

Automated static structural FEA environment in MATLAB using Solid Edge and Femap

A. SZÁNTÓ

University of Debrecen, Faculty of Engineering, andras.szanto.0503@gmail.com

Abstract. Nowadays, CAD and FE software are getting integrated to each other. However, these new integrated pieces of software cannot be customized enough to analyze different optimization methods or model behavior. In this paper, a MATLAB environment is developed that connects to C# and VB programs, which use Solid Edge and Femap API functions, in order to automate the geometry modification and the FE model building and solving functions. A simple static structural size optimization problem is solved with the fmincon solver of MATLAB.

Introduction

Nowadays, the software dedicated to only CAD or FEA are getting more versatile with additional modules. For example, Solid Edge has FEA and optimization modules, while Ansys has a built-in parametric CAD software. However, these built-in environments are not opened enough to test new methods regarding to optimization or model analysis. Not to mention the fact that in a numerical computing environment, like MATLAB, many tools are available to perform optimizations and other algorithms. Contrary to this, in the built-in software environments one usually cannot even use programming languages.

In this paper, C# and VB programs are created which use Solid Edge and Femap API functions. These programs communicate with MATLAB via TCP. In MATLAB, several classes are created to simplify the environment. A simple size optimization problem is solved to verify the environment.

The method of generating new geometry and re-meshing it for the FEA is one approach. There are other approaches [1,2] which modify directly the mesh to obtain the geometry change. Analysis and comparison of these methods are not presented in this paper. This paper contains a size optimization example to verify the FEA and optimization MATLAB environment. Other shape and size optimizations are presented in these papers [3,4].

1. Overview

A brief overview of the workflow of the FEA with geometry modification is the following.

1. A Solid Edge geometry is needed. The API has functions capable of creating new geometric features. However, it is much easier to design the geometry in the Solid Edge GUI and the API is

only used for the modification of parameters that are in the Variable Table. The geometry modification API function is in a C# program and called from MATLAB with a class function.

2. The modified geometry is exported in a specified format (.stp in this case) to a specified path. With this method, the whole geometry needs to be re-meshed. There are other methods [1,2] that directly modify the mesh, without changing the CAD model.
3. The Femap model of API functions corresponding to the FE model of the geometry is written in a VB program. This function is called from MATLAB with a class function with an array of inputs for the Femap model.
4. Once the FEA based on the modified geometry is finished, a result file is saved in binary format to a specified path. File writing is also in the VB function. Several results are saved corresponding to elements and nodes. For example, stresses, positions, nodes of elements, and the IDs of the nodes and elements so that the identification of nodes and elements in the Femap GUI is not a problem.
5. Results are stored in a MATLAB class. The objective and constraint functions that are needed for the optimization are defined in this class. In this case, the objective is the mass of the geometry and the constraint is the maximum stress. These functions are defined so that they are compatible with fmincon.

MATLAB also offers tools for FEA called PDE Toolbox. However, I preferred to use the Femap API because the PDE Toolbox does not have functions for precise mesh control.

