

# Matlab GUI Application for Teaching Control Systems

ANTONIOS S. ANDREATOS and ANASTASIOS D. ZAGORIANOS

Department of Aeronautical Engineering

Hellenic Air Force Academy

Dekeleia, Attica, TGA-1010

GREECE

[aandreatos@gmail.com](mailto:aandreatos@gmail.com), [aandreatos@hafa.gr](mailto:aandreatos@hafa.gr), [azagorianos@gmail.com](mailto:azagorianos@gmail.com)

*Abstract:* - This paper presents the development and use of a user friendly GUI tool for teaching Automatic Control Systems, based on Matlab. The ACTA-II tool –as we have called it– is going to be used in automatic control systems course, offered to the students of the Hellenic Air Force Academy of Greece. ACTA-II tool facilitates the design of Automatic Control Systems. In order to demonstrate its functionality we present the solution of a typical control system of an aircraft, step by step.

*Key-Words:* - Control systems, Aircraft control, Poles and zeroes, root loci, graphical user interface, Matlab.

## 1 Introduction

An Automatic Control Systems course is offered to the third-year cadet students of the Hellenic Air Force Academy of Greece, Dept. of Aeronautical Engineering. The above course runs through both semesters of the particular year. The average student population ranges from 25 to 30 students.

At the beginning of the course, students have a background knowledge of basic mathematics, physics, mechanical, electrical and aeronautical engineering, electronics and programming. The introductory control systems course covers the following topics:

- Input-output and state space analysis of single-input single-output (SISO) control systems;
- Quantitative and qualitative analysis of MIMO control systems;
- The Root-Locus and frequency domain control design techniques;
- State space design method;
- Optimal control; and Adaptive control.

This course focuses on teaching and analysing dynamic stability and control of unstable multivariable dynamic systems; in particular, emphasis is given on the systems of modern aircrafts and aerospace structures. These engineering systems demonstrate high structural instability. One of the most significant design objectives is to develop a flight controller for achieving improved system stability and automatic control. The control systems course has the following *objectives*:

- To teach cadets the fundamentals of automatic control systems design and analysis, as well as the design methods of computer-aided control;

- To support learning by computer-aided control design simulation examples;
- To minimize the gap between control theory and practice, by teaching control implementation;
- To prepare cadets for the “Aircraft Feedback Control Systems” course; this course is offered during the fourth year of studies at the aforementioned department. Furthermore, to facilitate cadets to computer-aided design and testing of automatic flight control systems of modern aircrafts, used by the Hellenic Air Force.

Today, during the process of teaching the fundamentals principles of control, the emphasis is not given on tedious calculations, but rather on offering engineering education, focused on the design of real systems, by utilising efficient software tools. Computer-aided control design and control implementation are appropriate tools, because they improve the efficiency of learning the fundamental principles of control and the efficiency of industrial control practices. The performance, stability and control characteristics analysis of dynamic systems can be done on the basis of appropriate Matlab and Simulink models.

This paper presents a comprehensive and user-friendly tool called the Automatic Control\_Systems Teaching Assistant no.2 (abbreviated herein as **ACTA-II**), which performs complex Automatic Control Systems calculations and graphs in a user-friendly fashion.

## 2 Matlab in Engineering / Science Education

Matlab is one of the most popular platforms for scientific and engineering computing. This is due to the inherently implemented plethora of tools for

most engineering/ scientific fields, and to its high computing level implementation which offers the advantage of a sharp learning curve to a beginner. Therefore it is being used around the globe in teaching engineering and science courses such as electronics [Attia, 1999; Dastfan 2007; Andreatos & Michalareas, 2008], simulation, signal and image processing, circuit theory, filter design [Andreatos & Michalareas, 2008], control systems [Uran & Jezernik, 2008; Chatzikos, 2003; Biran & Breiner, 1999; Cheever & Li, 1998], communication theory, electrical machines [Popescu et al. 2006] etc. It has also been proposed as an e-assessment tool [Andreatos & Michalareas, 2008; Andreatos & Michalareas, 2008b].

Since Matlab has already inherently implemented powerful functions and toolboxes for most scientific fields, it is a strong candidate for programming educational applications in a corresponding wide range of subjects. In order to analyse the feasibility of Matlab for such applications we have examined the following issues:

- Computational power and plethora of commands for scientific and engineering computing.
- Powerful and easy-to-use plotting features.

