|  |  |
| --- | --- |
| **分类号密级** | 校徽 |
| **UDC** |
|  |

**PID自优化的实现**

**学生姓名 张理官 学号 16090022053**

**指导教师 赵波**

**院、系、中心 工程学院机电工程系**

**专业年级 机械设计制造及其自动化2016级**

**日期 2019 年 6 月 25 日**

**中 国 海 洋 大 学**

1. **目的与意义**

这次matlab大作业我所选择的题目是“PID自优化的实现”，虽然知道难度会挺大的，但是我还是抱着浓厚的兴趣进行了学习和实践。

在现代控制理论中，PID控制一直是一个重点，它在实际生活中也处处可见，平衡车、无人机、机器人、液位控制、炉温控制等等。而经过了许多年的发展，PID参数的调节方法也得到了不断地补充，从一开始的试凑法，到后来的临界比例度法，再后来出现的模糊控制法和基于神经网络的自适应调参法。总之，PID的参数一直不是一个简单的公式就能够的出来的，而经过经验和不断趋近得出来的最优解。

在我接触的同学里面，大家使用PID时大多数都采用的试凑法，这种方式及其繁琐和漫长，得出来的参数往往无法适应各种情况。比如平衡车上多加了一块小电路板，可能就会导致又一轮漫长的调参。

而事实上我们应该学会去用更加高效和科学的方法去优化我们的PID，试问，如果做一个控制系统的调节器，却没有对这个系统进行深入的了解和数学的建模，如何能够在茫茫的参数对中最快的找到合适的参数？

因此，我的目的就是找到一种更好的方法代替现在普遍使用的试凑法，也许在认识到新方法的便利之后，大家才会摒弃旧方法甚至更进一步开发更新的方法。

同时，我在这个过程中也学习到了很多控制系统的知识，回顾了之前上课所学到的知识，开拓了一些新的领域的知识，尤其是熟练掌握了matlab一些重要的功能例如符号运算、句柄调用、动画制作等等。

1. **设计理论依据**

**传递函数定义：**

把具有线性特性的对象的输入与[输出](https://baike.so.com/doc/6782547-6999006.html" \t "https://baike.so.com/doc/_blank)间的关系，用一个[函数](https://baike.so.com/doc/5368923-5604762.html" \t "https://baike.so.com/doc/_blank)(输出波形的[拉普拉斯变换](https://baike.so.com/doc/5566254-5781363.html" \t "https://baike.so.com/doc/_blank)与输入波形的拉普拉斯变换之比)来表示的，称为传递函数。

**动态性能指标（阶跃响应）**

上升时间（rise time）tr

指响应从终值10%上升到终值90%所需的时间；对于有振荡的系统，也可定义为响应从零第一次上升到终值所需的时间。上升时间tr是系统响应速度的一种度量。上升时间越短，响应速度越快。