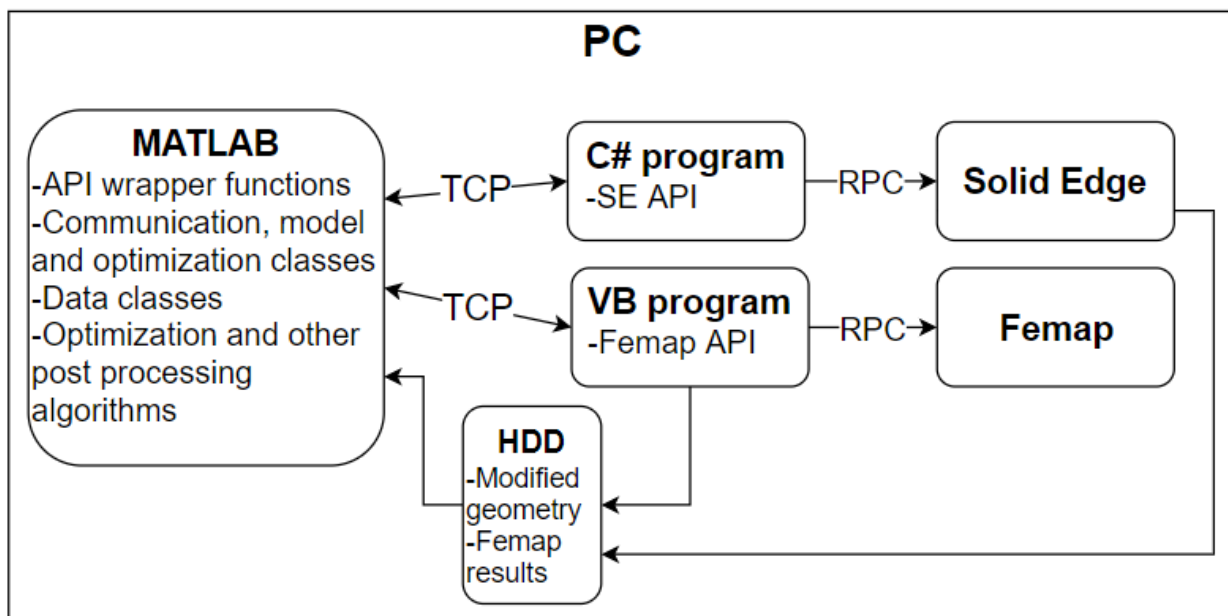


Figure 1: The sketch of the communication between the software/programs

2. C# and VB programs with API functions

The main parts of these programs are the TcpListener classes. TCP is a communication protocol that can be used for – apart from other applications – two-way communication between different programs

on a PC. The different messages are identified by a message ID in this program (a number created from the first 8 bytes of the message). With this method, any API function or model of API functions with any input can be called from MATLAB. The MATLAB functions can be thought as wrapper functions in a way.

3. MATLAB classes

Several MATLAB classes are created to simplify the FEA environment in MATLAB.

- SE_Femap class: it is used for the communication with the C# and VB programs. It runs different Solid Edge API functions and Femap model of API functions and reads the results from the binary file.
- ModelManager class: it stores the Solid Edge and Femap model properties for several models. Every model has a Solve, Objective and Constraint function in this class which can be customized.
- Opt class: the automation of the optimization and data save is realized in this class.
- Data classes: the FEA results for one model, the optimization results and the 1-2D analysis results are also stored in classes with practical functions for the objectives, constraints and other functions.

4. Size optimization example with MATLAB's fmincon

Detailed analysis of different optimization settings and methods are not presented in this paper. This section contains a solution for one size optimization problem with 2 variables. The geometry of the model with *var1* and *var2* variables can be seen in *Figure 2*. In the FE model (*Figure 3*), there is an X-Y force on the circular face of the cylinder. On the opposite side (face of the cross shape) there is a fixed boundary. The fmincon solver settings are in *Table 1*. The maximum stress as a constraint is 100MPa. The results of the optimization are in *Figure 4* and *Figure 5*. The output message of the optimization is the following: '*Local minimum possible. Constraints satisfied*'. Further analysis is needed to verify the optimization results.

