CTA Science Alert Generation: integration with science tools

V. Fioretti (INAF OAS) on behalf of the INAF/OAS RTA group:

A. Bulgarelli (coordinator), N. Parmiggiani, A. Di Piano, G. Stratta, G. De Cesare, L. Baroncelli

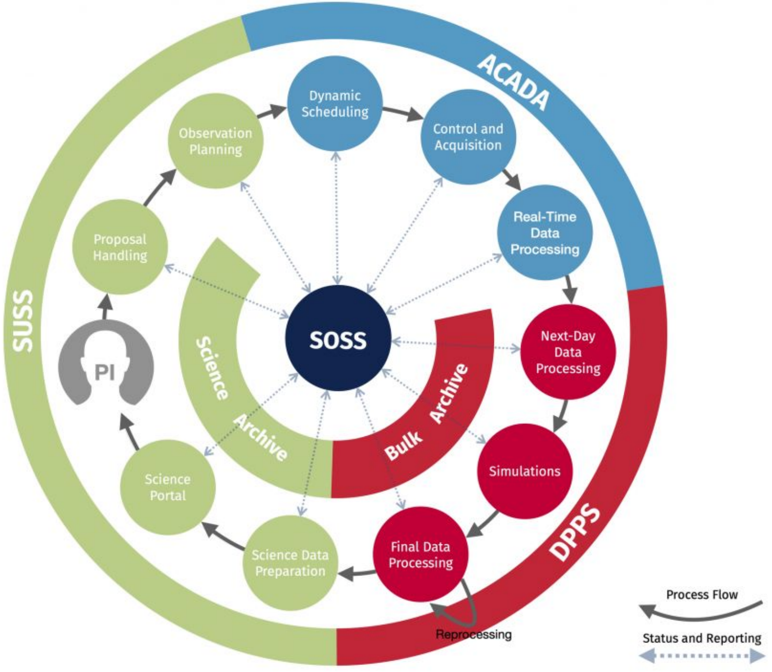
Gammapy Co-Working Week, November 2020

ACADA/Science Alert Generation

CTA ACADA

The Array Control and Data Acquisition (ACADA) is a software system of CTA, responsible for the supervision and control of telescopes and calibration instruments at both CTA array sites, including the efficient execution of scheduled and dynamically triggered observation.

The Science Alert Generation (SAG) Pipeline is an ACADA subsystem coordinated by INAF.

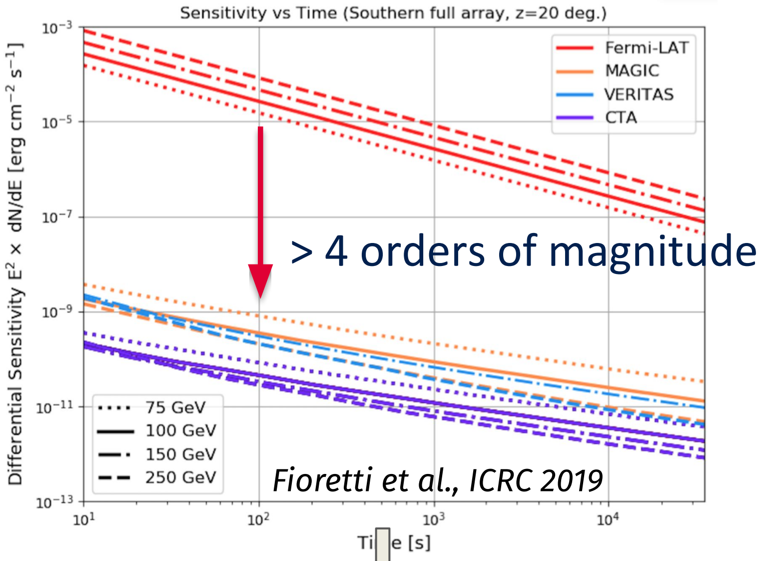


credits: https://www.cta-observatory.org/

ACADA/SAG: requirements

NB: Real-Time-Analysis = SAG

- The **SAG/RTA** is a **software system** that analyses CTA data **during the observation**.
- **On-site** with the telescopes.
- The SAG must be capable of **issuing candidate science alerts with a latency of 20s** since data becomes available to ACADA.
- The SAG must **search for transient** phenomena on different timescales **from 10 seconds to 30 minutes**.
- The **sensitivity** of the analysis is required not to be worse than the one of the final analysis by more than a factor of 2.



With SAG we are able to detect transients in real-time

The CTA SAG is the key system in the context of **multi-messenger** and **multi-wavelength** astronomy.

