PROCEEDINGS OF SCIENCE

The ASTRI Mini-Array Cherenkov Data Pipeline

M. Mastropietro,[a,*] **S. Lombardi,**[a,b] **F. Lucarelli,**[a,b] **F. Visconti,**[a,b] **E. Fedorova,**[a,c] **L. A. Antonelli**[a] **for the ASTRI Project**[d]

[a] *INAF – Osservatorio Astronomico di Roma, Via Frascati 33, I-00078 Monte Porzio Catone (Roma), Italy*

[b] *ASI – Space Science Data Center, Via del Politecnico s.n.c., I-00133 Roma, Italy*

[c] *Taras Shevchenko National University of Kyiv, Astronomical Observatory, Observatorna str. 3, 04053 Kiev, Ukraine*

[d] *http://www.astri.inaf.it/en/library/*

E-mail: michele.mastropietro@inaf.it, saverio.lombardi@inaf.it

The ASTRI Mini-Array is an international project led by INAF and devoted to imaging atmospheric Cherenkov light for very high-energy γ-ray astronomy. The project is deploying an array of nine imaging atmospheric Cherenkov telescopes of 4-m class at the Observatorio del Teide (Tenerife, Canary Islands). The Cherenkov data pipeline is in charge of reducing and analysing the scientific data recorded during the ASTRI Mini-Array observations by means of the `A-SciSoft` software package. The pipeline will be integrated in the data processing system cluster, which will leverage a workload manager for automatic data processing and archival at the offsite ASTRI data center. In this contribution we describe how we designed the multiple types of pipelines capable of analyzing both Monte Carlo and real data of the ASTRI Mini-Array.

The 38th International Cosmic Ray Conference (ICRC2023)
26 July – 3 August, 2023
Nagoya, Japan

*Speaker

PoS(ICRC2023)774

1. Introduction

The usual data processing of Cherenkov telescope data is composed of several steps. The images of Cherenkov events captured by the telescopes need to be calibrated and reduced; subsequently, the information of the primary particle shall be reconstructed from this processed data. With the *calibration* step it is meant the conversion from raw analog-to-digital converter counts to photo-electrons for each pixel using dedicated calibration coefficients; the *reduction* step is performed by cleaning the image from the background and computing several parameters characterising the image (the standard Hillas parameters, [1]). Finally, the most delicate step is the *reconstruction*, where by means of a set of simulations, the nature of the primary particle impacting the atmosphere is assessed and both its energy and arrival direction are estimated. In the following we describe how we designed the processing steps and the infrastructure used to run the pipeline.

2. Data model

The data model of the ASTRI Mini-Array (described in detail in [2]) is the context for the design of the Cherenkov Data Pipeline (CDP). Following the data model, it is assumed that the pipeline receives as input FITS data files and produces FITS files as outputs. The CDP shall be able to process both *Real data* and *Monte Carlo data*. These two represent the main use cases for the design of the CDP, and they will be described in the following.

Real data analysis The data processing system will run the CDP for the calibration, reduction and analysis of the Cherenkov raw data acquired during the ASTRI Mini-Array observations. During the processing, the CDP will use further information (engineering and auxiliary data; metadata) related to the observations stored in the science archive at the end of each observing run. From the reduction of real Cherenkov data, the CDP pipeline will produce:

- science-ready data (event-lists and observation-related Instrument Response Functions, IRFs);
- data-filtering tables (Good Time Intervals, GTI);
- automated generated science products (skymaps, detection plots);
- data quality reports and plots.

During the real data processing, intermediate data products will also be produced and, if necessary, archived.

Two levels of real data processing scenarios are foreseen: the short-term standard analysis pipeline, to be run at the end of each observing run, and the long-term standard analysis pipeline.

The short-term analysis pipeline will make use of pre-computed calibration factors (CAL1a, generated by the calibration software system, [3]) and coarse IRFs, and runs up to the production of preliminary science products. The long-term analysis pipeline will produce consolidated science-ready data and IRFs, intended for final analysis and publication of results.

The short-term analysis will be performed at the off-site ASTRI data center, in an automated way. The analysis will start as soon as scientific data have been transferred and archived in the off-site permanent ASTRI bulk archive. Only during the commissioning and science verification