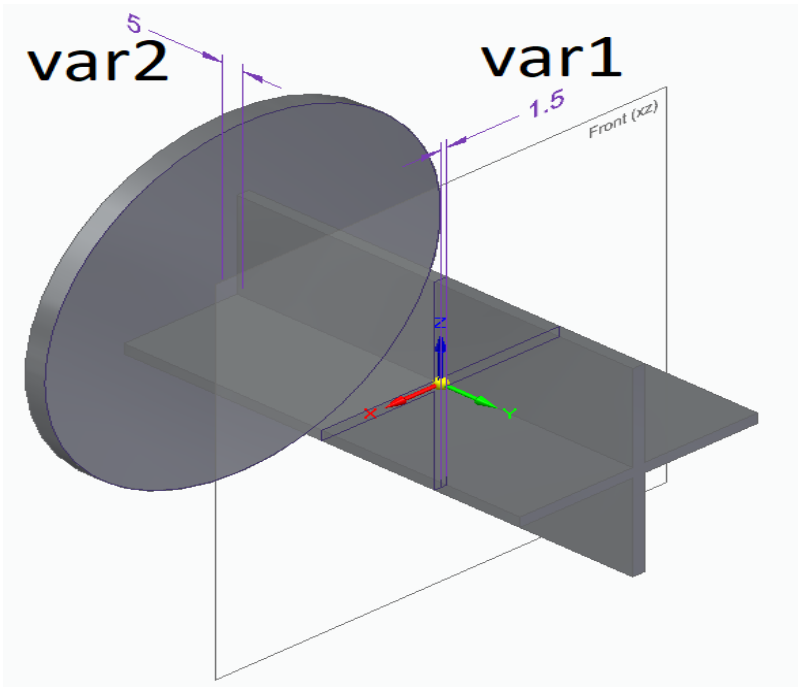


Figure 2: The geometry in Solid Edge with the variables [mm]

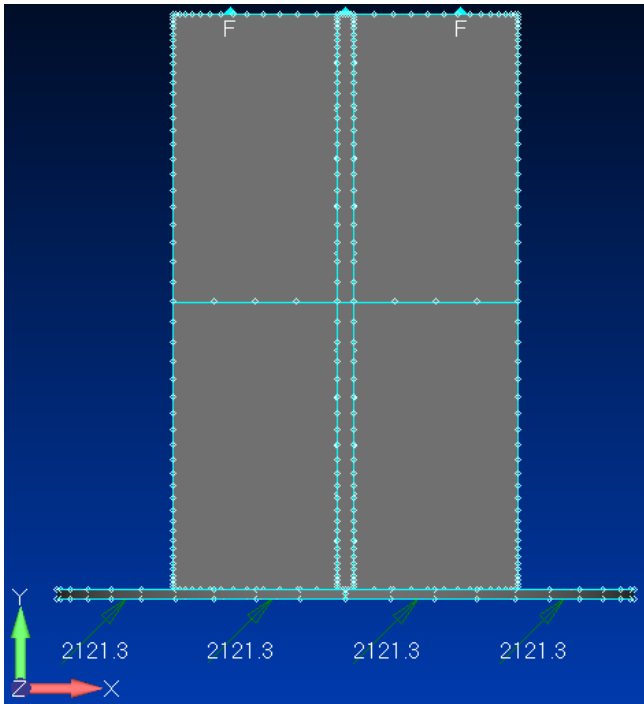


Figure 3: Load and boundary condition for the FE model in Femap

Solver	Algorithm	Constraint Tolerance	Step Tolerance	Finite Difference Step Size	Lower bound X	Upper bound X	Start X0	Optimal X
fmincon	SQP	5e-2	5e-2	1e-1	[1.5,1.5]	[5,10]	[1.5,1.5]	[2.72,4.68]

