**Download FastX 2**: <https://webstore.illinois.edu/shop/product.aspx?zpid=2642>

**Download UIUC VPN**: Connect VPN to the UofI campus network: vpn.illinois.edu

**Use FastX to connect to our machines**:

i) Create an SSH connection under the + sign

ii) Name can be arbitrary. Host: ssm29, Port:22, User: your NetID

iii) The username and password will be the same as that of your NetID.

**Environment Setup**

Install Miniconda version 4.10.3 for Linux (or the latest version with Python 3.8.10)

<https://docs.conda.io/en/latest/miniconda.html>

**Installing FEniCS library**

FEniCS provides a computing platform in python (and C++) to quickly implement the finite element method to solve partial differential equations.

Open a terminal and type:

conda create -n fenicsproject -c conda-forge fenics

source activate fenicsproject

You will likely need to replace one package in the conda-forge installation:

conda install -c conda-forge mpi4py=3.0.3

Other useful packages to install:

conda install -c conda-forge notebook (jupyter notebook)

conda install -c conda-forge matplotlib

conda install -c conda-forge scipy

conda install -c conda-forge mshr

conda install -c conda-forge gmsh

**To run a code**

Open terminal and first type:

* conda activate fenicsproject
* export OMP\_NUM\_THREADS=1 (this is to make the parallelization work properly)

***Tip***: Add the following line to your .bashrc file to map the above two commands to the keyword *fenics*, which you can run whenever you open a new terminal window:

alias fenics=”conda activate fenicsproject; export OMP\_NUM\_THREADS=1”

Then go to the directory where the code script is saved and run the code

mpirun -np <cpus> python <scriptname.py>

You can choose the number of cpus to be between 1 and 24, or the maximum number of cores available (which can be obtained with the command nproc)

**Visualizing the results**

To visualize results of FEM simulations, we use the software paraview (<https://www.paraview.org/download/>).

* Download the linux version after logging into FastX. This version works well: Paraview-5.7.0-MPI-Linux-Python2.7
* After extracting the downloaded files (I suggest into a folder such as ~/tools), run the following commands to run paraview:



A helpful tutorial for Paraview (the first 2 videos in particular):

<https://www.youtube.com/watch?v=OxcU6Upz5ZA&list=PLvkU6i2iQ2fpcVsqaKXJT5Wjb9_ttRLK-&index=1>

***Typical Paraview workflow / Commonly Used Filters***:

* Open (should be directly below File) XDMF file generated by fenics script, choose the default XDMF reader, Apply
* How to change background color (optional):
  + Click the gear button in the Properties panel
  + Select a color under the Background header under View (Render View) near the bottom (I recommend either white or a light grey)
  + Click the “Save current view settings” button next to View (Render View)
* How to change the default colormap (optional):
  + View -> Color Map Editor -> Choose preset (heart next to Mapping Data)
  + I suggest Blue to Red Rainbow (need to switch from Default to All on top)
  + Click the “Save current color map settings as default for all arrays” button on bottom of Color Map Editor panel
* Select the desired output variable (“Temperature”, “Alpha”, “Velocity”, etc), and component (if any), to display in the dropdown menu on top
* Can view mesh by switching from Surface to Surface With Edges in the dropdown menu on top
* Filters (under toolbar):
  + Plot Over Line – view all output variables at sampled points along a line. Select endpoints on mesh using Ctrl+1 and Ctrl+2
  + Calculator – Perform operations on variables (ie. compute Von Mises stress from stress tensor, compute norm of vector variable, etc). Make sure to name result
  + Contour – show isolines for a scalar variable. Next to Value Range, first Remove all Entries, then Add a Range of Values (ie. can add 20 equally spaced contour lines between min and max of variable). For contours of a component of a vector variable, use the Calculator filter to select the component
  + Warp by Vector – used to visualize displacement of an elasticity problem
* Filters (under Filters -> Alphabetical):
  + Reflect – Useful for extending visualization of a symmetric geometry

***Tip***: Create a shortcut on the taskbar panel to easily open Paraview, with the command:

<PATH TO PARAVIEW INSTALLATION>/bin/paraview-mesa paraview