The Non-affine Fiber Network Plugin for Finite Element Analysis

Ryan R Mahutga, Victor H Barocas, & Patrick W Alford

Department of Biomedical Engineering

University of Minnesota, MN, USA

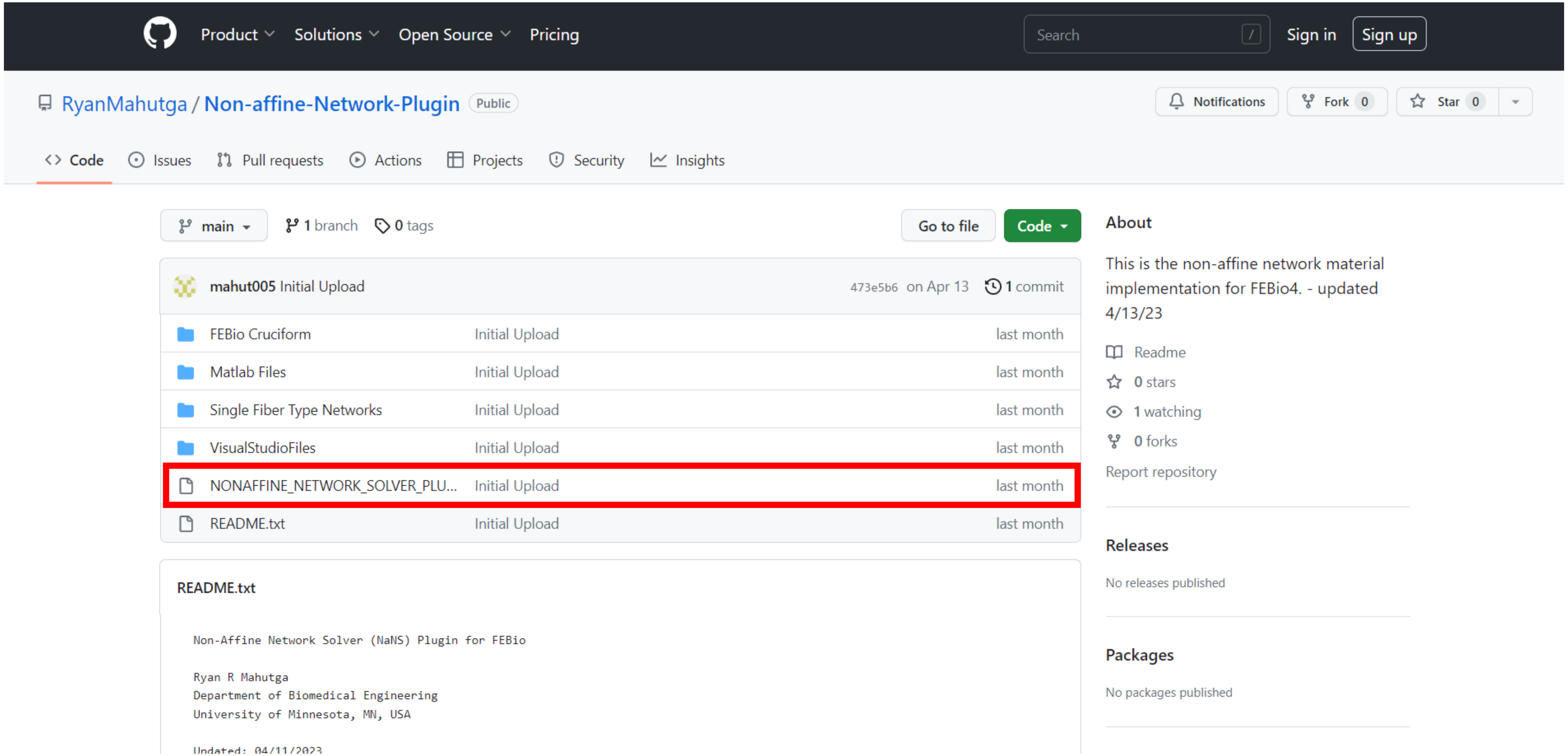
May 17, 2023

Getting Started

What You’ll Need:

1. FEBio Studio 2 + the FEBio SDK
2. Text Editor (Notepad++)
3. MATLAB (any newish version should work, I used 2021b)

Download the Plugin



RyanMahutga / Non-affine-Network-Plugin

mahut005 Initial Upload 473e5b6 on Apr 13 1 commit

FEBio Cruciform	Initial Upload	last month
Matlab Files	Initial Upload	last month
Single Fiber Type Networks	Initial Upload	last month
VisualStudioFiles	Initial Upload	last month
NONAFFINE_NETWORK_SOLVER_PLU...	Initial Upload	last month
README.txt	Initial Upload	last month

README.txt

```
Non-Affine Network Solver (NaNS) Plugin for FEBio

Ryan R Mahutga
Department of Biomedical Engineering
University of Minnesota, MN, USA

Updated: 04/11/2023
```

