Finanziato dall'Unione europea
NextGenerationEU

Ministero dell'Università e della Ricerca

Italiadomani
PIANO NAZIONALE DI RIPRESA E RESILIENZA

INAF
ISTITUTO NAZIONALE DI ASTROFISICA

REAL-TIME IMPLEMENTATION OF HEIMDALL DETECTION PIPELINE FOR FRB AND VALIDATION AGAINST OFFLINE MODE

Final Meeting

Martedì 30 Giugno – Giovedì 02 Luglio

OAC - Osservatorio Astrofisico di Arcetri

PNRR NG-CROCE

Valentina Cesare

Table of contents

1. Objectives

Data rates from current and future radiotelescopes are increasingly larger (e.g., scientific data rate from ASKAP = 75 PB/year; Chapman, 2013. ➡ SKA = 700 PB/year; Hartley, 2023).

REQUIRED **GPU COMPUTATION** ON **HIGH PERFORMANCE COMPUTING (HPC)** INFRASTRUCTURES + CHANGE OF PARADIGM OF **DATA MANAGEMENT**:

OFFLINE ANALYSIS = DATA SAVED TO DISK

- I/O saturation;
- Storage issues;
- Possible loss of scientific informations for (sub-)ms events, e.g., radiotransients, and difficulty in their follow-up observations in radio and multi-messenger perspectives.

REAL-TIME ANALYSIS = CONTINUOUS DATA FLOW IN RAM MEMORY

- Keep up with data acquisition rates;
- Reduction of disk space and throughput needs;
- Access to raw and not compressed data: scientific information preserved;
- Immediate feedback on Radio Frequency Interference;
- Automatic alerts.

Finanziato dall'Unione europea NextGenerationEU
Ministero dell'Università e della Ricerca
Italiadomani PIANO NAZIONALE DI RIPRESA E RESILIENZA
INAF ISTITUTO NAZIONALE DI ASTROFISICA

1. Objectives

- One of the targets of Nothern Cross Radiotelescope is the detection of radiotransents:
  - ❖ Pulsars (Hewish, et al. 1968)
  - ❖ Rotating RAdio Transients (RRATs) (Keane & McLaughlin, 2011)
  - ❖ **Fast Radio Bursts (FRB)** (Lorimer, et al, 2007) ➡ **Greatest challenge:** time scales [sub-ms – days]; rare events; tendency not to repeat; random directions; sources difficult to locate.

Real-Time analysis essential to prevent loosing important scientific information!!!

- Dispersion Measure (DM) has to be kept into account! A radio signal from a source propagating to Earth interacts with free electrons in the interstellar medium, producing a frequency-dependent time delay Δt (Barsdell, et al., 2012):

$$\Delta t = k_{\mathrm{DM}} \mathrm{DM}(\nu_1^{-2} - \nu_2^{-2})$$

The DM along the line of sight in $\mathrm{pc/cm^3}$ is:

$$\mathrm{DM} \equiv \int_0^d n_{\mathrm{e}} dl$$

n_{e} = Election number density [cm^{-3}]; d = Distance to the source [pc] → not known: search over a large DM trial grid **(algorithm with high numerical parallelism degree)**.

Finanziato dall'Unione europea NextGenerationEU
Ministero dell'Università e della Ricerca
Italiadomani PIANO NAZIONALE DI RIPRESA E RESILIENZA
INAF ISTITUTO NAZIONALE DI ASTROFISICA

2. The Employed Pipeline

Objective: obtain a compact pipeline to efficiently manage data in the real-time flow with minimum HPC resources consumption.

1) Raw data flow management
- Data-Rates >
- Low-level Hardware (FPGA)

2) Scientific data flow management
- Data-Rates <
- High-level Hardware (GPU)

FPGA boards + FPGA

Radio Antennas

Analogue Front End + Low Noise Amplifier

Pre ADUs
ADU → Analogue to Digital Converter + Down Converter
Time aligning
Channelization or F-Engine
Preliminary RFI
Pachetization

Link aggregation for an efficient and scalable throughput → Network Switch

HPC cluster with GPU nodes

Beamformer (GPU)

Detection pipeline = Heimdall **(CPU + 1 GPU, single beam, open source, executed on container = portable)**

ML-based FRB classifier (CPU, executed on container = portable)

Finanziato dall'Unione europea NextGenerationEU
Ministero dell'Università e della Ricerca
Italiadomani PIANO NAZIONALE DI RIPRESA E RESILIENZA
INAF ISTITUTO NAZIONALE DI ASTROFISICA

2. The Employed Pipeline

Objective: obtain a compact pipeline to efficiently manage data in the real-time flow with minimum HPC resources consumption.

1) Raw data flow management

- **Data-Rates >**
- **Low-level Hardware (FPGA)**

2) Scientific data flow management

- **Data-Rates <**
- **High-level Hardware (GPU)**

FPGA boards + FPGA

Radio Antennas

Analogue Front End + Low Noise Amplifier

Pre ADUs
ADU → Analogue to Digital Converter + Down Converter
Time aligning
Channelization or F-Engine
Preliminary RFI
Pachetization

Link aggregation for an efficient and scalable throughput → Network Switch

MY WORK

HPC cluster with GPU nodes

Beamformer (GPU)

Detection pipeline = Heimdall **(CPU + 1 GPU, single beam, open source, executed on container = portable)**

ML-based FRB classifier (CPU, executed on container = portable)

Finanziato dall'Unione europea
NextGenerationEU
Ministero dell'Università e della Ricerca
Italiadomani
PIANO NAZIONALE DI RIPRESA E RESILIENZA
INAF
ISTITUTO NAZIONALE DI ASTROFISICA

OFFLINE MODE

Heimdall: https://sourceforge.net/projects/heimdall-astro/

1) Beamformer

Data output written to disk in SIGPROC filterbank (.fil) format.

2) Heimdall detection pipeline:

Loop of nsamps_gulp data elements per iteration

Data injestion — **CPU**

Host-To-Device

GPU MODULES

- RFI mitigation
- Dedispersion = Correction for dispersion
- Baseline removal
- Normalization
- Matched filtering
- Event detection = "Giants" finding
- Event merging

Device-To-Host

.cand files production (e.g., TOA, DM, SNR)

ML-based FRB classifier — **CPU**

Finanziato dall'Unione europea NextGenerationEU
Ministero dell'Università e della Ricerca
Italiadomani PIANO NAZIONALE DI RIPRESA E RESILIENZA
INAF ISTITUTO NAZIONALE DI ASTROFISICA

OFFLINE MODE

Heimdall: https://sourceforge.net/projects/heimdall-astro/

1) Beamformer

Data output written to disk in SIGPROC filterbank (.fil) format.

2) Heimdall detection pipeline:

Data injestion — **CPU**

Host-To-Device

GPU MODULES

- RFI mitigation
- Dedispersion = Correction for dispersion
- Baseline removal
- Normalization
- Matched filtering
- Event detection = “Giants” finding
- Event merging

Device-To-Host

.cand files production (e.g., TOA, DM, SNR)

Loop of nsamps_gulp data elements per iteration

ML-based FRB classifier — **CPU**

REAL-TIME MODE

1) Creation of PSRDADA Ring Buffer:

`dada_db -k dada -b <b> -n <n>`

<b> = size of a block: <n> number of blocks

PSRDADA: Standard format in radioastronomy.
⇓
PORTABILITY.

Finanziato dall'Unione europea NextGenerationEU

Ministero dell'Università e della Ricerca

Italiadomani PIANO NAZIONALE DI RIPRESA E RESILIENZA

INAF ISTITUTO NAZIONALE DI ASTROFISICA

OFFLINE MODE

Heimdall: https://sourceforge.net/projects/heimdall-astro/

1) Beamformer

Data output written to disk in SIGPROC filterbank (.fil) format.

2) Heimdall detection pipeline:

Loop of nsamps_gulp data elements per iteration

- Data injestion — CPU
- Host-To-Device
- GPU MODULES:
  - RFI mitigation
  - Dedispersion = Correction for dispersion
  - Baseline removal
  - Normalization
  - Matched filtering
  - Event detection = "Giants" finding
  - Event merging
- Device-To-Host
- .cand files production (e.g., TOA, DM, SNR)

ML-based FRB classifier — CPU

REAL-TIME MODE

1) Creation of PSRDADA Ring Buffer:

`dada_db -k dada -b <b> -n <n>`

<b> = size of a block: <n> number of blocks

PSRDADA: Standard format in radioastronomy.

⇓

PORTABILITY.

2) Heimdall detection pipeline = Consumer:

- **It waits for the data from the beamformer.**
- Same structure as the offline mode. **Only difference: DATA SOURCE.**

Heimdall: https://sourceforge.net/projects/heimdall-astro/

OFFLINE MODE

1) Beamformer:
Data output written to disk in SIGPROC filterbank (.fil) format.

2) Heimdall detection pipeline:

Loop of nsamps_gulp data elements per iteration

Data injestion — **CPU**

Host-To-Device

GPU MODULES
- RFI mitigation
- Dedispersion = Correction for dispersion
- Baseline removal
- Normalization
- Matched filtering
- Event detection = "Giants" finding
- Event merging

Device-To-Host

.cand files production (e.g., TOA, DM, SNR)

ML-based FRB classifier — **CPU**

REAL-TIME MODE

PSRDADA: Standard format in radioastronomy. ⇓ PORTABILITY.