*Graphical User Interface capabilities of Matlab.* Matlab has a Graphical User Interface designer “Guide” [Smith, 2006] through which is relatively easy to get familiar with and requires little knowledge of programming. The addition of elements such as plots, buttons and sliders is straightforward. Matlab provides an easy way to build GUIs with elements such as check boxes, radio buttons, list boxes, pushbuttons etc. [Smith, 2006], and link them to build-in Matlab functions.

Moreover, in [Andreatos & Michalareas, 2008] another important issue is examined: *Licensing regarding multi-user programs.* When a Matlab application is designed as a Matlab program (*M-file*) meaning that it requires Matlab to be installed in order to run, apart from the license required for the application developer, a license is required for each different user that is going to run the application. In the case of educational applications involving the use of the program by tens or hundreds of students, the number of licenses required becomes a drawback as far as the use of Matlab is concerned. In [Andreatos & Michalareas, 2008], there has been presented a way to overcome this problem, which is to convert applications to Matlab executables using the *mcc* build-in compiler. No extra licenses are required for the use of these executables by additional users.

### 3 Description of the ACTA-II tool

The purpose of the ACTA-II tool is to display the classic control theory system stability criteria graphically, quickly, easily and accurately. This is done by exploiting the powerful computational and graphical functions of Matlab. It is constructed using the embedded “GUIDE” Matlab tool for building GUIs.

Figure 1 shows the start-up window; the user is prompted to enter the numerator and denominator polynomial coefficients in the appropriate boxes. Then he/she can select a pull-down menu (Fig. 1). The name of each item gives information on the actions carried out. There options are available: Bode diagrams; Nyquist diagram; Nichols chart; root-locus diagram and root-locus with a coordinate finder tool; the latter gives information not only on the coordinates of a point in the root-locus but also on the amplification, in order for the user to find the maximum value.

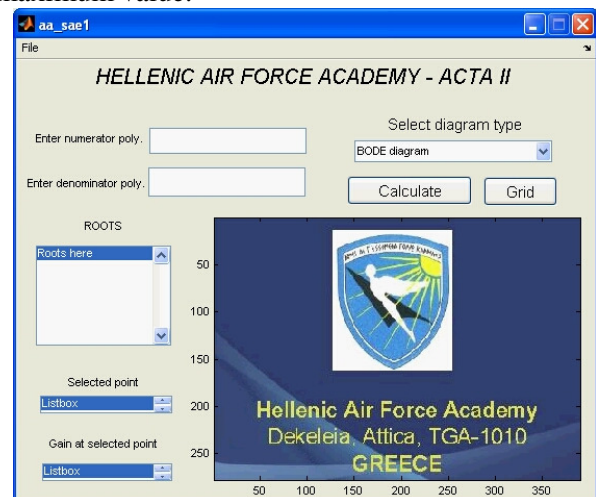


Fig. 1 Introductory screen of the ACTA-II tool

### 4 Use of the ACTA-II tool

The ACTA-II tool is intended to be used by both instructors and students, in a different way by each user group. Let us present the use by each group.

#### 4.1. Use by instructors

The ACTA-II tool is very useful to instructors when demonstrating the various stability criteria in class; it is extremely useful when we want to experiment on specific coefficients of the system and get results fast. Also, it can be used for designing and fine-tuning problems for assignments or exams.

#### 4.2. Use by students

The ACTA-II tool is also very useful to students for solving assignments and designing optimal control systems.

#### 4.3. Example

Let us now demonstrate the use of ACTA-II tool through a step by step example. Consider a rigid

supersonic aircraft, linear dynamics of which are described by a simple single input single output transfer function (see below), where K is a gain factor depending on airplane structure and flight parameters. The satisfactory airplane dynamic stability and trajectory control is achieved by a single loop feedback control system, which provides the desired airplane robustness and performance properties across a large range of operating conditions. The purpose of this problem is to evaluate the proper gain value to provide the desired airplane performance. The blocks of the forward path are:

$$\frac{1}{s(s+1)} \cdot \frac{1}{s+2} \cdot \frac{K(s+3)}{s^2+2s+2} \quad (1)$$

Using Matlab we can find that the equivalent open-loop transfer function is:

$$SYS_{O-L} = \frac{s+3}{s^5+5s^4+10s^3+10s^2+4s} \quad (2)$$

The numerator and denominator polynomials in Matlab format are: [1 3] and [1 5 10 10 4 0] respectively.

