
Subject: Confusion about nthread and ncore
Posted by [chuxiangning](#) on Thu, 20 Nov 2014 18:04:08 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi All

It has always been confusing to me how IDL handles nthread and ncores.

I have an i7 which has 4 physical cores and 8 threads.

Matlab recognizes i7 as 4 cores and uses 4 threads. IDL recognizes i7 as 8 threads.

In my opinion, in a computation-heavy program, 4 threads should be used instead of 8, because using 8 threads actually slows down the program.

Is this correct?

Subject: Re: Confusion about nthread and ncore
Posted by [Sergey Anfinogentov](#) on Fri, 21 Nov 2014 11:59:45 GMT
[View Forum Message](#) <> [Reply to Message](#)

It depends on the code.

The best way is to make an experiment using your computer and your code.

You can measure the execution time of your program with default settings then you can change threads count via "cpu,TPOOL_NTHREADS = 4" and ,measure the execution time again.

- > Hi All
 - > It has always been confusing to me how IDL handles nthread and ncores.
 - > I have an i7 which has 4 physical cores and 8 threads.
 - > Matlab recognizes i7 as 4 cores and uses 4 threads. IDL recognizes i7 as 8 threads.
 - > In my opinion, in a computation-heavy program, 4 threads should be used instead of 8, because using 8 threads actually slows down the program.
 - > Is this correct?
-

Subject: Re: Confusion about nthread and ncore
Posted by [chuxiangning](#) on Tue, 25 Nov 2014 17:38:45 GMT
[View Forum Message](#) <> [Reply to Message](#)

Thanks for your suggestion!

I've tested my program under different cases by setting nthreads to different values.

(cpu,tpool_nthreads = 3)

I have an i7 cpu, which has 4 cores and 8 threads(2 per cpu).

It turns out the program runs faster when I set nthread to 4 instead of 8 (IDL default). This result is expected anyway.

As a suggestion to IDL, I would suggest they change the default value of nthreads according to number of cores.

nthread	time (sec)
1	7.923
2	4.458
3	3.451
4	3.055
6	3.162
8	3.298

Subject: Re: Confusion about nthread and ncore

Posted by [victor.cazcarra](#) on Thu, 27 Nov 2014 09:05:51 GMT

[View Forum Message](#) <> [Reply to Message](#)

El martes, 25 de noviembre de 2014 18:38:48 UTC+1, chu xiangning escribió:

> Thanks for your suggestion!

> I've tested my program under different cases by setting nthreads to different values.
(cpu,tpool_nthreads = 3)

> I have an i7 cpu, which has 4 cores and 8 threads(2 per cpu).

> It turns out the program runs faster when I set nthread to 4 instead of 8 (IDL default). This result is expected anyway.

> As a suggestion to IDL, I would suggest they change the default value of nthreads according to number of cores.

>

>

> nthread time (sec)

> 1 7.923

> 2 4.458

> 3 3.451

> 4 3.055

> 6 3.162

> 8 3.298

Hi,

I think that it always depends on your program. if your program can do some works in parrallel automatically, for example if you work with big matrices IDL do some parallelization so it is better not using all the threads.

I was working with IDL_bridge trying to split a big loop to work in parallel and for me it worked better using the number of threads instead of the number of cores.

I am not a hardware expert but I guess that it also depends on the structure of your processor.

In any case, It is a good idea always testing your program and choose the best compromise between number of threads and time.

Subject: Re: Confusion about nthread and ncore
Posted by [chuxiangning](#) on Sun, 30 Nov 2014 00:40:25 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Thursday, November 27, 2014 1:05:52 AM UTC-8, Víctor Cazcarra wrote:

> El martes, 25 de noviembre de 2014 18:38:48 UTC+1, chu xiangning escribió:

>> Thanks for your suggestion!

>> I've tested my program under different cases by setting nthreads to different values.

(cpu,tpool_nthreads = 3)

>> I have an i7 cpu, which has 4 cores and 8 threads(2 per cpu).

>> It turns out the program runs faster when I set nthread to 4 instead of 8 (IDL default). This result is expected anyway.

>> As a suggestion to IDL, I would suggest they change the default value of nthreads according to number of cores.

>>

>>

>> nthread time (sec)

>> 1 7.923

>> 2 4.458

>> 3 3.451

>> 4 3.055

>> 6 3.162

>> 8 3.298

>

> Hi,

>

> I think that it always depends on your program. if your program can do some works in parrallel automatically, for example if you work with big matrices IDL do some parallelization so it is better not using all the threads.

>

> I was working with IDL_bridge trying to split a big loop to work in parallel and for me it worked better using the number of threads instead of the number of cores.

>

> I am not a hardware expert but I guess that it also depends on the structure of your processor.

>

> In any case, It is a good idea always testing your program and choose the best compromise between number of threads and time.

Thanks for weighing in.

Yes it depends on the programs.

In some of my program, I also have large matrix and IDL automatically use parallel calculation.

The problem is, by setting CPU to use number of cores instead of number of threads, the program runs faster.

It is also true if you run several IDLs together.