ACADA/SAG: Workflow

Data Reconstruction

Check the status of the observation

Data quality monitoring

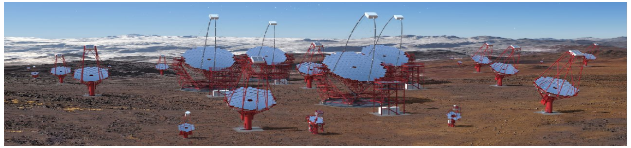
To provide feedbacks during observation to **external** (from other observatories) **science alerts**, to change the observation strategy

Science monitoring

To provide a scientific feedback during the observation

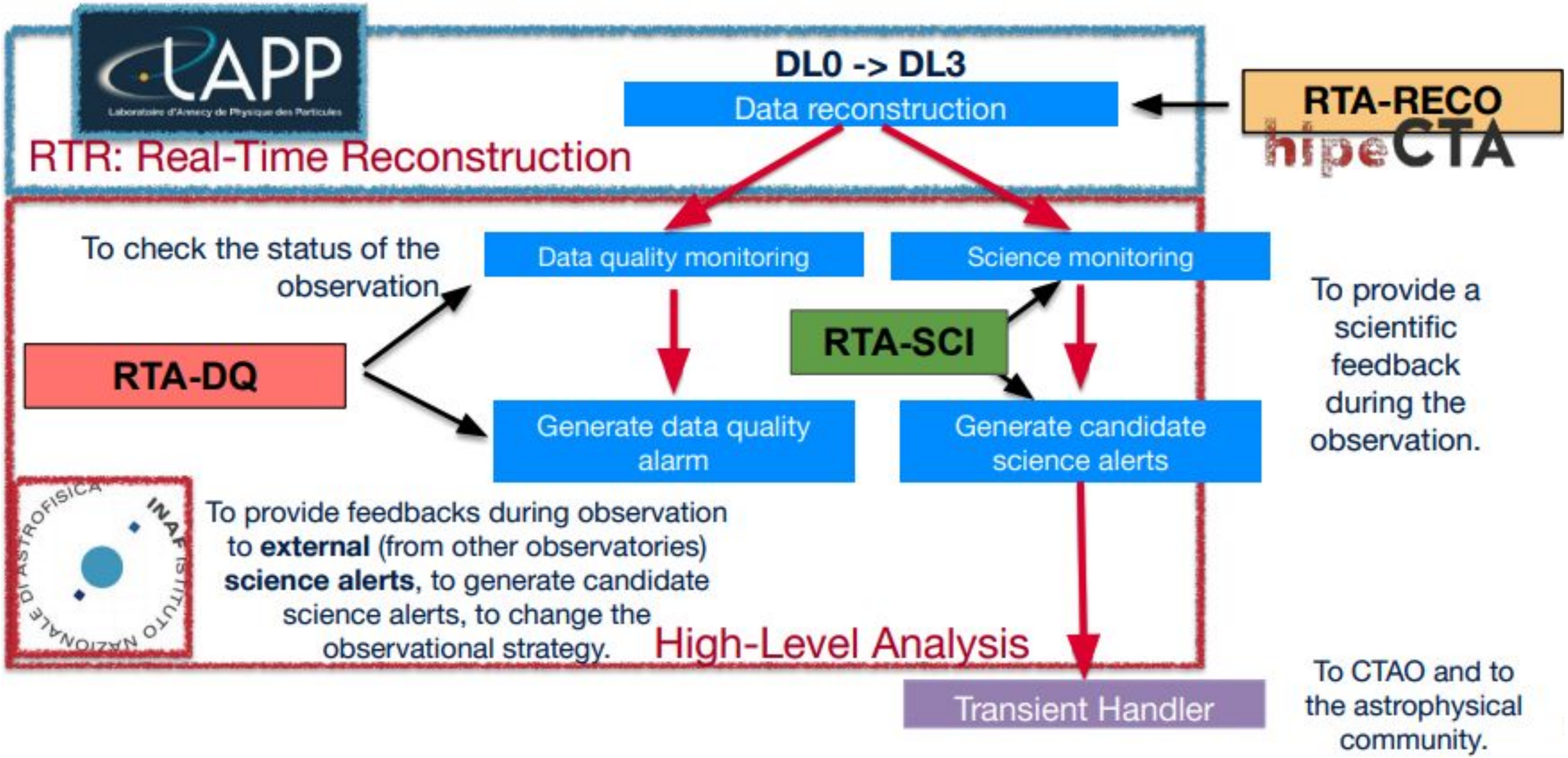
Generate candidate science alerts

ACADA/Transient Handler



To alert the astrophysical community immediately

ACADA/SAG: Workflow



SAG RTA-SCI
High Level science analysis

RTA-SCI

The RTA-SCI module starts from the CTA DL3 photon list, performing:

