Unable to get a cisTEM job to parallelize on multiple nodes in cluster

Tue, 01/16/2018 - 17:03

saikat

Unable to get a cisTEM job to parallelize on multiple nodes in cluster

Hi,

I am trying to run cisTEM after requesting an interactive session over mutiple nodes in our cluster. The jobs are just running on processors on single node and not using the other nodes.

The file system in the cluster is gpfs and we are using PBS job submission.

Do you have any suggestions for effectively running cisTEM jobs using mutiple nodes.

Thanks

timgrant

Hi,

If using a PBS cluster, can't you just qsub all the jobs as single jobs and not worry about running an interactive session?

I think if you want to run in an interactive session this way, you will need to have a run profile that explicitly ssh's to the machines directly, however each time you start a session the machines will likely be different, so this isn't really optimal.

Tim

Tue, 01/16/2018 - 17:16 <u>(Reply to #2)</u>

saikat

Hi Tim,

Thanks for your response.

So far I have been running the job using the GUI after requesting the interactive session.

What is the best way to launch it as conventional jobs? Do you have any template job file which you can share.

Thanks

saikat

job

Hi Tim,

The reason I asked the above question regarding job file is because we were trying earlier to launch jobs on mutiple nodes using the cisTEM GUI with the following job template file:

#!/bin/csh
#PBS -l nodes=3:ppn=40
module load cistem
$variable

We would specify this job file through the Manager Command at the Settings window of cisTEM.

Unfortunately this didn't help get the job launched on multiple requested nodes.

So any suggestions will be really useful.

Thanks

timgrant

Hi,

I think you need to launch each job as 1 node, 1 ppn. I have not done this on PBS explicitly, but I imagine you need to make a script called "submit_cistem_job.b" or something that looks something like :-

#!/bin/csh
#PBS -l nodes=1:ppn=1
/path_to_cistem_binaries/$1 $2 $3 $4

Then your run command would be something like "qsub submit_cistem_job.b $command"

Your manager command would depend on whether you were running on the head node in which case it would just be "/path_to_cistem_binaries/$command", or you may need to change this to ssh to the head node etc, depending on exactly where you are running the gui.

You will probably need to edit this somewhat, but I hope it gives you the general idea.

Please feel free to ask questions if this doesn't make sense.

Cheers,

Tim

saikat

Thanks Tim

Thanks Tim

I will try it and let you know.

-Saikat

saikat

Hi Tim,

A quick question regarding all the $1,$2,$3 and $4. I guess $1 is the command itself but what are $2,$3,$4?

Is there a one-liner command that will get ride of the commas.

Thanks

Sat, 02/03/2018 - 16:52 <u>(Reply to #7)</u>

rnavaza

A way to make cisTEM work with Torque

Hi,

I got cisTEM to work on multiple nodes in a cluster managed with Torque. Here's the way to do it:

0) Verify that the cluster works : same home folder for your user account on each node, ssh without password between the submit host and the nodes, qsub command in PATH, and so on...

1) Now create a script (that I called submit.sh) with the content:

```
#!/bin/bash
cat - <<EOF | qsub
#!/bin/bash
#PBS -N cisTEM.${1}
#PBS -l nodes=1:ppn=1
/your_path_to_cisTEM_binaries/${@}EOF
```

2) Give that file the executable rights:

```
chmod 755 submit.sh
```

3) In cisTEM settings, add a new "Run Profile" with the following parameters :

Manager Command: `/your_path_to_cisTEM_binaries/$command`
Gui Address: `Automatic`
Controller Address: `Automatic`
Command -> Edit:
    Command: `/your_path_to_your_script/submit.sh $command`
    No. Copies: `(Total number of CPUs you want to allocate in the whole cluster)`
    Delay (ms): `10`

That's all. Hope this helps ;-)

Fri, 08/31/2018 - 13:09 <u>(Reply to #8)</u>

davidgenemorgan

cisTEM and TORQUE

Hi,

I am finally trying to get cisTEM to play nicely with several of the clusters here at IU and have a number of questions...

In the scenario that uses a submit.sh script and TORQUE, what does "No. Copies" control? I can see how it would tell cisTEM the number of submit.sh commands to run, but it also might do nothing or something totally different...

Also, with our clusers, I need to estimate the amount of CPU (#PBS -l walltime=XX:XX:XX) needed. I realize that with some clever scripting, I can add any number of additional command line arguments to the submit script (always leaving "the rest" as the command to run). Are there ways to estimate how long various jobs in cisTEM will take per core?