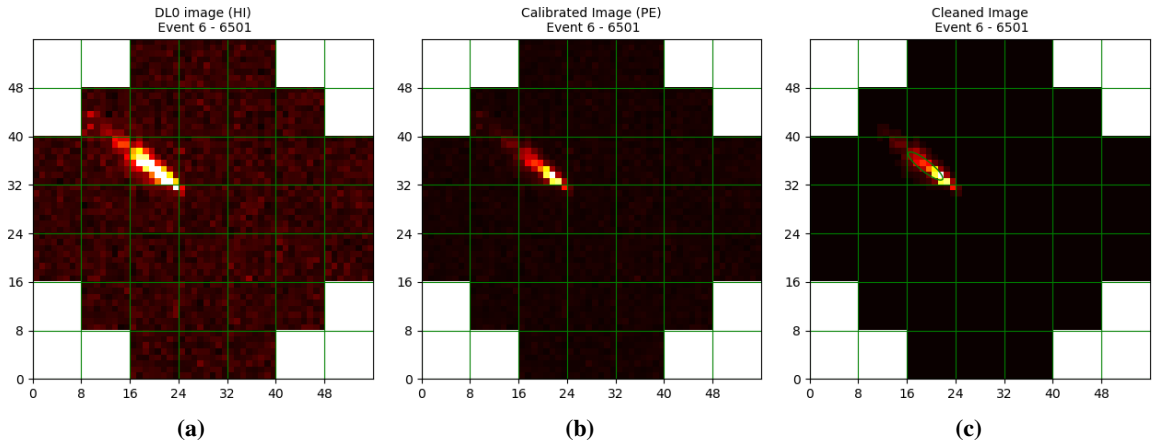

Figure 1: Monte Carlo simulated point-like photon event as captured by the ASTRI camera. The different output shown corresponds to high gain ADC counts (fig. 1a), calibrated photoelectrons (fig. 1b), and cleaned image with parameterization (fig. 1c). Simulation parameters are E_0 = 32.3 TeV and height of first interaction of 10 km.

phase, it is foreseen to manually run the CDP onsite, to immediately check run consistency and the expected technical and science performance requested during the Cherenkov observations. Instead, the long-term analysis will always run off-site at the ASTRI data center. Both processes can run an automated data quality check and instrument monitoring after the observations.

Monte Carlo analysis The Cherenkov Data Pipeline will also be used to reduce and analyse Monte Carlo (MC) simulated data, needed to generate look-up-tables (LUTs) and low-level (global) instrument response functions (IRFs) to be used during the real data processing, as well as to evaluate the ASTRI Mini-Array science performance.

3. Pipeline architecture

3.1 Calibration step

We start by processing a set of DL0 FITS files coming from the telescopes, containing the raw images of all the nine telescopes. The raw Cherenkov camera data, containing for each triggered event (`EVT0` data) the full information available per each camera pixel (integrated signal amplitude in analog-to-digital counts (ADC) and arrival time), are calibrated separately for each telescope in order to extract and convert the signal into physically meaningful units (photo-electrons). See the output of the calibration step in Fig. 1b. The conversion coefficients (`CAL1a` data) can be either extracted from specific camera calibration data – processed with the calibration software system – or, alternatively, directly retrieved from the Calibration Data Base (CALDB)[1]. The software module implementing these functionalities is called `astrical`.

[1] https://heasarc.gsfc.nasa.gov/docs/heasarc/caldb/

3.2 Reduction and reconstruction step

The characterisation of each Cherenkov event triggered by the system is done both on a telescope-wise and an array-wise basis. The calibrated images of each triggered telescope firstly undergo a cleaning procedure (imaging cleaning) aimed at removing pixels which most likely do not belong to a given Cherenkov shower image. See the output of the cleaning step in Fig. 1c. After this step, the Hillas parameterization [1] of each cleaned image is performed. Using then suitable single-telescope look-up-tables (`LUT1`), calculated beforehand by means of MC simulated Cherenkov events, the main air Cherenkov shower parameters (arrival direction, energy reconstruction, and event classification) are estimated on a telescope-wise basis. The telescope-wise fully reconstructed events, owing to the same stereo triggered event as identified by the stereo event builder, [4], are merged and a set of basic array-wise image parameters (such as the geometrical estimation of the shower arrival direction, the maximum height, and the impact parameters relative to each telescope) are calculated. See the merging of the events and the calculation of stereo parameters in Fig. 2. Array-wise LUTs (`LUT2`) are then applied to the merged data and array-wise shower parameters for the gamma/hadron separation, energy reconstruction, and arrival direction estimation are available for each fully reconstructed stereo event. In particular, the gamma/hadron discrimination parameter is evaluated for each triggered stereoscopic array event. The software modules implementing these functionalities are called `astricleanpar` for the cleaning and parametrization, `astrimer` for merging and calculating array-wise parmeters, `astriluts` for generating LUTs and `astrireco` to apply both single-telescope and array-wise LUTs to the data.

