

The Study of Pinion – Rack Mechanism Using Matlab and Spyder-Python

Ciprian Ion Rizescu^(✉) and Dana Rizescu

Department of Mechatronics and Precision Engineering,
University POLITEHNICA of Bucharest, Splaiul Independenței nr. 313, sector 1,
060042 Bucharest, Romania
{ciprian.rizescu, dana.rizescu}@upb.ro

Abstract. The study presents a comparison between a graphic simulation of a pinion-rack mechanism operating developed both in Matlab and PYTHON environments. Also some remarks, concerning the software's efficiency and an economic analysis efficiency-cost of the two software, are presented in this study which have an identical use/purpose.

Mainly, the idea of materialization of such a work has come out of necessity of programs usage that imply low purchase costs. From the point the view of acquisition, SPYDER-PYTHON is a open source e.g. free software. Spyder is the Scientific PYthon Development Environment. Both programs, MATLAB and PYTHON, can achieve certain accounts, can employ matrix and implement utility programs the difference between them are the used functions, because some of them don't match. MATLAB is a broader program from the point of view of functions against PYTHON, but from the point of view of cost software PYTHON is the advantageous solution.

Using PYTHON software in achieving performance and make an animation of pinion rack mechanism, it was noticed that the only differences between this program and MATLAB are certain functions with different syntaxes. In the present study it stands out in particular that both software are similar in terms of programming language and interface.

Each environment has advantages and disadvantages and their use is identical in terms of performances compared to results. Considering that PYTHON environment uses a particular programming language MATLAB is much used in industry and academia worldwide.

The study of present paper can be enhanced with new researches concerning the description and functionality of the programs as well as the presentation and improvement of the toolbox sites belonging PYTHON program, in order to achieve more complex animation in terms of the movement mechanisms.

Keywords: PYTHON · SPYDER · ANACONDA · MATLAB · Pinion-rack mechanism · Simulation

1 Introduction

This paper deals with a mechanism simulation: pinion rack mechanism using both PYTHON and MATLAB programs. Also, in the frame of the paper, there will be highlighted the main advantages and disadvantages of using PYTHON with respect to

MATLAB. In principle, the idea of such work arise from the need to use some programs involving the minimum cost of purchase. From this point of view, PYTHON is a free software. Both of these programs, MATLAB and PYTHON, can perform certain calculations, may use the matrix and deploy algorithms, the difference between them being the functions used because some of them do not match. MATLAB is a much more comprehensive from the point of view of the functions in relation to the PYTHON, but from the point of view of the cost price, the PYTHON solution is advantageous.

Using the PYTHON to achieve performance and animate slider crank mechanism, it was noted that the only differences between this program and MATLAB, there are certain functions with different syntax checker.

2 MATLAB Programming

The MATLAB programming is a graphical representation, numerical calculation, graphic differentiation or data read on the graphs based on certain applications encountered in practice. It should be stressed that the numerous software currently existing in the market, the most appropriate for the analysis of the static and dynamic mechatronics systems is the program MATLAB through the spread of its pronounced in both the academic environment as well as in the industrial uses.

The MATLAB programming has several commands for the numerical analysis, for the purpose of calculating the matrix vector and, for generating and processing the signals for the analysis of the control systems and for the integration of the numerical systems linear or nonlinear media. Must be and effective controls for generating the performance of the two-dimensional and three-dimensional, what can easily be customized. Writing models calls the syntax checker and notations very similar to those used to define the analytical models, which makes that the programming itself to be quite easy. In the working area of the MATLAB (Workspace) can be executed all controls, can be displayed index, results [1].

In Fig. 1 is presented a graphic user interface dedicated to design a XY table with high positioning accuracy.

To generate the graphical interfaces are used either of the six items of control for interactive graphical interface used by MATLAB when programming with objects for the control of interactive graphics. The six predefined elements of interactive control are [1]:

1. push button;
2. radio button;
3. sliders;
4. popup menu;
5. editable text;
6. check box.