1) Creation of PSRDADA Ring Buffer:
```
dada_db -k dada -b <b> -n <n>
```
<b> = size of a block: <n> number of blocks

Asynchronous processes

3) Beamformer = Producer:
Data output **written to RAM** in the PSRDADA ring buffer.

2) Heimdall detection pipeline = Consumer:
- **It waits for the data from the beamformer.**
- Same structure as the offline mode. **Only difference: DATA SOURCE.**

Finanziato dall'Unione europea NextGenerationEU
Ministero dell'Università e della Ricerca
Italiadomani PIANO NAZIONALE DI RIPRESA E RESILIENZA
INAF ISTITUTO NAZIONALE DI ASTROFISICA

Heimdall: https://sourceforge.net/projects/heimdall-astro/

OFFLINE MODE

Loop of nsamps_gulp data elements per iteration

1) Beamformer:
Data output written to disk in SIGPROC filterbank (.fil) format.

2) Heimdall detection pipeline:

Data injestion — **CPU**

Host-To-Device

GPU MODULES
- RFI mitigation
- Dedispersion = Correction for dispersion
- Baseline removal
- Normalization
- Matched filtering
- Event detection = "Giants" finding
- Event merging

Device-To-Host

.cand files production (e.g., TOA, DM, SNR)

ML-based FRB classifier — **CPU**

REAL-TIME MODE

PSRDADA: Standard format in radioastronomy. ⇓ PORTABILITY.

1) Creation of PSRDADA Ring Buffer:
`dada_db -k dada -b <b> -n <n>`
<b> = size of a block: <n> number of blocks

Asynchronous processes

3) Beamformer = Producer:
Data output **written to RAM** in the PSRDADA ring buffer.

2) Heimdall detection pipeline = Consumer:
- **It waits for the data from the beamformer.**
- Same structure as the offline mode. **Only difference: DATA SOURCE.**

ML-based FRB classifier — **CPU**

3. Testbed Setup – Hardware and Software Stack

Hardware

- 1 single node (med-node001)
- CPU: Dual-socket Intel Xeon Gold 5416S @ 2.0 GHz (Turbo up to 4.0 GHz), 32 cores / 64 threads total (16 cores per socket, Hyper-Threading enabled)
- RAM Memory: 2 TiB
- GPU: 2 x 46068 MiB VRAM GPU NVIDIA L40S. 92136 MiB total GPU memory of the node.

Software

- **GPU Parallelization frameworks:**
  - **Dedispersion:** dedisp library (Jameson & Barsdell, https://github.com/ajameson/dedisp).
  - **All the other modules:** Thrust C++ Template Library (Hoberock & Bell, 2010) (part of NVIDIA CUDA toolkit).
- Heimdall installed on **Docker container (portability + flexible management of data dependencies):**
  - Operating System: Rocky Linux 9.7
  - Prerequisites:
    - dedisp 1.1
    - PSRDADA 1.2.0
    - CUDA 12.8.1

3. Testbed setup – Tests and Objectives

TESTS OBJECTIVES

- **Results accuracy comparison (offline vs real-time)**
- **Performance comparison (offline vs real-time, preliminary assessment)**

Finanziato dall'Unione europea NextGenerationEU

Ministero dell'Università e della Ricerca

Italiadomani PIANO NAZIONALE DI RIPRESA E RESILIENZA

INAF ISTITUTO NAZIONALE DI ASTROFISICA

3. Testbed Setup – PSRDADA pipeline built for the tests

PSRDADA writer (sequential on CPU) instead of beamformer (GPU ported).
Compiled with:

```
gcc writer_p.c -o writer_p \\
-I/<PSRDADA_INST_PATH>/include \\
-L/<PSRDADA_INST_PATH>/lib \\
-lpsrdada
```

a) Reading data bytes stream of SIGPROC file in a buffer, bypassing SIGPROC file header.

b) Definition of PSRDADA Ring Buffer Header with a specific function.

c) Writing while loop:
The buffer is cyclically written in the ring buffer, after the header location, in chunks of size `CHUNK_SIZE (CS)`.

REAL-TIME MODE

1) Creation of PSRDADA Ring Buffer:

```
dada_db -k dada -b <b> -n <n>
```

`<b>` **= size of a block:** `<n>` **number of blocks**

Asynchronous processes

3) *WRITER* = Producer:
Data output **written to RAM** in the PSRDADA ring buffer.

2) Heimdall detection pipeline = Consumer:

- **It waits for the data from the beamformer.**
- Same structure as the offline mode. **Only difference: DATA SOURCE.**

ML-based FRB classifier | **CPU**

Finanziato dall'Unione europea NextGenerationEU
Ministero dell'Università e della Ricerca
Italiadomani PIANO NAZIONALE DI RIPRESA E RESILIENZA
INAF ISTITUTO NAZIONALE DI ASTROFISICA

4. Testbed Setup – SIGPROC pipeline built for the tests

OFFLINE MODE

1) Writer:
It writes an output file identical to an input SIGPROC filterbank (.fil) file.

Built for a fair performance comparision.

Loop of nsamps_gulp data elements per iteration

2) Heimdall detection pipeline:

- Data injestion — **CPU**
- **Host-To-Device**
- **GPU MODULES**:
  - RFI mitigation
  - Dedispersion = Correction for dispersion
  - Baseline removal
  - Normalization
  - Matched filtering
  - Event detection = "Giants" finding
  - Event merging
- **Device-To-Host**
- .cand files production (e.g., TOA, DM, SNR)

ML-based FRB classifier — **CPU**

3. Testbed setup – Employed Files and Parameter Space

- 2 classes of files:
    - 11 files with 12 injected FRB having known Time Of Arrival (TOA), DM, and SNR (**synthetic files**).
    - 11 files with 1 true FRB having known TOA and DM (**true files**).
- Default parameters combination:
    - `nsamps_gulp` = 2^{18} **(for offline and real-time)**
    - `<b>` = 128 MiB, `<n>` = 8, `CS` = 8 MiB **(only for real-time)**
- Exploring the parameter space for 1 synthetic file (n. 1) and 1 true file (n. 10):

Test N.	nsamps_gulp	<b> [MiB]	<n>	CHUNK_SIZE [MiB]
(1)	2^{18}	128	8	8
(2)	2^{19}	128	8	8
(3)	2^{20}	128	8	8
(4)	2^{21}	128	8	8
(5)	2^{21}	256	4	8
(6)	2^{21}	512	2	8
(7)	2^{18}	512	2	512
(8)	2^{21}	4096	2	4096

For both SIGPROC and PSRDADA

Only for PSRDADA

Constant size for the PSRDADA ring buffer.

`nsamps_gulp`, `<b>`, and `CS`, aligned.

- In each case: (1) Comparison of found results vs reference results; (2) Comparison of execution time vs acquisition time; (3) Comparison of execution time offline vs execution time real-time

Finanziato dall'Unione europea
NextGenerationEU

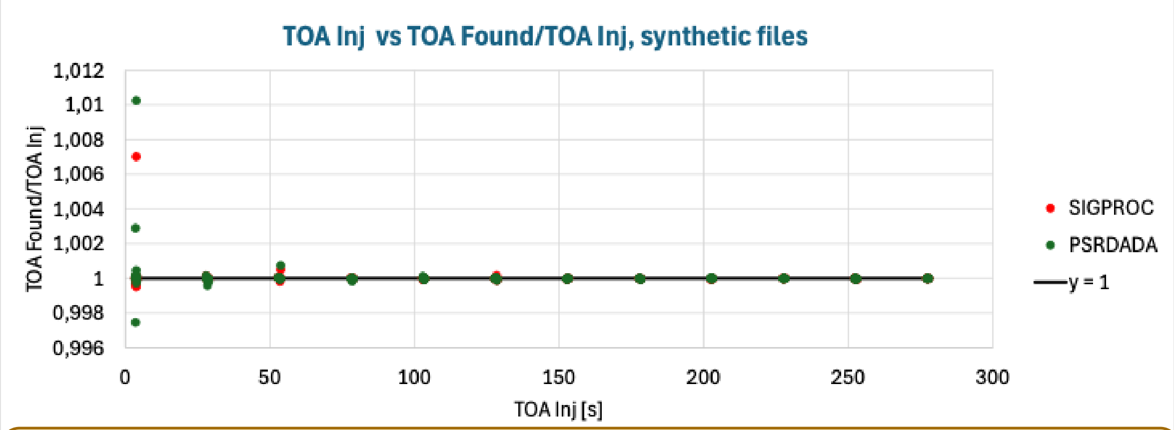
Ministero dell'Università e della Ricerca

