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

(a) The definition of the example (loading, geometry):

The definition of the example is in the preprocesing.m file. On that file, there is a switch command where the entire example and its parameters defined. First, the nodes coordinates of the geometry, the code load, its degrees of freedom…

In addition, there is an explanation (%text line) in the main_incremental_iterative.m file where every example is briefly explain.

(b) The choice of solution method (Newton's method with or without line-search):

In main_incremental_iterative.m file, at the beginning, there is a definition of some parameters. On the %Equilibrate ones you can choice the solution method. So, when you run main_incremental_iterative you can define: option.method=0 (vanilla-N-R) or option.method=1 (N-R) or option.method=11 (modified N-R).

Also buckling.m file has this parameters definition.

(c) The implementation of the solution method:

In Equilibratin.m the implementation of the different solution methods is specified.

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

This implementation can be found in Energy.m file.

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

In main_incremental_iterative.m file after calculating x variable a rand(x) value is added to this with the next expression:

Line 50   `x=x+rand(size(x))*.001; %random perturbations`"

2. Run the following tests

(a) Example 0 without line-search and for mod1.force = -3e0 and mod1.force = 3e0.

Exemple 0: upsetting of a block, dead.

In the next tables, figures obtained with Matlab are shown.
It is easy to observed in this case the proportionally and also the symmetry of the response due to the load.
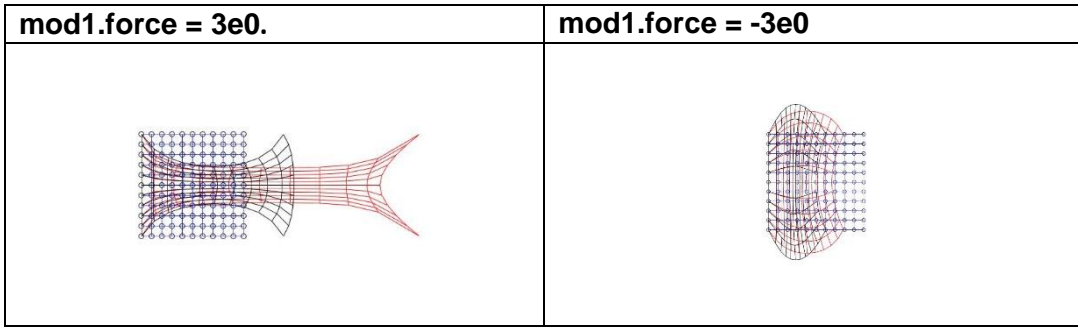In the last table, the stability of the solution is shown.

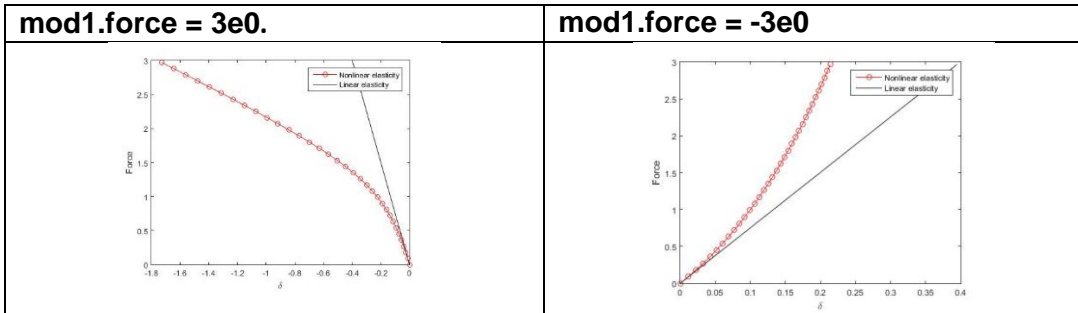| mod1.force = 3e0.	mod1.force = -3e0

Table 1: Deformed shape.

mod1.force = 3e0.	mod1.force = -3e0

Table 2: Force-delta.