Thanks for any help someone can provide.

Thu, 09/06/2018 - 09:39 (Reply to #9)

I've been using cisTEM for a few reconstrunctions and that's what I noticed.

First I'll group cisTEM's "Actions" in two categories :

- The "flat" category (with only one cycle) : "Align Movie", "Find CTF" and "Find Particles"
- The "cyclic" category (with multiple cycles) : "2D Classify", "Ad-initio 3D" and "Auto-Refine"

"No. Copies" controls the number of "submit.sh" (alas qsub) commands that will be launched for one action, PER CYCLE.

Each cycle can be split in two steps :

1. The "preprocessing" step that will be made by the GUI using only 2 CPUs (it's not "cluster friendly" because it's the GUI that's doing it).
2. The "qsub" step that will do the calculations on the cluster.

For the "cyclic" category actions I didn't have any "qsub" step that reached more than a few minutes, BUT the preprocessing step may take DAYS with big particles !! I advise you to run the cisTEM GUI on the quickest frontal server that you have in the cluster.

For the "flat" category actions, your set of Movies/Micrographs will be divided in "No. Copies". The time that each qsub job will take to finish is dependent of the number of movies/micrographs and the "No. Copies".

Obviously you'll need to modify the submit.sh script to fit your needs, as I did. In mine I suppressed the output (there's a lot per action) and I added an option for memory allocation.

Here's my script :

```
#! /bin/bash
# physical memory per process (possible suffixes are
b, kb, gb, tb)
pmem=
# maximun time per qsub job (format is HH:MM:SS)
walltime=
while getopts ":m:w:" OPTION
do
   case "${OPTION}" in
      m) pmem="${OPTARG}";;       w)
walltime="${OPTARG}";;
   esac
done
```

```
shift $((OPTIND - 1))
cat - <<EOF | qsub
#! /bin/bash
#PBS -N cistem.${1}
#PBS -e /dev/null
#PBS -o /dev/null
#PBS -l nodes=1:ppn=1${pmem:+#PBS -l pmem=${pmem}}
${walltime:+#PBS -l walltime=${walltime}}
/your_path_to_cistem_binaries/${@}
EOF
```

And the new command that would work for almost everything should be :

```
/your_path_to_your script/submit.sh -m 5gb -w 00:30:00 $comman
```

... unless you run into a very big particle at high resolution that requires 28GB of RAM per CPU. But you won't know unless you try it.

Thu, 09/06/2018 - 10:56 <u>(Reply to #10)</u>

davidgenemorgan

cisTEM and TORQUE

Thanks for the detailed reply. I had come up with a very similar but slightly different script - glad to know I am on the right track!

On my system, I think I have proved that "No. copies" = 2 works best for the classification, ab initio model building and refinement. When I ask for more, the same job appears to take considerably longer (though the load on the queue affects this). Also, many of the spawned jobs are totally devoid of output (as painfull as it was, I have run several jobs with full output in an attempt to tease out exactly what is happening), and that would seem to indicate that there was no point in spawing them in the first place. I have _not_ systematically searched this parameter space, but I did try things like No copies = cores/node and some numbers based on that (both hyperthreading and cutting down the request by factors of two). Does that fit with your observations?

In terms of scaling with data set size, I'm sure that as the box size of the particles increases, the amount of processing time will also increase. So for a data set of 2000 particle images, 200 x 200 boxes will take longer per cycle (I will here refer to an individual qsub command as a cycle) to process than 125 x 125 boxes.

However, it's not clear to me what happens when the data set of 200 x 200 boxes increases from 2000 particle images to 40000. It will obviously take much more CPU, but how cisTEM approaches that situation is totally unknown. It could be that cisTEM always "chunks out" (say) 200 particles to be processed per cycle (and that 2000 particles would be run in 10 cycles). That is what the original EMAN did in terms of how it parallelized operations. It could also be that cisTEM is doing something completely different, like simply breaking the particle dataset into 25 pieces (running one piece per cycle) regardless of the number of particles. That seems less likely, but since virtually everything is hidden under the hood in cicTEM, it's simply impossible to know. I am slowly (and empirically) figuring out some of this, but I'd be grateful if you (or anyone) can cast any light on how things work.

And finally, just to clarify my situation, I have neither control nor knowledge of who might be using cisTEM on the university cluster. However, I will definitely be the person asked to deal with any and all problems users will encounter. This means that I am currently trying to anticipate "use models" of how cisTEM will be run so that common problems can be handled more easily.