About

This is the non-affine network material implementation for FEBio4. - updated 4/13/23

Readme

0 stars

1 watching

0 forks

Report repository

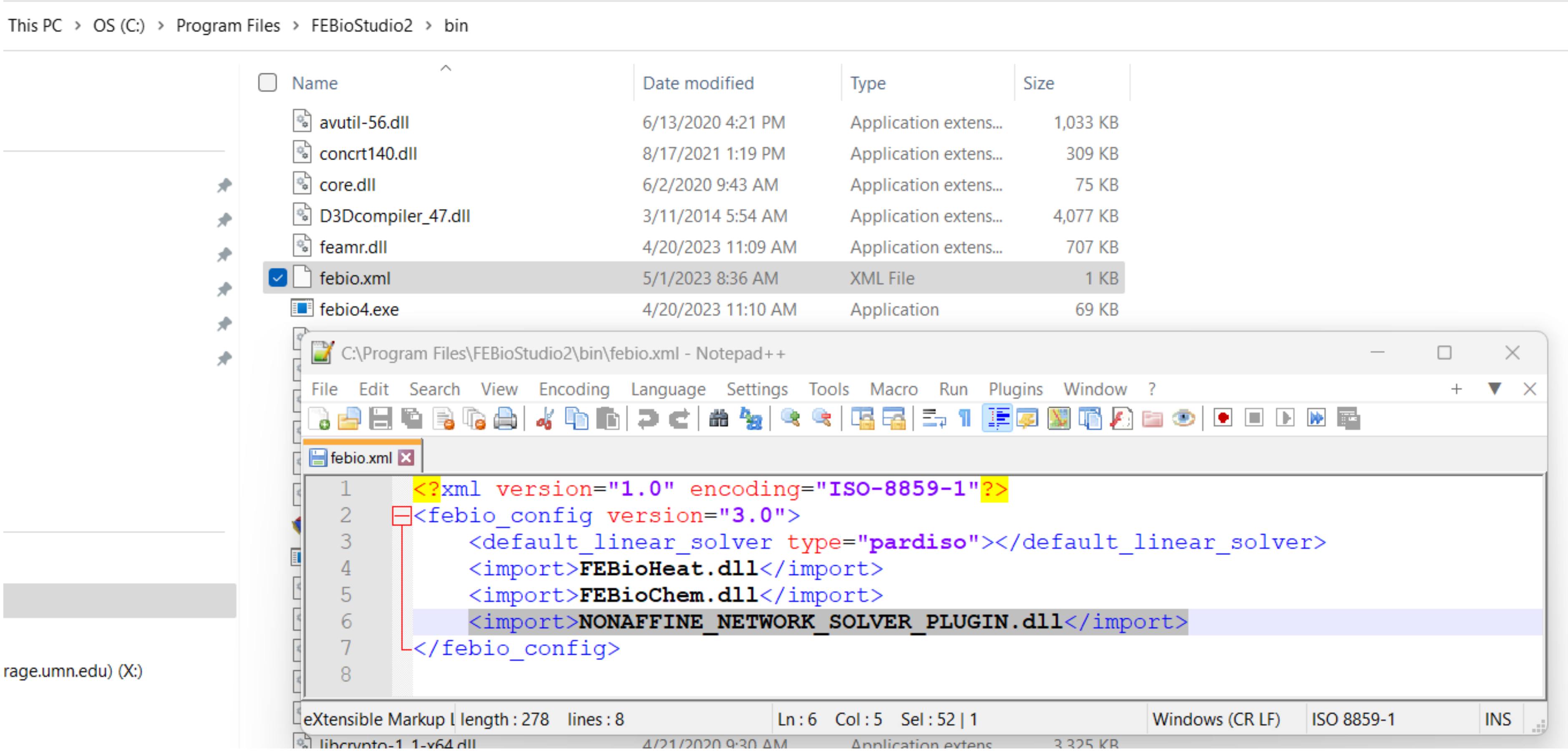
Releases

No releases published

Packages

No packages published

Modify the febio.xml File

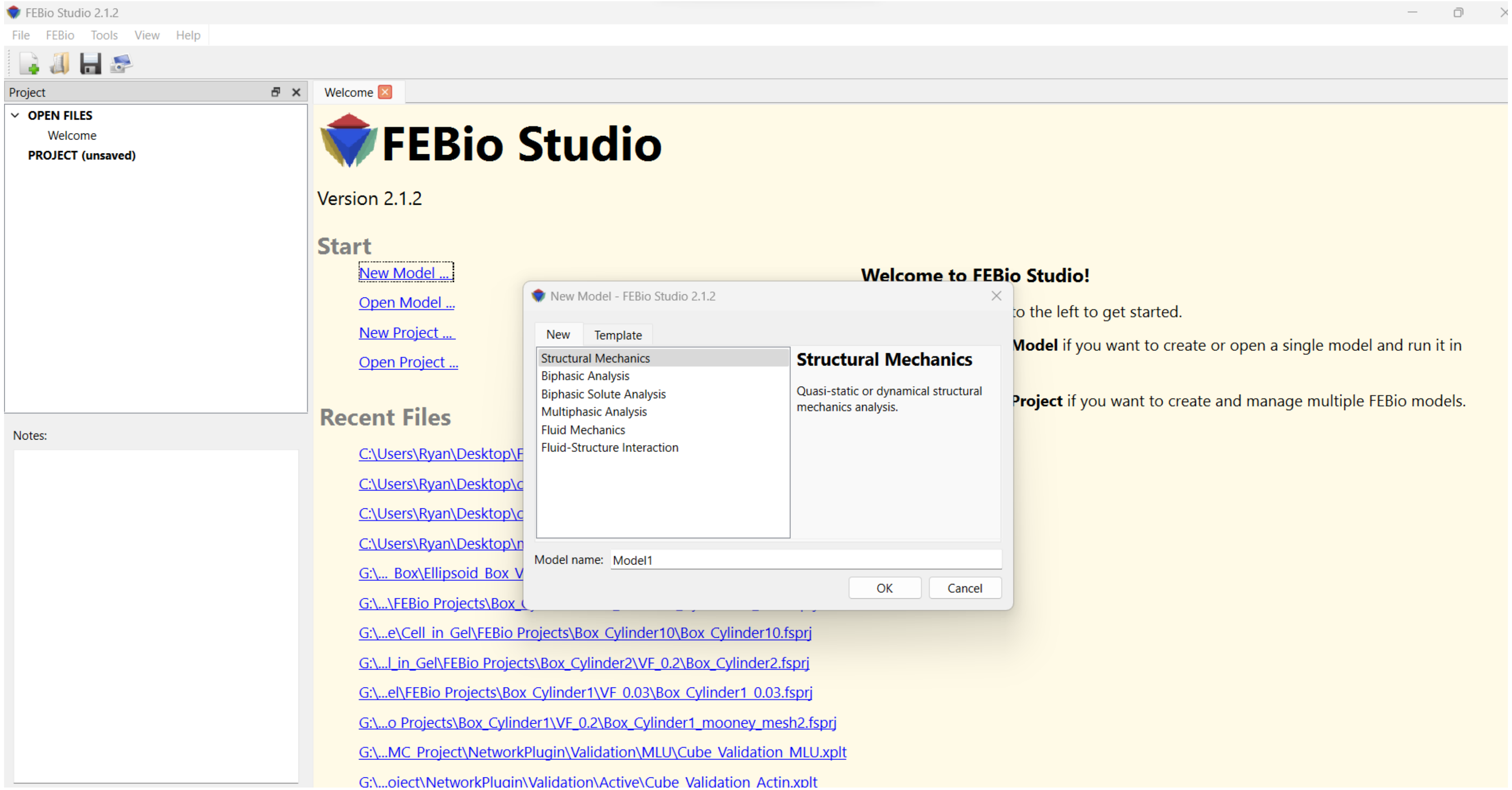


This PC › OS (C:) › Program Files › FEBioStudio2 › bin
C:\Program Files\FEBioStudio2\bin\febio.xml - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
febio.xml
rage.umn.edu) (X:)
eXtensible Markup l length : 278 lines : 8 Ln : 6 Col : 5 Sel : 52 | 1 Windows (CR LF) ISO 8859-1 INS

Name	Date modified	Type	Size
avutil-56.dll	6/13/2020 4:21 PM	Application extens...	1,033 KB
concrt140.dll	8/17/2021 1:19 PM	Application extens...	309 KB
core.dll	6/2/2020 9:43 AM	Application extens...	75 KB
D3Dcompiler_47.dll	3/11/2014 5:54 AM	Application extens...	4,077 KB
feamr.dll	4/20/2023 11:09 AM	Application extens...	707 KB
☑ febio.xml	5/1/2023 8:36 AM	XML File	1 KB
febio4.exe	4/20/2023 11:10 AM	Application	69 KB
libcrypto-1_1-x64.dll	4/21/2020 9:30 AM	Application extens...	3,325 KB

```xml
<?xml version="1.0" encoding="ISO-8859-1"?>
<febio_config version="3.0">
    <default_linear_solver type="pardiso"></default_linear_solver>
    <import>FEBioHeat.dll</import>
    <import>FEBioChem.dll</import>
    <import>NONAFFINE_NETWORK_SOLVER_PLUGIN.dll</import>
</febio_config>

