

ctools - Bug #2660

csphagen "No point spread function information found in response." for nthreads<>1

08/06/2018 02:03 PM - Specovius Andreas

Status:	Closed	Start date:	08/06/2018
Priority:	High	Due date:	
Assigned To:	Knödlseher Jürgen	% Done:	100%
Category:		Estimated time:	0.00 hour
Target version:	1.6.0		

Description

When executing csphagen with parameter nthreads=1 everything seems to work fine.
When executing csphagen with parameter nthreads not set (default 0) or set to 0 or >1 an error occurs (see below!).
This looks like something does not work properly in the multithreaded case!

ValueError Traceback (most recent call last)

<ipython-input-20-1ece58d054b9> in <module>()

```
50 # phagen['nthreads'] = 1
51 phagen['use_irf_bkg'] = False
--> 52 phagen.execute()
53
54 print("- %s" % phagen.obs())
```

/home/woody/caph/sn0533/git/ctools_install/lib/python2.7/site-packages/ctools/tools.pyc in _execute(self)

```
1539 self.logFileOpen()
1540 self._read_ahead(True)
-> 1541 self.run()
1542 self.save()
1543 csobservation.execute = _execute
```

/home/woody/caph/sn0533/git/ctools_install/lib/python2.7/site-packages/cscripts/csphagen.pyc in run(self)

```
587 args = [(self, '_process_observation', i)
588          for i in range(self.obs().size())]
--> 589 poolresults = mputils.process(self._nthreads, mputils.mpfunc, args)
590
591 # Construct results
```

/home/woody/caph/sn0533/git/ctools_install/lib/python2.7/site-packages/cscripts/mputils.pyc in process(nthreads, function, args)

```
100
101 # Run function in parallel
--> 102 results = pool.map(function, args)
103
104 # Close pool and join
```

/apps/python/2.7-anaconda/lib/python2.7/multiprocessing/pool.pyc in map(self, func, iterable, chunksize)

```
251 ""
252 assert self._state == RUN
--> 253 return self.map_async(func, iterable, chunksize).get()
254
255 def imap(self, func, iterable, chunksize=1):
```

/apps/python/2.7-anaconda/lib/python2.7/multiprocessing/pool.pyc in get(self, timeout)

```
570 return self._value
571 else:
--> 572 raise self._value
573
574 def _set(self, i, obj):
```

ValueError: *** ERROR in GCTAResponseIrf::psf_delta_max(double&, double&, double&, double&, double&): Invalid value. No point spread function information found in response.
Please make sure that the instrument response is properly defined.

History

#1 - 09/07/2018 05:07 PM - Knödlseider Jürgen

- Assigned To set to Knödlseider Jürgen
- Target version set to 1.6.0

Can you tell me what type of PSF you are using? Best would be to have some test data since I suspect that something goes wrong with the pickling.

#2 - 09/08/2018 10:09 AM - Knödlseider Jürgen

- Project changed from GammaLib to ctools
- Target version deleted (1.6.0)

#3 - 09/09/2018 11:58 AM - Specovius Andreas

user#3 wrote:

Can you tell me what type of PSF you are using? Best would be to have some test data since I suspect that something goes wrong with the pickling.

I am using a table psf (-> loaded as GCTAPsfTable object).

What data do you need exactly for testing? Only the psf response file(s)?

#4 - 09/10/2018 09:25 AM - Knödlseider Jürgen

- Target version set to 1.6.0

I would need of one the PSF response files to make sure that the pickling works as expected.

#5 - 09/10/2018 11:01 AM - Specovius Andreas

I just sent you one of the H.E.S.S. table psf response files via email!

#6 - 09/10/2018 11:25 AM - Knödlseider Jürgen

- Status changed from New to In Progress
- % Done changed from 0 to 50

Looks like the issue was that the default PSF filename extension that is used if no extension name is specified was set to "POINT SPREAD FUNCTION" instead of "PSF_2D_TABLE" in GCTAPsfTable.hpp.

In addition, the pickling of @@ does not work for a PSF table, which is likely the origin of the problem:

```
>>> irf=gammalib.GCTAResponseIrf()
>>> irf.load_psf('hess_psf_table_047802.fits')
>>> print irf
=== GCTAResponseIrf ===
Caldb mission .....:
Caldb instrument .....:
Response name .....:
Energy dispersion .....: Not used
```

```

Save energy range .....: undefined
=== GCaldb ===
Database root .....: /usr/local/gamma/share/caldb
=== GCTAPsfTable ===
Filename .....: hess_psf_table_047802.fits
Number of energy bins .....: 32
Number of offset bins .....: 6
Number of delta bins .....: 144
Log10(Energy) range .....: 0.00999999977648258 - 100 TeV
Offset angle range .....: 0 - 2.5 deg
Delta angle range .....: 0 - 0.665605783462524 deg
>>> dump=pickle.dumps(irf)
>>> obj=pickle.loads(dump)
>>> print obj
=== GCTAResponseIrf ===
Caldb mission .....:
Caldb instrument .....:
Response name .....:
Energy dispersion .....: Not used
Save energy range .....: undefined
=== GCaldb ===
Database root .....: /usr/local/gamma/share/caldb

```

#7 - 09/10/2018 11:29 AM - Knödseder Jürgen

The pickling method was only implemented for the case of a calibration database. I'll change that.

```

%pythoncode {
    def __getstate__(self):
        state = (self.rspname(), self.caldb())
        return state
    def __setstate__(self, state):
        if state[0]:
            self.__init__(state[0], state[1])
        else:
            self.__init__()
}

```

#8 - 09/10/2018 12:27 PM - Knödlseider Jürgen

- % Done changed from 50 to 60

I'm making progress on the issue. To be sure that all other IRF components can be correctly digested by GammaLib, could you please also send an effective area file and a background file? Thanks in advance.

#9 - 09/10/2018 01:17 PM - Specovius Andreas

I just sent you one of the H.E.S.S. effective area and background response files via email!

#10 - 09/10/2018 03:06 PM - Knödlseider Jürgen

- Status changed from *In Progress* to *Pull request*

- % Done changed from 60 to 100

Thanks. I changed a bit the code so that all files are now digested correctly. I also added a pickling test.

I'm currently integrating the code.

#11 - 09/10/2018 03:18 PM - Knödlseider Jürgen

- Status changed from *Pull request* to *Feedback*

The code is now merged into devel (changes only in GammaLib). Could you please check if csphagen now also works for nthreads!=1?

#12 - 09/11/2018 08:46 AM - Specovius Andreas

For me csphagen runs now with nthreads<>1 without problems.

Good job!

#13 - 09/11/2018 09:40 AM - Knödlseider Jürgen

- Status changed from *Feedback* to *Closed*