How to get started with OpenFOAM at SHARCNET

Isaac Ye, High Performance Technical Consultant
SHARCNET, York University
isaac@sharcnet.ca
Outlines

- Introduction to OpenFOAM
- Compilation in SHARCNET
- Pre/Post-Processing with Paraview
- Running jobs
- Compiling user-defined local solver
Introduction to OpenFOAM

- WHAT CAN DO?
- PROGRAM STRUCTURE
- THINGS TO KNOW TO RUN IN SHARCNET CLUSTERS
What can do?

OpenFOAM is a C++ toolbox for the development of customized numerical solvers, and pre-/post-processing utilities for the solution of continuum mechanics problems, including computational fluid dynamics (CFD).
OpenFOAM Structures

Open Source Field Operation and Manipulation (OpenFOAM) C++ Library

Pre-processing
Utilities
Meshing Tools

Solving
User Applications
Standard Applications

Post-processing
ParaView
Others e.g. EnSight

- Vinyl
- basic
- compressible
- DNS
- financial
- incompressible
- multiphase
- combustion
- discreteMethods
- electromagnetics
- heatTransfer
- lagrangian
- stressAnalysis

[isaac@orc-login2:~solvers] ls

http://www.openfoam.org/features/standard-solvers.php
Login node is not for running a job
‘ParaFoam’ is not compilable

Should use ‘sqsub’ – scheduler
Paraview must be used
Installation in SHARCNET

- DOWNLOAD
- SETUP THE ENVIRONMENT
- COMPILATION
Download the current release

1. Create the main project directory
   [1] mkdir /work/$USER/OpenFOAM

2. Change directory and download using ‘wget’

3. Extract the downloaded files

4. Download ‘boost’
   (you can download it on your PC and upload it to your account)
   [8] cp /home/isaac/boost_1_55_0.tar.gz .
   [9] tar zxvf boost_1_55_0.tar.gz
Setup the environment (bashrc)

1. bashrc

[1] cd /work/$USER/OpenFOAM/OpenFOAM-3.0.1/etc
[2] vi bashrc

    module unload intel
    module unload openmpi
    module load gcc/4.9.2
    module load openmpi/gcc-4.9.2/std/1.8.7

    export NEWHOME=/work/isaac

and change all ‘HOME’ into ‘NEWHOME’ in bashrc

Note: Please do not set ‘HOME=/home/$USER’ which will slow down all performance.
Setup the environment (boost)

1. CGAL.sh
[1] cd /work/$USER/OpenFOAM/OpenFOAM-3.0.1/etc/config
[2] sed -i -e 's=boost-system=boost_1_55_0=' CGAL.sh

2. makeCGAL
[1] cd /work/$USER/OpenFOAM/ThirdParty-3.0.1
[2] sed -i -e 's=boost-system=boost_1_55_0=' makeCGAL
Job running environment

1. modules (OpenFOAM is optimized with Gcc)

module load gcc/4.9.2
module load openmpi/gcc-4.9.2/std/1.8.7

2. of_301 script (setting up the right environment for OpenFOAM 3.0.1)

It is better to make an alias to execute the OpenFOAM environment

[1] vi ~/.bashrc
[2] " alias of301='source /work/$USER/OpenFOAM/OpenFOAM-3.0.1/etc/bashrc' "
Setup the environment

[isaac@orc-login2:~] module list
Currently Loaded Modulefiles:
  1) torque/2.5.13
  2) moab/7.1.1
  7) ldwrapper/1.1
  3) intel/12.1.3
  4) openmpi/intel/1.6.2
  5) mkl/10.3.9
  6) sq-tm/2.5
  8) user-environment/2.0.1

[isaac@orc-login2:~] of301

[isaac@orc-login2:~] module list
Currently Loaded Modulefiles:
  1) torque/2.5.13
  2) moab/7.1.1
  7) gcc/4.9.2
  1.1
  3) mkl/10.3.9
  4) sq-tm/2.5
  5) ldwrapper/
  7) gcc/4.9.2
  6) user-
  8) openmpi/gcc-4.9.2/std/1.8.7
Setup the environment (Checking!)

