Submitting Parallel Matlab Jobs to a Cluster

Ed Hall

Matlab Parallel Computing jobs can be submitted to the ITC Linux cluster by first logging onto the cluster front-end node from an X-term window on your local computer and starting up Matlab. PCT jobs have to be submitted from within Matlab and the example scripts show how to setup and submit the jobs.

Matlab Remote Display from Cluster

Download and install NX client for your computer
http://www.nomachine.com/select-package-client.php

Install Cisco VPN client on your computer
http://www.itc.virginia.edu/network/vpn/

Once you have the NX login window, set the login configuration as follows:

- host ← fir.itc.virginia.edu

- Desktop ← Unix, Gnome,WAN

If logging in from off-grounds, the Cisco VPN will have to be started first to authenticate your computer with the on-grounds network.

Once you are logged into the cluster, you start Matlab with the command 'matlab' in a terminal window and the desktop interface will be displayed on your local machine.

Passwordless ssh Connections

You will need to configure your cluster account so that the Matlab PCT can use passwordless ssh to log into the cluster compute nodes using the following procedure:

1. If you don't have a key pair (id_dsa, id_dsa.pub in .ssh), run

    ```
    ssh-keygen -t dsa
    ```

2. Give it no passphrase

3. If you don't already have an authorized_keys2 file, from your home directory run the commands

    ```
    cd  .ssh
    cp  id_dsa.pub authorized_keys2
    ```

You can test if this works by logging into a compute node directly from the cluster front-end node (generally discouraged) with the command

```
ssh lc4-compute-2-3
```

Example Matlab Files

The files in this folder are organized into three groups:

Function or script M-files that can be run where each group interactively from the Matlab desktop to illustrate various Parallel Computing Toolbox constructs.

`colsum.m` Example function M-file using message passing

`expensive_confun.m` An expensive constraint function used in `parallel_optim1.m`

`expensive_objfun.m` An expensive objective function used in `parallel_optim1.m`

`mpi_example1.m` Example using spmd construct and message passing

`myRand.m` Simply a wrapper around rand

`parallel_optim1.m` Parallelized optimization using `fmincon` and `ga` commands

`parfor_example1.m` Using `parfor` to test Girko's law

`pcalc.m` Simple example using the `parfor` contruct to compute maximum eigenvalues

`solver_large1.m` Using the `spmd` contruct to solve a linear system

`spmd_example1.m` Example using spmd to estimate Pi in parallel

`spmd_example2.m` Second example using spmd and distributed arrays

`import_configs.m` Script to programmatically import parallel configuration files

`destroy_finjobs.m` Script to destroy intermediate directories (Job*) left over from parallel Matlab job run on the cluster

Job Submission script M-files that submit a job through PBS Pro to the cluster using some function M-file from above.

`dist_submit2.m` For Matlab parallel jobs where each worker runs independently of the others. Calls the function `myRand.m` .

`matlabpool_submit1.m` For Matlab parallel jobs requiring a pool of Matlab workers, e.g. using the function `solver_large1.m` with the `spmd` construct for a data parallel application.

`matlabpool_submit2.m` For Matlab parallel jobs requiring a pool of Matlab workers, e.g. using the function `pclac.m` with the `parfor` construct for a task parallel application.

`parallel_submit2.m` For Matlab parallel jobs that explicitly incorporate commands for the workers to communicate with each other. Calls the functikon `colsum.m` .

`matlabpool_submit2b.m` Script to submit a Matlab parallel job and save the job id to a file so that you can log into the cluster after the job has completed and retrieve any results that were sent to the command window

`matlabpool_retrieve2b.m` Script to load a Matlab parallel job saved job id and retrieve any results that were sent to the command window. Otherwise, you should save results to a file and exit Matlab once you have submitted the job.

Configuration Files that must be loaded into the configuration manager (Parallel menu on Desktop) for the job. The configuration files allow you to incorporate PBS directives specific to that job.

pbsproconfig_standard_2011a.mat A template configuration file that can submit a job of up to 16 Matlab workers to the PBS Pro `standard` queue of ITC's linux clusters for jobs that don't have large memory requirements. It is used in the above job submission scripts.

pbsproconfig_nopreempt_2011a.mat A template configuration file that can submit a job of up to 16 Matlab workers to the PBS Pro `nopreempt` queue of ITC's linux clusters for jobs that require large memory (> 4 GB). It is used in the above job submission scripts.

.