Visualizing a PID Controller and Trade-offs in Tuning Parameters

Kevin-Patxi Le Bras

Abstract—A PID (proportional-integral-derivative) controller is the most commonly implemented control algorithm. A PID controller's three inputs vary the output behavior of the control system. This paper introduces an interactive tool to visualize the trade-offs in the selected parameters in a 3D space. This was accomplished by taking data from simulated control experiments and analyzing it for three characteristics: speed, efficiency, and stability. Speed is defined as the time it takes to reach steady state. Efficiency is defined as the total power consumed in reaching steady state. Stability is defined as the percent overshoot of the system. The collected characteristics will then be mapped to a 3D space proportional to its input PID parameters. The 3D space will be visualized as a collection of three slices, which can be moved by the user. The goal of the visualization is to aid in further tuning of the PID controller.

Index Terms—PID Visualization Controls

I. INTRODUCTION

The motivation to pursue this topic is from my own experience tuning controllers. The PID controller is one of the most common control algorithms used, but tuning a controller takes patience. It is easy to spend hours fine-tuning parameters with little progress.

A. Current Visualization Techniques

2D step response plots of feedback control systems are helpful in determining the behavior of a single set of PID controller settings but lack the information on how a change to the inputs could affect the output. Animated versions of the 2D step responses plots exist that help visualize how changing a single one of the PID controls affects the control output, but it is difficult to understand the whole problem in this space. Another approach is to visualize the control system in the frequency domain as a collection of poles and zeroes. This type of visualization helps, but it is difficult to visualize what changing a control's input will do to the output in this space, as it does not behave linearly. A small movement can have a huge or very small effect on the controller. An example of a step response plot and a root locust can be seen in figure 1.

A 3D space in which the three performance characteristics can be visualized would then be ideal. You could visually see the effect your inputs have on the performance as well as see what potential changes to the parameters would have on the system.

One technique for dealing with multiple of the papers found that addresses the problem of visualizing multiple data sets in the same visualization is "Visualizing multiple fields on the same surface." by Taylor Russel [3]. Russel approaches the problem in a few different ways. The first way he describes is by simply visualizing all data sets on the same plane but use separate visualization techniques for each set of data. This approach works well for a few data sets but becomes way to cluttered after 3 or 4 sets of data are stacked on top of one another. Another one of solutions described is called data driven spots (DDS). The basic concept of DDS is that each set of data is represented with a texture glyph, and the presence of the texture is determined by the values of the data. The main idea is that every set of data does not need to be displayed everywhere. Instead, we only select areas of the data set that are significant and display only those parts. This allows you to layer many more sets of data.

The biggest challenge for this project is to find a way to visualize all three collected characteristics in the same space. There are many ways to approach this problem.

One of the proposed methods is to create a 3D glyph and have the performance values control the dimension and other values of the glyph. For example, we could have energy denote size, speed by hue, and stability through altering geometry of the glyph. This visualization technique is straightforward but it could become cluttered quickly, and the interesting data could be difficult to see.

Another approach is to transform one of the characteristics into a velocity vector field. With this vector field you could then create path lines through the space. These path lines would all converge nowhere the characteristic is highest. The other two characteristics could then be encoded as the hue, brightness, or even transiency of the path line. This visualization technique is straightforward but could become cluttered quickly, and the interesting data could be difficult to see.

Another approach is to transform one of the characteristics into a velocity vector field. With this vector field you could then create path lines through the space. These path lines would all converge nowhere the characteristic is highest. The other two characteristics could then be encoded as the hue, brightness, or even transiency of the path line. This is interesting but it places a lot of focus on the characteristic you chose to make the path lines with. One could convert all of the performance characteristics into a velocity vector field and display them all in the same space. Each data set would...
Fig. 2. Mass Spring Damper Model

yet another possibility is to use marching cubes [2] to display an isometric surfaces of one or multiple characteristics in the same space. This is similar to the approach the path line approach but using surfaces instead.

II. METHODOLOGY

A. Model and Data Collection
To create data an idealized mass,spring,damper model was used. The dynamics of the system can be expressed in the following form.

\[ M \ddot{x} + b \dot{x} + kx = F \]

A PID controller was then set up to provide feedback to the system. A fixed set of PID values were then run through this controller and the results were analyzed for the following three variables described below.

B. Variables and Design Criteria
There are six variables in our visualization space, three inputs and three outputs. The three inputs are P(proportional error), I (Integrated error), and D(derivative error). The three outputs to the system are efficiency, stability, and speed. Efficiency is calculated as the sum of all accelerations [1].

\[ efficiency = \sum acceleration \]

Speed is defined as the time it takes for the function to settle within 5 percent of the goal.[1]

\[ speed = \frac{1}{t_{5\%}} \]

Stability is defined as the percent the step response over shoots the goal.[1]

\[ stability = \%overshoot \]

To ensure that the visualization created is a useful tool, the following criteria was defined. First, the user must easily be able to determine if the design criteria are met. Secondly, steps tasked to improve the designed controller must be clear. The user should also be able to determine how the controller is performing overall. Lastly movement in the visualization space should map linearly to changes to the controllers selected variables.

C. Color Mapping
To visually distinguish the 3D variables I assign them each a color. Efficiency is mapped to Green, speed is mapped to red, and stability is mapped to blue. These three colors were chosen because of their contrast with each other. While displayed in the same space the colors are easily distinguishable.

D. Experiments with 3D Space
The first iteration of this tool use a 3d rendering of contour lines and stream lines. The three contour volumes are of the efficiency, speed, and stability minim criteria. The stream lines seen in this space travel through the calculated derivative of the efficiency, speed, and stability data sets. Figure 3 is an example of the data visualized in this way. The problems with this visualization are very apparent. Firstly, The visualization is very cluttered. It becomes very difficult to extract much useful information from it. The volumes in the foreground obscure the insides of the volumes. It is also difficult to determine ones position in 3D space without being able to actively move the model.

After creating this model it was clear that a more simplified visualization was needed. Drawing inspiration from "The eyes have it: A task by data type taxonomy for information visualizations." by Ben Shneiderman[5] a new visualization was created. Shneiderman recommends displaying 3D data sets in 2D slices and having a slider move the data through the 3rd dimension.

III. RESULTS AND DISCUSSION

A. PID visualization tool
The final visualization developed for this project is a collection of four visualizations. All four visualizations are controlled with three sliders representing the inputs of the system, P, I, and D. The first three visualizations are 2D slices of the 3D space. Each slice depicts a contour of the minimum requirement three outputs variables in red green and blue. There are also three arrows pointing out from the current position in each of these windows. These arrows represents the direction of the derivative of the efficiency, speed, and stability variables. Each window is controlled by one of the three sliders mapped to P, I, and D. When the P slider is moved the corresponding slice of I and D space, is moved through the volume. When the I and D sliders are moved the PD and PI slices are moved respectively. The fourth window displayed shows a simple 3D representation of the space with a
cursor moving around and a contour plot of the three variables, efficiency, stability, and speed, projected on to the X,Y,Z planes respectively.

The resulting tool can be seen in figure 4.

B. Future Improvements

The tool works well but it is far from perfect. Some improvements I would like to see include filtering the data to smooth out the derivatives, adding the ability to change the minimum requirements of the controller during the simulation, adding the data points during the simulation so that you can see the data evolve.

REFERENCES