



yanghq13

已关注

0



5

5



(

【运动控制】Apollo6.0的LpfCoefficients解析

原创 yanghq13 2021-07-19 12:02:20 235 收藏 5 版权

分类专栏: Apollo 6.0 文章标签: 自动驾驶



Apollo 6.0 专栏收录该内容

7 订阅

6 篇文章

订阅专栏

【运动控制】Apollo6.0的LpfCoefficients解析

LpfCoefficients解析

1 digital_filter_coefficients.h

2 digital_filter_coefficients.cc

2.1 LpfCoefficients

2.2 LpFirstOrderCoefficients

3 感谢

.pfCoefficients解析

知乎: 【运动控制】Apollo6.0的LpfCoefficients解析

digital_filter_coefficients.h

```
1 #pragma once
2 #include <vector>
3 #include "cyber/common/log.h"
4
5 namespace apollo {
6 namespace common {
7
8 void LpfCoefficients(const double ts, const double cutoff_freq,
9                     std::vector<double> *denominators,
10                    std::vector<double> *numerators);
11
12 void LpFirstOrderCoefficients(const double ts, const double settling_time,
13                             const double dead_time,
14                             std::vector<double> *denominators,
15                             std::vector<double> *numerators);
16
17 } // namespace common
18 } // namespace apollo
19
```

digital_filter_coefficients.cc

2.1 LpfCoefficients

pf: lowpass filter

从代码分析,采用的是二阶巴特沃斯低通滤波器,推导如下

传递函数如下

$$H(s) = \frac{\omega_N^2}{s^2 + 2 * 0.707 * \omega_N * s + \omega_N^2}$$

其中，阻尼比为0.707。

采用双线性变换，T为采样周期

$$s = \frac{2}{T} \frac{1 - z^{-1}}{1 + z^{-1}}$$

为了简便，设 $\alpha = \frac{\omega_N T}{2}$ ，将上式带入传递函数，可以得到

$$H(z) = \frac{\frac{\alpha^2(1+2z^{-1}+z^{-2})}{1+\sqrt{2}\alpha+\alpha^2}}{1 + \frac{2(\alpha^2-1)}{1+\sqrt{2}\alpha+\alpha^2}z^{-1} + \frac{1-\sqrt{2}\alpha+\alpha^2}{1+\sqrt{2}\alpha+\alpha^2}z^{-2}}$$

```
1 void LpfCoefficients(const double ts, const double cutoff_freq,
2                     std::vector<double> *denominators,
3                     std::vector<double> *numerators) {
4     denominators->clear();
5     numerators->clear();
6     denominators->reserve(3);
7     numerators->reserve(3);
8
9     double wa = 2.0 * M_PI * cutoff_freq; // Analog frequency in rad/s
10    double alpha = wa * ts / 2.0;          // tan(Wd/2), Wd is discrete frequency
11    double alpha_sqr = alpha * alpha;
12    double tmp_term = std::sqrt(2.0) * alpha + alpha_sqr;
13    double gain = alpha_sqr / (1.0 + tmp_term);
14
15    denominators->push_back(1.0);
16    denominators->push_back(2.0 * (alpha_sqr - 1.0) / (1.0 + tmp_term));
17    denominators->push_back((1.0 - std::sqrt(2.0) * alpha + alpha_sqr) /
18                            (1.0 + tmp_term));
19
20    numerators->push_back(gain);
21    numerators->push_back(2.0 * gain);
22    numerators->push_back(gain);
23 }
```

2.2 LpFirstOrderCoefficients

这里的滤波器主要由两个部分组成：

采样系统 -为0阶保持器，传递函数如下所示：

$$G_0(s) = \frac{1 - e^{-sT}}{s}$$

一阶滤波系统，传递函数如下所示：

$$G_1(s) = \frac{1}{s + a}$$

写成数字控制器的形式：

$$G(s) = G_0(s)G_1(s) = (1 - e^{-sT})\left(\frac{1}{s(s + a)}\right)$$

代入离散变换公式得：

$$G(z) = (1 - z^{-1}) \left(\frac{z(1 - e^{-aT})}{(z - 1)(z - e^{-aT})} \right)$$