Check if right modules and environment are set *whenever you log in.*

```bash
[isaac@orc-login1:..] module list
Currently Loaded Modulefiles:
1) torque/2.5.13                             4) sq-tm/2.5
2) moab/7.1.1                                5) ldwrapper/1.1
3) mkl/10.3.9                                6) user-environment/2.0.1
                                            7) gcc/4.9.2
                                            8) openmpi/gcc-4.9.2/std/1.8.7

[isaac@orc-login1:..] echo $WM_PROJECT_DIR
/work/isaac/OpenFOAM/OpenFOAM-3.0.1

[isaac@orc-login1:..] echo $FOAM_INST_DIR
/work/isaac/OpenFOAM
```
Submitting a compilation job

[1] cd /work/$USER/OpenFOAM/OpenFOAM-3.0.1

Check ‘log’ and ‘error’ to check the status and errors
Pre/Post Processing

- MESH GENERATION
- PARAVIEW
Tutorial test

1. Lid-driven cavity case
   (http://cfd.direct/openfoam/user-guide/cavity/#x5-40002.1)

[1] mkdir -p $FOAM_RUN
[2] cp -r $FOAM_TUTORIALS $FOAM_RUN

    [isaac@orc-login1:..] ls -lrt
    total 12
    drwxrwxr-x 2 isaac isaac 4096 Mar 29 00:31 0
    drwxrwxr-x 2 isaac isaac 4096 Mar 29 00:31 system
    drwxrwxr-x 2 isaac isaac 4096 Mar 29 00:31 constant

[4] blockMesh
   - Generating mesh
   - Needs to run in the case directory

\[ U_x = 1 \text{ m/s} \]

\[ d = 0.1 \text{ m} \]

Figure 2.1: Geometry of the lid driven cavity.
Basic case structure

- **Text based**: Contains initial values and boundary conditions for all variables
- **Mesh**: Contains things that are constant
  - Mesh
  - Material properties
  - Environmental constants
- **System**: Specifies what methods and schemes to use and controls for the simulation
  - Timestep
  - Numerical schemes
  - SIMPLE/PISO controls
  - Simulation start/end time
  - Additional runtime calculations
‘blockMesh’ result

Basic statistics
- Number of internal faces: 0
- Number of boundary faces: 6
- Number of defined boundary faces: 6
- Number of undefined boundary faces: 0

Checking patch -> block consistency

Creating points with scale 0.1
- Block 0 cell size:
  - i: 0.005 .. 0.005
  - j: 0.005 .. 0.005
  - k: 0.01 .. 0.01

Mesh Information
- boundingBox: (0 0 0) (0.1 0.1 0.01)
- nPoints: 882
- nCells: 400
- nFaces: 1640
- nInternalFaces: 760

Patches
- patch 0 (start: 760 size: 20) name: movingWall
- patch 1 (start: 780 size: 60) name: fixedWalls
- patch 2 (start: 840 size: 800) name: frontAndBack

End
Mesh generation

BlockMesh
Generate simple structured meshes based on blocks [http://www.openfoam.org/docs/user/blockMesh.php](http://www.openfoam.org/docs/user/blockMesh.php)

→ SnappyHexMesh
Create unstructured meshes based on complex surface geometries (e.g. stl) [http://www.openfoam.org/docs/user/snappyHexMesh.php](http://www.openfoam.org/docs/user/snappyHexMesh.php)

→ Other meshers
conversion to OpenFOAM format from most popular formats possible
Prepare a ‘case’ for Paraview

[1] cd $FOAM_RUN/tutorials/incompressible/icoFoam/cavity

[2] touch cavity.openFOAM

[3] Open the file cavity.openFOAM in Paraview
   (select file type "all files" and then type “openFOAM”)


Note: ParaFoam is not supported on SHARCNET login node.
Connecting to Visualization machine

Click the icon
Connecting to the Visualization machine

Connecting to the Visualization machine

Click for terminal window
Starting Paraview
Starting ParaView
Mesh in Paraview
Running OpenFOAM

- DAMBREAKFINE CASE
- SUBMITTING A JOB IN SERIAL AND PARALLEL
Running a serial job

[1] of301

(call the right environment using the alias or command line directly)


Lid-driven cavity case,

http://cfd.direct/openfoam/user-guide/cavity/#x5-40002.1)