```

Create a Model



FEBio Studio

Version 2.1.2

C:\Users\Ryan\Desktop\F

C:\Users\Ryan\Desktop\c

C:\Users\Ryan\Desktop\c

C:\Users\Ryan\Desktop\n

G:\... Box\Ellipsoid_Box_V

G:\...\FEBio Projects\Box_C

G:\...e\Cell_in_Gel\FEBio Projects\Box_Cylinder10\Box_Cylinder10.fsprj

G:\...l_in_Gel\FEBio Projects\Box_Cylinder2\VF_0.2\Box_Cylinder2.fsprj

G:\...el\FEBio Projects\Box_Cylinder1\VF_0.03\Box_Cylinder1_0.03.fsprj

G:\...o Projects\Box_Cylinder1\VF_0.2\Box_Cylinder1_mooney_mesh2.fsprj

G:\...MC_Project\NetworkPlugin\Validation\MLU\Cube_Validation_MLU.xplt

G:\...oject\NetworkPlugin\Validation\Active\Cube_Validation_Actin.xplt

Welcome to FEBio Studio!

to the left to get started.

Model if you want to create or open a single model and run it in

Project if you want to create and manage multiple FEBio models.

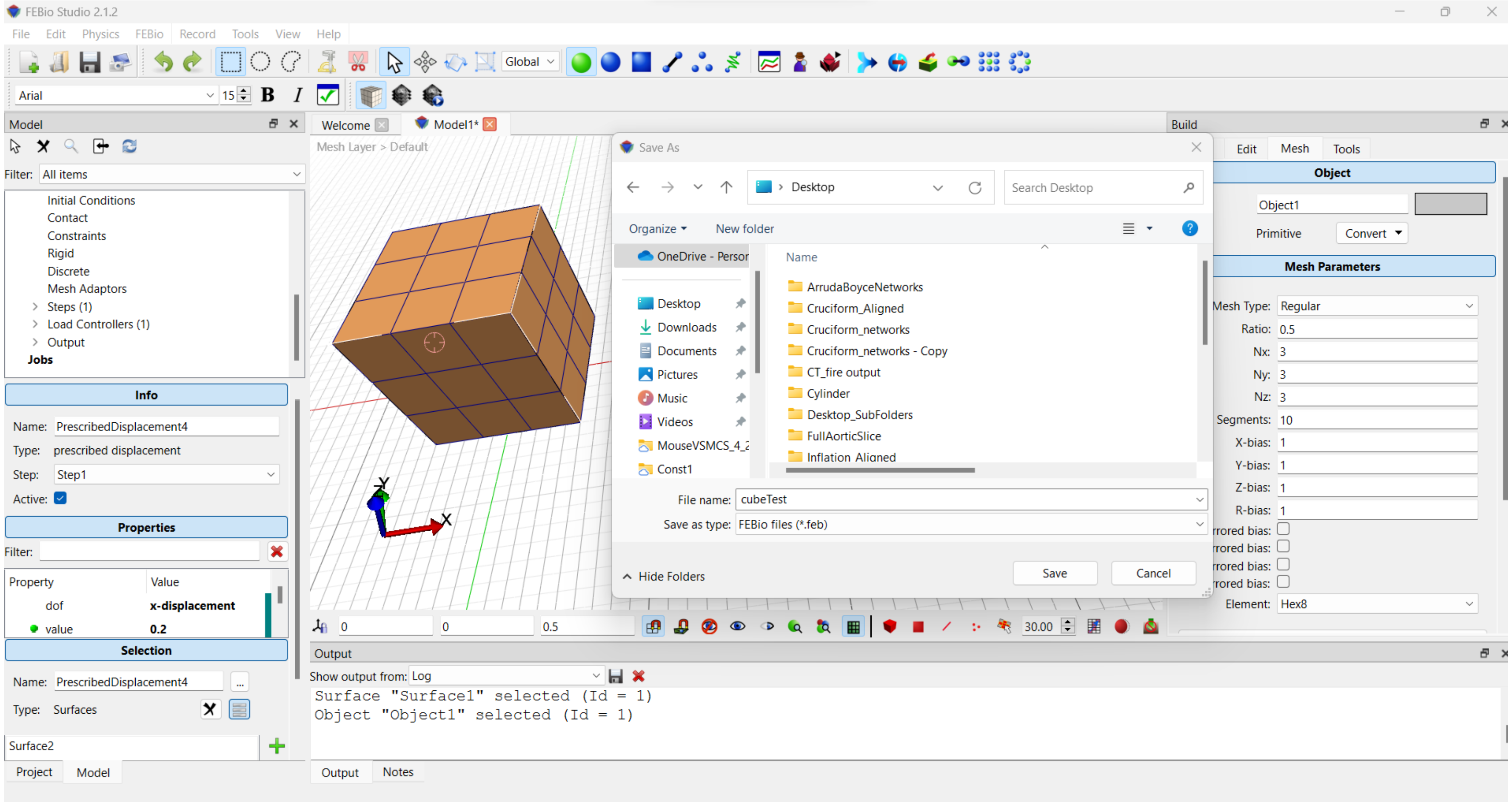
Structural Mechanics

Structural Mechanics

Quasi-static or dynamical structural mechanics analysis.