The user has to enter the polynomials in the appropriate GUI textboxes, without the brackets and with the terms separated by commas. Then he/she has to select the criterion from the pop-down menu and press the ‘Calculate’ button; the tool automatically calculates and displays the appropriate diagram in the bottom-right window, as well as the roots in the middle left window (see Figure 2). The ‘Grid’ button toggles the grid on the diagram.

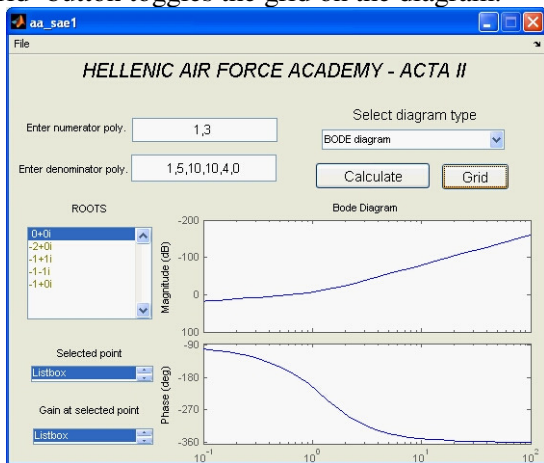


Fig. 2. Bode diagrams & roots calculated by the ACTA-II tool

Typical questions in such problems are: first to plot the root-locus diagram and second to find the proper values of system gain K so that the desired dynamic stability of the system (typically an aircraft) is achieved [Zagorianos, 2000; Paraskevopoulos, 1993]. Using the ACTA-II tool we select the option

“root locus” and we get the diagram shown in the following Figure 3.

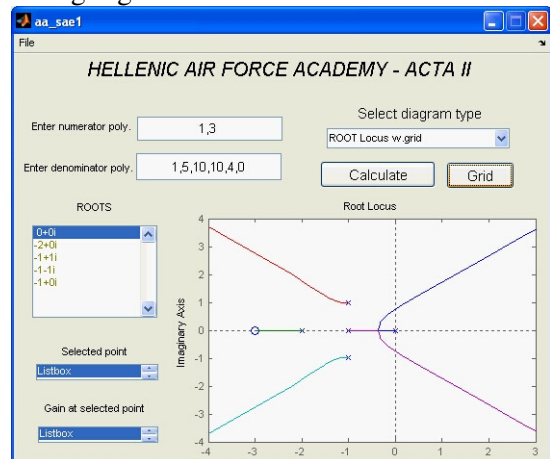


Fig. 3. Root locus diagram

Then we select the option “root-locus with a coordinate finder tool” and the coordinate finder tool appears on the diagram. This tool evaluates the gain K at a selected point by the mouse and displays the point coordinates as well as the gain in appropriate boxes located on the bottom left of the GUI (Fig. 4).

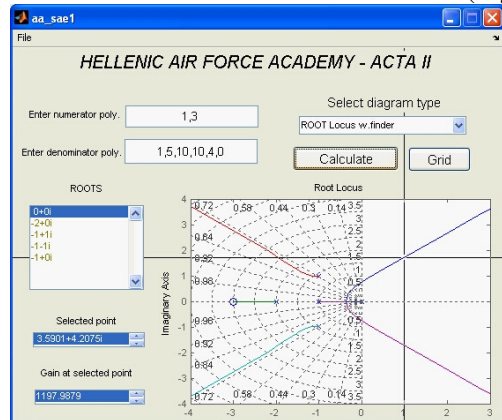


Fig. 4. Root locus diagram with finder tool

By moving the cursor over the desired area (which in our case is located on the left of the y-axis) and clicking on the optimal area (denoted by a circle in Fig. 5) we get the corresponding value of K as well as the step response for that value of K (see Fig. 6).

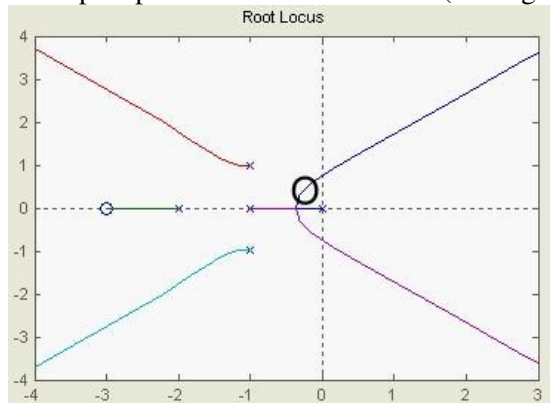
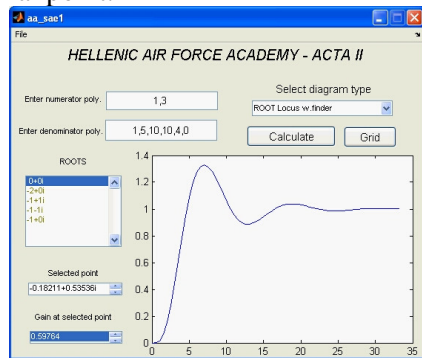


Fig. 5. Optimal area for system stability.

