OPENFOAM Best Practices for Intel® Cluster Ready

HPC ADVISORY COUNCIL
NETWORK OF EXPERTISE
BEST PRACTICES

1. Introduction

The following best practices document is provided as courtesy of the HPC Advisory Council.

2. Application Description:

OpenFOAM (Open Field Operation and Manipulation) CFD Toolbox is a free, open source CFD software package produced by a commercial company, OpenCFD Ltd. It has a large user base across most areas of engineering and science, from both commercial and academic organizations. OpenFOAM has an extensive range of features to solve anything from complex fluid flows involving chemical reactions, turbulence and heat transfer, to solid dynamics and electromagnetics. The presentation provides information on OpenFOAM performance, how to optimize and compile the code for highest performance and efficiency, and the effect HPC cluster components (HW and SW) have on OpenFOAM performance.

3. Version Information:

Download OpenFOAM and Third-Party source pack at:

http://www.openfoam.com/download/source.php

4. Prerequisites:

The instructions from this best practice have been tested with the following configuration:

4.1 Hardware Used:

The instructions from this best practice have been tested on the HPC Advisory Council, Dell PowerEdge M610 blade server based cluster.

- Dell PowerEdge M610 14-node cluster
- Intel Xeon X5670 CPUs
- Memory: 24GB per node
- Mellanox ConnectX-2 QDR InfiniBand Adapters
- Mellanox QDR InfiniBand Switch

4.2 Software Used:

- OS: Intel® Cluster Ready Platform, using CentOS 5 Update 4
- Application: OpenFOAM
- Compilers: Intel, GNU 4.4.0
- MPI: Intel MPI 4, Open MPI 1.4.2, MVAPICH2-1.5, Platform MPI 7.1
- Benchmark workload:

5. Pre-setup requirements

% . $HOME/OpenFOAM/OpenFOAM-1.7.1/etc/bashrc

Make sure that you use GNU compilers 4.4 version and not the version that comes with RHEL or CentOS by default.

6. Building OPENFOAM

Many of the steps below are outlined in the Download page: http://www.openfoam.com/download/source.php

Create a directory

% mkdir $HOME/OpenFOAM

% cd ~/OpenFOAM

% tar xvfz tar xvfz OpenFOAM-1.7.1.gtgz

% tar xvfz ThirdParty-1.7.1.gtgz

Package Dependencies

Depending how you install your OS, you may need this package:

libXt-devel

Otherwise you might run into this at compile time:

/usr/bin/ld: cannot find –liberty

Makefile changes for GNU compilers

Make sure you have GCC 4.4 installed. For RHEL, you can install gcc44-4.4.0-6.el5 and gcc44-c++-4.4.0-6.el5 using yum. Then make changes to these files.

% cd $HOME/OpenFOAM/OpenFOAM-1.7.1/wmake/rules/linux64Gcc

% vim c

```
cc          = gcc44 -m64
```

% vim c++

```
CC          = g++44 -m64
```

Makefile changes for Intel compilers

% vim $HOME/OpenFOAM/OpenFOAM-1.7.1/etc/settings.sh

: ${WM_COMPILER:=Icc}; export WM_COMPILER

Makefile changes for Platform MPI

% vim $HOME/OpenFOAM/OpenFOAM-1.7.1/etc/bashrc

: ${WM_MPLIB:=MPI-MVAPICH2}; export WM_MPLIB

% vim mplibHPMPI

Makefile changes for compilers and Open MPI

% vim $HOME/OpenFOAM/OpenFOAM-1.7.1/etc/bashrc

: ${WM_MPLIB:=MPI-MVAPICH2}; export WM_MPLIB

% vim mplibOPENMPI

Makefile changes for compilers and Intel MPI

% vim $HOME/OpenFOAM/OpenFOAM-1.7.1/etc/bashrc

: ${WM_MPLIB:=IMPI}; export WM_MPLIB

% vim $HOME/OpenFOAM/OpenFOAM-1.7.1/etc/settings.sh

IMPI)

export MPI_HOME=/application/intel/impi/intel64

export MPI_ARCH_PATH=$MPI_HOME

_foamAddPath $MPI_ARCH_PATH/bin

_foamAddLib $MPI_ARCH_PATH/lib

export FOAM_MPI_LIBBIN=$FOAM_LIBBIN/impi

;;

Makefile changes for MVAPICH2

Make sure that MVAPICH2 has the name MPI included otherwise OpenFOAM will not be able to detect it.

% vim $HOME/OpenFOAM/OpenFOAM-1.7.1/etc/bashrc

: ${WM_MPLIB:=MPI-MVAPICH2}; export WM_MPLIB

% vim $HOME/OpenFOAM/OpenFOAM-1.7.1/etc/settings.sh

MPI-MVAPICH2)

export MPI_HOME=/application/mvapich2-1.5-gnu

export MPI_ARCH_PATH=$MPI_HOME

_foamAddPath $MPI_ARCH_PATH/bin

_foamAddLib $MPI_ARCH_PATH/lib

export FOAM_MPI_LIBBIN=$FOAM_LIBBIN/mvapich2

;;

Build the sources

% Allwmake

Verifying the installation

$ cd /home/pak/OpenFOAM/OpenFOAM-1.7.1/bin

$ ./foamInstallationTest

7. Running OpenFOAM

Copy over the tutorial and run the first example

Create a project directory within the $HOME/OpenFOAM directory named <USER>-1.7.1 (e.g. chris-1.7.1 for user chris and OpenFOAM version 1.7.1) and create a directory named run within it, e.g. by typing:

Copy the tutorial examples directory in the OpenFOAM distribution to the run directory. If the OpenFOAM environment variables are set correctly, then the following command will be correct:

% mkdir -p $FOAM_RUN

% cp -r $FOAM_TUTORIALS $FOAM_RUN

Run the first example case of incompressible laminar flow in a cavity:

% cd $FOAM_RUN/tutorials/incompressible/icoFoam/cavity

% blockMesh

% icoFoam

% paraFoam

8. Running OpenFOAM in Parallel

8.1 Create a new case called "damBreakFine"

% cd $FOAM_RUN/tutorials/multiphase/interFoam/laminar

% mkdir damBreakFine

% cp -r damBreak/0 damBreakFine

% cp -r damBreak/system damBreakFine

% cp -r damBreak/constant damBreakFine

8.2 Increase the mesh density and mesh grading in file:

constant/polyMesh/blockMeshDict /blockMeshDict

blocks

(

hex (0 1 2 3 4 5 6 7) (1000 1000 1) simpleGrading (1 1 1)

);

8.3 Change the deltaT for the steps

In system/controlDict, adjust deltaT to 0.0005 to give

1000 steps.

Also change the writeInterval to control the rate data get written.

```
endTime         0.5;
deltaT          0.0005;
writeInterval   20000;
```

8.4 Specify the number of Subdomains for a parallel run

Add "system/decomposeParDict" file for case of interest, then modify the "numberOfSubdomains" and the method of Coeffs to the to the number of processes you plan to run:

```
numberOfSubdomains=168
simpleCoeffs
{
   n            ( 4 6 7 );
   delta        0.001;
}
% decomposePar -force
```

8.5 use Open MPI to launch the parallel job

```
% mpirun -np 168 –hostfile ~/hostfile.14 interFoam
-parallel
```

HPC ADVISORY COUNCIL
NETWORK OF EXPERTISE

350 Oakmead Pkwy, Sunnyvale, CA 94085
Tel: 408-970-3400 • Fax: 408-970-3403
www.hpcadvisorycouncil.com