

# PROCEDURE FOR RUNNING THE C++ CODE ON ARC

## 1 Latest version(s)

- PIGFEM4.6: only records the stress-strain response.
- PIGFEM4.6.1: records the stress-strain response and the full sensitivity analysis.

## 2 Installation and compilation

Upload the file on ARC system and unzip it through the following command:

```
unzip PIGFEM4.6.1.zip
```

Then, orderly load the modules needed for installing the required libraries:

```
module load gcc mvapich2 gsl hdf5 openblas
```

```
module load metis atlas python
```

```
module load boost netcdf cmake parmetis fftw valgrind
```

Now go to the directory of the code and run the setup:

```
chmod 755 setup
```

```
./setup
```

This will take several minutes. It installs Petsc, Metis, and Gsl libraries and then compiles the code. After the compilation is completed successfully, you will see an executable file named “Main” is created in the code’s directory.

The main function is inside Tests/Main.cpp. Anytime a change is made inside the main function or other subroutines, the code needs to be compiled again by the following command:

```
make
```

Anytime a change is to be made, make sure the gcc library is loaded before compiling the code.

### Notes:

To activate the penetration algorithm, go to ”src/Material.OPCohesive.cpp”, line 125. Comment out ”penetrationn=0;”.

The address to the input file must be defined in the last line of Job.sh file, in front of “-in”. Line 81 of Tests/Main.cpp is for turning the damage on/off by changing the flag to true/false.

### 3 Job Submission

This section is for job submission on DragonsTooth cluster. Prior to the job submission, you must go to the code directory containing Main, Job.sh, Input,etc. The file job.sh is used to submit a job to the queuing system on ARC. If you go to the file, you will see the information needed to be entered for a job submission. In front of -p we define the type of allocation we are using, which is normal\_q. --ntasks=1 means that we are requesting only one processor. scancel sIn front of -A we define the allocation’s name. In front of -t we specify the walltime after which the job will be terminated. The amount of memory we are requesting is specified via --mem-per-cpu=50G. After those information was entered/modified, the job can be submitted to the queuing system through the following command:

```
sbatch ./Job.sh
```

This will return a message with the job ID (which is a number).

Here are some usefull commands for jobs’ management:

- Check a job status: **squeue -v JobID**
- Check the status of all of the jobs you have submitted: **squeue --users=Username**
- Cancel a job: **scancel JobID**
- Get detailed information about a running job: **scontrol show job JobID**

### 4 Update a version without installation

Replace the new “src”, “include”, and “Test” folders with the old ones, and remove everything in “obj” folder in the old version. Then, load the gcc compiler and compile the code using the following commands:

```
module load gcc/6.1.0
```

```
make
```

Add output folder:

```
./Main -in "Input/fibers.csv" -OUT_FOLDER "output_folder_address" -sensitivity true
```