Model name: Model1

Export the Model .feb File



Info

Name: PrescribedDisplacement4

Type: prescribed displacement

Step: Step1

Active: ☑

Properties

Filter:

Property	Value
dof	x-displacement
value	0.2

Selection

Name: PrescribedDisplacement4

Type: Surfaces

Surface2

File name: cubeTest

Save as type: FEBio files (*.feb)

Object1

Primitive: Convert

Mesh Type: Regular

Ratio: 0.5

Nx: 3

Ny: 3

Nz: 3

Segments: 10

X-bias: 1

Y-bias: 1

Z-bias: 1

R-bias: 1

rrored bias: □

rrored bias: □

rrored bias: □

rrored bias: □

Element: Hex8

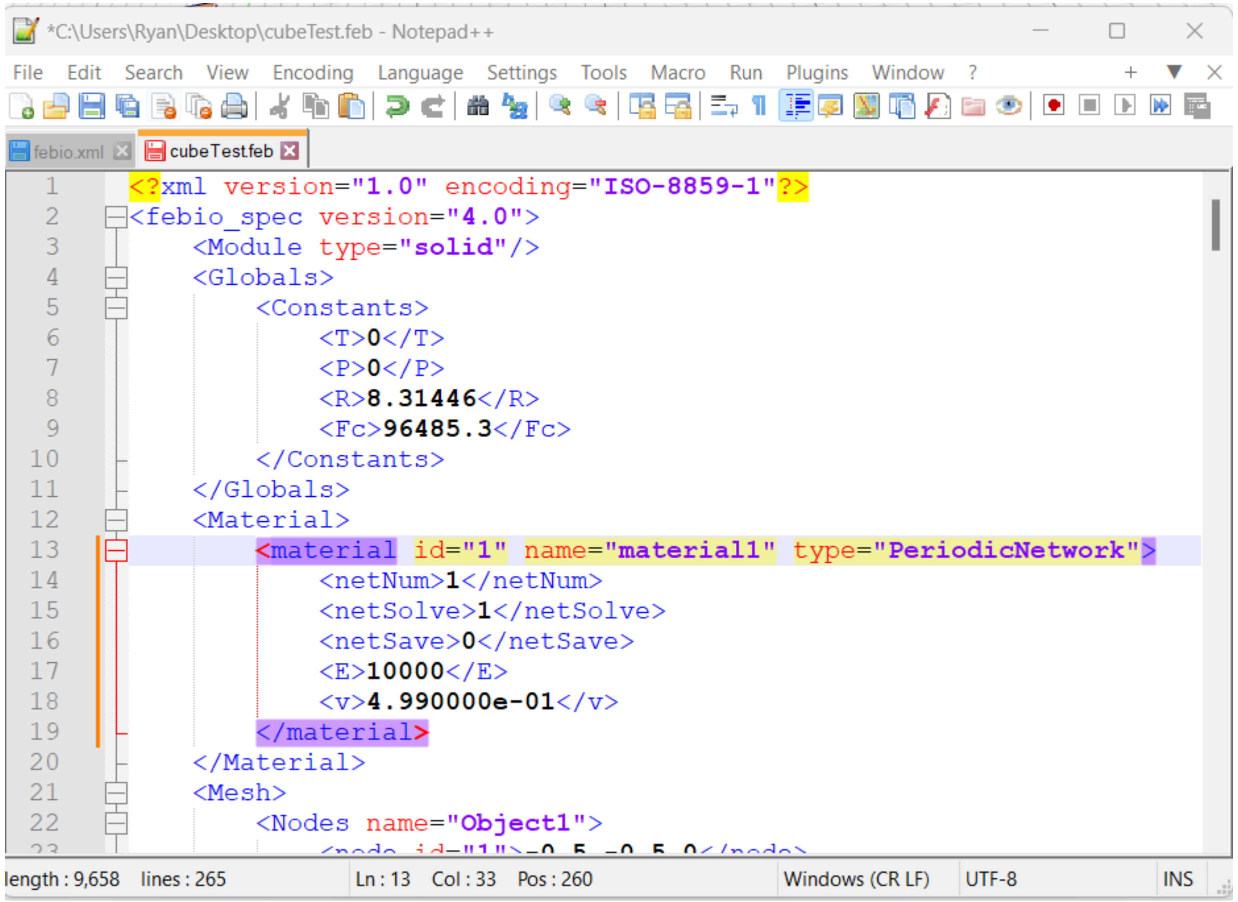
0 0 0.5

Show output from: Log

```
Surface "Surface1" selected (Id = 1)
Object "Object1" selected (Id = 1)
```

Edit the .feb File

- NetNum refers to the number appended to ‘PeriodicNetworkN.txt’ which defines which network to use for which material.
- NetSolve when true (1) solves the full network equilibrium when false (not=1) uses the affine approximation.
    - Affine approximation is faster and can be used when solving intermediate states you don’t really care about (i.e., one can use a load curve on this parameter so the full solution is only solved on the final iteration)
- NetSave when true (1) saves many text files of network data (fiber stresses, stretches, etc.). The NetSave feature overwrites data and only saves one file per element so beware of its use.
- E is the neoHookean modulus parameter and v is the Poisson’s Ratio of the material.