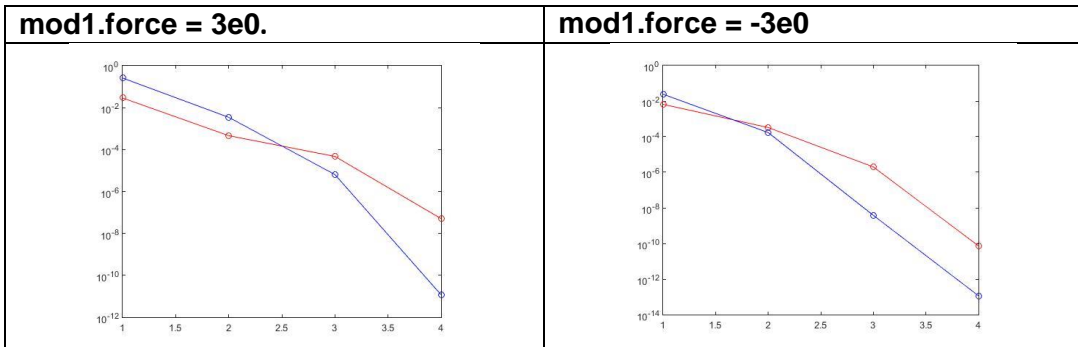| mod1.force = 3e0.	mod1.force = -3e0

Table 3. Stability

(b) Example 1 without line-search and for lambda=[1:.025:2] and lambda=[1:.01:0.5].

Exemple 1: upsetting of a block, imposed displacements

The results obtained in this calculation are similar as in example 0. Results are proportional and it is also easy to see the symmetry due to the sing of the lambda parameter.
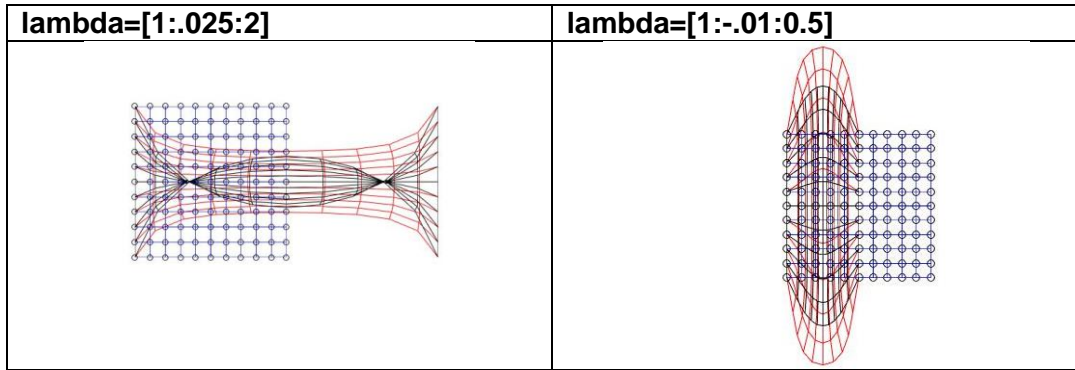
lambda=[1:.025:2]	lambda=[1:-.01:0.5]

Table 4. Deformed shape

lambda=[1:.025:2]	lambda=[1:-.01:0.5]

