

Guidelines for using TOMLAB on Linux Server

Yashar Pourrahimian and Hooman Askari-Nasab
Mining Optimization Laboratory (MOL)
University of Alberta, Edmonton, Canada

Abstract

This is a document on how to run and compile TOMLAB projects on the University cluster server. The servers are referred to as the AICT numerical and cluster servers.

1. Introduction

AICT operates and maintains a general purpose Linux cluster for research use by faculty, staff, and students at the University of Alberta. It offers a 64-bit environment with excellent performance for serial programs written in C/C++ and Fortran 77/90. MPI and OpenMP parallel programs are also supported. Users share the available resources by running jobs in batch mode.

The cluster is made up of one head node (login node) and 29 execution nodes, all running Linux. They are interconnected by standard Gigabit Ethernet (for NFS) as well as high speed Infiniband (for MPI). The head node contains two dual-core AMD Opteron 275 processors (four cores total) at 2.2GHz and 4GB of RAM installed. The execution nodes provide a total of 188 cores in the following configuration (AICT, 2011).

Access to the cluster and numerical servers are not automatic. Your supervisor must send an email to idadmin@ualberta.ca, Cc to research.support@ualberta.ca, requesting access on your behalf.

The remainder of this document outlines basic use of the cluster. It assumes you have prior experience working in a UNIX environment.

2. Required software

A basic Windows installation does not come with an ssh client program. Therefore, you should download and install [PuTTY](#) (terminal program) and [WinSCP](#) (file transfer program). You will also need to install an X display server program such as [Xming-mesa](#).

In addition to the above general software, you need TOMLAB software for Unix-like systems and MATLAB compiler. You can download these from [MOL](#) homepage. Copy these files in a folder titled "Reqsoftware".

You also need TOMLAB Linux license to run TOMLAB on the server.

3. Software installation on the cluster server

1. Run Xming
2. Login to the cluster server `cluster.srv.ualberta.ca` through WinSCP
3. Copy the folder (Reqsoftware) containing the `tomlab-lnx64-setup.bin` and `MCRInstaller.bin`, to the server. Drag the folder and put in WinSCP page. This folder is appeared in the following directory

→ /state/partition1/home/pourrahi

4. Run putty. Host Name: cluster.srv.ualberta.ca , In the putty configuration panel select tunnels under X11 from connection category then check “Enable X11 forwarding” and type “localhost:0” in Xdisplay location.
5. To see your folder in the cluster server use the command ‘ls’ as follows:

```
[pourrahi@cluster-login ~]$ ls
```

There is only one folder titled Reqsoftware.

6. Change directory to the folder:

```
[pourrahi@cluster-login ~]$ cd Reqsoftware
```

7. If you type ls, you will see there are two files, tomlab-lnx64-setup.bin and MCRInstaller.bin. We can only run executable files. You can convert files to the executable format as follows:

```
chmod +x file name
```

For example: `chmod +x tomlab-lnx64-setup.bin`

8. Type ls, it can be seen that the color of tomlab-lnx64-setup.bin file has been changed. This shows that this file is executable.
9. Execute the installer binary, tomlab-lnx64-setup.bin, as follows:

```
[pourrahi@cluster-login ~/Reqsoftware]$ ./tomlab-lnx64-setup.bin
```

If you are running the installer as a regular user, a warning may be displayed. Currently this warning can be ignored since a manual step. Press forward for the rest and at the final page; simply press the Finish button to exit the installer.

10. A new folder titled tomlab will have to appear in the following path:

→ /state/partition1/home/pourrahi

11. Copy the TOMLAB server license (tomlab.lic) from your computer to the tomlab folder through WinSCP.
12. Create a new folder titled MCR in the cluster server through WinSCP
13. Repeat Steps 7 to 8 for the installer binary, MCRInstaller.bin.
14. Execute the installer binary, MCRInstaller.bin, as follows:

```
[pourrahi@cluster-login ~/Reqsoftware]$ ./MCRInstaller.bin
```

Directory name: /state/partition1/home/pourrahi/MCR

15. A new folder titled InstallShield will have to appear in the following path:

→ /state/partition1/home/pourrahi

4. Compiling TOMLAB/CPLEX files on the numerical server

MATLAB compiler can be used to convert your MATLAB code to a stand-alone application. To compile, we need a copy of the TOMLAB/CPLEX required files. This is a list of files that you get when you run TOMLAB/CPLEX code, in order to learn how to access the files, study paper 301 in the first report of MOL (Askari-Nasab, 2009).

