Assignment 3: Non-linear Elasticity Block

The main objective of this assignment is to experiment with a 2D finite element code for a nonlinear elasticity.

What Newton's method does is the following:

1. Solve $J(x^k)\Delta x^k = -r(x^k)$.
2. Consider an energy-descent search direction, $s^k = \Delta x^k$ if $r(x^k)^T\Delta x^k \leq 0$, and $s^k = -\Delta x^k$ if $r(x^k)^T\Delta x^k > 0$.
3. Solve the 1D minimization problem $\min(\alpha<0)\ \pi(x^k + \alpha s^k)$.
4. Update $x^{k+1} = x^k + \alpha s^k$.

In this exercise we have to:

a) Identify in the code (file, lines) the following items:

The load and the geometry are defined in the preprocessing function, in which the initial mesh configuration is also defined. The lines of this definition are from line 8 to 59.

b) The choice of solution method (Newton's method with or without linesearch).

We can see that the choice of the solution method is done in the main_incremental_iteretive code. In line 20 of this code we have options.method that defines the different options, that are:

0: plain Newton-Rapshon
1: Newton-Rapshon with line search

And at line 21 we can activate the line search as we can see here:
options.linesearch=1; 0: off, 1: on

In addition to the main buckling code we have also similar options, on line 18 you can choose the type of problem:

options.method = 1

0: vanilla Newton-Raphson
1: Newton-Rapshon
11: Modified NR
2: L-BFGS
3: Conjugate Gradient

In line 20 you have the possibility of activate or not the options.linesearch line = 1;
0: off, 1: on.

And for method 3 it is automatically on.

c) The implementation of the solution method.

We can see the implementation of the solution method in the code called: Equilibrate.m file for vanilla Newton-Raphson, Newton Raphson and modified Newton Raphson

d) The implementation of the incremental-iterative strategy, with smart initial guesses for imposed displacements.

We can see the implementation of the incremental–iterative strategy in the main_incremental_iterative file from line 39 to line 71 and main_buckling file from line 65 to line 71.

e) The introduction of random perturbations in the initial guesses of the solution method.

```
x=x+rand(size(x))*.001; %random perturbations

% 0: upsetting of a block, dead load
% 1: upsetting of a block, imposed displacements
% 2: compression of a slender beam, imposed displacements
% 3: compression of a slender beam, dead load
% 4: arch, dead load at center of the arch
% 5: arch, dead load near the supports
```

We have to run also the following tests:

a) Example 0 without line-search and for mod1.force = -3e0 and mod1.force = 3e0.
b) Example 1 without line-search and for lambda=[1:.025:2] and lambda=[1: .01:0.5].
c) Examples 2 and 3 with and without line-search, with and without random perturbations.
d) Examples 4 and 5 with and without line-search, without random perturbations.

And after that we have to do a report and discuss a selection of results that illustrate the features of the nonlinear model vis a vis the following features of linear models:

- Proportionality of the response with respect to the loads.
- Symmetry of the response with respect to the sign of the loads.
- Uniqueness of solutions.
- Stability of the solutions.

Comment on the physical validity of the solutions given by both models. Finally, sum up your conclusions.

Report and discussion of results

Once we have run different examples it is appreciable that for non-linearity cases the response with respect to the loads are not proportional. We can do a comparison between the behavior of the non-linear case and the linear case. In matlab files the code gives a graph of both behaviors. We can see that in the non-linear case the deformation (response) is not proportional if we increase or decrease the load, whereas in the linear case if we increase the load, increase the deformation proportionality.

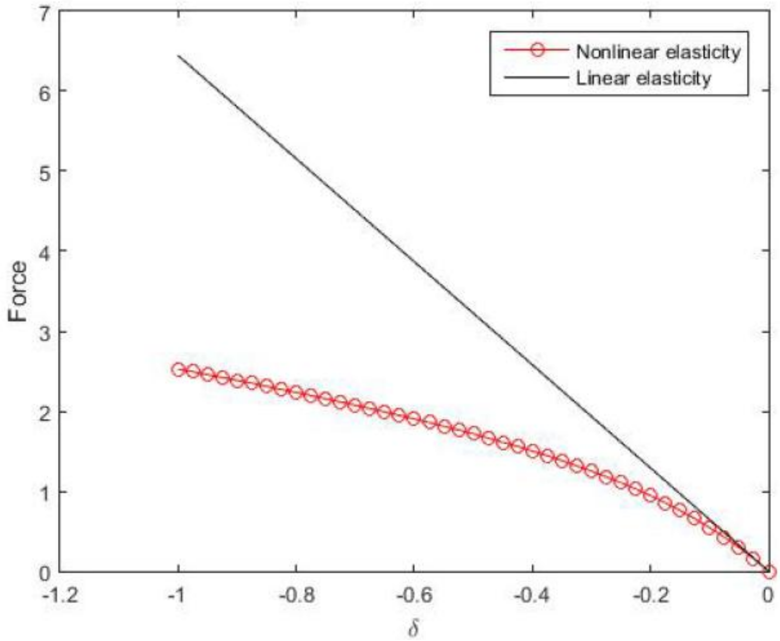

Figure1: Deformation-Force

Example 1 without line-search and with lambda= [1,.025,2]