As it is well know for the creation and control of the figures there are used the following MATLAB functions [1]:

- figure - creates a figure, leaving untouched previous figures;
- gcf - returns the identifier for the control of the characteristics of the current figure;
- clf - deletes the current figure;
- close - close the current figure.



Fig. 1. Matlab main graphic user interface - Romanian

The graphic user interface has two push buttons in order to select the language (marked with a flag). If in Fig. 1 the interface is presented in romanian, in Fig. 2 the interface is in English.

In Fig. 3 there was presented the computational window for basic parameters defined for pinion rack mechanism. This windows appears after selecting 'Calculation of pinion rack mechanism' from pop-up menu shown in Fig. 2. At first step, the operator should fill in the design parameters of the mechanism, in the left side of the window, such as: modulus [mm], pressure angle α [°], number of teeth, specific displacement X [mm], height of rack pitch line H [mm]. Then pressing 'Calculate' push button the results will be displayed in the right side of the window [2].

3 Python Programming

Python is a language that offers many features and has a fast learning curve for both programers who know languages such as C and Java and for beginners. Although it is an interpreted language, this is not a member in the path of its popularity and use in numerous projects.



Fig. 2. Matlab main graphic user interface - English

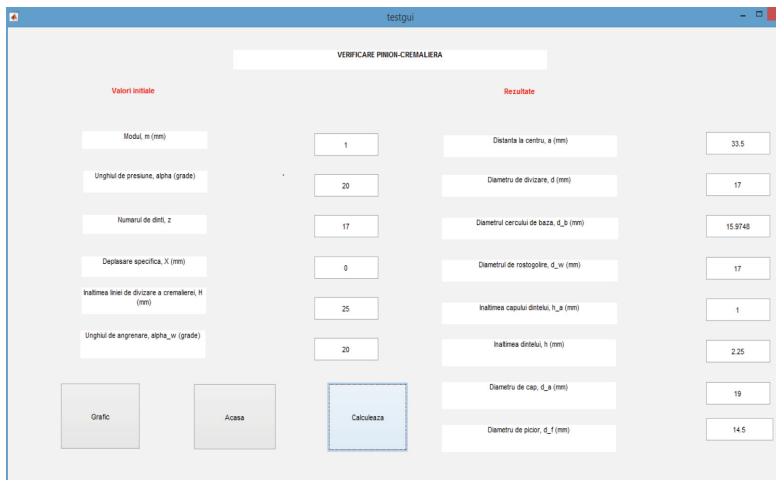


Fig. 3. Matlab graphic user interface – computing pinion – rack mechanism

The application is developed using the language Python 2.7 to which shall be added the libraries:

- NumPy - matrix calculation;
- PyQt4 - graphical user interface;
- Matplotlib – graphics.

From the point of view of the compatibility with the various operating systems, stability and the functionality of the application, are given by the status and subduced of the language for programming in the environment. There have been carried out runs on Windows 7 and Linux Lying 17.3 (Debian). Structure of the application can be divided into three components:

- * Kernel Code - *main.py*;
- * Component *Animation* - *pc_animation.py*;
- * Files auxiliary images.

The file, '*main.py*', represents the central part of the application of the corresponding classes GUI, formulas for the calculation of parameters specific to the tool and calling the code of animation, respectively of the auxiliary files. Whether you are satisfied dependencies, by running the file *main.py* (double-click on the file or executing in the terminal/cmd command `python main.py`) appears window shown in Fig. 4.



Fig. 4. Python main graphic user interface - English

The general aspects of the two programming environments reflect the fact that although they look similar at a first consideration, they differ very much by syntaxes that are used to create the programs and the most representative aspect is by the fact that many functions of the MATLAB have no correspondence in Python, even highlighted in the two programs used to the creation of a mechanism with pinion - rack mechanism, the environment of MATLAB programming being more complex in performing the functions, more powerful, but with certain disadvantage of the cost.