Table 1, shows an example of the files required by TOMLAB/CPLEX to compile the MILP on the server for a specific project. Table 1 should be used just as a guideline only.

Table 1. Files required by TOMLAB/CPLEX to compile the MILP on the Linux cluster

tomRun	PrintResult	optParamDef	checkAssign	LineParamDef
cplexTL	ProbCheck	DefPar	SOLGet	tomFiles
cplex	optParamSet	endSolveMini	defblbu	mkbound
cplexmex	mipAssign	checkType	cpx2retvec	iniSolveMini
cpxBuildConflict	tomlablic	ProbDef	cplexStatus	ResultDef
tomlabVersion	StateDef	Str2num	etime	getfield
blanks	xprint			

1. Copy all the required files and TOMLAB code in a folder. For example, the TOMLAB code is Run_BlockCave.m, we copy this file, and all required files in a folder titled Reqfunctions.
2. Login to the numerical server num.srv.ualberta.ca through WinSCP
3. Copy the folder, Reqfunctions, to the numerical server. Drag the folder and put in WinSCP page.
4. Run putty. Host Name: num.srv.ualberta.ca , In the putty configuration panel select X11 under SSH from connection category then check “Enable X11 forwarding” and type “localhost:0” in Xdisplay location.
5. Change directory to the folder:

```
o/pourrahi> cd Reqfunctions
```

6. To compile your TOMLAB code use the command 'mcc-matlab' as follows:

```
Pourrahi\ Reqfunctions> mcc-matlab -m -v Run_BlockCave.m
```

7. After compiling, new files are generated within the Reqfunctions folder. The name of one of them is similar to the TOMLAB code function, Run_BlockCave.m, but without extension. Copy this file on your computer desktop through WinSCP.
8. The TOMLAB code loads coefficient, upper bound, and lower bound matrices. You may use this code to load other matrices, too. Create a folder and copy all these matrices and the compiled file, which does not have extension in this folder, for example BC.

Table 2, shows the files which were copied in the BC folder for a block caving problem.

Table 2. List of files in BC folder

File	Description
A.mat	Coefficient matrix
b_L.mat	Lower bound matrix
b_U.mat	Upper bound matrix
c.mat	Matrix of objective function
Input.mat	A MATLAB file which is contain required input data
zarib.mat	A MATLAB file which is contain required data
Run_BlockCave	Compiled file without extension

9. Login to the cluster server through WinSCP
10. Copy the folder, BC, which described in previous step to the server. Drag the folder and put in WinSCP page.

5. Test your program

Before submitting the job on the cluster, run your code interactively and make sure it works.

1. Login to the cluster server through putty
2. Change directory to the folder (BC).
`[pourrahi@cluster-login ~]$ cd BC`
3. Convert the file without extension to the executable format as follows:
`chmod +x Run_BlockCave`
4. Execute the file without extension, as follows:
`[pourrahi@cluster-login ~/BC]$./Run_BlockCave &`

6. Batch jobs

Once you have successfully tested your program, you can submit the job on the cluster. A job is simply a shell script that invokes the program you want to execute.

Contention among jobs for the finite resources on a given node can lead to poor performance (or even a node crash). To avoid this, you should accurately specify your job's needs so that Portable Batch System (PBS) can reserve the necessary resources.

A convenient alternative to specifying resources on the command line is to introduce a PBS directive into your job script. A PBS directive is a shell script comment that begins with the string PBS. It must appear before the first executable statement.