Table 5. Force-delta
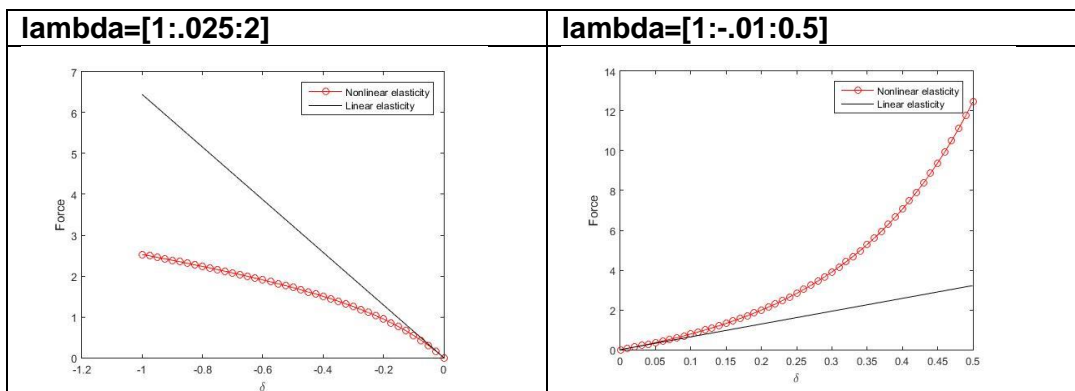
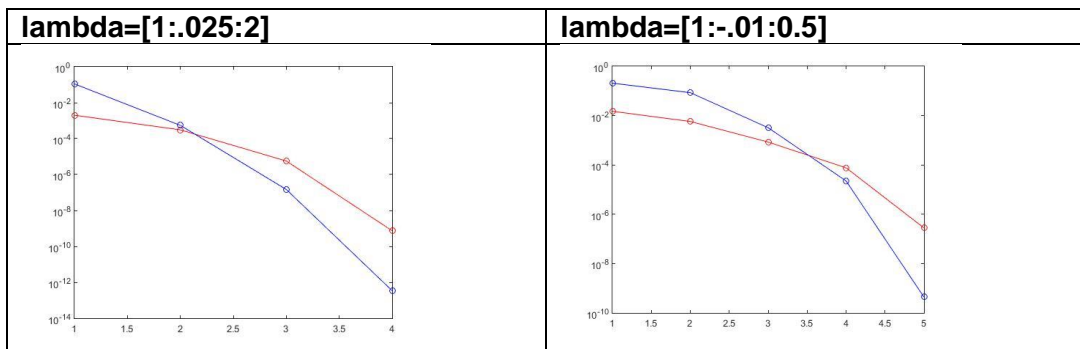| lambda=[1:.025:2]	lambda=[1:-.01:0.5]

Table 6. Stability

(c) Examples 2 and 3 with and without line-search, with and without random perturbations.

Example 2: compression of a slender beam, imposed displacements

Example 3: compression of a slender beam, dead load

Examples 2 and 3 were run with the same parameters on these calculations.

For the Example 2 very similar results are obtained with and without perturbations.

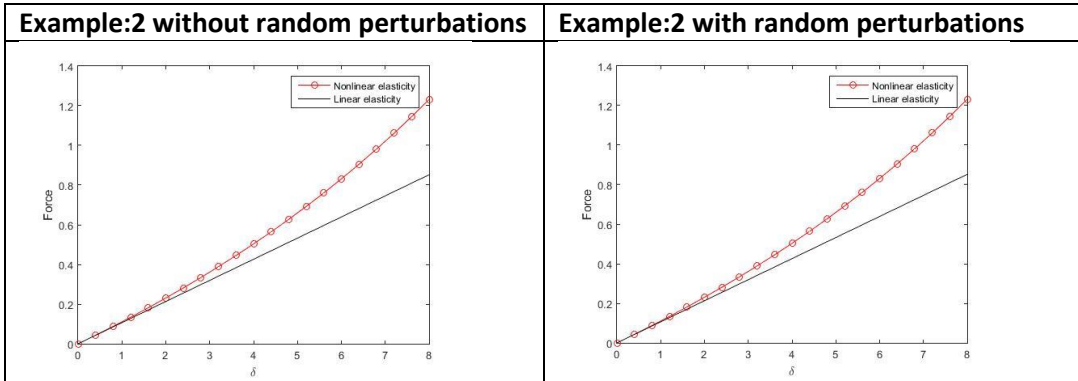| Example:2 without random perturbations	Example:2 with random perturbations

Table 8. Deformed Shape

Example:2 without random perturbations	Example:2 with random perturbations

Table 9. Force-delta

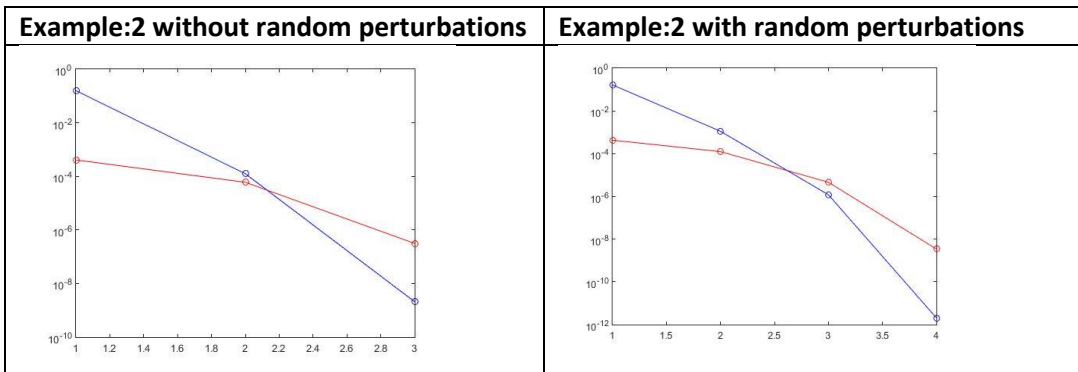| Example:2 without random perturbations	Example:2 with random perturbations

