

Teaching Iterative, Incremental Problem Solving Methods Using MATLAB GUIDE

Ryan Patton and JosAnn Duane
Engineering Education Innovation Center
The Ohio State University
Columbus, Ohio, 43210
Email: duane.1@osu.edu

Introduction and Objective:

This poster paper is one of a group of papers detailing different aspects of course design¹. Our objective for this poster paper² is teaching iterative and incremental problem solving methods using the MATLAB Graphical User Interface Development Environment (GUIDE)³. All students in the course have the opportunity to use software tools developed in the class with GUIDE. Students entering the course with a substantial background in MATLAB programming can elect to take on the “technical challenge” assignment of developing software tools using GUIDE.

Our goal is to achieve the stated learning objectives of the course¹ by teaching students how to apply learned MATLAB skills in a new environment. Achieving this goal requires careful consideration of: design workflow, component interaction, dynamic programming, and end user needs.

Method:

There are three main stages in teaching students how to program with GUIDE. Each stage builds on the previous stage, enabling students increase in their proficiency a step at a time. The first stage introduces the GUIDE interface that the students will be using. This includes the different GUIDE components such as: push buttons, check boxes, text fields, axes, and others. Also introduced is the Inspector window, which provides an easy way to set component properties, such as size, text, and color.

The second stage builds on the student’s prior knowledge of MATLAB by introducing the Callback function. Figure 3 contains a sample Callback function used to create a plot of: $y = 1/x$. Other similar examples show students how every component has a Callback function, which can be custom written for different tasks. Understanding the Callback function requires students to recall previous topics such as function files, input/output parameters, variable scope, and others.

The third and last stage introduces a small but important topic, the handle structure. This is a system level data structure which allows graphical components to interact with each other by sharing information. A review of the set() and get() commands will allow students to use this data structure to dynamically change a component’s properties during runtime. This is where the GUIDE user interface is especially effective in interactive design and analysis. A student’s

program can now dynamically react to user inputs, and multiple graphics components can work together to achieve problem specific goals.

Results:

The results presented in figures 1 through 10 are placed in color coded frames in alignment with the learning objectives of the course as follows:

1. THINK (coral): *Demonstrate ability in critical, creative and practical thinking through algorithm design, MATLAB software design and evaluation.* **For example:** Understanding how GUIDE can facilitate simulation, optimization, and data analysis.

2. USE TOOLS (purple): *Utilize MATLAB software tools to solve engineering problems.* **For example:** Developing software tools for performing basic numerical calculations interactively and displaying the results graphically.

3. DESIGN (lime): *Demonstrate the ability to create and design within the constraints of time, cost, quality, safety, and environmental impact.* **For example:** Developing software tools using GUIDE that make possible evaluation of multiple designs alternatives quickly and easily; GUIDE also has capability to facilitate development of interactive budgets and schedule charts for optimization of projects constraints such as time, cost, quality, safety, and environmental impact.

4. CONDUCT (turquoise): *Work individually, in pairs, and on teams to solve engineering design and analysis problems professionally and ethically.* **For example:** Understanding concepts of intellectual property, responsibility and accountability when writing applications software.

5. COMMUNICATE (orange): *Demonstrate skill in technical communication related to engineering and software development.* **For example:** Communicating results of multiple alternatives based on calculations performed using software developed with GUIDE.

These learning objectives are explained in more detail by the poster paper on goal directed course design¹.

The figures on this poster correspond to different aspects of the three stages mentioned above. Figure 1 is an introduction to the GUIDE interface with several basic components displayed. Figures 2 through 4 continue the example from Figure 1. The Callback function is found through the GUIDE interface in Figure 2. The Callback function code which utilizes concepts from different homework assignments appears in Figure 3. Finally, Figure 4 shows the results of student written code during runtime.

Figures 5 and 6 apply to the third stage utilizing the handle structure. Figure 5 is a sample Gantt Chart produced by non-interactive programming. By using the system level data structure, user

input can be shared across multiple components to create an interactive program as shown in Figure 6.

Figures 7, 8, 9, and 10 show how the MATLAB GUIDE interface can be used for interactive, incremental design⁴, of a Gantt chart. The Gantt shown is for construction of a wind farm. During each successive step in the iterative design process, the schedule is revised as the pieces are put in place and moved around electronically, until the optimum duration and sequence of all tasks is achieved.

Conclusion:

Using GUIDE is an excellent way for students to expand on their current knowledge of MATLAB. The MATLAB GUIDE tool primarily challenges students to solve engineering problems by thinking critically, and using interactive programming. This meets the first course learning objective, *Demonstrate ability in critical, creative and practical thinking through algorithm design, MATLAB software design and evaluation.*

In addition, this poster paper demonstrates how GUIDE, supports all five of the stated course the learning objectives. Students taking on the team “Technical Challenge” assignment have the opportunity to learn how to use GUIDE to develop MATLAB software tools. Their teammates then use these software tools in completing the wind farm design-build term project. This meets the second, third and fourth learning objectives: *Utilize MATLAB software tools to solve engineering problems; Demonstrate the ability to create and design within the constraints of time, cost, quality, safety, and environmental impact; and, Work individually, in pairs, and on teams to solve engineering design and analysis problems professionally and ethically.* Finally, the MATLAB graphics functions implemented through GUIDE foster achievement of the fifth learning objective, *Demonstrate skill in technical communication related to engineering and software development.*

References:

1. **Duane, JosAnn, Liu, Ye and Maynell, Laurie.** Value Creation in Engineering Education through Goal Directed Course Design. EEIC ASEE North Central Section Conference, April 5 and 6, 2013, The Ohio State University, Columbus, OH. [Online] April 5, 2013. [Cited: April 5, 2013.] <http://eeic.osu.edu/courses-services/asee-papers>.
2. **Patton, Ryan and Duane, JosAnn.** Teaching Iterative, Incremental Problem Solving Methods Using MATLAB GUIDE. EEIC ASEE North Central Section Conference, April 5 and 6, 2013, The Ohio State University, Columbus, OH. [Online] April 5, 2013. [Cited: April 5, 2013.] <http://eeic.osu.edu/courses-services/asee-papers>.
3. **The Mathworks.** Creating a GUI with GUIDE. Videos. [Online] [Cited: February 2, 2013.] <http://www.mathworks.com/videos/creating-a-gui-with-guide-68979.html>.
4. **Larman, Craig and Basil, Victor.** Iterative and Incremental Development: A Brief History. Computer. [Online] June 2003. [Cited: February 2, 2013.] <https://www.it.uu.se/edu/course/homepage/acsd/vt08/SE1.pdf>.