- science monitoring
- candidate science alert identification and communication to the ACADA Transient Handler

Pipeline time constraint: **5 seconds**

Observation time integration:

- fixed (e.g. time bins of 10 seconds for light curve production)
- variable (>10 seconds) integrating from the last acquired event (e.g. search for significant detection)

Interfaces:

- RTA-RECO and ACADA other subsystems
- Science tools (e.g. ctools, gammapy, ...) for the high level science analysis (maps, spectra, light curves, TS analysis, ...)

RTA-SCI and Science Tools

The RTA-SCI performance in high level science analysis has been explored:

- within the RTA-SCI pipeline prototype:
  - science tool: ctools
  - aim: testing the engineering functionality of the pipeline

- as stand-alone studies of the RTA ability in detecting transient events:
  - science tools: ctools, independent code, gammapy (very recently)
  - aim:
    - evaluating the sensitivity at short exposures
    - characterizing follow-up observations of GRBs (e.g. significance evolution, blind search, on/off wobble methods)

RTA-SCI Pipeline

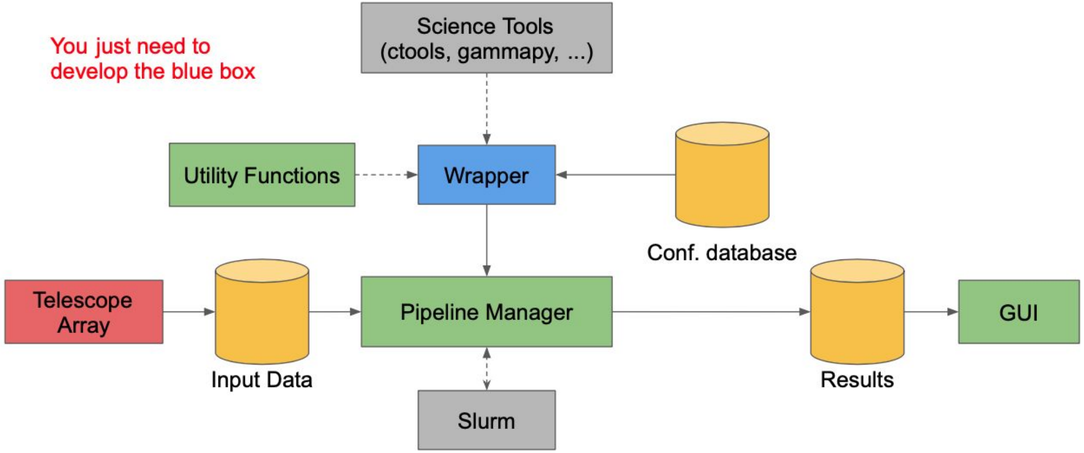
RTA-SCI pipeline functional schema

The RTA-SCI pipeline is interfaced to the Science Tools by means of a tool-dependant **python wrapper** that encodes the calls to:

- the DL3 database
- the configuration database and the utilities
- the Science Tools commands

The Science Tools are an external module.

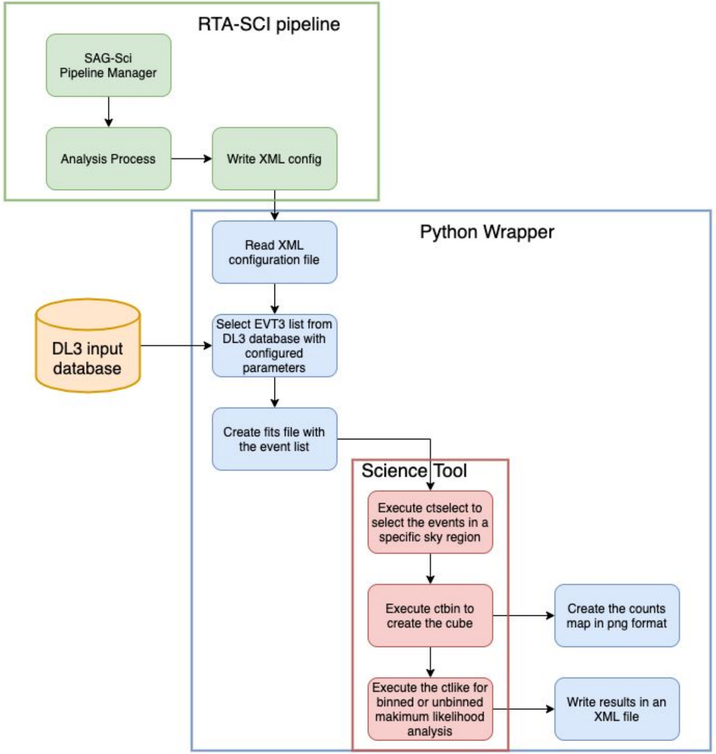
For more details: N. Parmiggiani, A. Bulgarelli, “CTA Real-Time Analysis Science Alert Generation Pipeline Logical Model and Interfaces”, CTA RTA-SCI Interface Document, 2019