3.3 Event selection step

The fully reconstructed stereo events are further processed to achieve the fully reduced data, applying both quality and gamma/hadron separation cuts. At this stage, the final gamma-like event-list (`EVT3`) is produced, extracted from the corresponding EVT2b, along with the corresponding reduced `IRF3` (from the corresponding `IRF2`), [5]. The reduced IRFs (`IRF3`) are generated by filtering the global IRFs (IRF2) over several parameters, weighted by the observation-related parameters (i.e., hardware settings/configurations, data-taking conditions, zenith and azimuth pointings) of the particular dataset that is being reduced by the CDP. The software module implementing these functionalities is called `astriana`.

4. A-SciSoft

The software tools developed to implement the pipeline architecture have been called ASTRI Scientific Software `A-SciSoft`. In the following we will describe the design choices as well as the technical features of the software tools.

4.1 Pipeline design and implementation

The main design choices of the pipeline have been driven by the following requirements:

- Modularity
- Ease of development

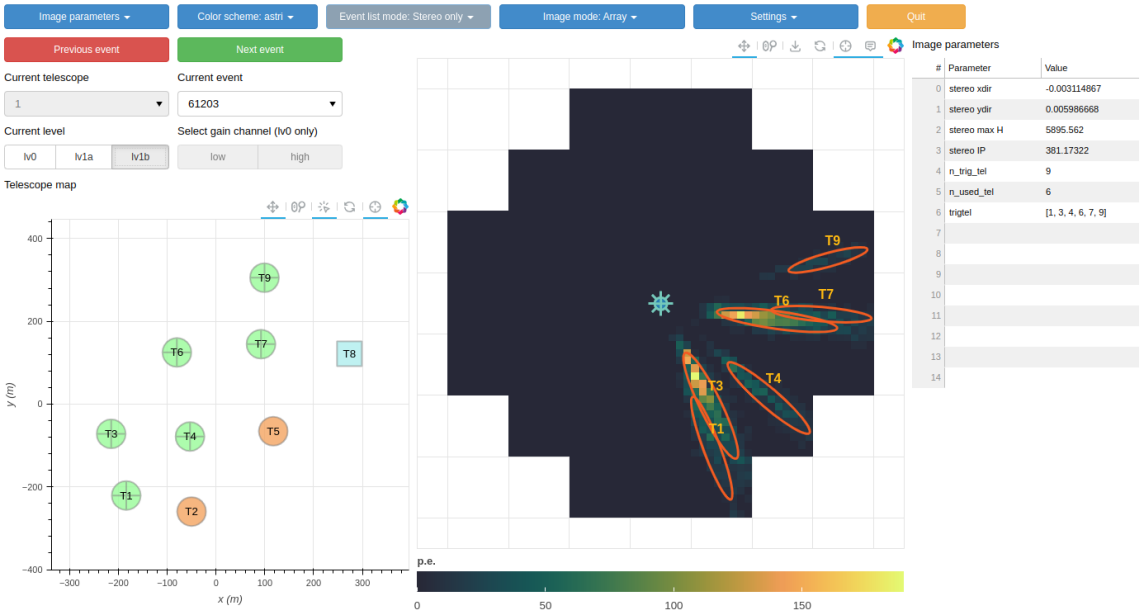

Figure 2: ASTRI Mini-Array visualizer of a Monte Carlo simulated photon triggered by all nine telescopes. In the bottom left panel the telescope positions are shown with different symbols indicating whether the captured image did not survive the cleaning step (T8) or the image was not considered part of a stereo event and thus not used to compute stereo parameters (T2 and T5). Finally a superposition of all the survived images and their parametrization is shown in the central panel as well as the computed stereo parameters for the reconstructed event.

- Fast data processing

Modularity is key for a multiple step pipeline. This way intermediate outputs can be inspected and multiple alternative steps can be defined in case a different processing is needed.

4.2 Programming language choices

The first part of the processing requires operations to be carried out per pixel, whereas after the reduction, only image related parameters are taken into account and the size of data to be processed is brought down by a factor 10^3. Therefore, in the first two modules of the pipeline we leverage the performance of a compiled language like C++, whereas all the other modules are written in python. C++ functions (like calibration, parametrization and cleaning) have been parallelized using OpenMP both per pixel and per events, hiding latencies. Nonetheless, we found I/O as the main source of latency. As stated, FITS is the format of the files used to save results. In particular we found out that `CCfits`, the Heasarc's standard wrapper to `cfitsio`, creates data bloating because of some internal data containers. Therefore, I/O critical parts of the software required the application of `cfitsio`'s low-level `fits_*_tblbytes` routines, obtaining less than half of the memory footprint when reading and writing files and about ×7 faster I/O with respect to `CCfits` [6, 7]. In addition, low level operations on raw data – like bytes swapping and unsignedness corrections (since the FITS format does not support a native unsigned integer data type) – have been parallelized using OpenMP, leading to a fast and very lean I/O system for `A-SciSoft`.

