`dgcgwave` : docker and anaconda source code access

Simon Shaw, May 2019

- https://www.brunel.ac.uk/people/simon-shaw
- http://people.brunel.ac.uk/~icsrsss/

Downloads

The files for this project are posted on **Brunel University London's** *figshare* under the collection entitled `dgcgwave` : **Discontinuous Galerkin Finite Element Approximation of Linear Viscodynamic Solids** at this link https://brunel.figshare.com/authors/Simon_Shaw/1177599. The collection's DOI is 10.17633/rd.brunel.c.4503650 (here https://doi.org/10.17633/rd.brunel.c.4503650).

Introduction

This note explains how to access and run the codes used to generate the numerical results in the paper:

An *a priori* error estimate for a temporally discontinuous Galerkin space–time finite element method for linear elasto- and visco-dynamics.

Simon Shaw. Comput. Methods Appl. Mech. Engrg. 351 (2019) 1–19

See https://www.sciencedirect.com/science/article/pii/S0045782519301549 or https://doi.org/10.1016/j.cma.2019.03.025.

There is a version based on using FEniCS (https://fenicsproject.org) inside a docker container (https://hub.docker.com/u/variationalform) and a version based on using FEniCS in Anaconda 3 (https://www.anaconda.com, Python 3.7 version). For the latter two folders (or directories) named 'anaconda' are available for download in both `zip` and `tgz` formats. Once downloaded they can be extracted in the usual way: `unzip anaconda.zip` and `tar xvf anaconda.tgz` . It is recommended that you first create a folder or directory called `dgcgwave` and work inside of that.

There are also two PDF's other than this one. `main.pdf` is the preprint version of the paper cited above, and `report.pdf` is a longer version containing more details and more results.

The next two sections explain how to reproduce the results.

The FEniCS docker image

The code in the FEniCS pull was used to get the results in the paper and the report. We used the 2017.1.0 FEniCS docker image started with

```
docker run -ti ... quay.io/fenicsproject/stable:2017.1.0
```

(here `...` indicates that superfluous details are omitted) on 20 December 2018. A custom image built for this paper can be pulled in docker with

```
docker pull variationalform/fem:dgcgwave
```

(see https://hub.docker.com/r/variationalform/fem) and then run with

```
docker run -ti variationalform/fem:dgcgwave
```

The command `cd fenics` followed by `./bigrun.sh -J 3 | tee runmeout.txt` will, for the suite of twelve test cases in the report, run `solver.py` to produce the error results up to $N_{xy} = \mathrm{int}(2^{3/2})$.

in the `results` directory. Examples 5, 10, 11 and 12 (resp.) of those in the report correspond to examples I, II, III and IV (resp.) in the paper. Use `-J 7` to go up to $N_{xy} = \mathrm{int}(2^{7/2})$ and so on (but it will take longer). The file `bigrun_np.sh` can be used with `mpi`.

The Anaconda3 environment

The code in the anaconda directory has slight modifications made to the files `bigrun.sh` and `solver.py` referred to above, so that the results could be reproduced in the FEniCS anaconda environment.

This Anaconda python 3.7 version was installed and tested on a Mac during 30 April — 3 May 2019. The FEniCS environment was created and then cloned to an environment called `dgcgwave` with:

```
conda create -n fenicsproject -c conda-forge fenics
conda update conda
conda create --clone fenicsproject --name dgcgwave
```

To use this `dgcgwave` environment download and extract the anaconda directory and change into it. Then re-create the environment from the `yaml` file with:

```
conda env create -f dgcgwave.yml
```

If that is successful you can activate the environment and run the code. This,

```
conda activate dgcgwave
./bigrun.sh -J 5
```

will generate results for `Nxy = 2,3,4,6` for examples 1 to 12 in the report. These will be in the directories `./results/m` for `m=1,2,...,12`. To get results for `Nxy > 6` you need to run `./bigrun -J M` for `M > 5`.