[3] sqsub -r 10m -o log -e error icoFoam

[isaac@orc-login1:..cavity] ls -lrt

```
total 162
drwxrwxr-x 2 isaac isaac  4096 Mar 29 00:31 0
drwxrwxr-x 2 isaac isaac  4096 Mar 29 00:31 system
drwxrwxr-x 3 isaac isaac  4096 Mar 29 00:35 constant
-rw-rw-r-- 1 isaac isaac     0 Mar 29 01:06 cavity.openFOAM
-rw-rw-r-- 1 isaac isaac     0 Mar 29 01:34 error
-rw-rw-r-- 1 isaac isaac     0 Mar 29 01:34 0.3
-rw-rw-r-- 1 isaac isaac     0 Mar 29 01:34 0.2
-rw-rw-r-- 1 isaac isaac     0 Mar 29 01:34 0.1
-rw-rw-r-- 1 isaac isaac     0 Mar 29 01:34 0.5
-rw-rw-r-- 1 isaac isaac     0 Mar 29 01:34 0.4
-rw-rw-r-- 1 isaac isaac 74232 Mar 29 01:35 log
```
Running a parallel job

1. Dambreak case (damBreakFine)

(http://cfd.direct/openfoam/user-guide/damBreak/#x7-610002.3.11)

2. Input conditions and boundaries must be set properly

(see the OpenFoam user guide: blockMeshDist, setFields, decomposePar)

[1] of301


[3] sqsub -r 1h -q mpi -n 4 -o log -e error interFoam -parallel

[isaac@orc-login1:..damBreakFine] sqjobs

jobid  queue state ncpus                nodes  time command
------- ------ ----- ----- -------------------- ----- -------
.. 7785272    mpi     R     4 orc[307–308,310–311]  350s interFoam -parallel
OpenFOAM at SHARCNET

Isaac Ye

SHARCNET General Interest Seminar Series

[isaac@orc-login2:..damBreakFine] ls -lrtl
total 674
-rw-rw-r--  1 isaac isaac      0 Mar 29 11:00 error
-rw-rw-r--  1 isaac isaac 629325 Mar 29 11:03 log

[isaac@orc-login2:..damBreakFine] more log

/*---------------------------------------------------------------------------*
| =========                 | OpenFOAM: The Open Source CFD Toolbox                     |
| \      /  F ield         | Version: 3.0.1                                           |
|  \    /   O peration     | Web: www.OpenFOAM.org                                    |
|   \  /    A nd           |                                                     |
|    \/     M anipulation  |                                                     |
*---------------------------------------------------------------------------*/

Build   : 3.0.1-6b88e07ba67e
Exec    : interFoam -parallel
Date    : Mar 29 2016
Time    : 11:02:34
Host     : "orc307"
PID      : 30332
Case     : /gwork/isaac/OpenFOAM/isaac-3.0.1/run/tutorials/multiphase/interFoam/laminar/damBreakFine
nProcs   : 4
Compiling local solver

– USER-DEFINED SOLVER
Example: myFoam

There is a need to compile your own version of solver attaching to the OpenFOAM. Here is an example study about how to.
(Note: the OpenFOAM environment must be loaded before this procedure.)

1] Make a local solver folder

```
mkdir $WM_PROJECT_DIR/solvers
```

2] Copy the existing solver

```
cd $WM_PROJECT_DIR/solvers
cp -r $FOAM_SOLVERS/incompressible/pisoFoam myFoam
```

It is a better practice to copy the existing solver structure and modify it.
Example: myFoam

3] Modify the name and etc accordingly.

```bash
cd $WM_PROJECT_DIR/solvers/myFoam
mv pisoFoam.C myFoam.C
```

4] Edit some description and application name in myFoam.C

```bash
vi myFoam.C
Application
  myFoam
Description
  my solver
```
Example: myFoam

5] Modify the makefile list of files

vi Make/files

myFoam.C

EXE=$(FOAM_USER_APPBIN)/myFoam

6] Compilation

wclean
wmake

[isaac@orc-login1:..myFoam] which myFoam

/scratch/isaac/OpenFOAM/OpenFOAM-3.0.1/platforms/linux64GccDPInt32Opt/bin/myFoam
Thank you!