RTA-SCI pipeline functional schema

- the RTA-SCI pipeline manager executes analysis runs within a CTA observation and it creates for each run an XML with all the information to run the analysis with a specific Science Tools

- the wrapper:
    - reads the XML file
    - selects the events from the database
    - create an event file with the data format required by the Science Tool (e.g. FITS)
    - runs the Science Tool commands
    - writes the output:
        - map image
        - XML file with results (e.g. significance, flux, position, etc.)



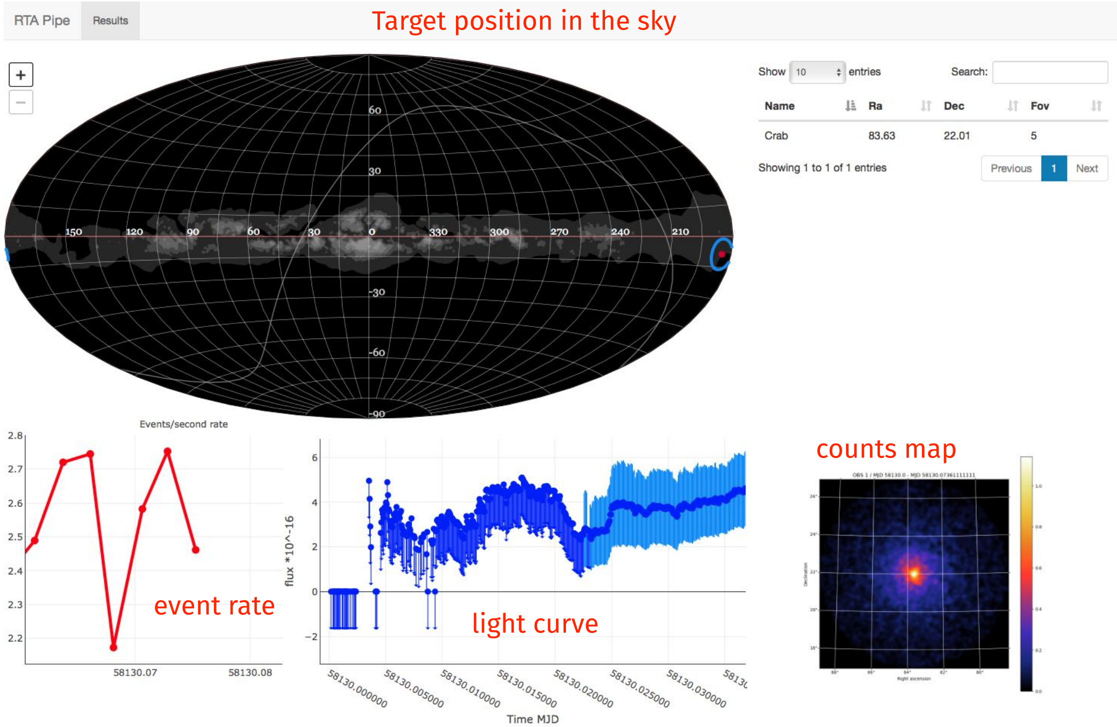
Credits: N. Parmiggiani

RTA-SCI pipeline: a working example

The RTA-SCI pipeline prototype has been currently interfaced and tested with ctools.

The results of the science analysis are showed in real-time.

A full benchmarking is yet to be performed, but the baseline execution times are few seconds for each run.