峰值时间 tp

指响应超过其终值达到第一个峰值所需的时间。上升时间tp是系统响应速度的一种度量。

调节时间（settingtime）ts

指响应到达并保持在终值±5%(或±2%)内所需的最短时间。调节时间ts是评价系统响应速度和阻尼程度的综合指标。

超调量（overshoot）：σ %

指响应的最大偏离量c(tp)与终值c(∞)之比的百分数，若c(tp)<c(∞)，则响应无超调。σ%评价系统的阻尼程度。

**PID控制**

比例P控制

比例控制是一种最简单的控制方式。其控制器的输出与输入误差信号成比例关系。

积分I控制

在积分控制中，控制器的输出与输入误差信号的积分成正比关系。。

微分D控制

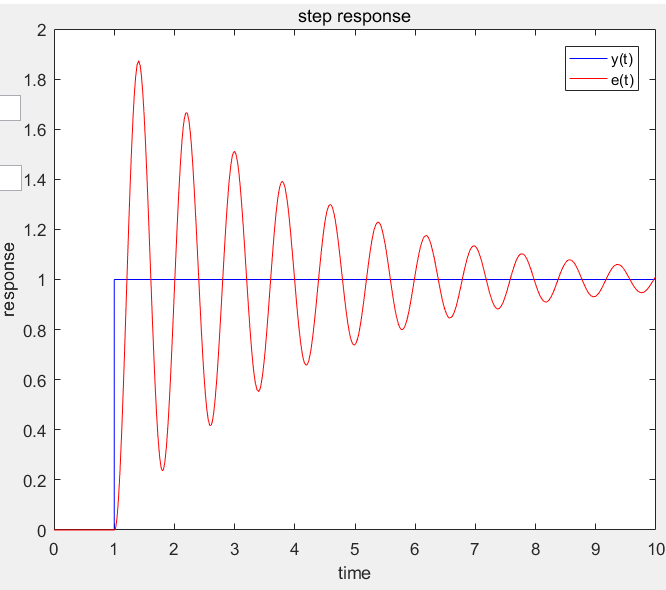
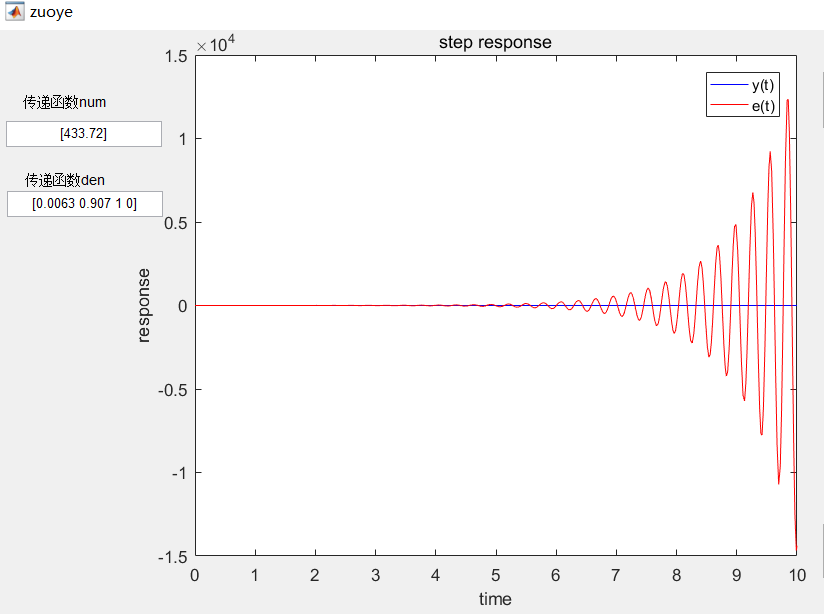
在微分控制中，控制器的输出与输入误差信号的微分(即误差的变化率)成正比关系。