Table 10. Stability

In Example 3 we observed different results between the two cases.
With random perturbation we can observed that the solution is not stable. A deeper analysis could be done to search for buckling modes.
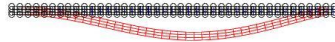
Example:3 without random perturbations	Example:3 with random perturbations

Table 11. Deformed Shape

Example:3 without random perturbations	Example:3 with random perturbations

Table 12.Force-delta

Example:3 without random perturbations	Example:3 with random perturbations

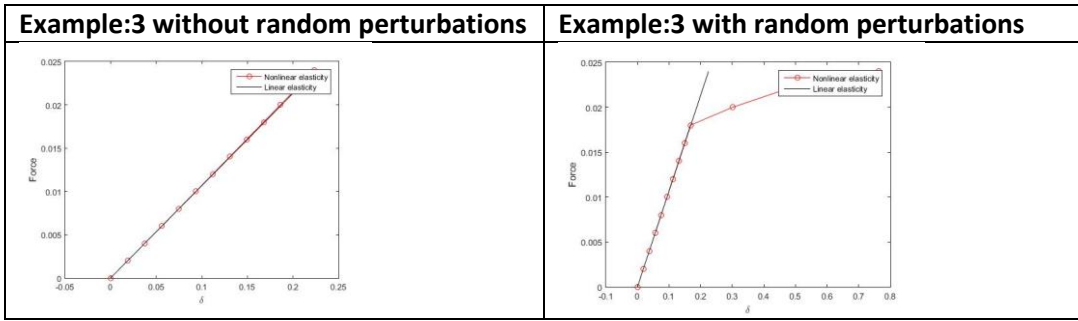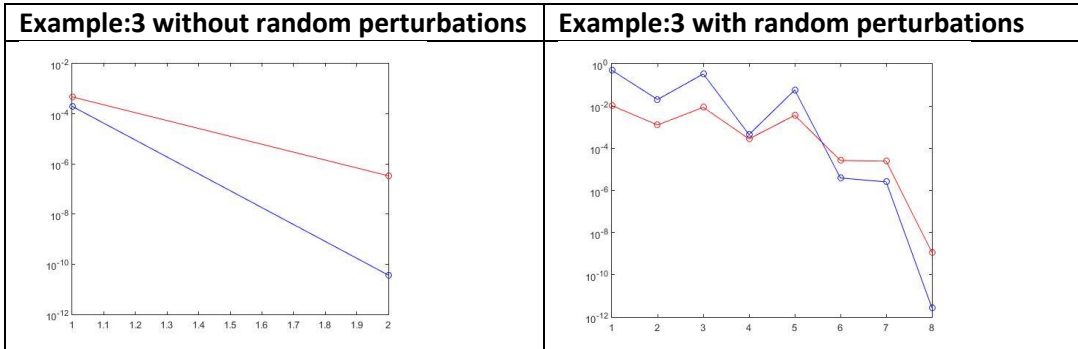*Table 13. Stability*

(d) Examples 4 and 5 with and without line-search, without random perturbations.

Example 4: arch, dead load at center of the arch

Example 5: arch, dead load near the supports

For example 4 and 5 no solution was found without line-search, so it wasn't the appropriate method to solve those problems.