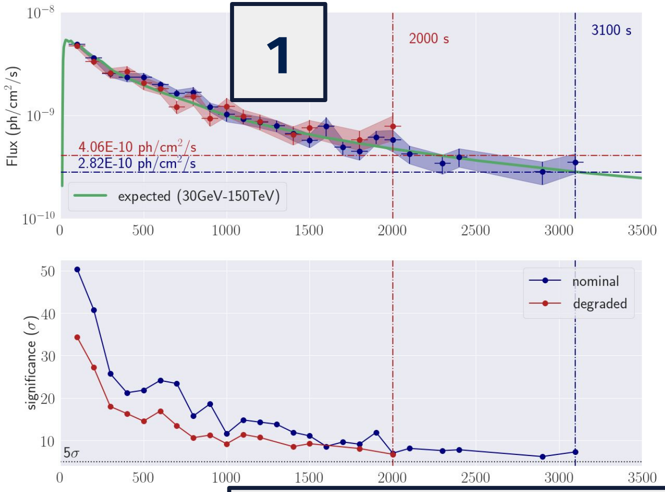
Engineering GUI for testing purposes. Credits: N. Parmiggiani

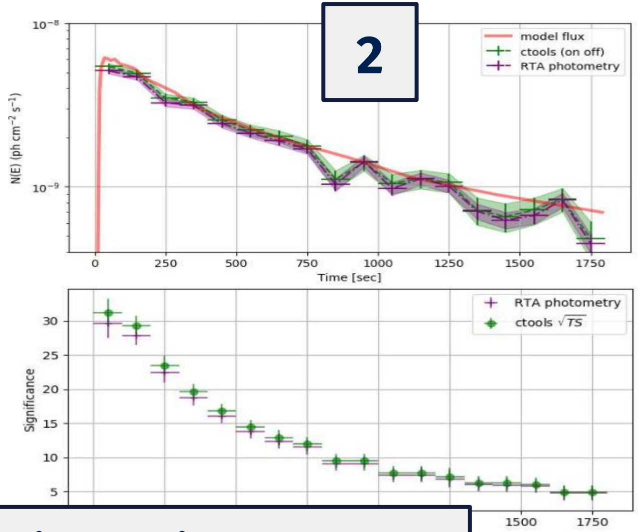
RTA scientific performance

High level analysis prototypes

1. Blind-search + full-FoV maximum likelihood pipeline with **ctools** software package for externally alerted follow-ups (A. Di Piano thesis)

2. RTA **photometry tool** pipeline (reflected/wobble methods) for targeted observations + comparison with ctools on/off analysis (S. Tampieri thesis)





Interface with CTA-ACADA-SAG is to be implemented

1. GRB analysis with ctools

Current Setup

- **CTA South full-array**
  - 30 GeV - 150 TeV integrated analysis
- **Degrees of freedom = 1**
  - Prefactor only free parameter
- **RTA sensitivity**
  - Degradation of the IRFs to achieve half nominal sensitivity

Blind-search: localisation

- **peak-search tool** : from ctools software package
  - Pointing retrieved from the merger sky map
    - **Pointing = max localisation probability**
  - No tiling strategies at the moment
    - **Localisation uncertainty ≈ FOV**

Detection: significance and flux

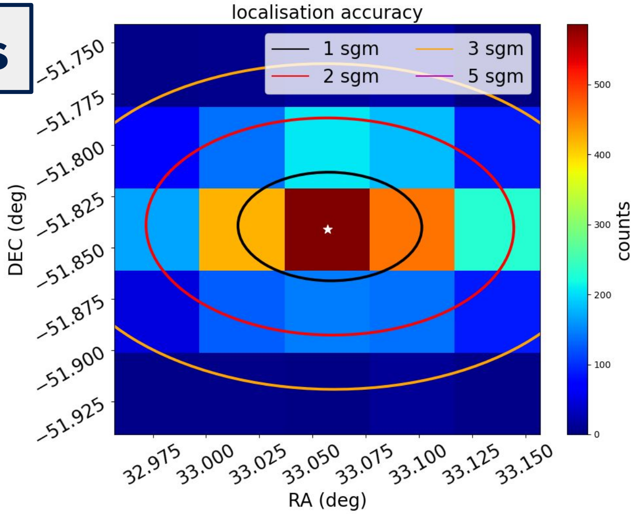
- **full FOV max likelihood** : from ctools software package
  - **Fixed position** retrieved during the blind-search
  - Input model: point-source simple power law
    - **Prefactor: free**
    - **Spectral index: fixed** (-2.48 at the moment)

Credits: A. Di Piano

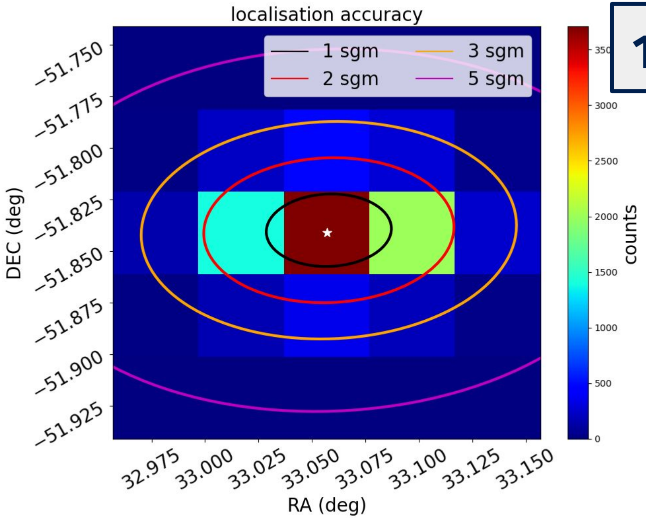
1. GRB analysis with ctools

Blind-search localisation and localisation uncertainty at very-short exposure time and at sensitivity limit