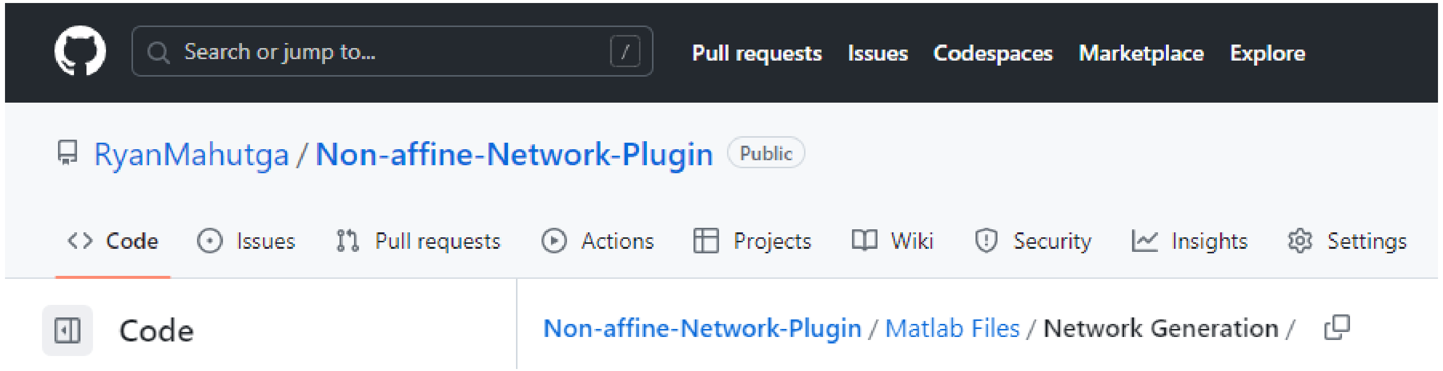
```xml
<?xml version="1.0" encoding="ISO-8859-1"?>
<febio_spec version="4.0">
    <Module type="solid"/>
    <Globals>
        <Constants>
            <T>0</T>
            <P>0</P>
            <R>8.31446</R>
            <Fc>96485.3</Fc>
        </Constants>
    </Globals>
    <Material>
        <material id="1" name="material1" type="PeriodicNetwork">
            <netNum>1</netNum>
            <netSolve>1</netSolve>
            <netSave>0</netSave>
            <E>10000</E>
            <v>4.990000e-01</v>
        </material>
    </Material>
    <Mesh>
        <Nodes name="Object1">
            <node id="1">-0.5,-0.5,0</node>
```

Edit the .feb (ADVANCED)

- Included in the Matlab Files folder on Github is a folder called FE Simulation Modification
  - Open main2.m – this code will allow you to take a base .feb file with a specified number of elements and break it into a different material for each element, and create a new network for each material.
  - The file writeFEBio2.m will need to be modified for specific simulations (including pressure, using different mesh types, including material axes, etc.

