

文章编号:1672-9315(2006)04-0511-04

基于 MATLAB 的 PID 控制器参数整定及仿真

何东健¹, 刘忠超², 范灵燕³

(1. 西北农林科技大学 信息工程学院; 2. 西北农林科技大学 机械与电子工程学院;

3. 西北农林科技大学 经济管理学院, 陕西 杨凌 712100)

摘要: PID 控制器结构和算法简单, 应用广泛, 但参数整定方法复杂, 通常用凑试法来确定。文中探讨利用 MATLAB 实现 PID 参数整定及仿真的方法, 并分析、比较比例控制、比例积分控制和比例微分控制, 探讨了 K_p , T_i , T_d 3 个参数对 PID 控制规律的影响。

关键词: MATLAB; PID 控制器; 参数整定; 仿真

中图分类号: TP 301

文献标识码: A

Parameter tuning and emulation of PID controller based on MATLAB

HE Dong-jian¹, LIU Zhong-chao², FAN Ling-yan³

(1. College of Information Engineering, Northwest A&F University; 2. College of Mechanical and Electronic Engineering, Northwest A&F University; 3. College of Economics and Management, Northwest A&F University, Yangling 712100, China)

Abstract: The control structure and algorithm of PID is easy and widely applicable, but its setting methods of parameter are multifarious. Generally utilize guessing and trying to fix. This article is convenient to tune PID parameters and emulate through MATLAB experiment. Analyze and compare the proportion control, the proportion integral control and the proportion differential control. Discuss the influence of three parameters K_p , T_i and T_d to the PID control rules.

Key words: MATLAB; PID controller; parameter tuning; emulation

0 引言

PID 控制器又称为 PID 调节器, 是按偏差的比例 P (ProPortional)、积分 I (Integral)、微分 D (Differential or Derivative) 进行控制的调节器的简称, 它主要针对控制对象来进行参数调节。PID 控制器问世至今, 控制理论的发展经历了古典控制理论、现代控制理论和智能控制理论 3 个阶段。在工业控制系统和工程实践中, 传统的 PID 控制策略依然被广泛采用。因为它算法简单、稳定性好、工作可靠、鲁棒性好, 在工程上易于实现。但 PID 控制器的参数整定方法复杂, 通常采用 PID 归一参数整定法和试凑法来确定, 费时、费力, 且不能得到最优的整定参数^[1,2]。针对这一问题, 文中探讨用 MATLAB 实现 PID 参数整定及仿真的方法及控制参数对 PID 控制规律的影响。利用 MATLAB 强大的计算仿真能力, 解决了利用试凑法来整定参数十分浩繁的工作, 可以方便、快速地找到使系统达到满意性能指标的参数。

1 PID 控制器的原理与算法

当被控对象的结构和参数不能被完全掌握,或得不到精确的数学模型时,应用 PID 控制技术最为方便。PID 控制器就是根据设定值与实际值的误差,利用比例(P)、积分(I)、微分(D)等基本控制规律,或者把它们适当配合形成有 PI、PD 和 PID 等的复合控制规律,使控制系统满足性能指标要求。

控制系统大多都有储能元件,这就使系统对外界的响应有一定的惯性,且能量和信息在传输和转化的过程中,由于管道、距离等原因也会造成时间上的延迟,所以,按偏差进行比例调节,很难取得理想的控制效果,因此引入偏差的积分(PI)调节以提高精度,引入偏差的微分(PD)来消除系统惯性的影响。这就形成了按偏差的 PID 调节系统^[3]。

图 1 是典型 PID 控制系统结构图。在 PID 调节器作用下,对误差信号分别进行比例、积分、微分组合控制。调节器的输出作为被控对象的输入控制量。

PID 控制算法的模拟表达式为

$$\mu(t) = K_p[e(t) + \frac{1}{T_i} \int_0^t e(t) dt + T_d \frac{de(t)}{dt}] \quad (1)$$

相应的传递函数为

$$G_c(s) = K_p(1 + \frac{1}{T_i s} + T_d s) \quad (2)$$

式中 K_p 为比例系数; T_i 为积分时间常数; T_d 为微分时间常数^[4]。

在传统的 PID 调节器中,确定 K_p 、 T_i 、 T_d 3 个参数的值,是对系统进行控制的关键。因此,控制最主要的问题是参数整定问题,在 PID 参数进行整定时,若有理论方法确定 PID 参数当然最为理想,但实际应用中,更多的是通过试凑法来确定 PID 的参数。而利用 MATLAB 强大的仿真工具箱的功能,可以方便地解决参数整定问题。

