Using Abaqus on Barossa
Step by Step User Guide

By
Wei-Liang Wu
ac3 Technical Support Officer
Office: Bld15.G28
Phone: Ex:5189
E-mail: ww02@uow.edu.au
Introduction

• *ac3* - The Australian Centre for Advance Computing and Communication
  – NSW State Government and 8 NSW-based universities (including UoW)
  – ATP at Redfern
Supercomputers at \textit{ac3}

- **Barossa**
  - Linux cluster
  - 147 dual nodes (294 CPUs)
  - 3GHz Pentinum 4

- **Clare**
  - SGI Origin 2400 system
  - 64 processors (400MHz)
  - 32GB of RAM

- **Hunter**
  - 2 NEC SX5 processors
  - Vector
My Responsibilities

• System level support
  – Make a request to use the *ac3* supercomputer
  – Arranging for access authorizations

• Application support
  – Parallelisation, and Optimisation of codes – porting existing codes to run on the supercomputer

• User level support
  – Advice and/or assistance in using various tools and techniques in high performance computing environment
Connect with barossa
Files transfer into barossa

- SSH Secure Shell Client 3.2.3
  (www.uow.edu.au/its/software/pcn.html)
Set PATH Environment 1

- Modify `.bash_profile`
```
# Set PATH Environment

# Get the aliases and functions
if [ -f ~/.bashrc ]; then
  . ~/.bashrc
fi

# User specific environment and startup programs

PATH=$PATH:$HOME/bin:/usr/local/abaqus/Commands:

export PATH
unset USERNAME
```

Prepare for Running abaqus

- Transfer input files into barossa (menangle.inp)
- Small job
  - abaqus job=menangle
  - Head node
    - OS
    - Manage the computers
  - Storage space is limited
Running a Big Job

- Computational nodes
- Scratch space
- Batch queue system (PBS)
- Steps:
  1. Create a python script
  2. Create a PBS file
  3. Submit the PBS job
Create a Python Script

```python
ifc -o <exe-name> <files> -lgoto_p4_512 /usr/local/lib/xerbla.o
or
g77 -o <exe-name> <files> -lgoto_p4_512 /usr/local/lib/xerbla-g77.o

A parallel version is also available, use -lgoto_p4_512p.

For further details see http://www.cs.utexas.edu/users/flame/goto/

The entire contents of this MOTD can be viewed at:

For assistance with this system, please email <help@ac3.com.au>, or
for urgent issues during working hours, telephone (02) 9209 4060.

[ww02@barossa ww02]$ ls
abaqus.pbs blast eltham menangle
[ww02@barossa ww02]$ cd menangle
[ww02@barossa menangle]$ ls
abaqus job=data cpus=2
Input file : menangle.inp
Job data.com not submitted.
[ww02@barossa menangle]$ ls
abaqus.pbs data data1 data.com data.err data.out menangle.inp
[ww02@barossa menangle]$ ls
```

Connected to barossa.ac3.com.au

SSH2 - aes128-cbc - hmac-md5 - none  80x24
Create a PBS file

```
#!/bin/sh
#PBS -o data.out
#PBS -e data.err
#PBS -l nodes=1:ppn=2
#PBS -l walltime=20:00:00
#PBS -l cput=20:00:00
#PBS -l mem=1024mb

mkdir -p /scratch/%PBS_JOBID/data
scp barossafs:menangle/* /scratch/%PBS_JOBID/data
cd /scratch/%PBS_JOBID/data
/usr/local/abaqus/Commands/abaqus python data.com
gzip /scratch/%PBS_JOBID/data/*
scp /scratch/%PBS_JOBID/data/* barossafs:menangle
```
Submit the PBS Job