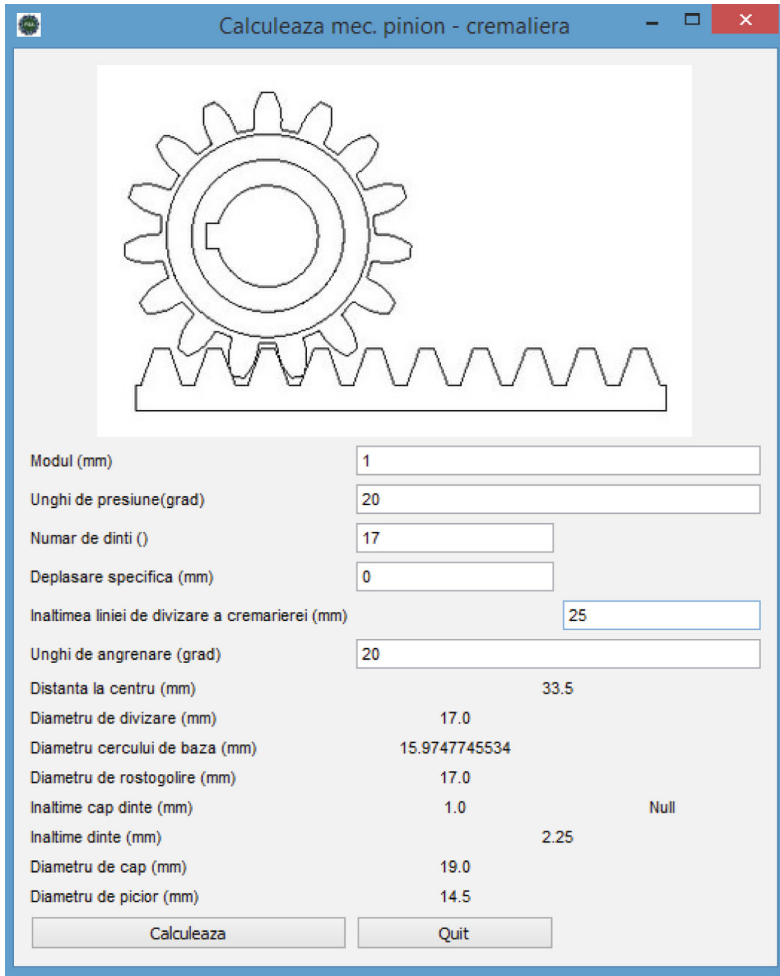


Fig. 5. Python graphic user interface – computing pinion – rack mechanism

In Fig. 5 there is presented the computational window for pinion – rack mechanism. The input arguments are the same as in Matlab application. The results are displayed in the lower part of the window as one can see.

The graphic user interfaces could display 2D and 3D representation of the mechanism using dedicated push buttons. The results are presented in Fig. 6 for Matlab and Python programs.

Finally there were developed simulations of mechanism operating for both environments: Matlab and Python. In Fig. 7 are shown the simulation of operating pinion – rack mechanism developed in Python (on the left side) and also the simulation developed in Matlab (on the right side).