10 s



100 s



Gaussian confidence regions for 1, 2, 3 and 5 sigmas (Sample size = 10^4. **Pixel size = 0.02°).**

Credits: A. Di Piano

2. GRB on/off analysis

General Setup

- **CTA South full-array**
  - 30 GeV - 150 TeV integrated analysis
- **Degrees of freedom = 1**
  - Prefactor only free parameter
- **Methods comparison**
  - independent rta photometry tool
  - On/off ctools

Detection: significance and flux

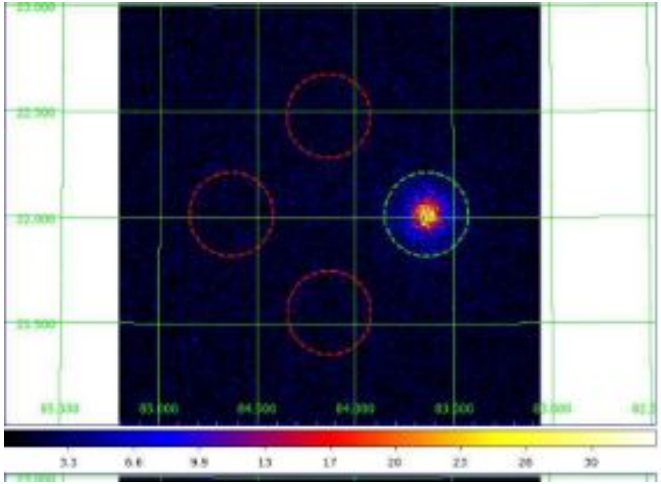
- **rta-photometry tool : analytical Li&Ma**
  - No model required
  - **N_s > 10**
  - **Comparable with ctools on/off**
  - Faster
- **csphagen : on/off from ctools software package**
  - **N_s > 10**
  - Input model: point-source simple power law
    - **Prefactor: free**
    - **Spectral index: fixed** (-2.48 at the moment)

Credits: S. Tampieri

2. GRB on/off analysis

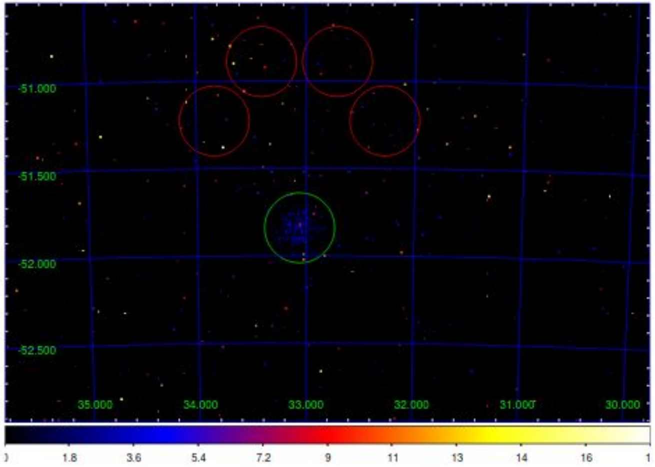
Implemented techniques for background estimation