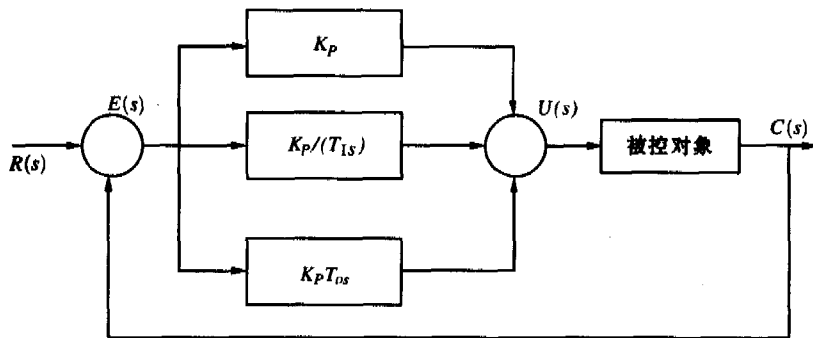


图 1 典型 PID 控制系统结构图

Fig. 1 Diagram of typical PID control system structure

2 电子心率起搏器系统

电子心率起搏器系统可以认为是一个实时监控装置。它是由起搏器和起搏导线构成。起搏器本身是一个脉冲发生器,由微电子电路和紧凑型电池构成。起搏器必须能够识别和感知心脏自主激动,当起搏器感知不到任何心脏自主跳动时,起搏器就会释放一个电脉冲,心脏的肌肉就会收缩一次。起搏器通过一根或两根起搏导线与心脏相连,起搏导线是一根很细的、外带绝缘层的导线,直接放置在右心室或右心房中。通过这根导线,电脉冲被传送到心脏。起搏导线也可以感知心脏的自主激动,并将这个信息传回给起搏器。

起搏器按照设定的频率每隔一段时间对心脏跳动情况进行监控。监控频率和时间间隔是根据人的正常心脏跳动频率来设计的。比如一个人的正常心率是 75 次/min,那么监控频率则设定为 75 次/min,两次监控的时间间隔为 0.8 s。起搏器工作时,自动时钟装置每隔 0.8 s 就发出信号引导监测心脏跳动,若过了 0.8 s 心脏还没有动,起搏器就会立刻发出电脉冲信号,对心脏进行电刺激,引发心脏跳动,保证心脏正常功能运转。

由起搏器的工作原理和设计过程可知,电子心率起搏器系统中模仿心脏的传递函数相当于一纯积分器,同时要把心脏跳动状态的电脉冲信号反馈传回心脏,为一单位负反馈。起搏器的增益及系统参数根据设计经验选择(图 2)。为了及时准确地控制起搏器正常工作,就需要在前端加入 PID 控制,而 PID 控制器参数选择是设计要解决的主要问题。

3 PID 控制器的 MATLAB 仿真

MATLAB 是美国 MathWorks 公司推出的一套高性能的数值计算和可视化软件,它集数值分析、矩阵运算、信号处理和图形显示于一体,构成了一个方便的、界面友好的用户环境。MATLAB 可以方便地设计漂亮的界面,具有丰富的函数库,和其他高级语言也具有良好的接口,同时也可以方便地实现与其他语言的混合编程,使它已经成为国际上最为流行的科学与工程计算的软件工具,进一步受到了科研工作者的欢迎^[5,6]。

PID 控制器的参数 K_p, T_i, T_d 分别对系统性能产生不同的影响。在控制过程中如何把 3 参数调节到最佳状态,需要深入了解 PID 控制中 3 参数对系统动态性能的影响。下面讨论起搏器 3 参量变化时对系统控制作用的影响。在讨论一个参量变化产生的影响时,设另外两个参量为常数。

