BUILD-RUN-DOCUMENT

Saurabh Sharma

30 November 2018

This document shows the steps needed to clone, build and run the FWI code and to install the prerequisite packages.

1 Pre-requisites

The prerequisite development tools needed can be installed using the following commands.

1. `sudo apt-get install git`
2. `sudo apt-get install qt5-default`
3. `sudo apt-get install libeigen3-dev`
4. `sudo apt-get install python2.7-dev`
5. `sudo apt-get install python2.7`
6. `sudo apt-get install python-tk`
7. `sudo apt-get install python-numpy`
8. `sudo apt-get install python-matplotlib`

2 Cloning the Repository

To clone the FWI repository using git,
`git clone -o redmine https://git.alten.nl/parallelized-fwi.git`
This will create a copy of the repository, in a folder named **parallelized-fwi**
Any branch as needed can then be checked out from inside the `parallelized-fwi` folder, e.g. the develop branch
`git checkout develop`

3 Build/Run

To build the project, first create a folder titled **build** outside the **parallelized-fwi** folder.
`NOTE: This folder should be exactly 1 level outside the parallelized-fwi folder.`
`mkdir Build`
`cd Build`
`cmake -DCMAKE_BUILD_TYPE=Release ../parallelized-fwi/`
`make -j4` (the flag -j is used to build in parallel)
Now, the individual scripts for the preProcessing and the processing part can be run as shown below:
`cd applications`
`cd preProcessing`
`./FWIPreProcess`
`cd ../processing`
`./FWIProcess`
The input parameters for the code are provided in the input card i.e. default.in. User can create his/her own input card with a new name e.g. `newCard.in`. To use this input card use the card name as an argument when running the executables, `./FWIPreProcess newCard` and `./FWIProcess newCard`.

For post-processing (i.e. generation of image using the estimated chi values), the python script `imageCreator_CMake.py` can be used. This script is located inside the **parallelized-fwi** folder and can used as,
`python imageCreator_CMake.py`
The pre-processing, processing and the image creation step can all be grouped together using the python wrapper `wrap_FWI_CMake.py` located inside the **parallelized-fwi** folder.
`python wrap_FWI_CMake.py`