**临界比例度法**

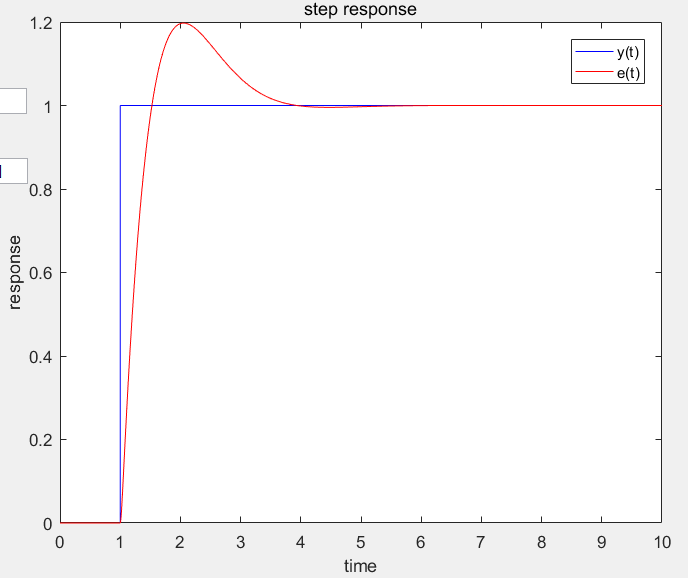
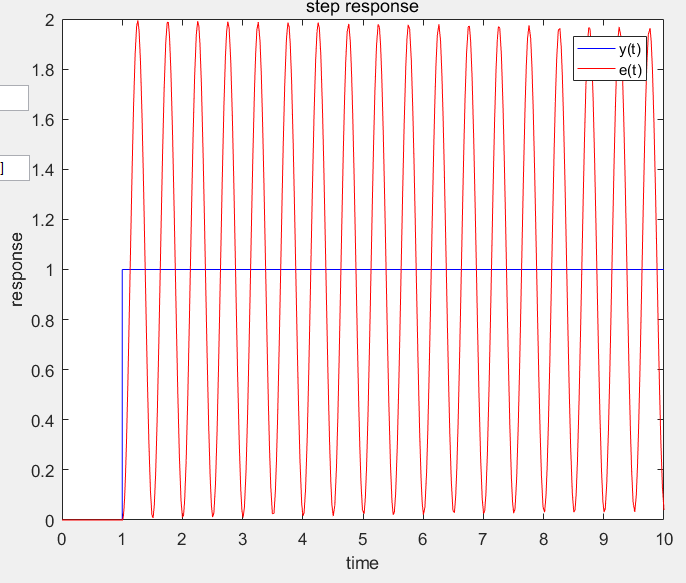
临界比例度法是目前使用较多的一种方法。它是先通过试验得到临界比例度∂k 和临界周期Tk,然后根据经验总结出来的关系求出控制器各参数值。具体作法如下:在闭环的控制系统中,先将控制器变为纯比例作用,即将T1 放在。∞位置上,TD 放在∞位置上,在干扰作用下，从大到小地逐渐改变控制器的比例度,直至系统产生等幅振荡(即临界振荡)。

临界比例度法比较简单方便,容易掌握和判断，适用于一般的控制系统。但是对于临界比例度很小的系统不适用。因为临界比例度很小，则控制器输出的变化一定很大，被调参数容易超出允许范围,影响生产的正常进行。临界比例度法是要使系统达到等幅振荡后，才能找出∂K 与TK,对于工艺上不允许产生等幅振荡的系统本方法亦不适用。