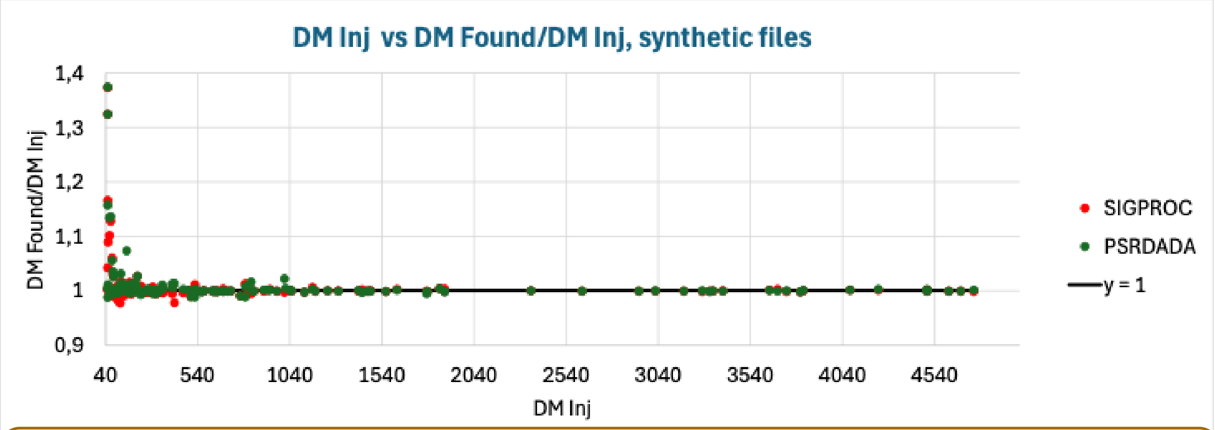
Italiadomani
PIANO NAZIONALE DI RIPRESA E RESILIENZA

INAF
ISTITUTO NAZIONALE DI ASTROFISICA

4. Results – Real-Time vs Offline validation – Synthetic files

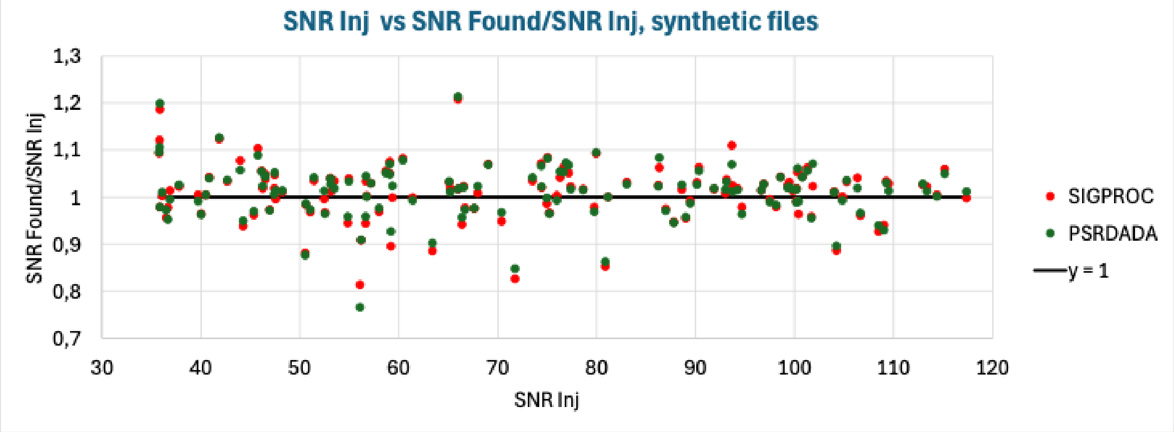
DEFAULT PARAMETERS





- TOA ratios ∈ [0.97,1.03] S: 100%; ∈ [0.995,1.005] S: 99.24%
- TOA ratios ∈ [0.97,1.03] P: 100%; ∈ [0.995,1.005] P: 99.24%

- DM ratios in [0.97, 1.03] S: 93.13%; ∈ [0.995,1.005] S: 74.05%
- DM ratios in [0.97, 1.03] P: 93.13%; ∈ [0.995,1.005] P: 74.81%

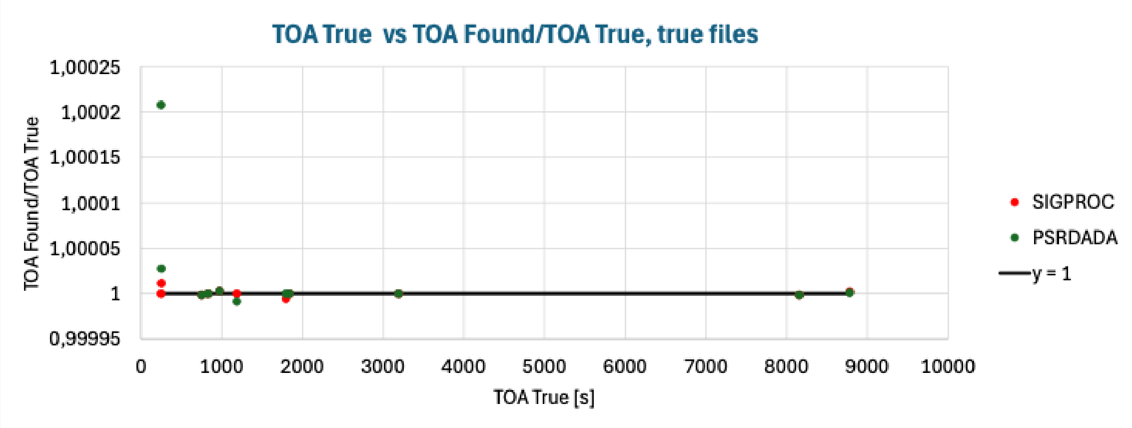


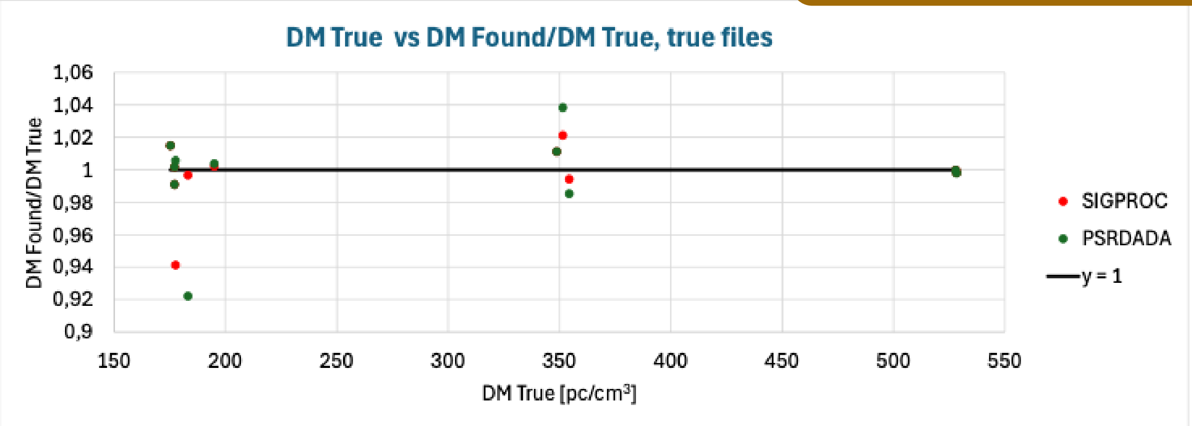
- SNR ratios in [0.97, 1.03] S: 43.51%; ∈ [0.9,1.1] S: 90.08%
- SNR ratios in [0.97, 1.03] P: 48.09%; ∈ [0.9,1.1] P: 93.13%

TOA Found/TOA Inj, DM Found/DM Inj, and SNR Found/SNR Inj distribute around 1, for both SIGPROC and PSRDADA.

DEFAULT PARAMETERS

4. Results – Real-Time vs Offline validation – True files





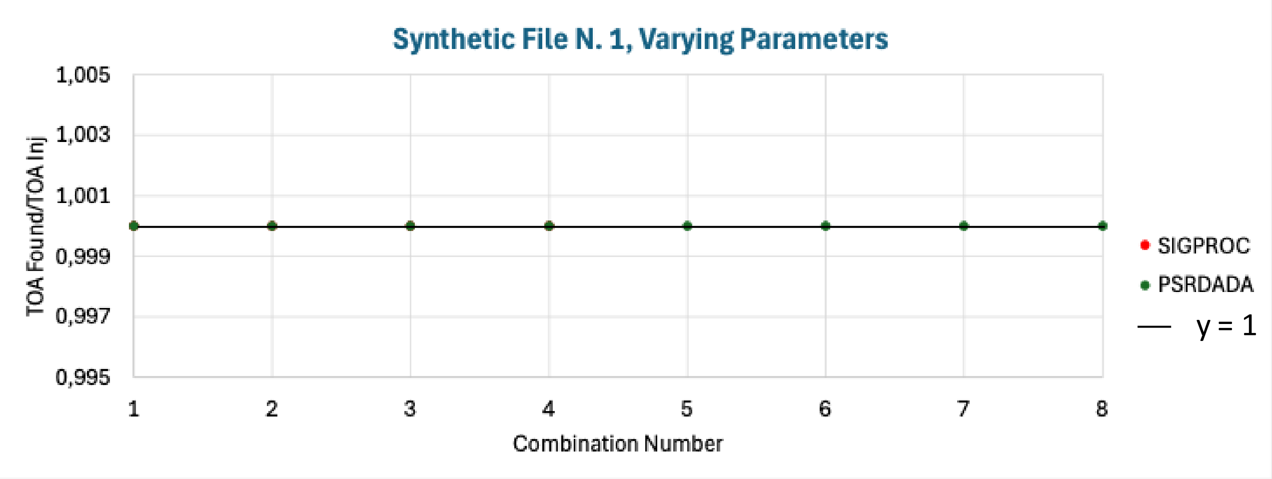
- TOA ratios ∈ [0.97,1.03] S: 100%; ∈ [0.995,1.005] S: 100%
- TOA ratios ∈ [0.97,1.03] P: 100%; ∈ [0.995,1.005] P: 100%