Again, thanks for your reply.

Thu, 09/06/2018 - 13:57 <u>(Reply to #11)</u>

rnavaza

cisTEM and TORQUE

Well, I processed 25000 particles of 512x512 with 128 copies. Compared to 16 or 32 or 64 copies, the time don't scale linearly but It does go faster; so I don't understand how 2 copies can be the best configuration for you.

You probably noticed that there's no point in modifying "nodes=1:ppn=1" because it will result in cisTEM and the cluster to be out of sync (cisTEM will request more CPUs than what it really uses).

Speaking about potential problems I can tell you about two ^^

The cluster have 8GB of RAM per CPU so I didn't run onto any RAM related problem until I processed a dataset of 5000 particles of 2048x2048. This dataset required more than 30GB per core for the "Ab-initio 3D"... The consequence is that I had to reboot a few nodes after they got stuck wihout any more RAM and swap... And I added the memory allocation option to the script.

The other issue is probably a bug and is not always true : While running Ab-initio or Auto-Refine actions, when, lets say, you configured a lot of copies to run at the same time, __ and some are running and others are still in queue __, if the ones running complete before the ones in queue start then the "job/action" will never complete the cycle. There's a way to work around it by rewriting the script, making it count all the calls to himself (using flock and a timer) and launching only one qsub for all the "copies" (using pbsdsh). I didn't implement it for now but the script would have to replace "nodes=1:ppn=1" by "procs=NumberOfCountedCopies".

Thu, 09/06/2018 - 19:28 <u>(Reply to #12)</u>

davidgenemorgan

cisTEM and TORQUE

Thanks for your input. I have just been experimenting with the tutorial data set and for example, the ab initio model generation needed 5+ hr (clock-time) with 33 copies and only about 2 with 17 (and both are much slower than running on a login node, but that's not a very good thing to advocate). In both the 33 and 17 copy cases, the vast majority of those copies generated empty output (stdout, stderr), which indicates to me that they never should have been run, aren't doing anything useful, etc. I'm not even sure why this happens, but it can't be good. With 2 copies and classification or refinement, I get about the same amount of clock-time as with 17 or 33 copies and none of these copies generate no output (while many of the copies with 17 or 33 do).

I am struggling to get the ab initio step to work with only 2 copies. It keeps hanging, though I don't think it is the reason you mention (though I could be wrong - is there some diagnostic output for that effect?). Hanging also happened with both 17 and 33 copies (and directly on the login node), so I don't think it is the copy number that is causing me trouble.

I might see a different copy number effect with a dataset different than the tutorial, and it's defnitely good to know that you get completely different results from what I have seen.

As a more general comment, it may be worth getting the cisTEM developers to think harder about dealing with queuing systems... And maybe if I understood how cisTEM parallelize things better, I'd be able to deal with some of these issues myself...

Tue, 09/11/2018 - 06:49 <u>(Reply to #13)</u>

I've been modifying the submit script to do what I've talked about previously (regrouping all the calls to the script and submit only one job).

The problem is that it only works correctly with the Moab scheduler, I couldn't get it do what it's supposed to do with Torque+Maui (unless downgrading Torque to version 2.5 and Maui to 3.2).

```
#! /bin/bash
# get the current number of instances of this script
that have been# launched by cisTEM and increment it
lockfile="/dev/shm/cisTEM.${@: -4:1}.${@: -1:1}"
lockcount=$(flock -x ${lockfile} /bin/sh -c "read
count <
${lockfile}
                                            let
'count = count +
1'                                          echo
\${count} >
${lockfile}
                                            echo
\${count}")
# only the first instance will launch the qsub
command, for others: Goodbye !if [ "${lockcount}" -ne
1 ]; then exit 0 ; fi
# wait long enough for cisTEM to finish submiting all
the "copies"
sleep 3
# now get the total number of CPUs requested by cisTEM
procs=$(cat ${lockfile})
rm -f ${lockfile}
#
###############################################################
#
# physical memory per process (possible suffixes are
b, kb, gb, tb)
mem=
# maximun time per qsub job (format is HH:MM:SS)
walltime='00:30:00'
while getopts ":m:w:" OPTION
do
   case "${OPTION}" in
      m) mem="${OPTARG}";;        w)
walltime="${OPTARG}";;
   esac
done
shift $((OPTIND - 1))
# submit the job with qsub
cat - <<EOF | qsub
```

```
#! /bin/sh
#PBS -N cisTEM.${1}
#PBS -j oe
#PBS -l walltime=${walltime}#PBS -l procs=${procs}
${mem:+#PBS -l pmem=${mem}}/usr/bin/pbsdsh
/your_path_to_cistem_binaries/${@}
EOF
```

For your problem about the processes that don't do anything, does the GUI says that all the processes have connected ?

