Running Matching Pursuit Algorithm (MPA) Jobs on Tyrone Cluster (SERC)

Vinay Shirhatti

Dr. Supratim Ray's Lab,

PRL, CNS,

Indian Institute of Science

February 2014

Introduction

This document describes, in short, the procedure to run jobs on the SERC Tyrone cluster to implement Matching Pursuit algorithm (MPA) on any relevant signal. The steps described here have been tested for running a job from a local Linux machine (Ubuntu 12.04 Precise Pangolin). Throughout this document the 'local machine' refers to your lab computer and the 'remote machine/server/cluster' refers to the cluster in SERC that you wish to run your jobs on. One may connect to the cluster from a terminal (i.e. command line) on a local machine and execute commands on it to run jobs. This simplifies and makes faster the procedure to organize data, codes and scripts and to run MPA and get back results for analyses.

Prerequisites

These are some prerequisites to run MPA on the cluster:

- Get an SERC Cluster Computation account. This would require atleast a couple of visits to the SERC office. When you get an account, you may request your account to be subscribed to the cluster(s) of your choice. In my case, I have an account on the Tyrone cluster and the instructions here follow accordingly.

- Install ssh client and server on your local machine. This would be required to communicate with the remote server i.e. with Tyrone in this case. The client is required to connect and copy files to the remote cluster from your local machine and the ssh server is required to copy files back to your machine from there.

  Install ssh client and server:
  ```
  >>sudo apt-get install openssh-client openssh-server
  ```

- Get MPA running on your local machine.
  Software available at: (not there anymore? Take it from lab drive instead?) (Distributed by: Epilepsy Research Laboratory at Johns Hopkins Medical Institutions (Dr. Supratim Ray))(Adapted from works by Dr. Stephane Mallat, Dr. Zhifeng Zhang, Dr. Piotr Franaszczuk, Dr. Christophe Jouny).

  Unzip the folder 'MP' to a suitable folder and open the command line terminal

  Change the directory to: path/MP/gabord-0.1
  ```
  >>cd path/MP/gabord-0.1/
  ```

  Execute the following:
  ```
  >>./configure
  >>make
  ```

  Copy the 'gabord' thus generated in the folder 'MP/gabord-0.1/src/' and replace the one in 'MP/source/' with this.
  The mpp codes in matlab should then be able to use these compiled codes and do MPA related computations.

  For the cluster, it wasn't required to specifically compile the MPA

code on the remote location each time. It was done just the first time (using ./configure and make as explained above), after which the code implemented successfully even if the entire code and the executable (gabord) were just simply copied from the local machine to it.

You are now ready to go!

Now let's do it!

Setting Up

Login to the Tyrone cluster.

`>>ssh username@cluster-id`

In this case: `>>ssh username@10.16.2.236`

You will be prompted to enter the password for this *username* to connect. One may surpass this password prompt (for subsequent logins) by entering these lines in a terminal on the local machine:

Bypass the password prompt:

`>>ssh-keygen -t rsa`

`>>ssh-copy-id -i ~/.ssh/id_rsa.pub username@10.16.2.236`

Open the .cshrc file.

`>>vi .cshrc`

Add the following lines to it (Press INSERT to enter the editing mode)

```
#Tyrone-cluster
set path=($path /opt/toque-305/bin)
set path=($path /opt/toque-305/sbin))
set path=(/opt/mvapich2-1.8-r5423/intel/bin $path)
```

Then close the .cshrc file (By pressing ESC and then typing *:wq*).

