

## GammaLib - Action #261

Feature # 226 (Closed): Parallelize maximum likelihood computation

### Study possible options for parallelization

07/02/2012 12:05 PM - Knödseder Jürgen

<b>Status:</b>	Closed	<b>Start date:</b>	06/16/2012
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assigned To:</b>		<b>% Done:</b>	100%
<b>Category:</b>		<b>Estimated time:</b>	35.00 hours
<b>Target version:</b>	Stage Jean-Baptiste Cayrou		
<b>Description</b>			
As a first step of the project we should study the possible options that exists for code parallelization.			
We recall the requirements:			
<ul style="list-style-type: none"><li>• no code dependencies (use only native C++ features)</li><li>• no limits due to Python Global Interpreter Lock (GIL)</li><li>• support concurrent memory access</li><li>• enabling / disabling during compile time</li><li>• selection of number of cores during setting of environment variable</li></ul>			

### History

#### #1 - 07/02/2012 12:07 PM - Knödseder Jürgen

- Description updated
- Estimated time set to 35.00
- Remaining (hours) set to 35.0

#### #2 - 07/02/2012 12:28 PM - Anonymous

- Status changed from New to In Progress

#### #3 - 07/02/2012 12:53 PM - Anonymous

Some information about Swig and threads:

<http://stackoverflow.com/questions/2510696/allowing-threads-from-python-after-calling-a-blocking-i-o-code-in-a-python-extend>

<http://matt.eifelle.com/2007/11/23/enabling-thread-support-in-swig-and-python/>

<http://code.activestate.com/recipes/52294-use-modules-generated-with-swig-in-a-multi-thread-/>

#### #4 - 07/05/2012 04:32 PM - Anonymous

- Status changed from In Progress to Resolved

#### #5 - 07/05/2012 04:59 PM - Anonymous

- % Done changed from 0 to 100

Finally, I will use the **OpenMP API** which allow multi-threading just with directive for compiler.  
For instance the following directive will share the loop on several threads.

```
#pragma omp for
```

```
for(int i=0;i<n;i++)  
{  
...  
}
```

It is easy to parallelize a code. When the option is disable the compiler does not care the pragma lignes and it works like for a mono thread code.

Many compilers implement the OpenMP API (with gcc, just add "-fopenmp" to active it).

Moreover it is possible to set the number of thread with an environment variable ( OMP\_NUM\_THREADS )

**#6 - 07/10/2012 03:54 PM - Anonymous**

- Status changed from Resolved to Closed
- Remaining (hours) changed from 35.0 to 0.0

**#7 - 07/28/2012 12:53 AM - Knödlseeder Jürgen**

- Target version deleted (Stage Jean-Baptiste Cayrou)

**#8 - 07/28/2012 12:53 AM - Knödlseeder Jürgen**

- Target version set to Stage Jean-Baptiste Cayrou