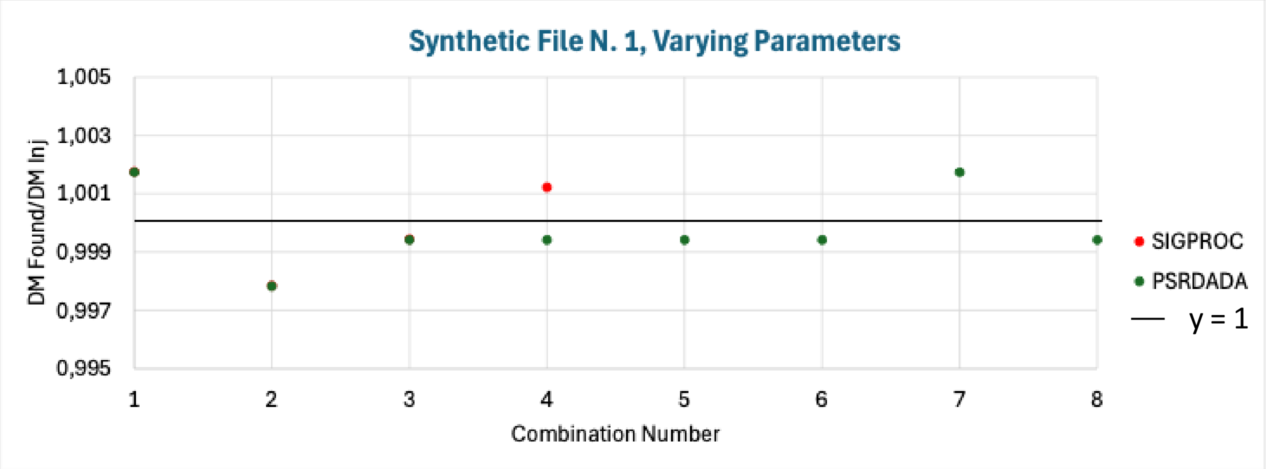
- DM ratios in [0.97, 1.03] S: 90.91%; ∈ [0.995,1.005] S: 45.45%
- DM ratios in [0.97, 1.03] P: 81.82%; ∈ [0.995,1.005] P: 36.36%

TOA Found/TOA Inj, DM Found/DM Inj, and SNR Found/SNR Inj distribute around 1, for both SIGPROC and PSRDADA.

VARYING PARAMETERS

4. Results – Real-Time vs Offline validation – True File N. 1





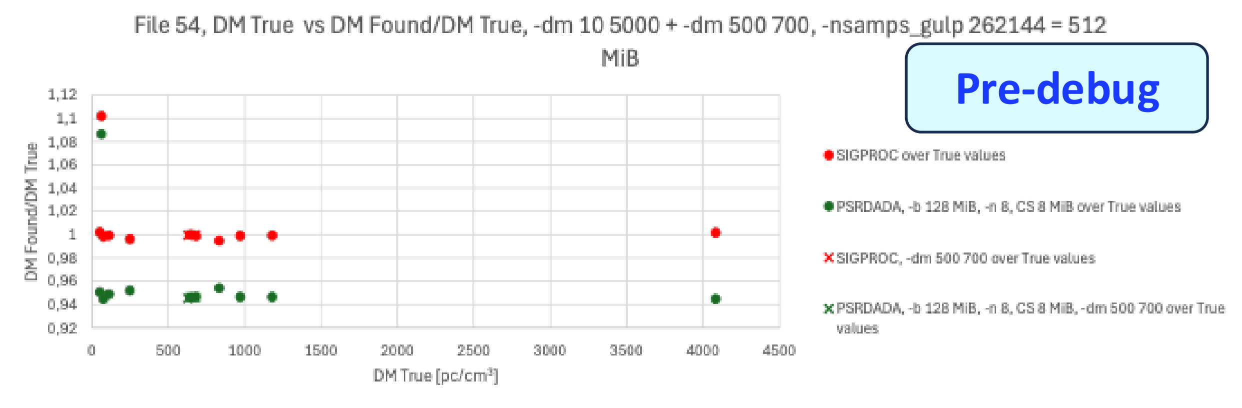
Parameters space variation does not affect the results (same consideration for the considered synthetic file).

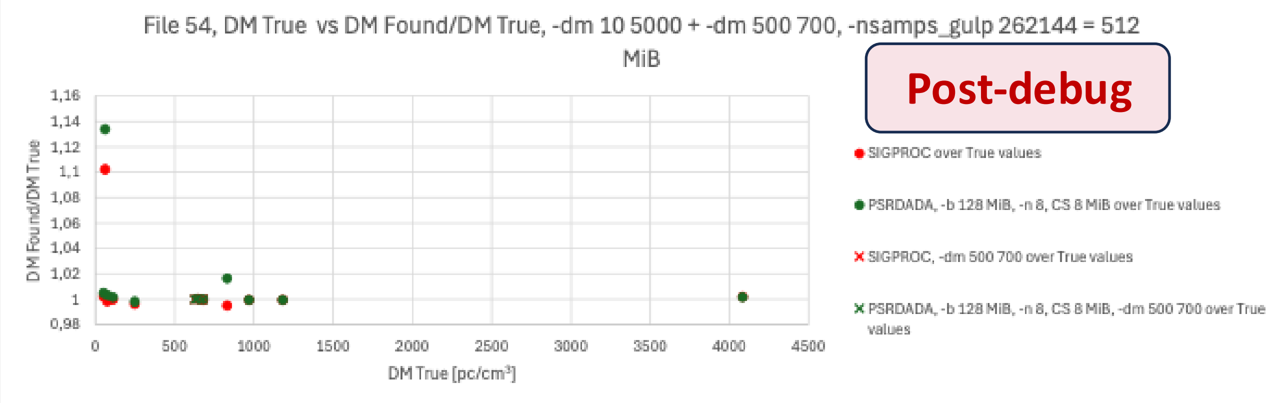
Finanziato dall'Unione europea NextGenerationEU
Ministero dell'Università e della Ricerca
Italiadomani PIANO NAZIONALE DI RIPRESA E RESILIENZA
INAF ISTITUTO NAZIONALE DI ASTROFISICA

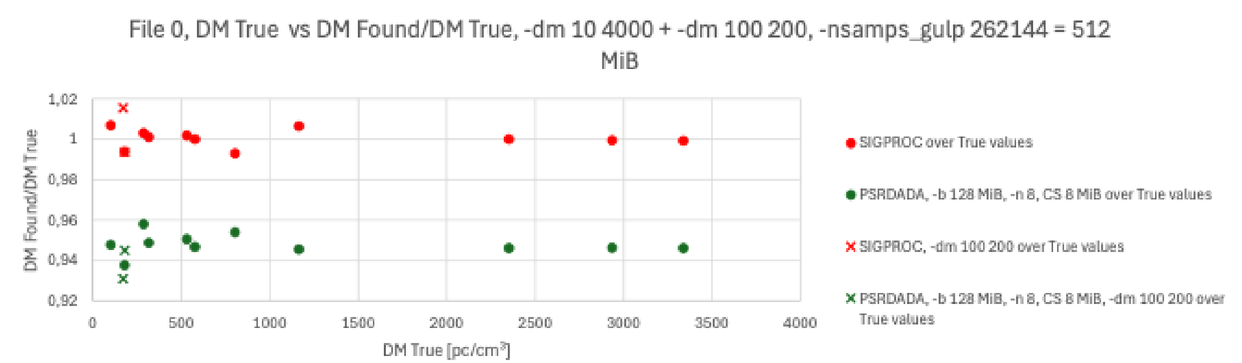
4. Results – Debugging of PSRDADA Header

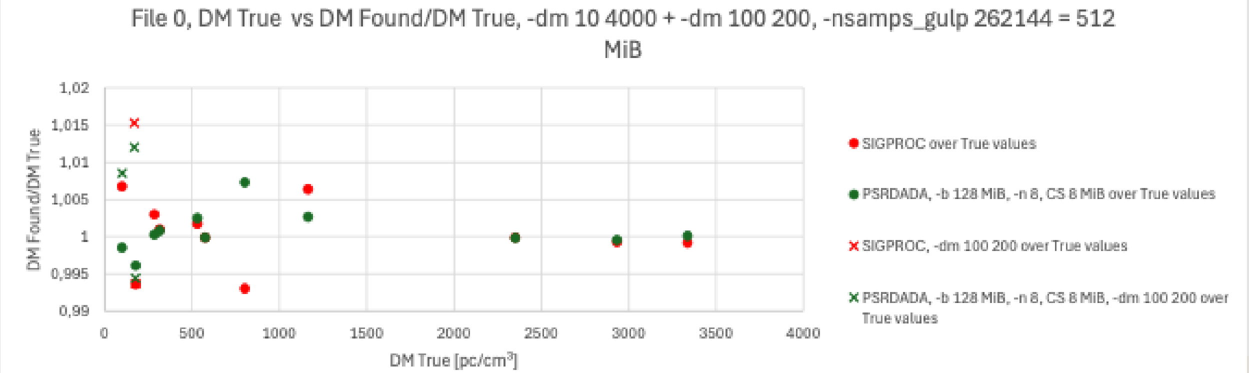
Some parameters in PSRDADA header are interpreted differently than in SIGPROC header:

- SIGPROC header needs the frequency of the first channel of the total bandwidth;
- PSRDADA header needs the central frequency of the total bandwidth
- The total bandwidth was taken with the wrong sign.