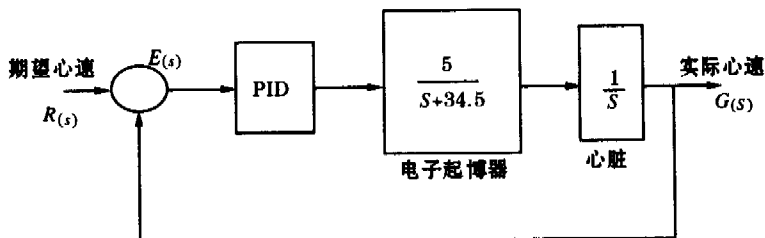


图2 电子心率起搏器系统结构图

Fig. 2 Diagram of electronic heart - rate instrument system structure

3.1 P 控制作用分析

设 $T_d = 0, T_i = \infty, K_p = 200 \sim 240$ 。

输入信号为阶跃函数,根据电子心率起搏器系统结构图,进行 MATLAB 程序仿真如下。

```
% P 控制作用程序
G1 = tf(5,[1 34.5]);           % 给定中间环节的传递函数形式
G2 = tf(1,[1 0]);
G12 = G1 * G2;
Kp = [200:20:240];
for i = 1:length(Kp)
    Gc = feedback(Kp(i) * G12,1);
    step(Gc),                  % 对单位阶跃函数的响应
    hold on
end
axis([0,5, 0.2,1.6]);
```

由运行该程序后系统的阶跃响应曲线(图3)可知,为了提高响应速度和调节精度,电子心率起搏器系统中 PID 控制器的参数 K_p 应选用 240。

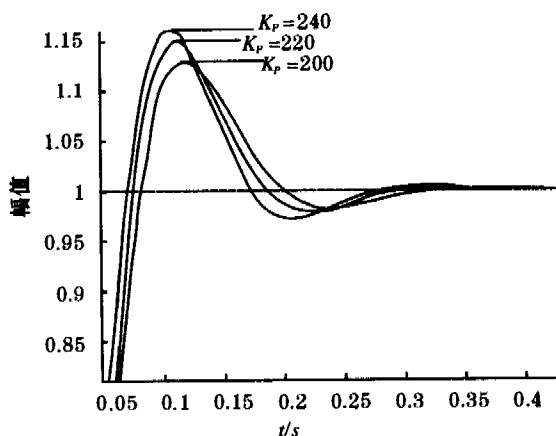


图3 电子心率起搏器系统 P 控制阶跃响应曲线

Fig. 3 Curve of electronic heart-rate instrument system P control step leap response

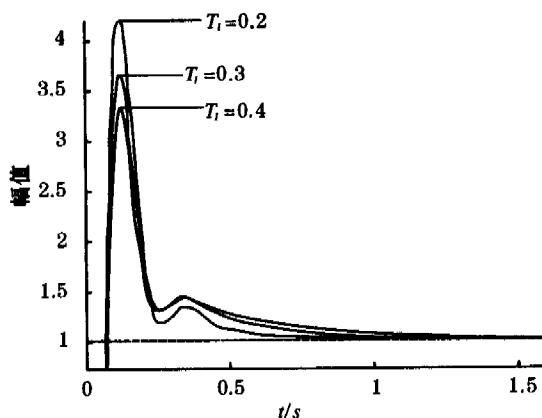


图4 电子心率起搏器 PI 控制阶跃给定响应曲线

Fig. 4 Curve of electronic heart-rate instrument system PI control step leap response

3.2 比例积分控制作用的分析

设 $K_p = 200$, 讨论 $T_i = 0.2 \sim 0.4$ 时对系统阶跃响应曲线的影响。根据电子心率起搏器系统结构图的数据, 进行 MATLAB 程序仿真如下。

```
% 比例积分控制作用程序
G1 = tf(5, [1 34.5]);           % 给定中间环节的传递函数形式
G2 = tf(1, [1 0]);
G12 = G1 * G2;
Kp = 200;
Ti = [0.2; 0.1; 0.4];
for i = 1:length(Ti)
    Gc = tf(Kp * [Ti(i) 1], [Ti(i) 0]);
    Gcc = feedback(Gc * G12, 1);
    step(Gcc),                  % 对单位阶跃函数的响应
    hold on
end
axis([0, 12, 0.2, 1.6]);
```

由运行该程序后系统的阶跃响应曲线(图4)可知, 为了消除系统的稳态误差, 提高系统的控制精度, 电子心率起搏器系统中 PID 控制器的参数 T_i 应选用 0.3。