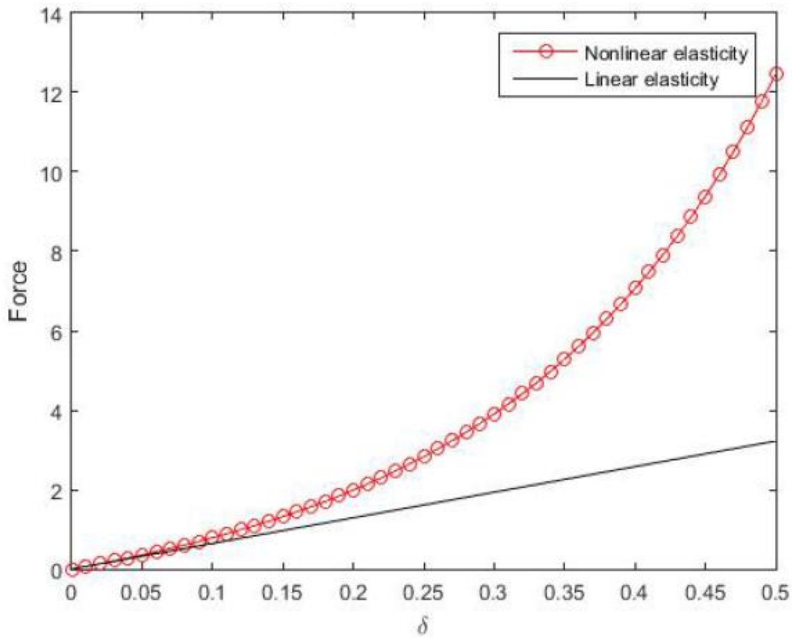

Figure 2: Deformation-Force

Example 1 without line-search and for lambda = [1:-.01:0.5]

We have to study also the symmetry of the response with respect to the sign of the load. In this part we have used the example 0 (upsetting of a block with Dead load). And we can see that for a linear model we have a symmetric response but for a non-linear model the response is asymmetric.