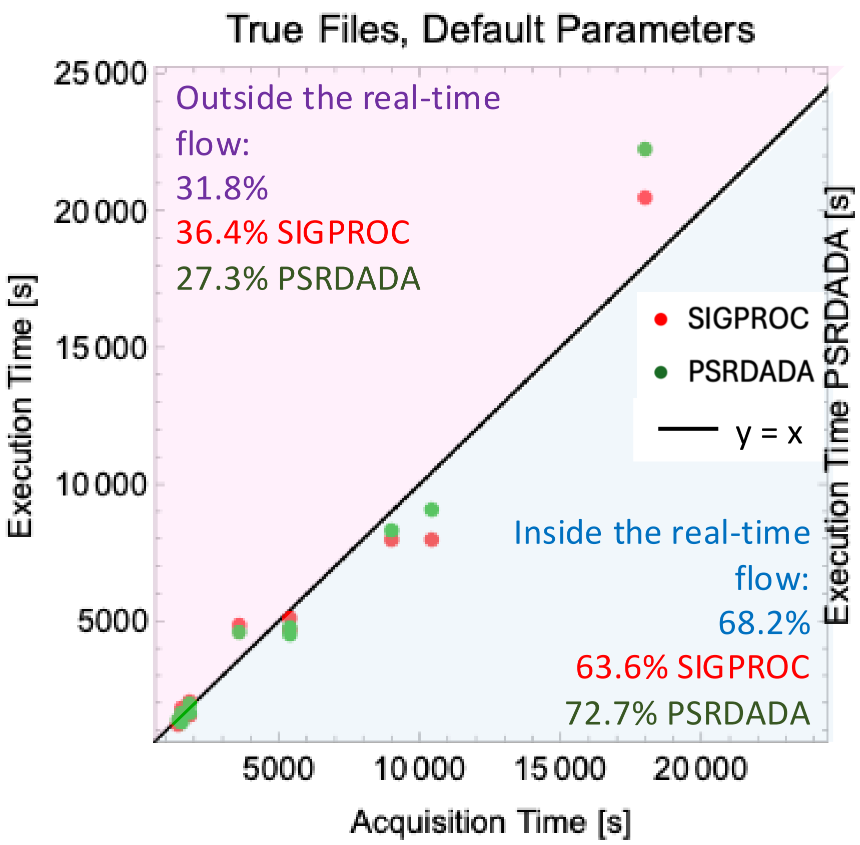


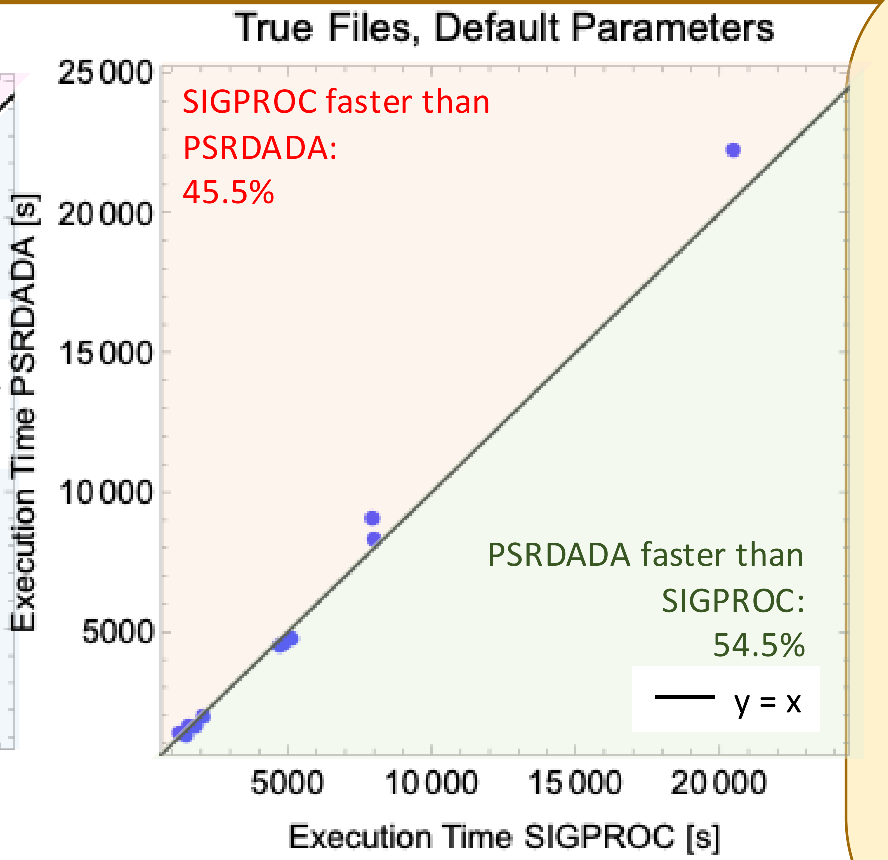


DM_{Found}/DM_{Inj} now distribute around 1 also for PSRDADA, that do not underestimate anymore the DM.

4. Results – Performance

DEFAULT PARAMETERS

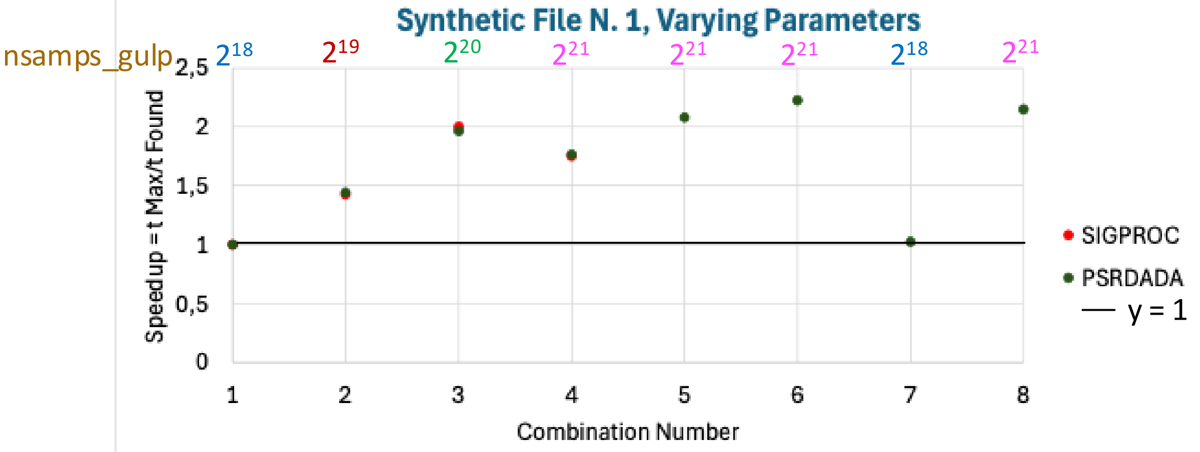


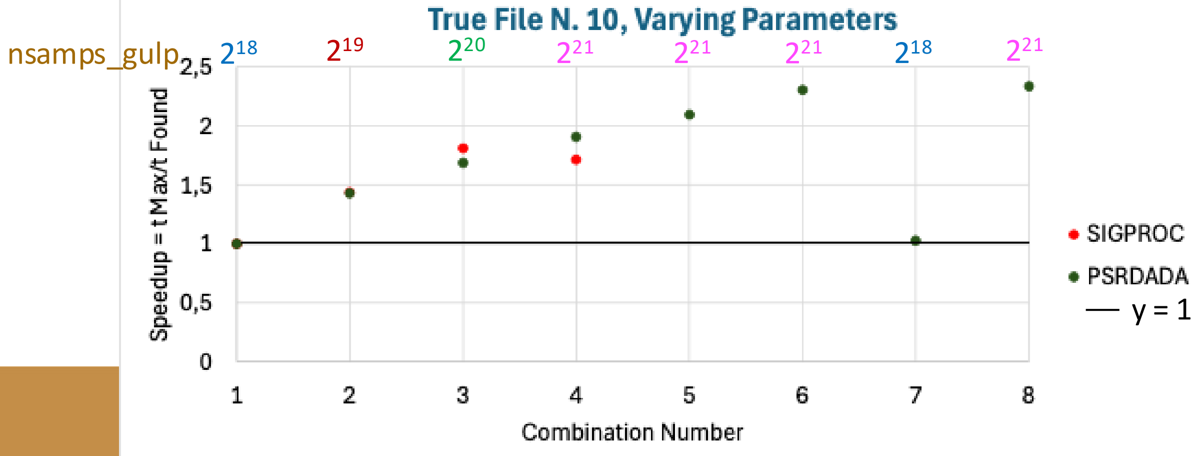


- $t_s = t_{\mathrm{writer,s}} + t_{\mathrm{Heimdall,s}}$
- $t_p = \mathrm{Max}(t_{\mathrm{writer,p}}, t_{\mathrm{Heimdall,p}})$
- **The larger number of points (PSRDADA more than SIGPROC) stay in the real-time flow.**
- Comparable times between SIGPROC and PSRDADA: **t_writer_s ~ 1-2% t_Heimdall_s ⇒ asynchronous writer and Heimdall in PSRDADA mode do not provide significant advantages with respect to SIGPROC mode.**
- **Different situation expected with the beamformer!**