1. **实际效果**



**图一 未加PID控制器时系统阶跃响应 图二 正在寻找系统临界震荡**

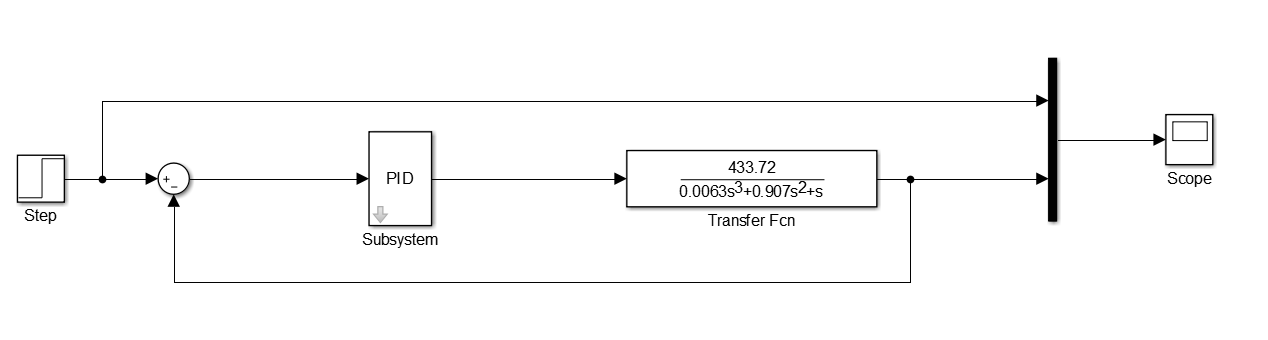


**图三 达到临界振荡 图四 计算出PID调节器各参数**

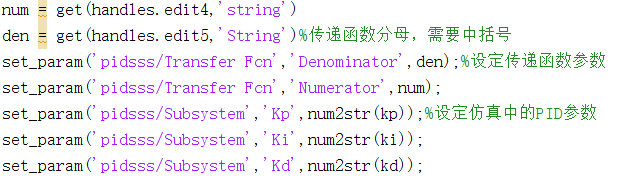
当然，这样得出的不一定是最优解，但是在这样的基础上，我们只需要在进行微调，就可以达到我们想要得到的最优解了。毫无疑问这样比从零开始或是根据经验开始调参数更加高效和准确！