Similar kind of results were obtained either for example 4 and 5 with line-search. We can see the different solutions and also the a really different behaviour compared to the Linear elastic calculation.
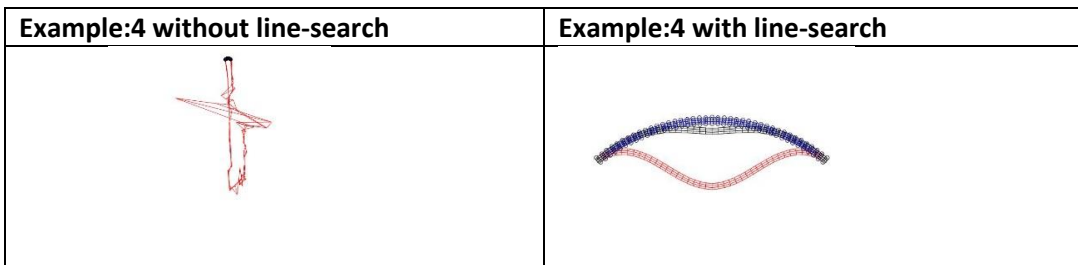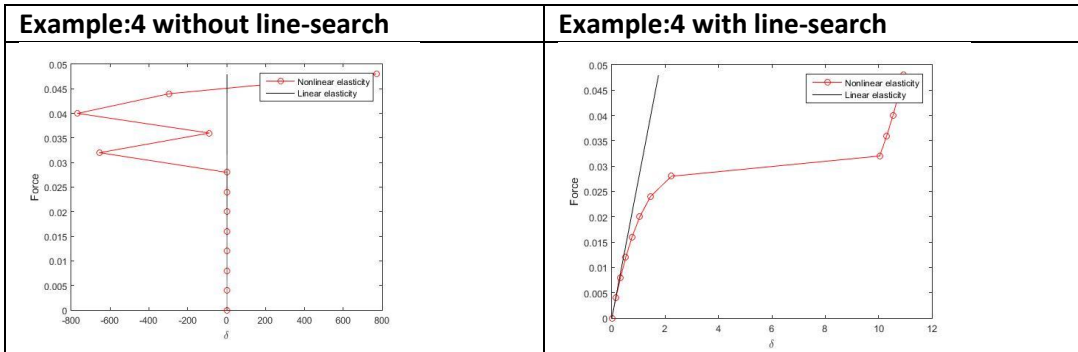
The solutions are stable.