rta photometry tool



Wobble

rta photometry tool and ctools



Reflection

Credits: S. Tampieri

Sensitivity at short exposures

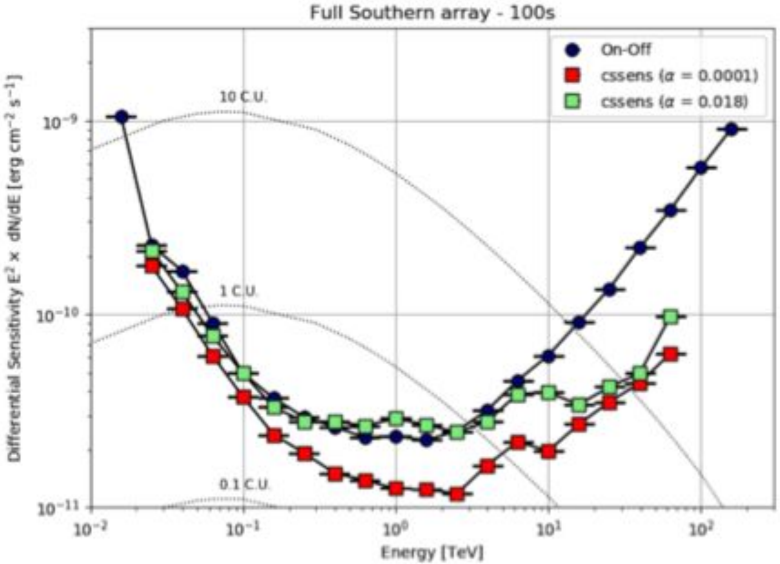
Full field Maximum Likelihood (ctools) vs On/Off

The performance of current analysis algorithms at short exposure has been studied within the ASWG Short term sensitivity group (coordinator: V. Fioretti):

- The full-field maximum likelihood technique predicts a sensitivity up to ~ 6 times better than what is obtained with the *On-Off* aperture photometry .

- The more optimistic prediction of the CTOOLS/*cssens* maximum likelihood script is given by the finer assumed knowledge of the background and the lack of a requirement on the minimum number of source counts. The two methods can converge if based on the same assumptions

- in gammapy:
    - are background systematics taken into account?
    - can the user require a minimum number of source events in the detection?



RTA-SCI and gammapy: summary

RTA-SCI and gammapy implementation

Next steps:

- RTA-SCI pipeline implementation:
  - the architecture is ready and we are available for developing a gammapy wrapper
  - we can provide the list of ctools algorithms currently implemented, to set-up a parallel analysis chain with gammapy within the pipeline
    - contact @Nicolò Parmiggiani

- comparison with current RTA science results using gammapy:
  - can gammapy … ?
    - Blind-search algorithm
    - Localisation performance near the sensitivity limit
    - Full FOV maximum likelihood
    - On/off analysis (wobble, reflection)
      - contact @Ambra di Piano
    - time variable models for transient detection
      - contact @Giulia Stratta and @Giovanni De Cesare

- … general questions? contact @Andrea Bulgarelli or @Valentina Fioretti

Thanks!

Back-up slides

Examples: GW follow-ups and divergent pointing

NB: RTA=SAG

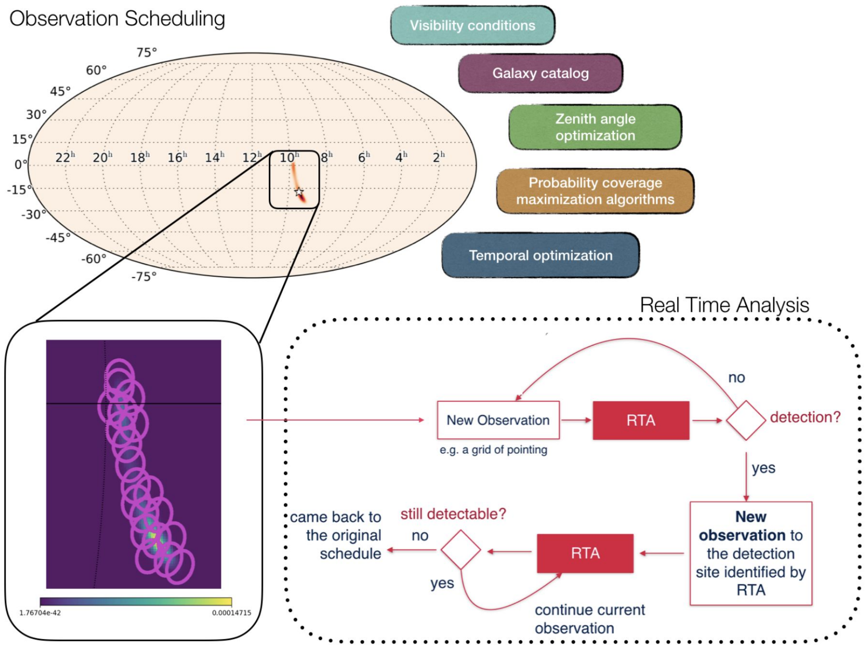
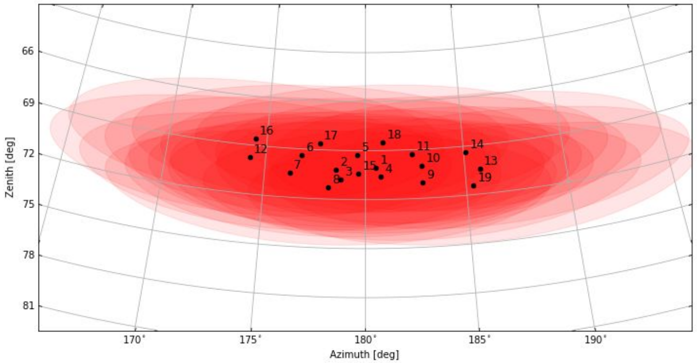
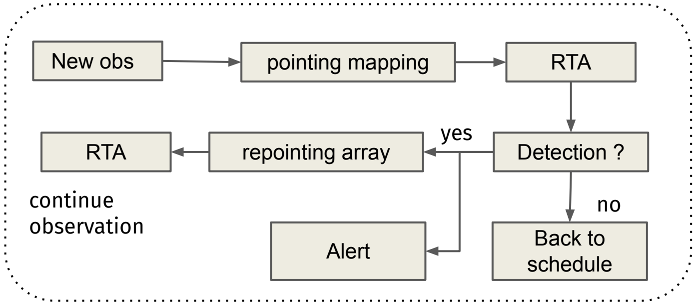


