GammaLib - Action #2466

Computation time of diffuse map models

04/27/2018 02:38 PM - Cardenzana Josh

Status:	Closed	Start date:	04/27/2018
Priority:	Normal	Due date:	
Assigned To:	Cardenzana Josh	% Done:	100%
Category:		Estimated time:	0.00 hour
Target version:	1.6.0		

Description

This concerns the computation of a given spatial diffuse model convolution with the IRFs in cta_irf_diffuse_kern_phi. Two changes could reduce the amount of time spent in the relevant methods:

- Caching variables (specifically GVector and GPhoton objects) in cta_irf_diffuse_kern_theta and cta_irf_diffuse_kern_phi in GCTAResponse_helpers in order to reduce the time used for allocating/deallocating memory. These objects are allocated/deleted ~25 times for each bin evaluated in ctmodel and profiling suggests caching would noticeably reduce the computation time.
- GModelSpatialDiffuseCube::cube_intensity involves an evaluation of the underlying cube at two energy values, but at the same sky position. This makes two calls to GSkyMap::dir2pix which is computationally expensive. Caching the pixel value would result in half as many calls to this function.

History

#1 - 04/27/2018 03:35 PM - Cardenzana Josh

- Status changed from New to Pull request

- % Done changed from 0 to 100

The above changes have been made. The result is about a factor of 2 reduction in the tests I ran using ctmodel and the diffuse IEM background model. Affected files:

- GSkyMap (.hpp & .cpp)
- GCTAResponse_healpers (.hpp & .cpp)

Pull branch:

GammaLib: joshcardenzana/gammalib/2466-diffuse_map_speed

#2 - 04/27/2018 03:54 PM - Cardenzana Josh

- Status changed from Pull request to In Progress

- % Done changed from 100 to 90

I mistakenly branched this change from my fix on issue #2463. I'll fix that before the pull.

#3 - 04/27/2018 04:16 PM - Cardenzana Josh

This branch now stands on its own. The computation times are unaffected.

Pull branch:

GammaLib: joshcardenzana/gammalib/2466-diffuse_map_speed

#4 - 04/27/2018 04:17 PM - Cardenzana Josh

- Status changed from In Progress to Pull request

- % Done changed from 90 to 100

#5 - 06/05/2018 03:12 PM - Knödlseder Jürgen

Improved GSkyMap::dir2pix caching by introducing a specific last sky direction for the caching.

#6 - 06/05/2018 03:58 PM - Knödlseder Jürgen

- Status changed from Pull request to Closed
- Target version set to 1.6.0

Merged into devel.

#7 - 06/06/2018 11:34 AM - Knödlseder Jürgen

- Status changed from Closed to Feedback

It turned out that the caching was not OMP thread safe. Furthermore, it is more universal to do the caching at the level of the GWcs::pix2dir() and GWcs::dir2pix() methods so that all clients can benefit from that.

So I move the caching implementation there and tried to make the core OMP thread safe. Still need to check whether any thread conflicts come up.

#8 - 07/20/2018 11:41 PM - Knödlseder Jürgen

- Status changed from Feedback to Closed