Table 1: Optimization settings and solution

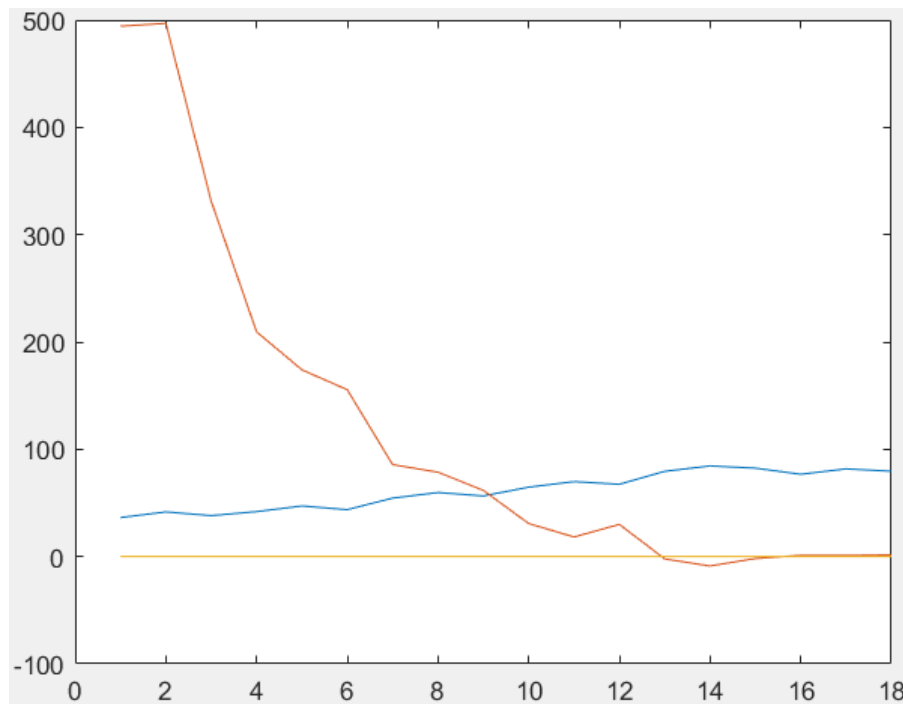


Figure 4: Objective [kg/100] (blue) and constraint [MPa] (red) versus the function count

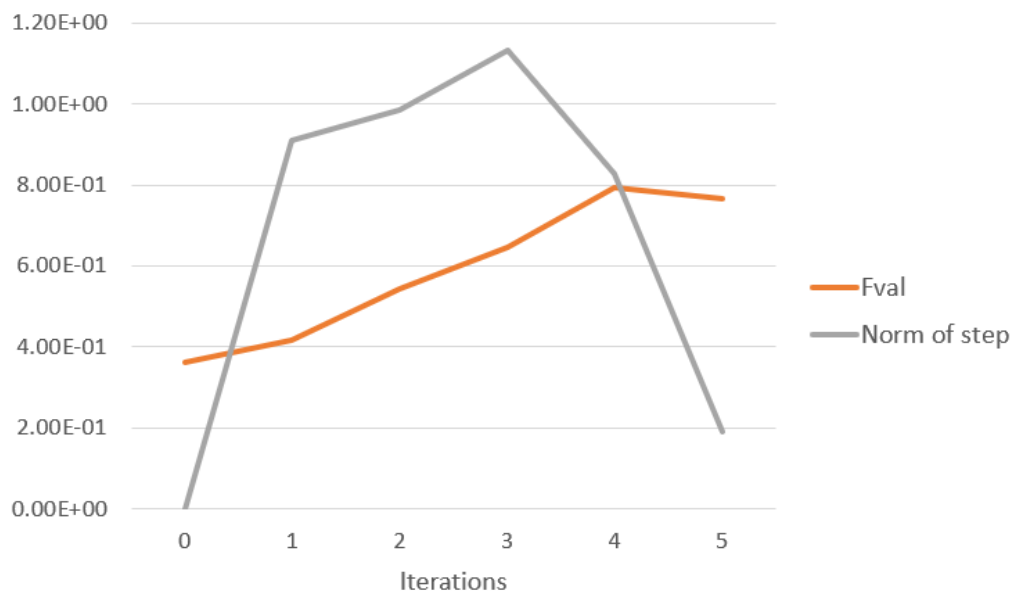


Figure 5: Fval [kg] and Norm of step [mm] versus the iterations

5. Conclusion

The motivation for starting to develop a MATLAB environment for FEA is described in the Introduction. The framework concepts for this environment are discussed in details in the main sections. In the latter part of the paper, an optimization example is presented. The created MATLAB environment for FEA works properly, MATLAB optimization algorithms can be used to solve FE optimization problems. Further analysis is needed regarding the optimization settings, methods and the FE model.

References

- [1] Lou, R. – Giannini, F. – Falcidieno, B. – Pernot, J.P. – Véron, P. – Mikchevitch, A. – el Marc, R., (2010) *Direct Modification of Finite Element Meshes Preserving Group Information*. International Journal of Shape Modeling, 16(1-2), pp.81-108.
- [2] Bonito, A. – Nochetto, R.H. – Pauletti, M.S. (2010) *Geometrically consistent mesh modification*. SIAM Journal on Numerical Analysis, 48(5), pp.1877-1899.
- [3] Morin, P. – Nochetto, R.H. – Pauletti, M.S. – Verani, M. (2010) *Adaptive sqp method for shape optimization*. In Numerical Mathematics and Advanced Applications 2009 (pp. 663-673). Springer, Berlin, Heidelberg.
- [4] Kaveh, A. – Khayatad, M. (2013) *Ray optimization for size and shape optimization of truss structures*. Computers & Structures, 117, pp.82-94.