Now we shall copy the *MP/* folder to our account on the cluster. Open a terminal on the local machine and navigate to the folder (using *cd* commands) where the *MP/* folder is located. Then type the following to the terminal:

`>>scp -r MP/ username@10.16.2.236:/localscratch/username/`

The */localscratch/username/* location is the local storage location provided to *username*. Run *./configure* and *make* and the rest of the related steps as described before and make sure that the MPA code runs properly on some data. For example, let's take a case where there is some data within the */localscratch/username/MP/data/* folder. The algorithm runs by executing 'gabord' within */localscratch/username/MP/source/* on */localscratch/username/MP/data/test/ImportData_SIG/GaborMP/local.ctl* file. This *local.ctl* contains information that is read by the MPA code during execution. Make sure that the paths defined in this *local.ctl* file are correct, in accordance with the paths within your account on the cluster.

Next, run MPA with the following command:

```
>>/localscratch/username/MP/source/gabord (space here)
/localscratch/username/MP/data/test/ImportData_SIG/GaborMP/local.ctl
```

This should create the MPA decomposition files (which look something like this -*mp0.bok.000, mp0.bok.001* etc.) inside the */localscratch/apmsvd/MP/data/test/GaborMP/* folder. If this happens, then your MPA code is running all fine and giving you the required output!

Preparing Data and Job Script

Now, one would like to implement the same steps, albeit on a number of data signals. To do that one may write a script which has all the command lines to implement MPA on all the data signals and submit this script as a job script to the Tyrone cluster. As this job is run, all the commands are implemented and we obtain the desired output data.

To prepare for this, the relevant data (which has all the input signals that have to be decomposed using MPA) has to be readied on your local machine and then it should be copied to the remote location. You may copy this data to your folder on */localscratch/username*.

Run prepareDataForOrchestra (modified file for tyrone) and generate the mpAnalysis files locally. Open a terminal (shortcut - Alt+Ctrl+T) and copy these mpAnalysis files to your cluster storage location. For example -

```
>>scp -r test/ apmsvd@10.16.2.236:/localscratch/apmsvd/MP/data/
```

You might have to manage this data, since it might contain files that aren't required for MP computation. Try to reduce the size of the chunk that you are copying to your cluster storage.

The mpAnalysis folder has the relevant data and a command script (com-

mandFile.sh) that can be submitted as the job script. Refer this link for information on job scripts.

Here's an example job script (don't include the lines in the parentheses, these are just explanations) :

```
#!/bin/sh
#PBS -N mpjob (mpjob is the name of the job, for identification in the job
queue on the cluster)
#PBS -l nodes=4:ppn=32:regular (This implies that you are requesting
128 CPUs/processors for running your job on the cluster)
#PBS -l walltime=24:00:00 (This implies that you are requesting 128 CPUs/processors
for running your job on the cluster. Refer the link mentioned above for other
options)
cd /localscratch/apmsvd/MP/source/ (Navigate to the folder where the
executable is located)
NPROCS=`wc -l < $PBS_NODEFILE`
HOSTS=`cat $PBS_NODEFILE | uniq | tr '\n' "," | sed 's|,$||'`
mpirun -np $NPROCS --host $HOSTS gabord path to local.ctl >path to
the verbose tyroneOutput file (This line is repeated for every signal
you want to work upon. One has to provide the corresponding local.ctl as
the input file and can read the outcome of the job run in output file, if it is
specified)
```

Running the Job

Open another terminal and login to your cluster account using ssh as described earlier. Navigate to your mpAnalysis folder and submit the job.

```
>> qsub commandFile.sh
```

You may use the following commands to check the job status and the cluster queue.

```
>> qstat (gist of all jobs)
```
OR
```
>> qstat -a (details of all jobs on the cluster)
```
OR
```
>> qstat -q (queue status on the various cores of the cluster)
```

Once the job has completed, you can copy the computed files back to your local system. Try copying only the newly created files and avoid redundant copying operations (for eg. don't copy back the header files, control files, original signal data files etc. Just copy the MP related files that have been freshly computed).
Something like this may work:

>>scp -r *path to the files/folders to be copied* you@10.120.10.xyz:*path to storage folder* *(Make sure you have ssh server installed on your machine, otherwise the above command won't work)*

* * *