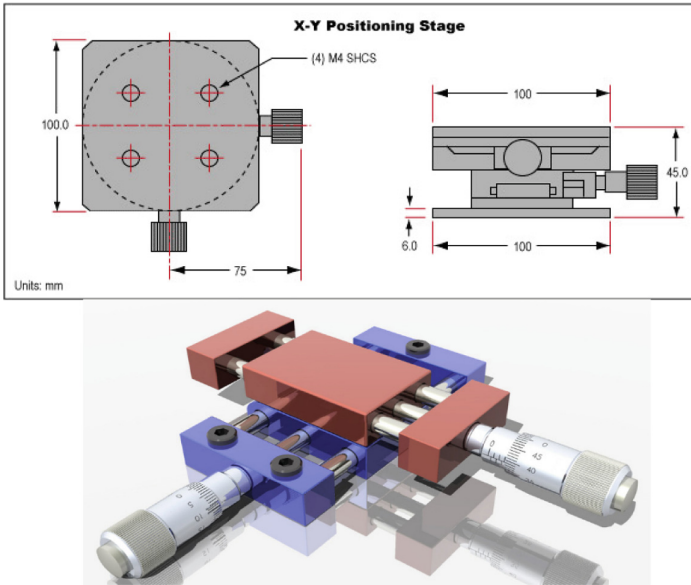


Fig. 6. Graphic user interface - 2D representation and 3D representation

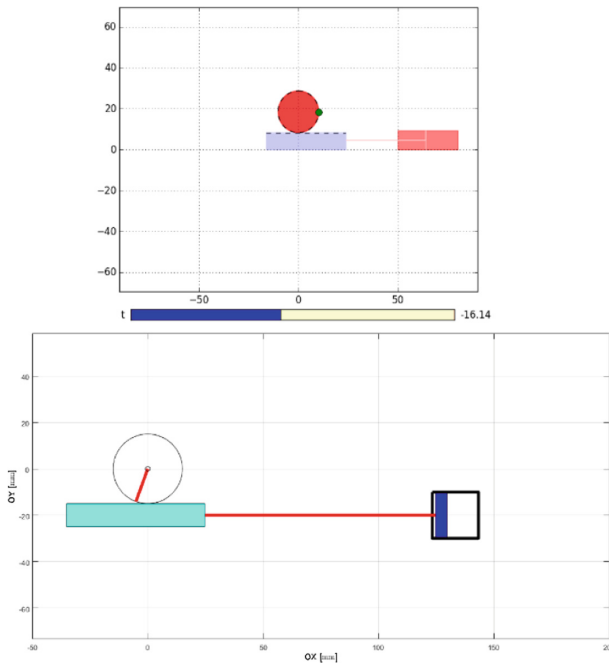


Fig. 7. Python simulation (left) versus Matlab simulation (right)

4 Conclusions

The work highlights the importance of using PYTHON program from the point of view of the completion of the animations, as well as the pinion rack mechanisms or other type of animations and opens up new directions regarding the execution of the GUI interfaces or other applications.

The similarities between the two environments, MATLAB and PYTHON, are given in the writing of functions, but also of the results of the graphics. The two software does not have as a result important differences, which shows us that both programs can easily perform applications relatively simple, [3].

An important aspect to economy mode of approach the two software being at different ends when we discuss about the purchase price of the MATLAB programming having a high cost and the environment of programming PYTHON having a zero cost being an open source, [4–7].

Each environment presents the advantages and disadvantages, and their use is the performance of even the same, as well as the comparison of the results but with a certain threshold of the complexity of the programs, complexity is given no more and no less than the PYTHON environment. Therefore, MATLAB environment is much more used in the fields of industrial and academic ones, at the global level, [8].

The study of the work can be enhanced with new research relating to the description and functionality of the program as well as the presentation and improve the toolbox sites that belong to the PYTHON program, in order to achieve animations, most complex from the point of view of the machinery movement.

References

1. Rizescu, D., Miu, S., Rizescu, C.I.: *Complemente de mecatronică*. Printech, București (2000)
2. Prisăcaru, G.H., Bercea, M., Gramescu, B., Ciupe, V.: *Mecatronica aplicata*, Ed. Academiei Oamenilor de Stiinta din Romania, București (2011)
3. Rizescu, C.: *Software pentru sisteme mecatronice*. Printech, București (2013)
4. Scott Sievert. <http://stsievert.com/blog/2015/09/01/matlab-to-python/>
5. Reference for PyQt4. <http://zetcode.com/gui/pyqt4/>
6. Reference for Matplotlib. <http://matplotlib.org/contents.html>
7. Kevin, S.: Introduction to Python for econometrics, statistics and data analysis. https://www.kevinsheppard.com/images/0/09/Python_introduction.pdf
8. Charalampides, G., Dontu, O., Gheorghe, G.H.I., Besnea, D., Avarvarei, I., Ciobanu, R.: Considerations regarding the integration of CAD/CAM design and manufacturing using CNC processing machines. *Rom. Rev. Precis. Mech. Opt. Mechatron.* (38) (2010)