4. Results – Performance – Speedup

VARYING PARAMETERS





- `nsamps_gulp` (`nsg`) dominates the performance: typically, `nsg` > ⇒ Speedup > but performance degradation can be obtained from nsg = 2^{20} to nsg = 2^{21}.
- Doubling `nsg` does not doubles the Speedup (not a perfect strong scaling with `nsg`).
- Maintaining `nsamps_gulp` fixed, we generally obtain a better performance with `<b>` > and `<n>` < (not a general result).
- `nsg-<b>-CS` alignment can provide a slight performance improvement.
- Some combinations provide speedup ≳ 2x.
- Synthetic file: combinations 2-6; 8 **make entering in the real-time flow.** True file **maintains in the real-time flow.**

Test N.	nsamps_gulp	<b> [MiB]	<n>	CHUNK_SIZE [MiB]
(1)	2^{18}	128	8	8
(2)	2^{19}	128	8	8
(3)	2^{20}	128	8	8
(4)	2^{21}	128	8	8
(5)	2^{21}	256	4	8
(6)	2^{21}	512	2	8
(7)	2^{18}	512	2	512
(8)	2^{21}	4096	2	4096

5. Summary

- **Very good consistency** between SIGPROC and PSRDADA results, both with fixed parameters, varying files, and with fixed files, varying parameters: **PSRDADA accuracy verified**.
- PSRDADA runs in the real-time flow with larger times involved.
- Generally, **comparable times of PSRDADA mode with SIGPROC mode**: expecting a larger performance gain of PSRDADA vs SIGPROC when beamformer will replace the PSRDADA writer, **to better exploit the asynchronism between** Heimdall (consumer) and the writer (producer) processes **vs the sequential execution in SIGPROC**.
- **Real-Time behaviour** can be obtained with a proper combination of parameters. A possible optimal choice: nsg >, <b> >, <n> <, nsg − <b> − CS alignment.

6. Outlooks

- Results presented in a paper in preparation for HPEC conference (https://ieee-hpec.org; deadline: July 7th 2026, conference: September 14th – 18th 2026, virtual).
- Future deeper exploration of the parameter space (first work in Camilleri, et al., 2026; https://arxiv.org/abs/2603.16345). Deeper performance analisys with NVIDIA Nsight systems profiler.
- Work in progress with Adrian de Barro and Alessio Magro from University of Malta (meeting occured in January 2026): the writer will be replaced by the **beamformer** and the creation of PSRDADA ring buffer will no more be managed via command line but via a **process manager** (orchestration process of all the pipeline).

Finanziato dall'Unione europea
NextGenerationEU
Ministero dell'Università e della Ricerca
Italiadomani
PIANO NAZIONALE DI RIPRESA E RESILIENZA
INAF
ISTITUTO NAZIONALE DI ASTROFISICA

6. Outlooks

System aiming **high throughput** and **clear division of roles.**

JSON configuration file

NODE A
Beamforming pipeline (B; producer), executed on a container:
It produces data and it writes them via RDMA.
Possibly, it runs on **multiple GPUs**.

Asynchronous processes

GPU Direct RDMA:
Bulk data transfer between B and H

Redis Cache = Control plane coordination mechanism
(sharing of all orchestration state, configuration and progress metadata)

NODE B
Process Manager (PM), orchestration authority among all the processes:
It mediates everything else.

Heimdall pipeline (H; consumer), executed on a different container:
It reads data from PSRDADA ring buffer.

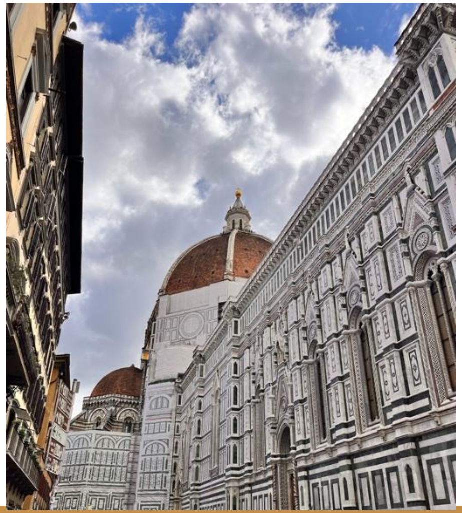
6. Outlooks

- From single-beam to **multi-beam**. → Requiring more GPU resources.
- Improvement of current ML-based classifier (FETCH), more updated version (work in progress of a PhD student and a master student).
- **More compact structure for containers** (currently Heimdall running on one container and FETCH on another container → possible employment of Apptainer) + **more up-to-date versions of the software**.
- Replace Heimdall with a pipeline having new ML-based modules, in place of current less efficient modules.

Finanziato dall'Unione europea
NextGenerationEU
Ministero dell'Università e della Ricerca
Italiadomani
PIANO NAZIONALE DI RIPRESA E RESILIENZA
INAF
ISTITUTO NAZIONALE DI ASTROFISICA

THANK YOU FOR THE ATTENTION! ☺
QUESTIONS?

Northern Cross radiotelescope, top view, (De Barro, et al., 2025).



EXTRA SLIDES

A. Testbed setup – Launching Commands Sequence

Launching commands sequence:

OFFLINE:

1) `time ./writer_s file_i.fil file_o.fil > output_w_s.txt`
2) `time heimdall -f file.fil -output_dir ./cand_s -dm 10 3000 -dm_tol 1.001 -boxcar_max 512 -v > output_s.txt`

- SIGPROC writer is built for a fair performance comparision.
- It simply writes ad identical output file given ad input file.

REAL-TIME:

1) PSRDADA ring buffer creation
2) `time heimdall -k dada -output_dir ./cand_p -dm 10 3000 -dm_tol 1.001 -boxcar_max 512 -v > output_p.txt`
3) `time ./writer_p file.fil > output_w_p.txt`

Blind search for DM: $[10{,}3000]\ \mathrm{pc/cm^3}$ range. Upper limit even higher with files with references DM higher than $3000\ \mathrm{pc/cm^3}$.

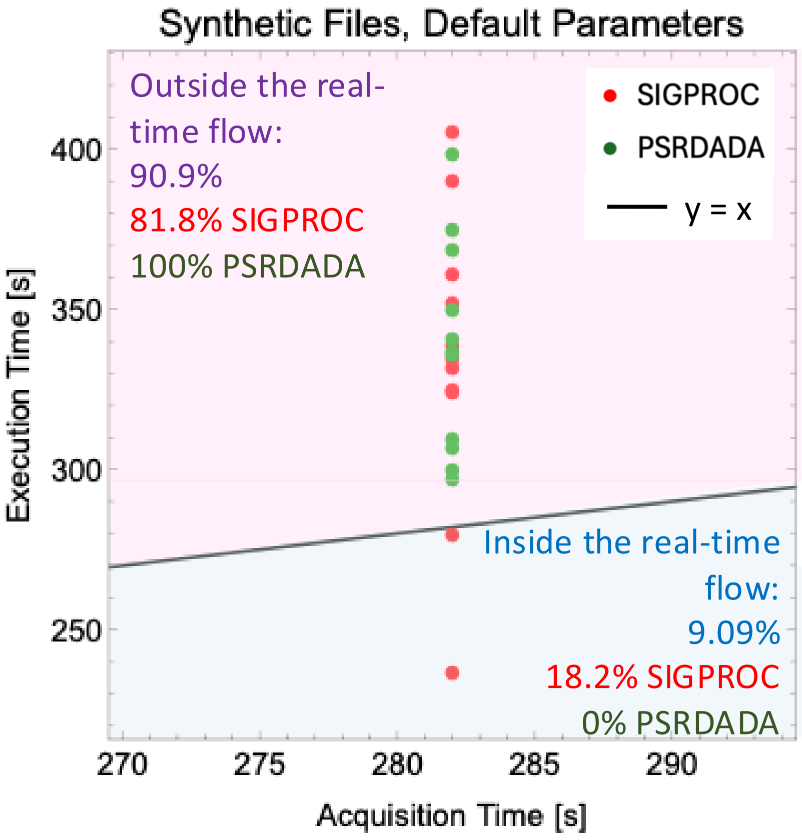
Finanziato dall'Unione europea NextGenerationEU | Ministero dell'Università e della Ricerca | Italiadomani PIANO NAZIONALE DI RIPRESA E RESILIENZA | INAF ISTITUTO NAZIONALE DI ASTROFISICA