Example:4 without line-search	Example:4 with line-search

Table 14. Deformed-shape

Example:4 without line-search	Example:4 with line-search

Table 15. Force-delta

Example:4 without line-search	Example:4 with line-search

Table 16. Stability

Example:5 without line-search	Example:5 with line-search

Table 17.Deformed-shape

Example:5 without line-search	Example:5 with line-search

Table 18. Force-delta

Example:5 without line-search	Example:5 with line-search
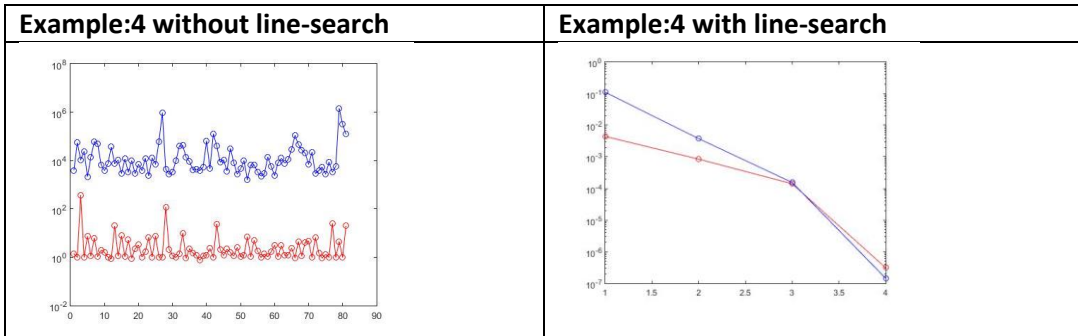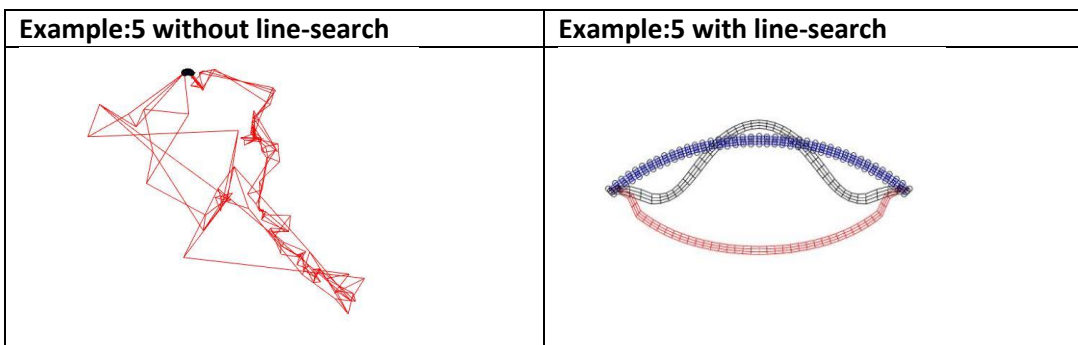
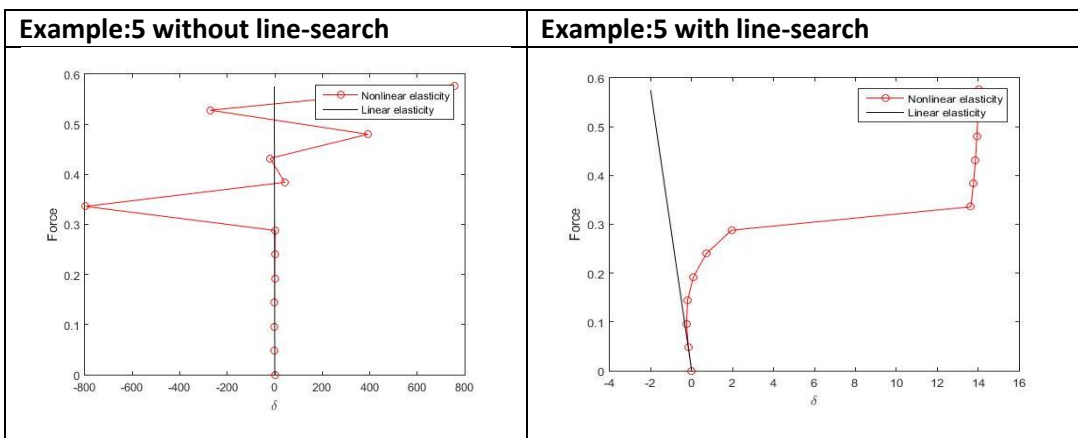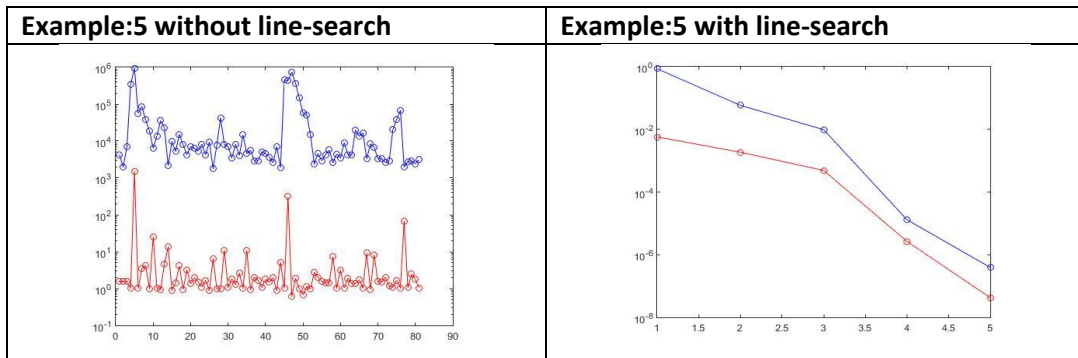*Table 19. Stability*

3. Conclusions

To sum up, with the first two examples (example 0 and example 1) we observed how the response were proportionally to the load and symmetric due to the load. The Newton-Raphson method used without line search was good to get the solution of the problem.

With example 2 and 3 the effects of the random perturbation to the solution has been observed. For example 3 some instability of the solution had been obtained.

In the computation of example 4 and example 5 the line-search method has been studied. We had graphic how useful was to use it to get a solution other way we weren't able to get with the Newton-Raphson method.