Spring 2023:
Computational and Variational Methods for Inverse Problems

Getting started with FEniCS

1. FEniCS overview

FEniCS is a powerful, open-source suite of tools for automated solution of partial differential equations (PDEs) using finite element (FE) method. Part of the power for FEniCS is the ease with which one can create FE solvers by describing PDEs using weak forms in nearly-mathematical notation. The FEniCS project can be found at: http://fenicsproject.org/ and comes with an extensive documentation and examples.

FEniCS includes a number of powerful features that include:

- Automated solution of variational problems;
- Automated error control and adaptivity;
- An extensive library of finite elements;
- High performance linear algebra through backends to such libraries as PETSc;
- Visualization via a simple interactive plotting function, as well as output in VTK format;
- Python and C++ interfaces;
- Extensive documentation (see for instance: https://fenicsproject.org/documentation/).

2. FEniCS resources

The documentation for FEniCS is extensive. Resources include:

- **The FEniCS Tutorial.** The book *Solving PDEs in Python: The FEniCS Tutorial Volume I* is the is the perfect guide for new users. The tutorial explains fundamental concepts of the finite element method, FEniCS programming, and demonstrates how to quickly solve a number of PDEs.

  The PDF version of the book can be downloaded (legally and for free) from
  https://fenicsproject.org/pub/tutorial/pdf/fenics-tutorial-vol1.pdf.

  Python scripts for all the examples described in the tutorial can be found at
  https://github.com/hplgit/fenics-tutorial/tree/master/pub/python/vol1

- **FEniCS Demos.** These documented demonstration programs are a great way to learn the different features in FEniCS. They come already packaged in FEniCS when you install it and are available on-line at:
  https://fenicsproject.org/olddocs/dolfin/latest/python/demos.html .

- **Programmer's References.** Some of the classes and functions in DOLFIN are more frequently used than others. The Python implementations are described in
  https://fenicsproject.org/olddocs/dolfin/latest/python/api.html .

- **UFL** UFL is the way that FEniCS expresses variational forms in Python. A complete documentation of the functionalities of UFL is provided here
  https://fenics.readthedocs.io/projects/ufl/en/2019.1.0/manual.html.

- **Getting Help.** See: https://fenicsproject.org/community/

Other resources, although a little outdated and not fully compatible with the latest versions of FEniCS, include:

- **The FEniCS Book:** All 732 pages of the FEniCS book (*Automated Solution of Differential Equations by the Finite Element Method*) can be downloaded (legally!) from here: http://launchpad.net/fenics-book/trunk/final/+download/fenics-book-2011-10-27-final.pdf. This is the comprehensive reference to FEniCS, along with many examples of the applications of FEniCS to problems in science and engineering. You will notice that the first chapter of the book is just the FEniCS Tutorial (with some minor editorial differences).

- **The FEniCS Manual.** This is a 200-page excerpt from the FEniCS Book, including the FEniCS Tutorial, an introduction to the finite element method, and documentation of DOLFIN and UFL: http://launchpad.net/fenics-book/trunk/final/+download/fenics-manual-2011-10-31.pdf. Since it's an excerpt from the FEniCS Book, you probably won't need it.

3. Jupyter notebooks and Cloud Access

For this class we will use the legacy version of FEniCS, FEniCS 2019.1. To simplify the use of FEniCS for the demo and assignment problems of this course, we deployed a FEniCS in the cloud using Docker and JupyterHub. This way, you can run the codes presented in class through your web browser, without running into installation issues or version discrepancies.

- Instructions to connect to the cloud resources, username and password will be posted on Canvas. Please do not exchange the user info.

- For instructions on how to use Jupyter notebooks see https://docs.jupyter.org/en/latest/running.html#running.

- It is recommended that you change your password after the first login. To do so, open a new Terminal and enter the Linux command `passwd`.

- If you prefer to use the JupyterLab interface (https://jupyterlab.readthedocs.io/en/stable/) rather than the Jupyter Notebook interface, you can manually change the url in the address bar from
  `<SERVER_IP>/user/<USER_ID>/tree`
  to
  `<SERVER_IP>/user/<USER_ID>/lab`,
  where `<SERVER_IP>` is the server IP address, and `<USER_ID>` is your user id.

- Pro Tip: You can create a git repository to save your work on GitHub using the Jupyter Terminal.

Cloud computing resources for this course were provided to us by the Advanced Cyberinfrastructure Coordination Ecosystem: Services & Support (ACCESS), which is supported by National Science Foundation, through Explore ACCESS allocation MTH230002.