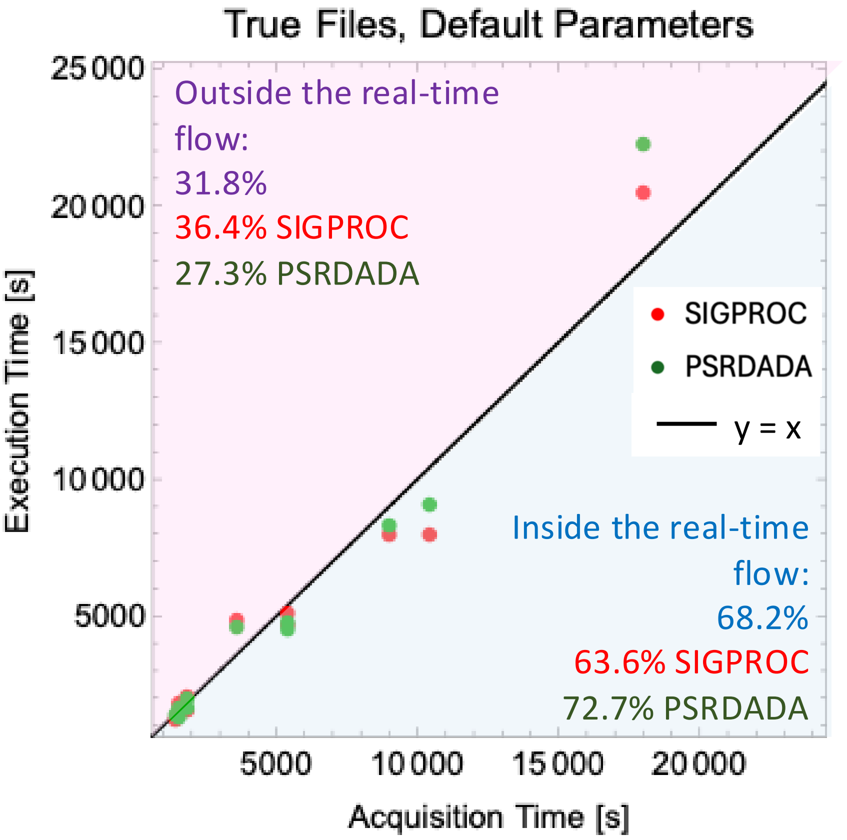
B. Results – Debugging of PSRDADA header

- Some parameters in PSRDADA header are interpreted differently than in SIGPROC header.
- In SIGPROC header, we have:
  - ❖ fch1 = 415.854456 MHz (frequency of the first channel of the total bandwidth)
  - ❖ foff = –0.014468 MHz (bandwidth of a single frequency channel)
  - ❖ nchans = 1024 (number of frequency channels)
  - ❖ ⇒ fchlast = fch1 + foff * (nchans – 1) = 401.053692 MHz (frequency of the last channel of the total bandwidth)
  - ❖ ⇒ BW = nchans * foff = –14.815232 MHz (total bandwidth)
- Bug introduced in PSRDADA header:
  - ❖ FREQ = fch1 BUT IT SHOULD BE **FREQ = 0.5 * (fch1 + fchlast) = 408.454074 MHz** (central frequency of the total bandwidth)
  - ❖ BW > 0 BUT IT SHOULD BE **BW < 0** (with the wrong sign, generating incoherence with foff, taken with the right sign)
  - ❖ Bandwidth before correction: [393.638842, 408.454074] MHz -> Bandwidth after correction: [401.053692, 415.854456] MHz
- Effect on DM:
  - ❖ Dispersion delay: $\Delta t = k_{\mathrm{DM}} \mathrm{DM}(\nu_1^{-2} - \nu_2^{-2})$ (Barsdell, et al., 2012). Since Δt keeps constant ⇒ $\mathrm{DM}_{\mathrm{Before}}/\mathrm{DM}_{\mathrm{After}} = \mathbf{0.945625}$ = **Underestimation factor**.

B. Results – Performance – Comparison with Acquisition Time

DEFAULT PARAMETERS

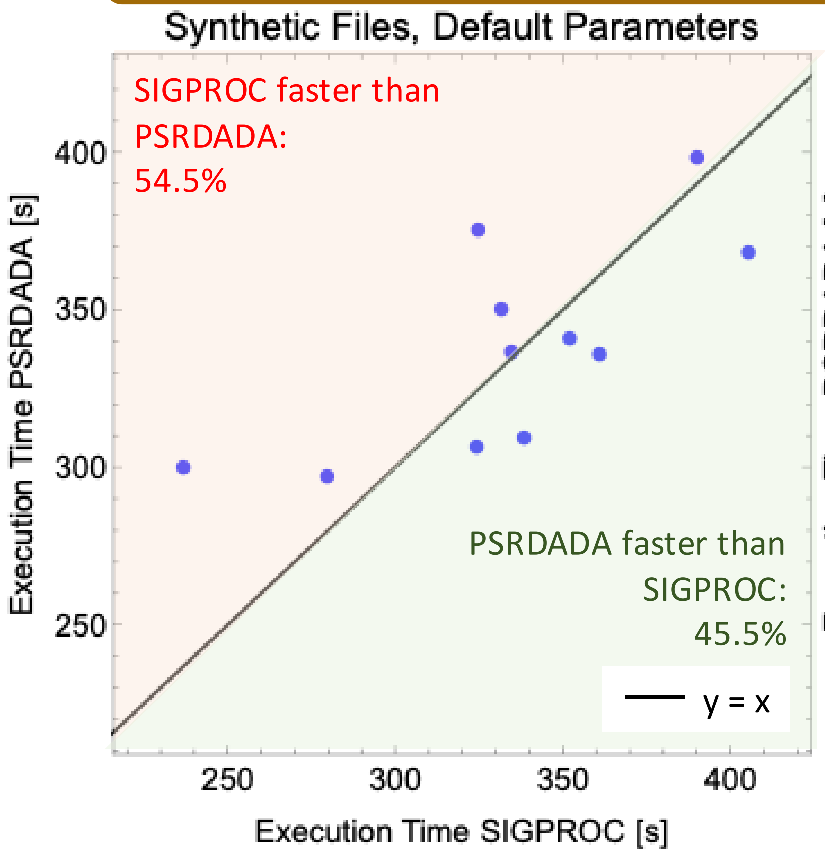


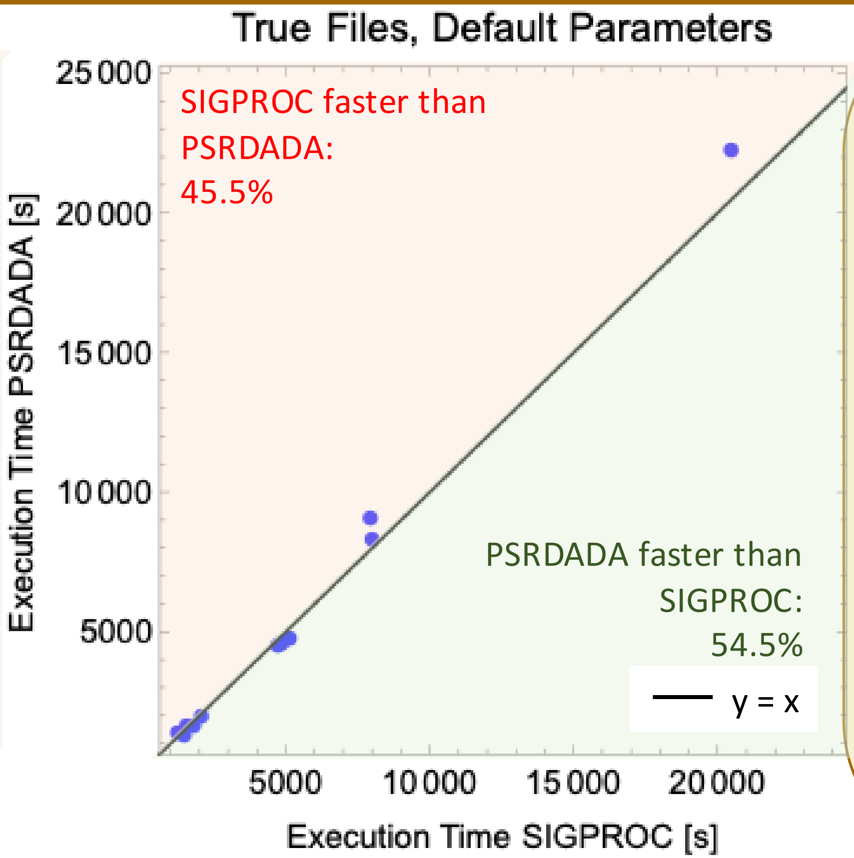


- $t_s = t_{\text{writer,s}} + t_{\text{Heimdall,s}}$
- $t_p = \text{Max}(t_{\text{writer,p}}, t_{\text{Heimdall,p}})$
- Synthetic files with acq. time = 4.7 min and exec. times ∈ [3.9,6.8] min.
- True files (real cases) with acq. time ∈ [23.9 min, 5 h] and exec. times ∈ [20.7 min, 6.2 h].
- **With smaller times, pipeline's overhead, PSRDADA synchronization operations, etc. can weight more over computation times, going outside the real-time flow region.**
- **With higher times (real cases), computation dominates and we have more points in the real-time flow region.**

B. Results – Performance – Offline vs Real-Time

DEFAULT PARAMETERS





- Generally, comparable times between SIGPROC and PSRDADA (more scattered with synthetic files): **t_writer_s ~ 1-2% t_Heimdall_s ⇒ asynchronous writer and Heimdall in PSRDADA mode do not provide significant advantages with respect to SIGPROC mode.**
- ~50% points PSRDADA faster than SIGPROC and ~50% points SIGPROC faster than PSRDADA, for both synthetic and true files.
- **Different situation expected with the beamformer!**

