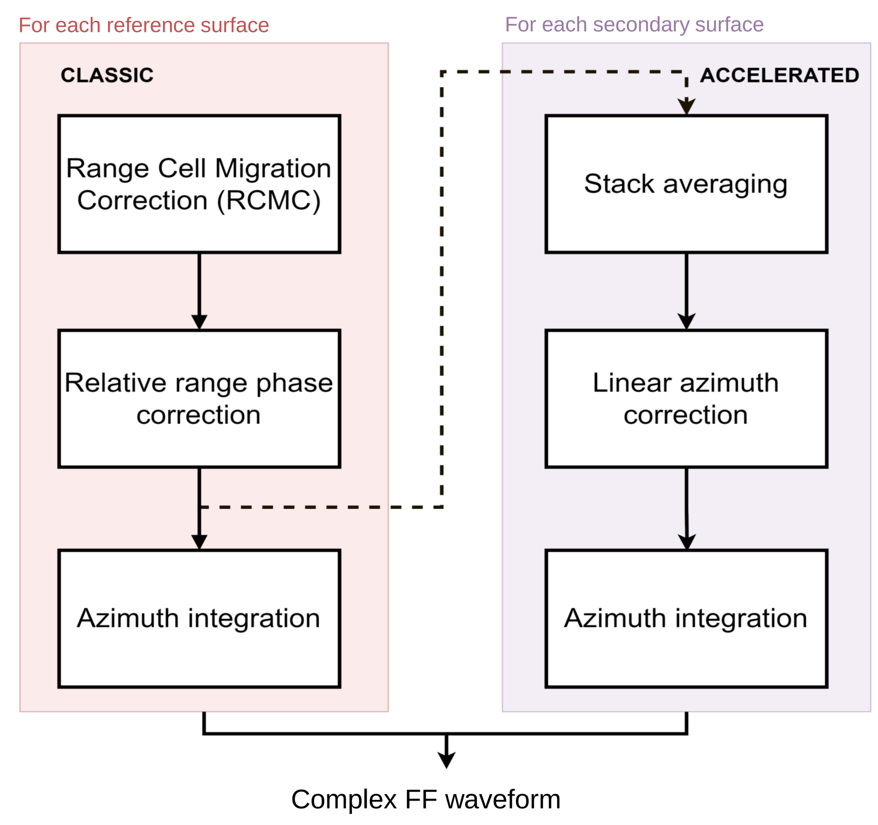
- the maximum theoretical along-track resolution
  - this opens the door to new applications where the Delay/Doppler resolution (~300 m) is insufficient, such as swell, inland waters, sea-ice leads…

However,

- the run-time performance of this processor is very poor (even with the utilisation of multi-threading)
  - the processing of a L1A granule of 10 minutes would take:
    - using a 1 thread: ~12 days
    - using 40 threads: ~7 hours

The following two points have been implemented to improve the run-time performance (without damaging the scientific performance):

1. **Processing using GPU** (elaborated in the sections below)
2. **Accelerations in the FF-SAR BP algorithms** [see Poster #247: "An Accelerated Processor for the Fully-Focussed SAR Back-projection Algorithm", 2]



DEVELOPMENT FOR GPUs

The **L1 FF-SAR BP processor**, within the S6 P4 L1 GPP,

- is coded in C++
- it operates with big matrices (of sizes around 20000 x 512 double-precision elements)

⇒ making it ideal for GPU processing.

isardSAT implemented a version of the **L1 FF-SAR BP processor**, mixing C++ and CUDA languages, so it works under NVIDIA GPUs. Our goals were to have it:

- Compatible with all NVIDIA architectures
- Fully scalable (number of cores)
- Working with double-precision

⇒ so we could use the processor in different environments, with GPUs of different characteristics, and without losing precision in the results.

The GPU parameters of interest are:

- **FP64 (double) performance** – the speed the GPU executes double-precision operations
- **Memory size** – the amount of data the GPU can store
- **Memory bandwidth** – the amount of data the GPU can access per clock cycle
- **Bus interface** – CPU-GPU memory transfer
- **Price** – how many we can afford

Our **L1 FF-SAR BP processor** does not need much data to work, so the main parameters would be the **FP64 performance** and the **price**.

The different GPUs we have used in our tests are:

GPU	FP64 performance [GFLOPS]	Memory size [GB]	Memory bandwidth [GB/s]	Bus interface	Price
NVIDIA GeForce GTX 1070 Mobile	202	8	256.3	PCIe 3.0 x16	220 €
NVIDIA Titan RTX	509.8	24	672	PCIe 3.0 x16	1750 €
NVIDIA A30	5161	24	933.1	PCIe 4.0 x16	5700 €

Check out the first cycle of L1 FF-SAR BP data (processed using GPU and algorithm accelerations) and L1 FF-SAR WK data

RUN-TIME PERFORMANCE COMPARISON

The same L1A granule of 10 minutes is processed using the **L1 FF-SAR BP processor**, single thread, same configuration, with the different GPUs:

Type of L1 FF-SAR processing	GPU used	Processing time	Processing ratio (processing time / data length)	Ratio wrt REFERENCE
Classic L1 FF-SAR BP (w/o algorithm accelerations)	No	12 days	1728.00	0.15
	NVIDIA GeForce GTX 1070 Mobile	10 hours	60.00	4.40
L1 FF-SAR BP with algorithm accelerations	No	44 hours	264.00	1.00 *[REFERENCE]*
	NVIDIA GeForce GTX 1070 Mobile	**36 minutes**	3.60	73.30
	NVIDIA Titan RTX	**24.5 minutes**	2.45	107.80
	NVIDIA A30	**19 minutes**	1.90	138.90

The results using the L1 FF-SAR BP with algorithm accelerations and processed in GPU are **very good** [see Poster 30YPRA #251: "A Performance Analysis of Fully Focussed SAR Processors: from Classic and GPU-accelerated Backprojection to Numerical and Closed-form Omega-K Algorithms", 3].

CONCLUSIONS

- The L1 FF-SAR BP processing using GPU in combination with the BP algorithm accelerations is a fantastic choice in order to have a very good quality of data close to Near Real Time.
- There is room for improving the run-time performance (with the same GPU and algorithms), e.g. by using a second thread to do the read/write to disk.
- Having more than one GPUs would scale the run-time performance linearly, e.g. two NVIDIA Titan RTX would achieve a processing ratio of 1.225, i.e., 10 minutes of data processed in 12.25 minutes.

REFERENCES

[1] A. Egido and W. H. F. Smith, "Fully Focused SAR Altimetry: Theory and Applications," IEEE, vol. 55, no. 1, pp. 392–406, 2017.
[2] P. Villalvilla et al., "An Accelerated Processor for the Fully-Focussed SAR Back-projection Algorithm." 30 Years of Progress in Radar Altimetry Symposium, 2-7 September 2024, Montpellier. Poster #247 / Abstract #360.
[3] S. Hernández et al., "A Performance Analysis of Fully Focussed SAR Processors: from Classic and GPU-accelerated Backprojection to Numerical and Closed-form Omega-K Algorithms". 30 Years of Progress in Radar Altimetry Symposium, 2-7 September 2024, Montpellier. Poster #251 / Abstract #379.