1. **实现过程**

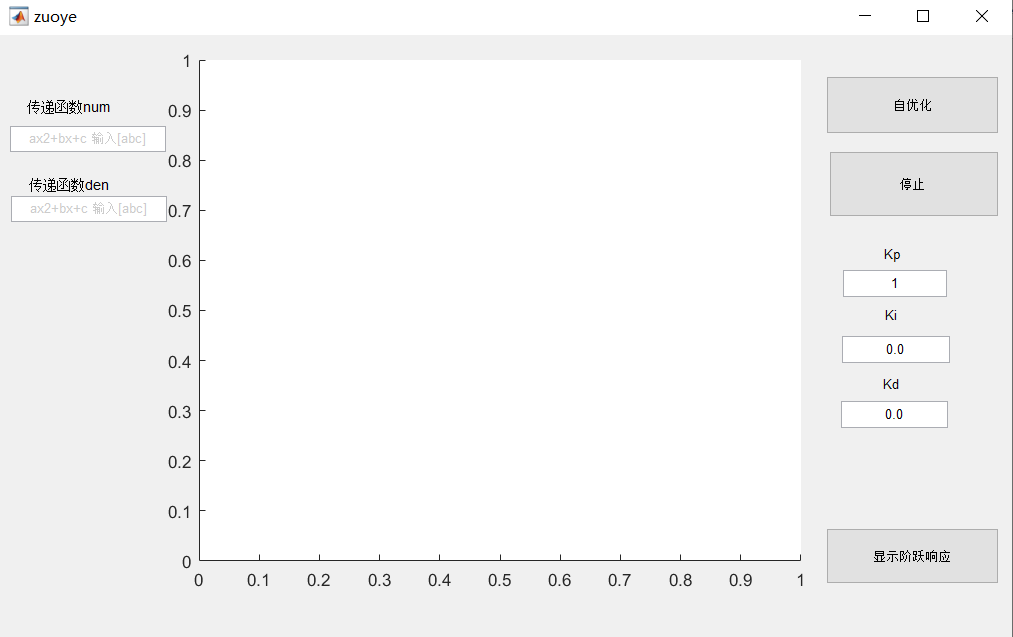
1.首先是simulink仿真



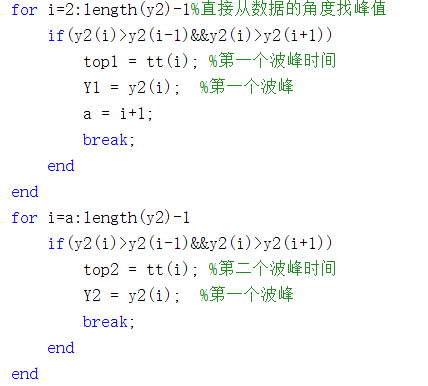
1. 实现simulink和m文件之间的交互



1. 制作gui界面，编写回调函数，实现手动调整PID参数



1. 设计判断临界振荡的计算方法



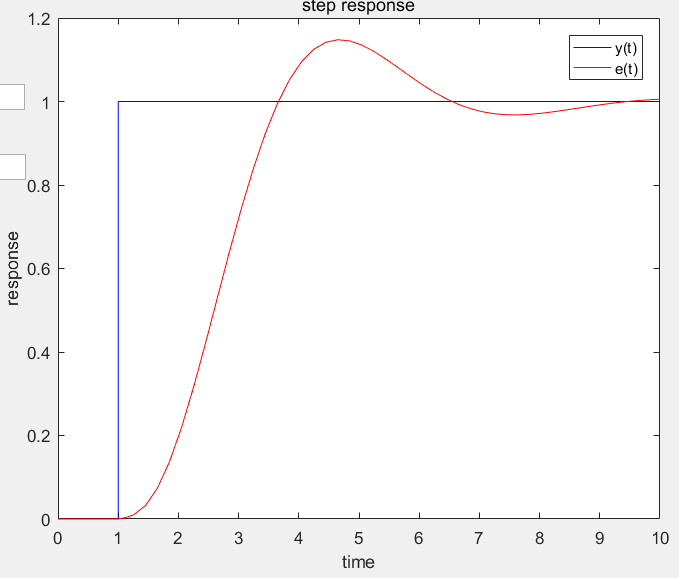
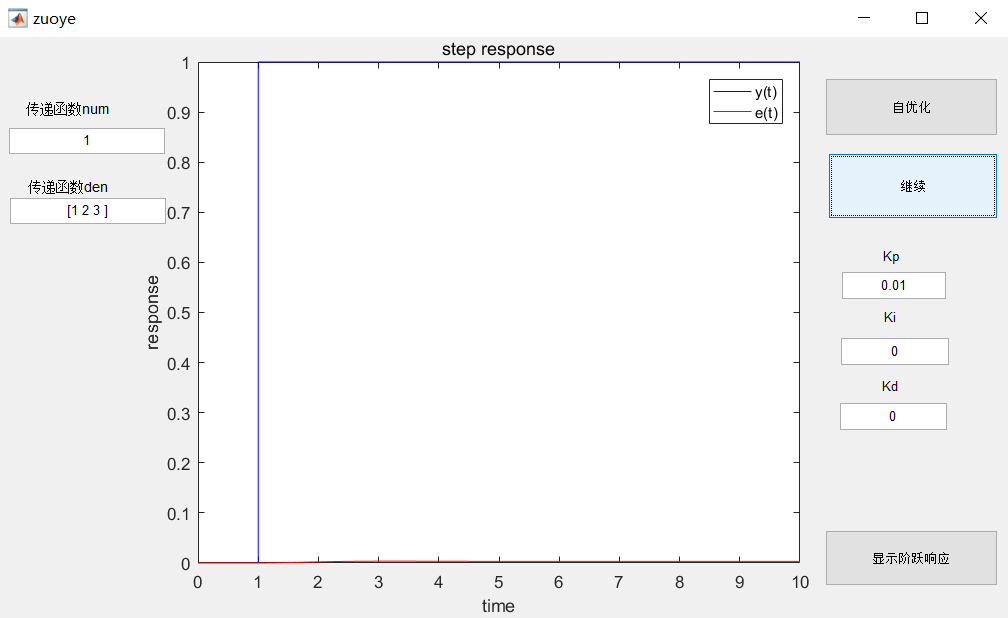
1. 应用循环和按键之间的互调，实现自优化和中断操作
2. 制作启动界面