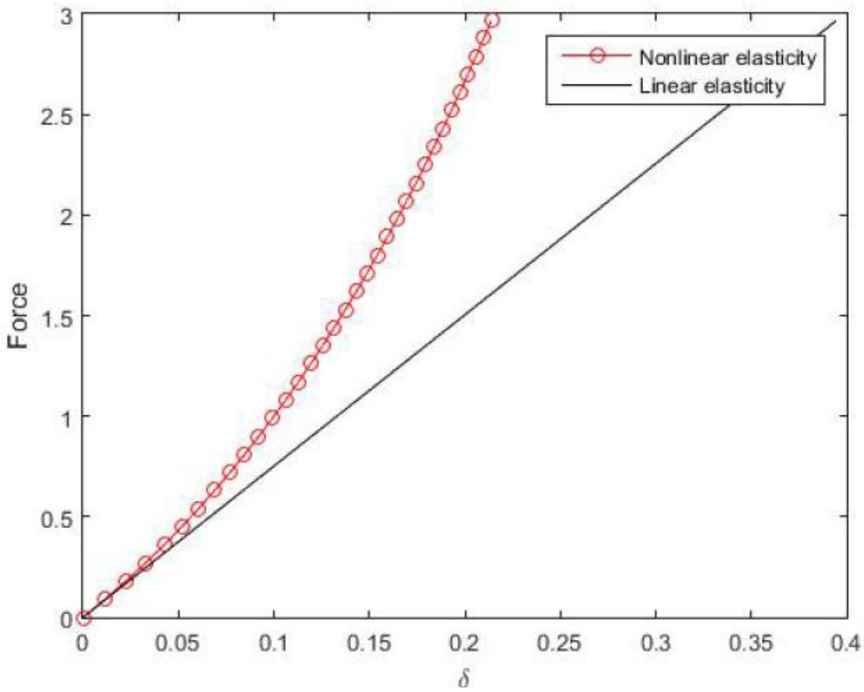

Figure 3: Example 0 without line-search and mod1.force=3e0

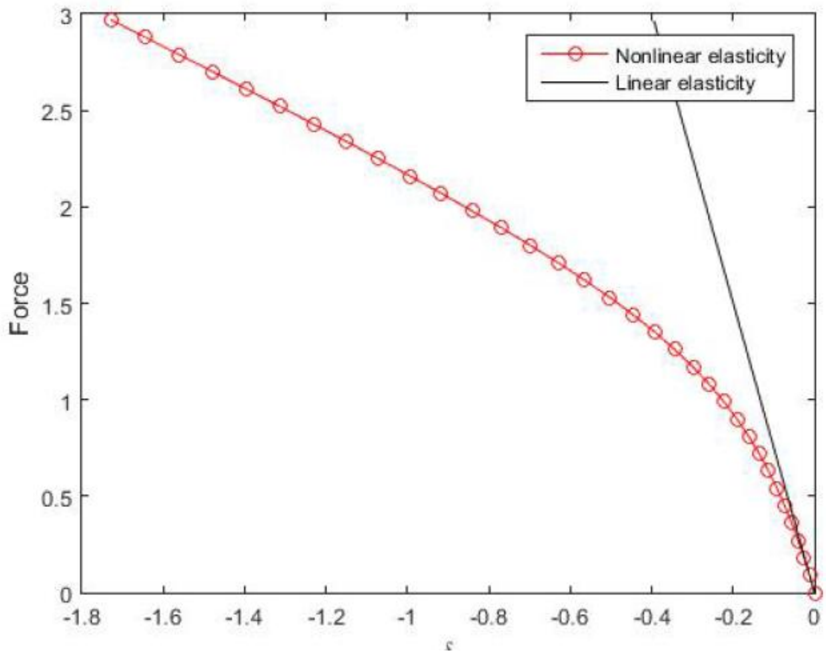

Figure 4: Example 0 without line-search and for mod1.force = -3e0

In conclusion we can say that the linear problems are problems that have unique solution. While this fact does not happen for non-linear problems as we can see solving different cases, we always get more than one solution if we use main_buckling or main_incremental_itereative files.

Here, we can see that there are several cases in which we have more than 1 solution (the solution is not unique in non-linear)

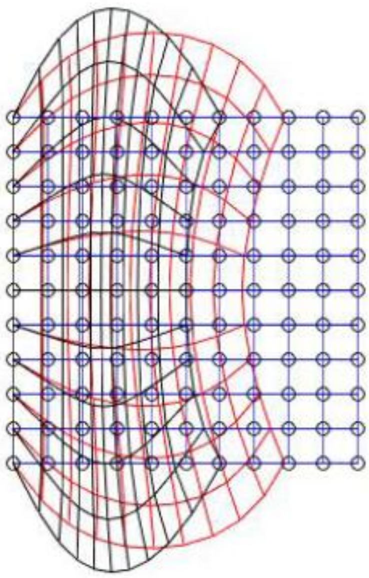

Figure 5: Example 0 without line-search, force= 3e0.

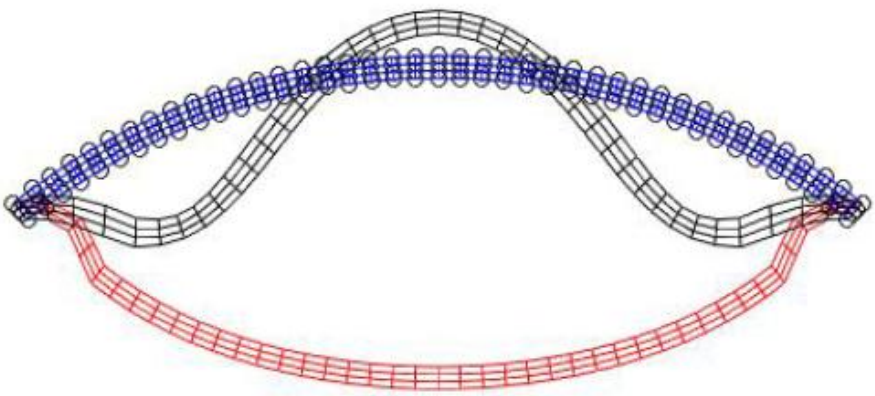

Figure 6: Example 5 with line-search (no random perturbations)

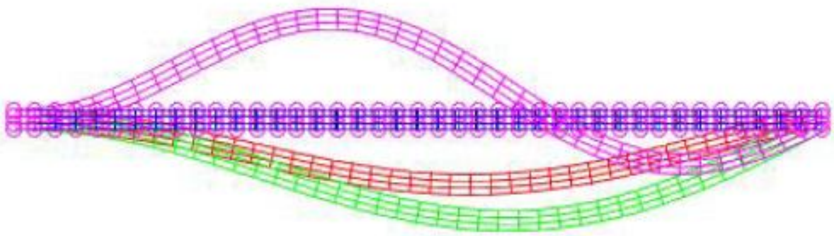

Figure 7: Example 3 using the main_buckling code with line-search (without random perturbations)