4.3 Containerization

We produced both `docker` and `apptainer`[2] cointainers for `A-SciSoft`. We set up both the container options in order to keep the possibility of using of the tools on multiple platforms, depending on the needs of the software experts testing the software in the validation phase. The containers are created with each tagged release by means of the INAF's Gitlab Continuous Integration platform, and are published on the same Gitlab containers registry for access to the collaboration. Especially during commissioning and science validation phases, this way of deployment allows a facilitated use of the software without the need of installing the dependencies. Also, strict versioning and tagging of containers allow for reproducible results. `apptainer` containers can also be used natively and easily on an HPC cluster, as shown in Section 5.

In addition to the compiled binaries for the `astrical` and `astricleanpar` modules, we provide a `conda` environment where to install the python `A-SciSoft` modules, which are made available through python entry points.

5. Infrastructure

The ASTRI Mini-Array data center in Rome hosts the data archive and the data processing system, [2]. It is composed by an HPC cluster and several virtual machines hosting user-oriented services running on a Kubernetes[3] cluster. The HPC cluster is currently made of 20 virtual machines with a shared filesystem with a net usable capacity of of 1.2 PB. It leverages OpenHPC[4] software suite based on SLURM[5], a cluster management and job scheduling system, on RockyLinux 8.7 operating system. We use the LMOD[6] software module system to install different software versions on the same system with the required dependencies, allowing for a flexible and reproducible software environment.

5.1 Task orchestration

The current prototype task orchestrator is based on Airflow[7] and SLURM. In particular, a custom-made in-house Airflow `SlurmOperator` has been designed to launch and monitor jobs on the HPC cluster. Multiple type of pipelines can be defined using Airflow's Direct Acyclic Graph (DAG) concept, writing simple python code. Each DAG is a pipeline as defined above in Section 3. Airflow has a nice user interface, can be installed in high availability mode (i.e. leveraging redundant computers in order to provide a minimum amount of down-time) on a Kubernetes cluster, can be configured with Role Based Access Control (RBAC) and allows expert and authorized users to define custom pipelines. Also, scheduling of a pipeline is made easy thanks to several triggers available in the airflow ecosystem (e.g., the appearance of a certain file in the filesystem, or a signal sent from the archive system). In addition, a dedicated PostgreSQL database collecting job logs and events is provided allowing for troubleshooting and inspection of faulty jobs.

[2] https://www.apptainer.org
[3] https://kubernetes.io
[4] https://openhpc.community
[5] https://slurm.schedmd.com
[6] https://lmod.readthedocs.io/
[7] https://airflow.apache.org

6. Summary and outlook

We described the procedures, the architecture and the technical details of the ASTRI Mini-Array Cherenkov data pipeline. The CDP is based on `A-SciSoft` and it has been designed to be modular, easily developed and optimized for fast data processing (with I/O optimization and parallelization). We leverage modern software development practices with Continuos Integration, testing and containerized deployment. The CDP is able to seamlessly reduce both real and Monte Carlo data and will be run on the ASTRI Mini-Array data center in Rome. The CDP has been tested on Monte Carlo data and real data coming from the ASTRI-Horn telescope, [8]. Thorough validation with real data is foreseen during the commissioning phase of the first three telescopes.

Acknowledgments

This work was conducted in the context of the ASTRI Project. We gratefully acknowledge support from the people, agencies, and organisations listed here: http://www.astri.inaf.it/en/library/. We acknowledge financial support from the ASI-INAF agreement n. 2022-14-HH.0. This paper went through the internal ASTRI review process.

References

[1] Hillas A. M. *Cherenkov light images of EAS produced by primary gamma*, 19th Intern. Cosmic Ray Conf-Vol. 3, 1985

[2] Lombardi, S. et al., Proc. 38th ICRC, Nagoya, Japan, PoS(ICRC2023)682 (2023).

[3] Mineo, T. et al., Proc. 38th ICRC, Nagoya, Japan, PoS(ICRC2023)733 (2023).

[4] Germani, S. et al., Proc. 38th ICRC, Nagoya, Japan, PoS(ICRC2023)597 (2023).

[5] Pintore, F. et al., Proc. 38th ICRC, Nagoya, Japan, PoS(ICRC2023)722 (2023).

[6] Lombardi, S. et al., *ASTRI data reduction software in the framework of the Cherenkov Telescope Array*, Software and Cyberinfrastructure for Astronomy V, SPIE, 2018

[7] Mastropietro, M. et al., *ASTRI SST-2M data reduction and reconstruction software on low-power and parallel architectures*, Software and Cyberinfrastructure for Astronomy IV, SPIE, 2016

[8] Lombardi, S., et al., A&A, 634, A22 (2020)