Finanziato dall'Unione europea NextGenerationEU

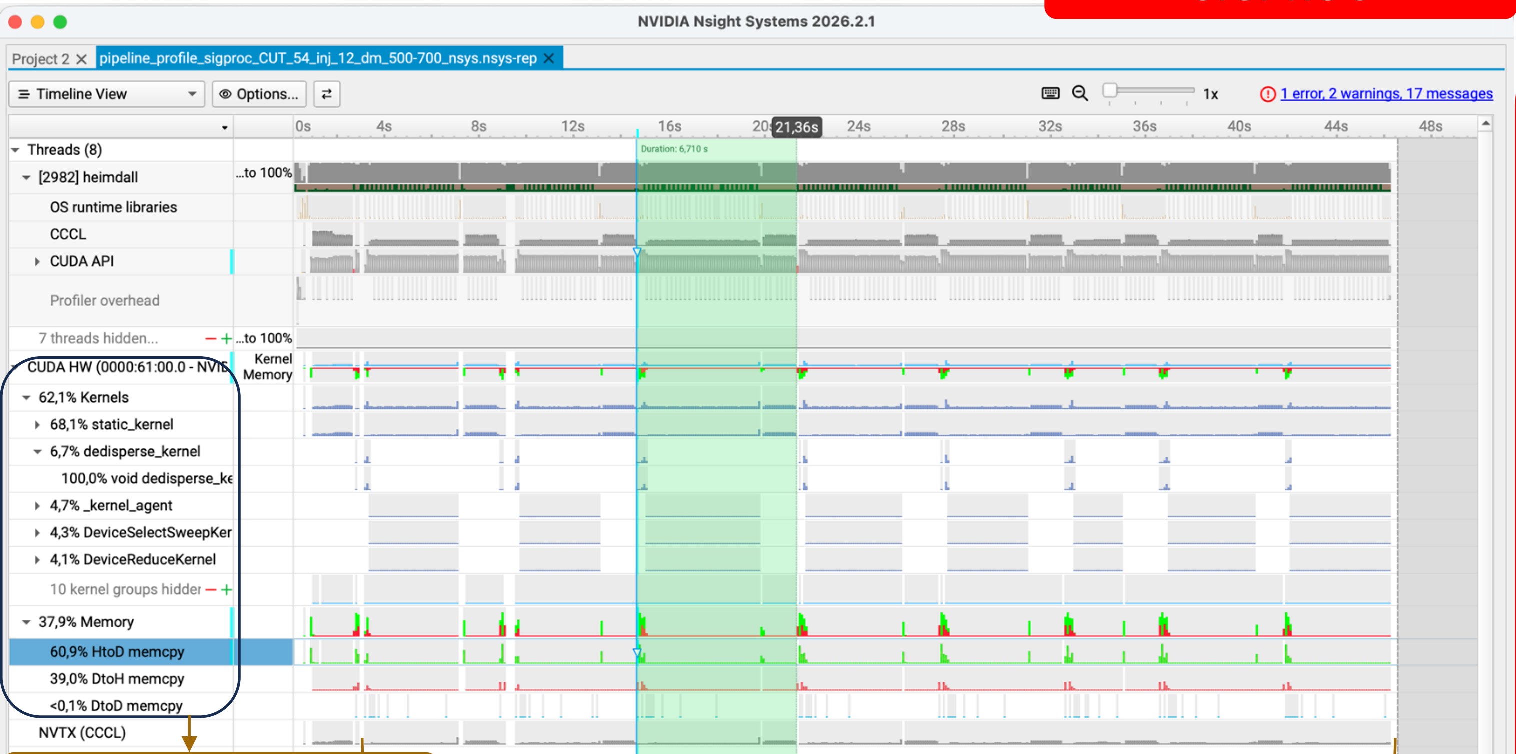
Ministero dell'Università e della Ricerca

Italiadomani PIANO NAZIONALE DI RIPRESA E RESILIENZA

INAF ISTITUTO NAZIONALE DI ASTROFISICA

B. Results – Profiling with NVIDIA Nsight Systems

SIGPROC



Compute-bound/partially memory-bandwidth-bound.

Cyclic structure of nsamps_gulp time samples.
Compute time dominating over transfer time (~1.4%). Continuous kernel execution (low overheads between kernel launches). Asynchronous transfers.

62.1% kernel computation, 37.9% memory operations.

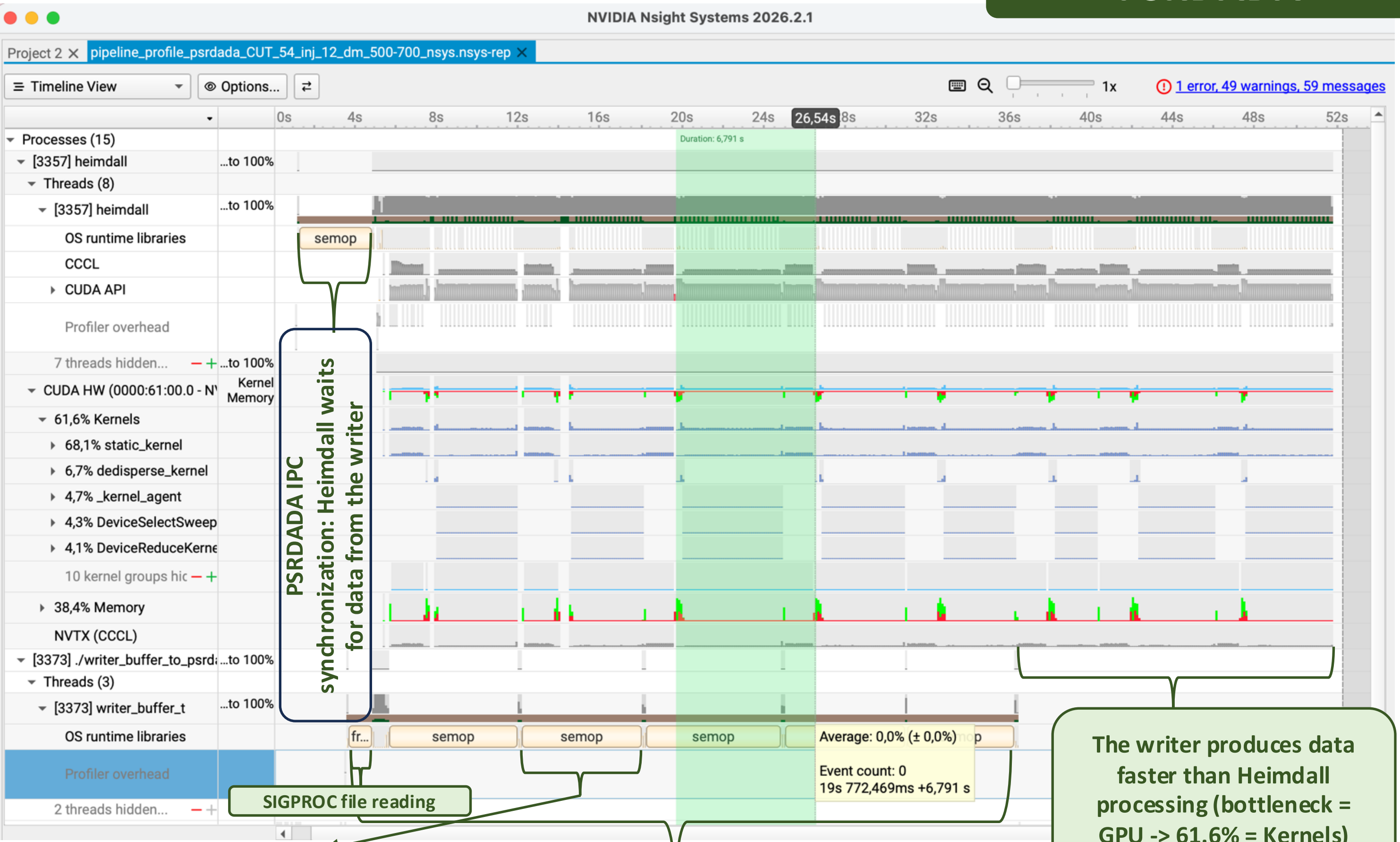
Main: nsamps_processed=262144 $= 2^{18}$
Executing pipeline on new gulp of 262144 samples...
Mem alloc time: 1.8e-05
0-DM cleaning time: 1.73399
Dedispersion time: 0.170899
Copy time: 0.0937979
Baselining time: 0.229877
Normalisation time: 0.235588
Filtering time: 0.327201
Find giants time: 3.86462
Process candidates time: 0.027326
Total time: 6.68432
Processed 262144 samples.

https://developer.nvidia.com/nsight-systems

Finanziato dall'Unione europea NextGenerationEU
Ministero dell'Università e della Ricerca
Italiadomani PIANO NAZIONALE DI RIPRESA E RESILIENZA
INAF ISTITUTO NAZIONALE DI ASTROFISICA

B. Results – Profiling with NVIDIA Nsight Systems

PSRDADA



PSRDADA IPC synchronization: Heimdall waits for data from the writer

SIGPROC file reading

Waiting to obtain a free block of the PSRDADA ring buffer to write

Writer

The writer produces data faster than Heimdall processing (bottleneck = GPU -> 61.6% = Kernels)

61.6% kernel computation, 38.4% memory operations.

```
Main: nsamps_processed=262144 = 2^18
Executing pipeline on new gulp of
262144 samples...
Mem alloc time:             1.7e-05
0-DM cleaning time:         1.74729
Dedispersion time:          0.173541
Copy time:                  0.0952495
Baselining time:            0.231617
Normalisation time:         0.238683
Filtering time:             0.332703
Find giants time:           3.89766
Process candidates time:    0.027352
Total time:                 6.74508
Processed 262144 samples.
```

https://developer.nvidia.com/nsight-systems