Create a Network

Navigate to MatlabFiles/Network Generation on GitHub



RyanMahutga / Non-affine-Network-Plugin

Non-affine-Network-Plugin / Matlab Files / Network Generation /

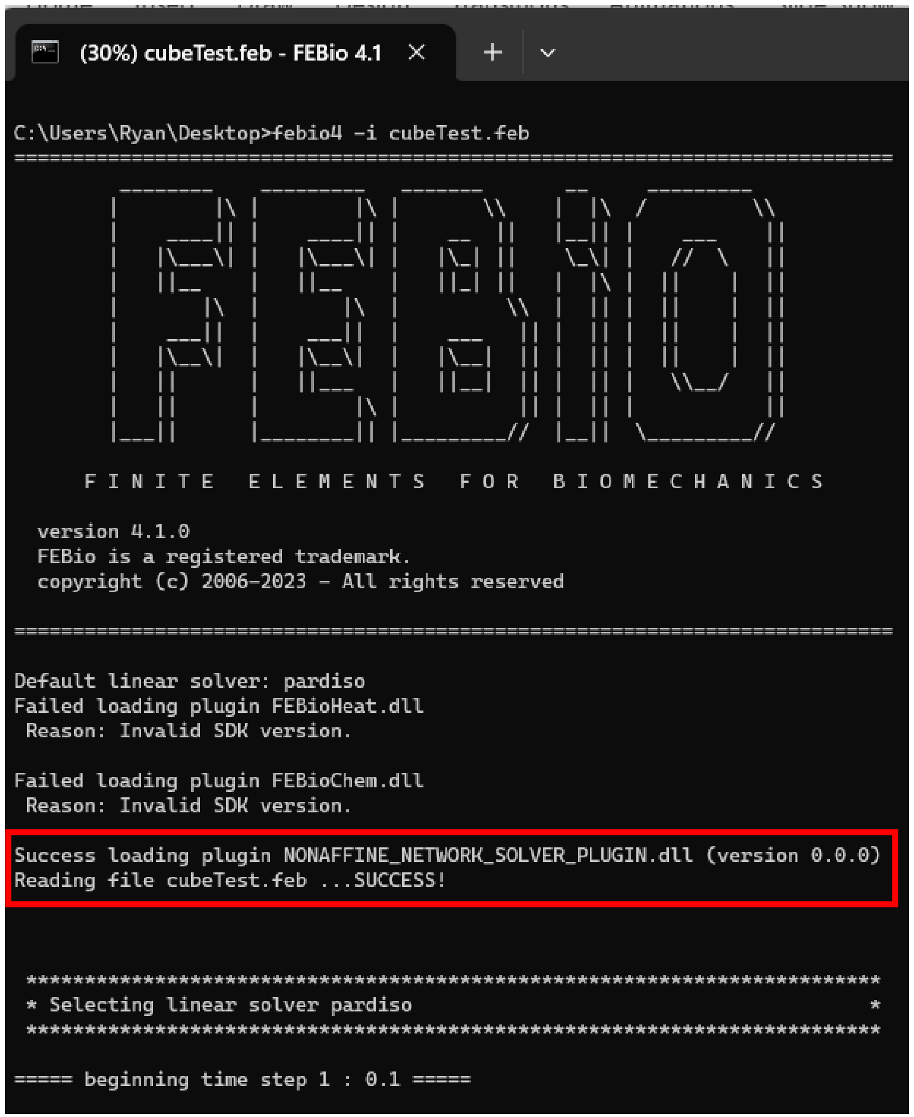
There are two files for creating networks

1. collagen_networks.m which creates single fiber type networks
2. createMLU.m which creates medial lamellar unit networks

Create a Network

- Inside the main network creation files, one will find several important sub-functions
  - periodicDelaunay.m – performs periodic Delaunay tessellation on a set of nodes and extracts fibers
  - periodicDelaunay2D.m – performsa 2D periodic Delaunay tessellation on a set of nodes and extracts fibers
  - removeDupes.m – checks for duplicate fibers and remove them
  - NetworkPare.m – pares the network connectivity to the specified value
  - MinDegreeReorder.m – reorders the nodes to improve computational efficiency
  - networkFeatures.m – calculates fiber lengths and applies fiber radii
  - WriteNet3.m – writes the network to a text file
  - solve_periodic_BCs2.m is used to calculate fiber boundary crossing locations for plotting netoworks
  - plot_net_single_fib_type.m plots a single network with fiber types color coded
  - plot_net_tile_fib_type.m plots a 3x3 grid of networks with defined normal direction (1,2,3)