You may also want to look at the "Controller Address" and "Gui Address" in the settings of cisTEM. With it you can force the use of a specific IP interface, with some caveats: it will only work as intended if the server running the GUI and the controller job is alway the same.

Wed, 08/19/2020 - 08:58 <u>(Reply to #14)</u>

abhisek

error in cistem batch script

Hi,

I have been trying to use the following script for my cistem on SGE:

#!/bin/bash

cat - <<EOF | qsub

#!/bin/bash

#PBS -N cisTEM.${1}

#PBS -l nodes=1:ppn=1

/path/cistem-1.0.0-beta/${@}

EOF

My manager command is: /path/cistem-1.0.0-beta/$command

command: /path/cistem-1.0.0-beta/submit.sh $command

However, I'm getting the following error:

....submit_cistemjob.b ctffind 172.26.32.8,127.0.0.1 3001
7885057717552310&' 10 times.
Error: A slave has disconnected before all jobs are finished.
All 10 processes are connected.
All slaves have re-connected to the master.
Error: A slave has disconnected before all jobs are finished.

Please advise.

Best,

Abhisek

Wed, 08/19/2020 - 14:47 <u>(Reply to #15)</u>

timgrant

Hi Abhisek,

This implies that the jobs are running, but then they are crashing. Does the cluster have the same path to the images as the GUI does?

Tim

Thu, 08/20/2020 - 09:28 <u>(Reply to #16)</u>

abhisek

Hi Tim,

The micrographs are being kept in /scratch/micrographs

And I'm running cistem and the script.sh from my $Home space.

When I imported the micrographs to cistem it should have created some link to work with right?

Abhisek

Thu, 08/20/2020 - 13:55 <u>(Reply to #17)</u>

timgrant

Hi Abhisek,

The files need to be accessible on the same path everywhere the job is being run. If they are not accessible in /scratch/micrographs on all of the machines, the job will crash.

Tim

abhisek

cisTEM disconnects after each iteration

Hi rnavaza,

Your code works like a charm in cluster.

However, one problem remains. After each reconstruction cycle the cisTEM is getting disconnected from the cluster and when it moves on to next iteration it re-sbmits all the jobs again on cluster. This thing goes on every iteration. When using a public cluster, a disconnection and re-connection lag can cause loosing a long queue.

Is there any way to permanently use (till the whole set of reconstruction finishes), say, 100 threads on a cluster and cisTEM will use these 100 for each cycles without disconnecting after each iteration.

Please suggest.

Thank you

Abhisek

Wed, 01/29/2020 - 08:05 <u>(Reply to #19)</u>

rnavaza

Hi abhisek,

What workload manager are you using ? Torque, SLURM ?

The problem is that cisTEM treats each cycle as a different job so a wrapper script isn't able to differentiate a new job from a new cycle. That will require changing the cisTEM source code.

Rafael.

Fri, 02/07/2020 - 17:29 <u>(Reply to #20)</u>

abhisek

Hi Rafael,

Sorry for the delayed response.

I'm using Sun Grid Engine to run cisTEM and using following script:

```
#!/bin/bash
cat - <<EOF | qsub
#!/bin/bash
#PBS -N cisTEM.${1}
#PBS -l nodes=1:ppn=1
/your_path_to_cisTEM_binaries/${@}EOF
```

The jobs (2D clssification) running fine, however, "Merging Class Average" step with 2 processes (default) takes an aweful amount of time to complete and sometimes hangs completely when using 100% of the particles (without any error message).

Please advise if I need to change something to make it run smoother.

Thank you

Abhisek

Sun, 02/09/2020 - 14:54 <u>(Reply to #21)</u>

rnavaza

If I remember correctly the

If I remember correctly the 'Merging Class Average' is done between the cycles of '2D Classification' jobs; if so then its the GUI procedd that's doing the calculations, no jobs are submitted to the cluster... And yeah I noticed that it takes a lot of time, but therés no way around it...

Source URL:https://cistem.org/unable-get-cistem-job-parallelize-multiple-nodes-cluster?page=0