To write a .pbs script file on the server use gedit command as follows:

```
[pourrahi@cluster-login ~]$ gedit
```

Write the following lines in the opened window and then save the script with a proper name, for example, Run in the created folder (BC). Table 3 shows description of each line (AICT, 2011).

```
#!/bin/bash -l
#PBS -N BC_NESW
#PBS -S /bin/bash
#PBS -l pvmem=2gb
#PBS -l nodes=1:ppn=8
#PBS -l walltime=168:00:00
#PBS -m bea
#PBS -M pourrahi@ualberta.ca

LD_LIBRARY_PATH=/home/pourrahi/tomlab;
LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:/home/pourrahi/tomlab/shared;
LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:/home/pourrahi/MCR/v711/runtime/glnxa64;

TOMLAB_LICENSE_FILE=/home/pourrahi/tomlab/tomlab.lic

cd /home/pourrahi/BC

./Run_BlockCave
```

Table 3. Description of commands in the script file

Line	Description
#PBS -N BC_NESW	Gives your job BC_NESW name
#PBS -S /bin/bash	This job is configured to use the recommended bash-shell login environment
#PBS -l pvmem=2gb	Specifies virtual memory of 2GB per-process
#PBS -l nodes=1:ppn=8	Is read as: one node multiplied by one processor per node equals eight CPUs required.
#PBS -l walltime=168:00:00	Specifies an elapsed time of 168 hours
#PBS -m bea	An email notification will be sent when the job (b)egins and (e)nds or is (a)borted
#PBS -M pourrahi@ualberta.ca	Your email address
LD_LIBRARY_PATH= /home/pourrahi/tomlab;	Define the library paths for TOMLAB, MATLAB Compiler runtime(MCR)
LD_LIBRARY_PATH= \${LD_LIBRARY_PATH}:/home/ pourrahi/tomlab/shared;	
LD_LIBRARY_PATH= \${LD_LIBRARY_PATH}:/home/ pourrahi/MCR/v711/runtime/glnxa64;	
TOMLAB_LICENSE_FILE= =/home/pourrahi/tomlab/tomlab.lic	Defines the directory of TOMLAB license
cd /home/pourrahi/BC	Changes the directory to the folder where your compiled file is
./Run_BlockCave	Runs your compiled file

Overestimating any of these parameters could potentially waste resources and cause jobs to be delayed, *including yours*, particularly when the cluster is busy. Therefore, specify the **minimum** requirements that will allow your program to run successfully.

If you omit a resource in the specification, a default value will be applied from the following list.

```
pvmem=512mb
nodes=1:ppn=1
walltime=24:00:00
```

The maximum resources available are

```
pvmem=126gb
nodes=32:ppn=4
nodes=3:ppn=16
walltime=168:00:00
```

7. Job submission

To run a job on the cluster, you must submit the created script for batch processing, thus

1. Change directory to the folder (BC).

```
[pourrahi@cluster-login ~]$ cd BC
```

2. Submit the job (the script file titled Run)

```
[pourrahi@cluster-login ~/BC]$ qsub Run
```

This will return a unique job id, and your job will take its place in a queue that includes all jobs from all the other users. It will wait there until the necessary resources become available.

Use the **qstat** command to view the status of jobs in the queue. Also try **qstatx**. Job status will be either running (R) or queued (Q). To remove a running or queued job that you previously submitted, use the **qdel** command with the numerical portion of the job id as the argument.

As an example, here is what to expect when you submit a job script called Run.

```
[pourrahi@cluster-login ~/BC]$ qsub Run
```

```
5480026.opteron-cluster.nic.ualberta.ca
```

```
[pourrahi@cluster-login ~/BC]$ qstatx -a
```

```
Mon Jan 31, 2011 11:51:02
```

```

--Requested----- --Used (mb)-----
Job Id  S Username Nodes   PVMem  Walltime  Mem    VMem    Walltime  Cputime
-----
.
.
.
5480020  R jes6      1:ppn=4   10gb  168:00:00  2459  2512   21:20:20  83:48:20
5480021  R jes6      1:ppn=4   10gb  168:00:00  3656  3715   2:36:54   9:47:59
5480022  Q jes6      1:ppn=4   10gb   48:00:00
5480023  Q jes6      1:ppn=4   10gb  168:00:00
5480024  Q jes6      1:ppn=1    4gb   48:00:00
5480025  Q jes6      1:ppn=1    4gb   48:00:00
5480026  Q pourrahi  1:ppn=1    2gb  168:00:00
-----
103 cpus used

```

8. References

- [1] AICT. (2011). AICT General Purpose Linux Cluster Retrieved 08/08/2011, 2011, from <http://www.ualberta.ca/CNS/RESEARCH/LinuxClusters/>
- [2] Askari-Nasab, H. (2009). *Guidelines for using TOMLAB on Linux clusters*. Paper presented at the MOL research report one, Edmonton.