Run the Simulation

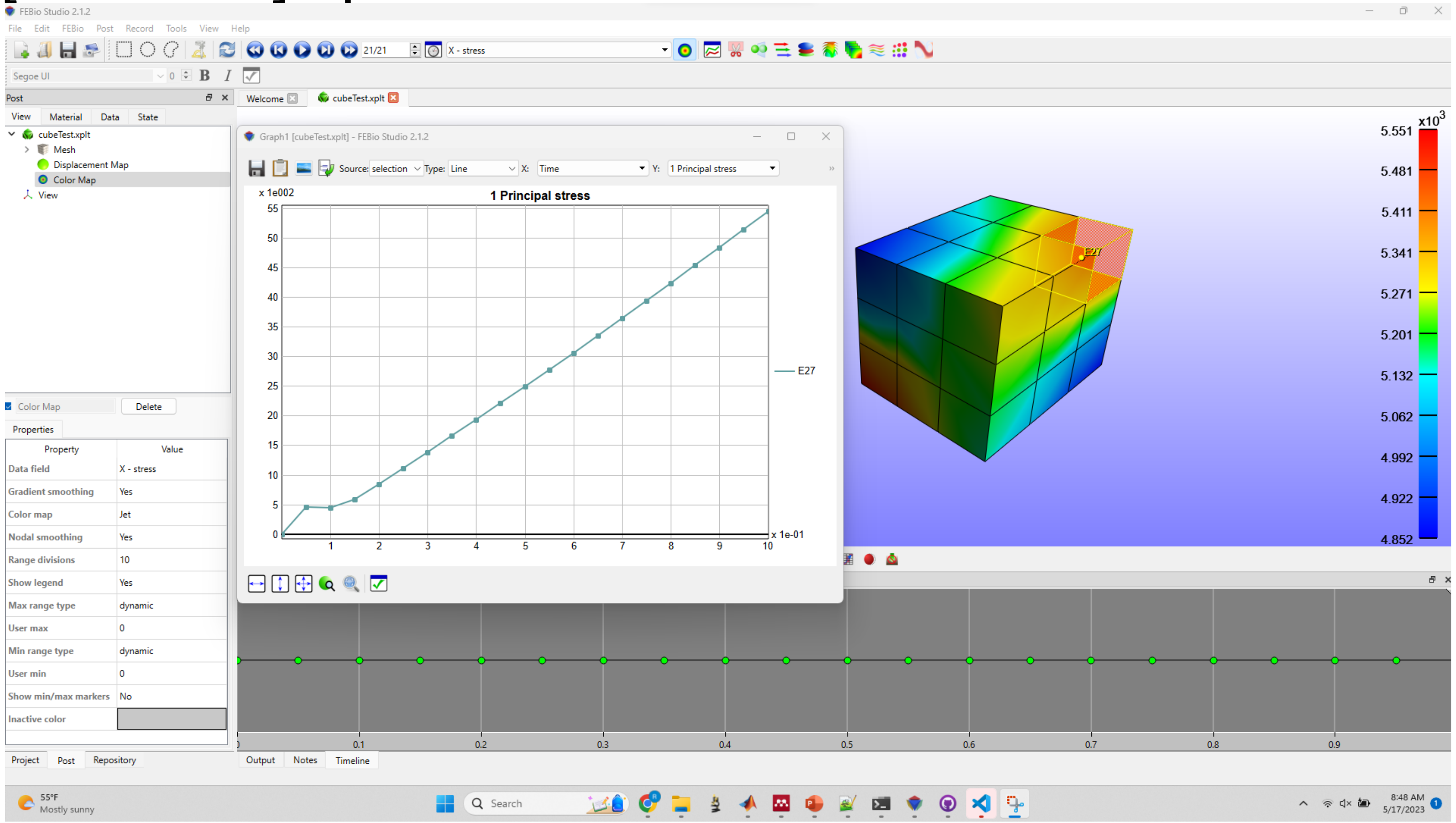


```
C:\Users\Ryan\Desktop>febio4 -i cubeTest.feb
===========================================================================
         ________    _________   _______       __     _________
        |        |\ |        |\ |       \\    |  |\  /         \\
        |    ____|| |    ____|| |    __  ||   |__|| |    ___    ||
        |   |\___\| |   |\___\| |   |\_| ||    \_\| |   //  \   ||
        |   ||__    |   ||__    |   ||_| ||   |  |\ |  ||    |  ||
        |       |\  |       |\  |         \\  |  || |  ||    |  ||
        |    ___||  |    ___||  |    ___   || |  || |  ||    |  ||
        |   |\__\|  |   |\__\|  |   |\__|  || |  || |  ||    |  ||
        |   ||      |   ||___   |   ||__|  || |  || |   \\__/   ||
        |   ||      |        |\ |          || |  || |           ||
        |___||      |________|| |_________//  |__||  \_________//

      F I N I T E   E L E M E N T S   F O R   B I O M E C H A N I C S

  version 4.1.0
  FEBio is a registered trademark.
  copyright (c) 2006-2023 - All rights reserved

===========================================================================

Default linear solver: pardiso
Failed loading plugin FEBioHeat.dll
 Reason: Invalid SDK version.

Failed loading plugin FEBioChem.dll
 Reason: Invalid SDK version.

Success loading plugin NONAFFINE_NETWORK_SOLVER_PLUGIN.dll (version 0.0.0)
Reading file cubeTest.feb ...SUCCESS!



 *************************************************************************
 * Selecting linear solver pardiso                                       *
 *************************************************************************

===== beginning time step 1 : 0.1 =====
```

** If febio4 is unrecognized as a command go to System Properties >> Advanced >> Environment Variables >> Path >> Edit >> New >> [location of FEBio Studio]

Visualize the Results

- Open the [filename].xplt file to view the results.



Graph1 [cubeTest.xplt] - FEBio Studio 2.1.2

Source: selection Type: Line X: Time Y: 1 Principal stress

Data field: X - stress

Property	Value
Data field	X - stress
Gradient smoothing	Yes
Color map	Jet
Nodal smoothing	Yes
Range divisions	10
Show legend	Yes
Max range type	dynamic
User max	0
Min range type	dynamic
User min	0
Show min/max markers	No
Inactive color	