Figure 1: Illustration of the gravitational waves follow-up program of CTA

from M. Seglar-Arroyo+ IRCR2019: https://arxiv.org/abs/1908.08393

Or with divergent pointing (see dedicated presentation) - observing the sky region at once



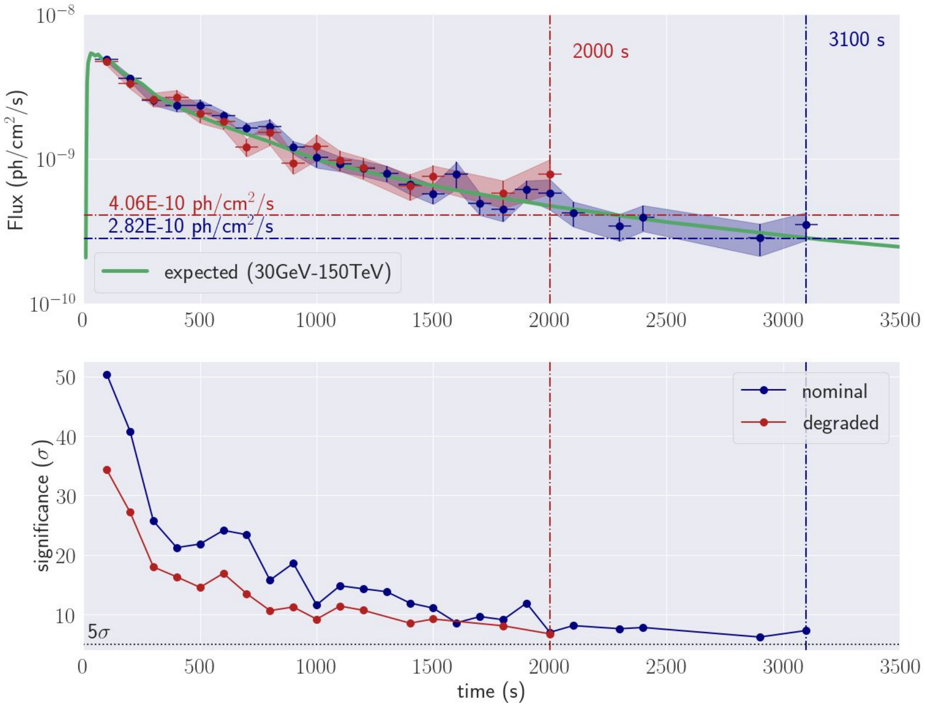


1. GRB analysis with ctools

100 s exposure example

- **Last positive detection**
  - Sigma ≥ 5

- Sensitivity comparison
  - **On-axis limit**



Credits: A. Di Piano

Next steps

Work in progress (A. Di Piano)

- **Visibility** implementation (both in simulations and analysis) → *tool ready*

Future tasks

- **Cumulative** exposure time (up to 30 mins)
- Varying the setup
- More refined IRF **degradation**
- Test with **newer templates** and **other source classes**

First inquiry steps with gammapy

work by G. Stratta and G. De Cesare

- We started from transient source simulations with gammapy (version tested 0.17) as first step for a preliminary study on the SAG requirements.
- A set of temporal models are available as gammapy classes for user scripts, but a power law temporal model (useful for e.g. gamma-ray burst simulations) is currently not available in gammapy.
- We have implemented this new model in our user code as a standard gammapy class PowerlawDecayTemporalModel(TemporalModel).
- Tests are still ongoing.
- A technical feedback in this workshop is welcome.