1. **解决的问题**

已解决的问题：

开始写出来的算法对第一第二个波峰只进行了比较，之比接近1是判断为震荡，但是从实际输入传递函数来看，会出现这样的情况：

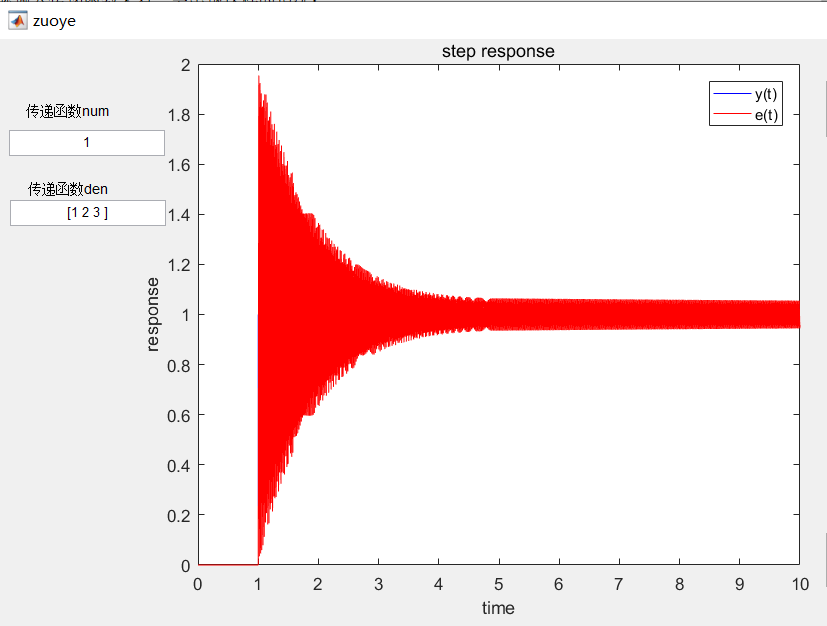
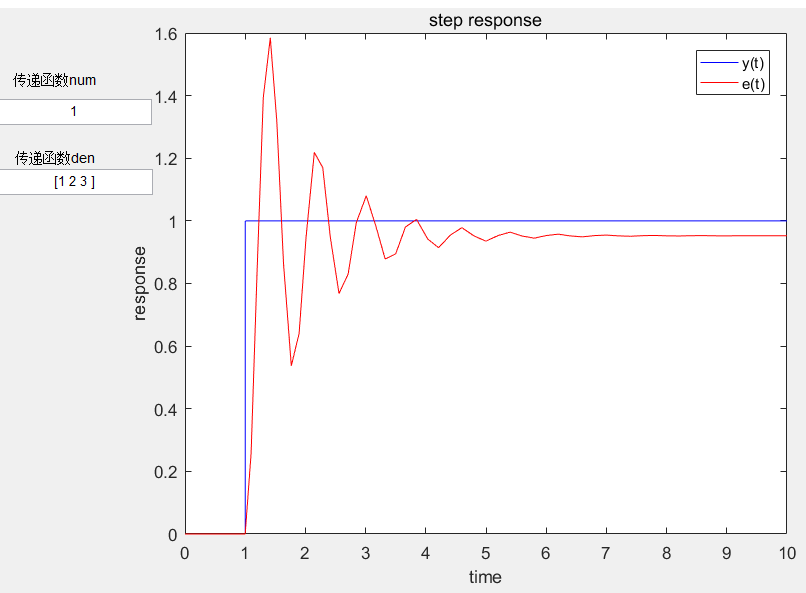


**图五 自优化判断临界振荡 图六 得出的效果图不理想**

例如这种传递函数，其分子太小以至于对于阶跃响应的反应非常小，就导致响应波变化不明显，程序误将它作为震荡情况。

解决方法：

对波峰波谷加以判断，波峰是否在阶跃输入终值之上，波谷是否在之下。



**图七 避开了之前的问题 图八 但是进入新的问题**

未解决的问题：

这个问题是临界比例度法客观存在的缺陷，即对于临界比例度很小的系统不适用。因为临界比例度很小，则控制器输出的变化一定很大，被调参数容易超出允许范围,影响生产的正常进行。临界比例度法是要使系统达到等幅振荡后，才能找出∂K 与TK,对于工艺上不允许产生等幅振荡的系统本方法亦不适用。

1. **总结**

这次做的这个小项目已经是能够解决一些常见的PID参数调控问题了，能够大幅度提升工作效率。固然存在不足，但是未来肯定会有更多优秀的算法涌现出来。能够早点接触到这些对我的帮助也很大。