

GammaLib - Change request #2657

Allow usage of theta2 cut IRFs in On/Off analysis

08/01/2018 11:28 PM - Knödlseider Jürgen

Status:	Closed	Start date:	08/01/2018
Priority:	Normal	Due date:	
Assigned To:	Knödlseider Jürgen	% Done:	100%
Category:		Estimated time:	0.00 hour
Target version:	1.6.0		
Description			
<p>The H.E.S.S. community uses effective area components that have a theta2 cut applied (see https://gamma-astro-data-formats.readthedocs.io/en/latest/irfs/point_like/index.html). It should be possible to use these IRFs in csphagen (and ultimately in GCTAOnOffObservation without recomputing the PSF containment fraction.</p> <p>Such an IRF can be recognised using the following characteristics:</p> <ul style="list-style-type: none">• HDUCLAS3 keyword is POINT-LIKE• Presence of RAD_MAX keyword <p>So if a POINT-LIKE effective area is encountered, the code should not correct the ARF with the PSF containment fraction. The code should verify that the RAD_MAX keyword is consistent with the radius of the On region, and if this is not the case, throw an exception. Ideally, the code should also verify consistency with the On region centre, but it seems that this information is not available. Also, this logic only works for circular On regions, and an exception should be thrown if the On region is not circular. And finally, the code only works for point sources, so an exception should be thrown in case that a non point-source model is detected.</p> <p>All the checking could be implemented in the GCTAOnOffObservation::set method, and if everything is okay, a flag could be passed to GCTAOnOffObservation::compute_arf to switch off the theta2 cut. This should probably be reported in the ARF for diagnostics. A possibility is to add a RAD_MAX keyword in the ARF file in case that the IRF had such a parameter. This means that the GARf class needs to be modified.</p> <p>Reading of the HDUCLAS3 and RAD_MAX keywords needs to be implemented at the GCTAAeff2D level. From there they can be accessed by the GCTAOnOffObservation::set method.</p>			

History

#1 - 08/03/2018 12:10 PM - Knödlseider Jürgen

- Status changed from New to In Progress
- % Done changed from 0 to 10

Implement handling of RAD_MAX keyword in GCTAAeff2D class. If the keyword is present in the IRF the value will be read into a dedicated data member that is accessible via the GCTAAeff2D::rad_max() method. If a value is present it will be written back in a FITS table by the GCTAAeff2D::write() method.

#2 - 08/03/2018 03:02 PM - Knödlseider Jürgen

- Status changed from In Progress to Feedback
- % Done changed from 10 to 100

Everything was implemented as described about, and I did some checking using a faked IRF. The RAD_MAX value is written in the ARF file so that track is kept of what was done.

I put the issue now on Feedback since the implementation needs to be tested using real HESS data.

#3 - 12/12/2018 11:38 AM - Knödlseider Jürgen

- Status changed from Feedback to Closed