The step response represents system behaviour in abrupt manipulation (manoeuvres) during the flight. The ideal step response should have small overshoot, low natural frequency and relatively high dumping ratio, for the aircraft to retain stability and behave as smoothly as possible. Thus we need to try several values before we can end up with a good operational point.



**Fig. 6. Gain K and step response at selected point**

The ACTA-II tool provides also the open-loop and closed-loop transfer functions for the selected K (in the command window), as shown in Figures 7 and 8.

```
Transfer function:
      0.5976 s + 1.793
-----
s^5 + 5 s^4 + 10 s^3 + 10 s^2 + 4 s
```

**Fig. 7. Open-loop transfer function as produced by Matlab**

```
Transfer function:
      0.5976 s + 1.793
-----
s^5 + 5 s^4 + 10 s^3 + 10 s^2 + 4.598 s + 1.793
```

**Fig. 8. Closed-loop transfer function as produced by Matlab**

Hence, the ACTA-II tool greatly facilitates the flight control system design.

## 5 Conclusion

The tool presented here intends to demonstrate a GUI tool for teaching and learning automatic control systems using Matlab. ACTA-II is an appropriate tool for Engineering Education problems, where there is no unique solution but many good solutions. Moreover, it gives learners the possibility to experiment with different sets of parameters and stability criteria. The ACTA-II tool can be distributed to the students as an executable created by the Matlab compiler, if the students do not have the whole software installed.

This application shows that Matlab's powerful tools can give the instructor the ability to create educational tools not only for control systems but

also for many other majors, with moderate coding, using appropriate Matlab built-in functions [Andreatos & Michalareas, 2008; Andreatos & Michalareas, 2008b].

### References:

- [1] J. O. Attia, Teaching Electronics with Matlab, in *Proc. 26th Annual Conf. Frontiers in Education*, vol. 2, Nov. 1996, pp. 609-611.
- [2] A. Dastfan, Implementation and Assessment of Interactive Power Electronics Course, *WSEAS Trans. on Advances in Engineering Education*, issue 8, Volume 4, August 2007, pp. 166-171.
- [3] A. Andreatos and G. Michalareas, Engineering education e-assessment with Matlab; Case study in electronic design, in *Proc. of Int'l Conf. on Engineering Education '08*, Heraklion, Greece, July 2008.
- [4] A. Andreatos and G. Michalareas, Facilitating E-Assessment with Matlab, in *Proc. ED-MEDIA, World conference on educational multimedia, hypermedia & telecommunications*, Vienna, Austria, June 30- July 4, 2008 (b).
- [5] S. Uran & K. Jezernik, Virtual Laboratory for creative control design experiments, *IEEE Trans. on Education*, vol. 51(1), Feb. 2008, pp. 69-75.
- [6] E. Chatzikos, *Matlab 6 for Engineers*, Thessalonica, Greece: Tziola, 2003.
- [7] A. Biran & M. Breiner, *Matlab 5 for Eng-neers*, 2nd ed. Addison Wesley Longman, 1999.
- [8] E. Cheever and Y. Li, A Tool for Construction of Bode Diagrams from Piecewise Linear Asymptotic Approximations, *International Journal of Engineering Education*, 1998, <http://www.ijee.dit.ie/OnlinePapers/Interactive/Cheever/EACWebPaper/index.htm>.
- [9] M. Popescu, A. Bitoleanu & M. Dobriceanu, Matlab GUI application in energetic performances analysis of induction motor driving systems, *WSEAS Trans. on Advances in Engineering Education*, issue 5, vol.3, May 2006, pp. 304-311.
- [10] S. T. Smith, *Matlab Advanced GUI Development*. Indianapolis: Dog Ear publishing, 2006.
- [11] A. D. Zagorianos, *Aircraft Flight Dynamics and Automatic Flight Control*, Dekeleia, Athens, Greece: Hellenic Air Force Academy, 2000 (in Greek).
- [12] P. N. Paraskevopoulos, *Solved Problems of Automatic Control Systems with Applications*, Athens, Greece: National Technical University of Athens, 1993 (in Greek).