- qsub, qstat, qdel, pbs showq

```
#PBS -e data.err
#PBS -l nodes=1:ppn=2
#PBS -l walltime=98:00:00
#PBS -l cput=98:00:00
#PBS -l mem=1024m

mkdir -p /scratch/%PBS_JOBID/data
scp barossafs:menangle/* /scratch/%PBS_JOBID/data
cd /scratch/%PBS_JOBID/data
/usr/local/abaqus/Commands/abaqus python data.com
gzip /scratch/%PBS_JOBID/data/*
scp /scratch/%PBS_JOBID/data/* barossafs:menangle

[ww02@barossa menangle]$ ls
abaqus.pbs  data data1 data.com  data.err  data.out  menangle.inp
[ww02@barossa menangle]$ qsub abaqus.pbs
36640.barossa.ac3.com.au
[ww02@barossa menangle]$]
```
Example

• Menangle-Freq.inp (from Ben Lake, a PhD student of Professor Michael West)
  – Analyse a railway bridge vibration
  – Extract the first 1000 natural frequency modes

• worner
  – 200 eigenvalues
  – 30 hours

• barossa
  – 1000 eigenvalues
  – 19756 sec (about 5 hours)
THE LANCZOS EIGENSOLVER IS USED FOR THIS ANALYSIS

NUMBER OF EIGENVALUES = 1000
HIGHEST FREQUENCY OF INTEREST = 0.10000E+19
MAXIMUM NUMBER OF STEPS WITHIN RUN = 35
BLOCK SIZE FOR LANCZOS PROCEDURE = 7
THE EIGENVECTORS ARE SCALED SO THAT THE LARGEST DISPLACEMENT ENTRY IN EACH VECTOR IS UNITY

THE ANALYSIS HAS BEEN COMPLETED

ANALYSIS SUMMARY:
TOTAL OF
1 INCREMENTS
0 CUTBACKS IN AUTOMATIC INCREMENTATION
0 ITERATIONS
857 PASSES THROUGH THE EQUATION SOLVER OF WHICH
51 INVOLVE MATRIX DECOMPOSITION, INCLUDING
0 DECOMPOSITION(S) OF THE MASS MATRIX
1 REORDERING OF EQUATIONS TO MINIMIZE WAVEFRONT
0 ADDITIONAL RESIDUAL EVALUATIONS FOR LINE SEARCHES
0 ADDITIONAL OPERATOR EVALUATIONS FOR LINE SEARCHES
23458 WARNING MESSAGES DURING USER INPUT PROCESSING
1 WARNING MESSAGES DURING ANALYSIS
0 ANALYSIS WARNINGS ARE NUMERICAL PROBLEM MESSAGES
0 ANALYSIS WARNINGS ARE NEGATIVE EIGENVALUE MESSAGES
0 ERROR MESSAGES

JOB TIME SUMMARY
USER TIME (SEC) = 3347.7
SYSTEM TIME (SEC) = 2366.0
TOTAL CPU TIME (SEC) = 5713.8
WALLCLOCK TIME (SEC) = 19756
Why use barossa?

• Efficiency
• Multi-jobs
  – Run individual jobs on the same time
• Anytime
  – No time restriction
Clare

• Security token ($150)
• Connect with clare
  $ telnet clare.ac3.com.au
• Upload/Download files
  $ ftp ftp.uow.edu.au
• Batch queue system (PBS)
Parallelisation Techniques

- There are many different parallelisation methods for FORTRAN or C/C++ code
- OpenMP compiler directives
  - Manual parallelisation
  - Automatic parallelisation
Running FORTRAN 1

• To run your code on one processor
  $ f77 test.f –O3
  $ nohup timex a.out &
  $ ps

• -O3 option optimises code

• timex reports the CPU used by your program
Running FORTRAN 2

• To run your code on multi-processors
  – Before running your job
    $\text{setenv OMP\_NUM\_THREAD } n$
    Where $n$ is the number of processors
    (default: 6 processors)
  – Compile using
    $\text{f77 test.f –O3 –apo}$
  – -apo flag is the automatic parallelisation option
Running FORTRAN 3

- Clare - $f77
- Barossa - $g77
- Batch queue system
  - $ qsub test.pbs

    #PBS -l ncpus=1
    #PBS -l walltime=96:00:00
    #PBS -l cput=96:00:00
    #PBS -l mem=256m

    ./a.out