3.3 比例微分调节作用的分析

设 $K_p = 200$, $T_i = 0.2$, 讨论 $T_d = 100 \sim 700$ 时对系统阶跃响应曲线的影响。根据电子心率起搏器系统结构图, 进行 MATLAB 程序仿真如下。

```
% 比例微分调节作用的程序
G1 = tf(5, [134.5]);           % 给定中间环节的传递函数形式
G2 = tf(1, [1 0]);
G12 = G1 * G2;
Kp = 200; Ti = 0.2;
Td = [100; 300; 700];
for i = 1:length(Td)
    Gc = tf(Kp * [Ti * Td(i) Ti 1], [Ti 0]);
    Gcc = feedback(Gc * G12, 1);
    step(Gcc),                  % 对单位阶跃函数的响应
    hold on
end
```

```
axis([0, 12, 0.2, 1.6]);
```

由运行该程序后系统的阶跃响应曲线(图5)可知, 为了改善系统的动态性能, 如减小超调量, 缩短调节时间, 电子心率起搏器系统中 PID 控制器的参数 T_d 应选用 400。

4 结 论

1) 利用 MATLAB 对 PID 参数进行整定和仿真, 省去了传统方法反复修改参数, 反复试运行, 方便、快捷、省时、直观。

2) 增大比例系数 K_p 将加快系统的响应, 有利于减小静差, 但是过大会使系统有较大的 (下转第 523 页)

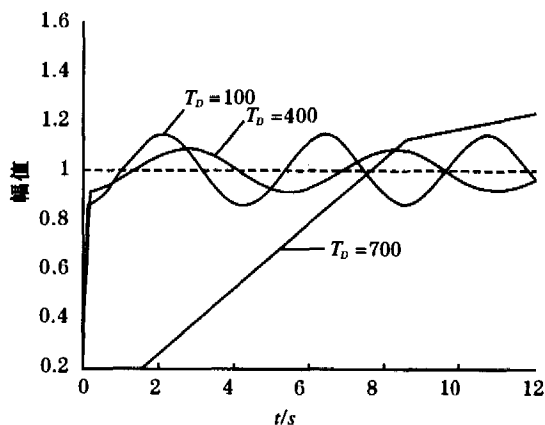


图5 电子心率起搏器 PID 控制阶跃给定响应曲线

Fig. 5 Curve of electronic heart-rate instrument system PID control step leap response

参考文献:

- [1] 李为民,俞巧云. 光点定位中的曲面拟合迭代算法[J]. 光学技术, 2004, 30(1): 1-3.
- [2] 张志勇. 精通 MATLAB 6.5 版[M]. 北京:北京航空航天大学出版社, 2003.
- [3] 刘贤德. CCD 及其应用原理[M]. 武汉:华中理工大学出版社, 1990: 1-15.
- [4] 杨 焯,裴继红,杨万海. 实时光点检测与跟踪方法研究[J]. 红外与毫米波学报, 2001, 20(4): 1-4.
- [5] 杨志文. 光学测量[M]. 北京:北京理工大学出版社, 1995: 354-358.
- [6] Castleman K R. 数字图像处理[M]. 北京:清华大学出版社, 1998: 200-229.

~~~~~  
(上接第 514 页)

超调,使稳定性变坏; $K_p$  取值过小,会使系统的动作缓慢。

3) 增大积分时间  $T_i$  有利于减小超调,减小振荡,使系统的稳定性增加,但系统静差消除时间变长;若  $T_i$  过小,系统的稳态误差将难以消除,导致系统不稳定。

4) 增大微分时间  $T_d$  有利于加快系统的响应速度,使系统超调量减小,稳定性增加。但  $T_d$  不能过大,否则会使超调量增大,调节时间较长;若  $T_d$  过小,同样超调量也增大,调节时间也较长。

**参考文献:**

- [1] 赖寿宏. 微型计算机控制技术[M]. 北京:机械工业出版社, 2003: 90-111.
- [2] 刘明俊,于明祁. 自动控制原理[M]. 长沙:国防科技大学出版社, 2000.
- [3] 陈 辉,邵 林. 基于 MATLAB 的数字 PID 控制器仿真[J]. 连云港职业技术学院学报, 2004, 19(2): 31-32.
- [4] 齐剑玲,曾玉红,刘慧芳. PID 调节器的仿真研究[J]. 海淀走读大学学报, 2004, 8(1): 69-71.
- [5] 王沫然. MATLAB 与科学计算[M]. 第二版. 北京:电子工业出版社, 2003.
- [6] 薛定宇. 控制系统计算机辅助分析——Matlab 语言及其应用[M]. 北京:清华大学出版社, 1996.