最终我们可以得到：

$$G(z) = \frac{(1 - e^{-aT})}{(z - e^{-aT})}$$

上述离散公式与apollo代码中的一致，其中T为采样时间，与采样频率相关；参数 $a = 2\pi f$ ， f 为滤波器带宽，即为截止频率。

代码相关

代码里令 $\omega = 1/\text{settlingtime}$ ，按照上一步的公式推算猜测为估计的截止频率，单位为rad/s，不太明白是什么意思

```

1 void LpFirstOrderCoefficients(const double ts, const double settling_time,
2                               const double dead_time,
3                               std::vector<double> *denominators,
4                               std::vector<double> *numerators) {
5     // sanity check
6     if (ts <= 0.0 || settling_time < 0.0 || dead_time < 0.0) {
7         AERROR << "time cannot be negative";
8         return;
9     }
10
11     const size_t k_d = static_cast<size_t>(dead_time / ts); // 死区
12     double a_term;
13
14     denominators->clear();
15     numerators->clear();
16     denominators->reserve(2);
17     numerators->reserve(k_d + 1); // size depends on dead-time
18
19     if (settling_time == 0.0) {
20         a_term = 0.0;
21     } else {
22         a_term = exp(-1 * ts / settling_time);
23     }
24
25     denominators->push_back(1.0);
26     denominators->push_back(-a_term);
27
28     numerators->insert(numerators->end(), k_d, 0.0); // 插入几阶的零
29     numerators->push_back(1 - a_term);
30 }

```

感谢


感谢P&C组的张天宇同事对一阶低通滤波器的解析改进。



请发表有价值的评论， 博客评论不欢迎灌水，良好的社区氛围需大家一起维护。





评论

- 

flytree1991

博主，请教一下。为啥这里一阶滤波器还要考虑保持器呢？ 而二阶不需要考虑？


回复 4 月前 ...


- 

flytree1991 回复 yanghq13

哦哦 明白你的意思了 谢谢

回复 4 月前 ...


- 查看全部 4 条回复

- 二阶巴特沃斯低通滤波器(BLPF)

05-08
- 二阶巴特沃斯滤波器通频带内的频率响应曲线最大限度平坦，没有起伏，而在阻频带则逐渐下降为零。在振幅的对数特性上，它的幅频特性在通频带内保持0dB/十倍频程，在阻频带内以-40dB/十倍频程的速率下降。在阻频带外，它的幅频特性以-40dB/十倍频程的速率下降。在阻频带内，它的幅频特性以-40dB/十倍频程的速率下降。在阻频带外，它的幅频特性以-40dB/十倍频程的速率下降。
- Simulink中构造时变传递函数的四种方法

最新发布

Matlab_Fans技术博客

940
- 原由传递函数是针对线性时不变（Linear Time-Invariant, LTI）系统定义的，严格来说，时变传递函数的说法是不正确的。时变传递函数的定义是针对非线性系统定义的，它描述了系统在时变输入下的输出特性。时变传递函数的定义是针对非线性系统定义的，它描述了系统在时变输入下的输出特性。时变传递函数的定义是针对非线性系统定义的，它描述了系统在时变输入下的输出特性。
- 【运动控制】Apollo6.0的mpc_controller解析_yanghq13..

9-15
- 【运动控制】Apollo6.0的mpc_controller解析mpc_controller解析1 Init1.1 输入1.2 动力学模型初始化1.3 前馈
- apollo 运动规划算法解析_yangfan111的博客

11-1
- 参考:apollo 坐标系说明https://blog.csdn.net/davidhopper/article/details/79162385apollo 运动规划模块详解h
- apollo 6.0 规划算法解析分享

steve_shen0810的博客

1796
- 我在知乎开通专栏啦~ 专注分享自动驾驶规划控制算法，欢迎同方向的同学们一起学习交流~~~ 欢迎点入下面
- 双极点低通滤波器-二阶巴特沃斯滤波器

u013492900的博客

1601
- 双极点低通滤波器1 描述 1 描述 最近看到apollo的滤波器代码，猛一看没整明白，因此就想推导一下公式。代
- 车辆动力学及控制_Apollo控制算法车辆动力学模型分析（三）

weixin_39631519的博客

76
- 欢迎关注微信公众号《不想做科学家的工程师不是好码农》根据上一篇文章《Apollo控制算法车辆动力学模型
- 二阶巴特沃斯低通滤波器仿真TLV2252

weixin_30242907的博客

974
- 输入频率 10Hz和1KHz混合波形，通过仿真可以看到输出波形中，高频信号1kHz已经滤除。从输出结果中可以
- 【运动控制】Apollo6.0的leadlag_controller解析

weixin_44041199的博客

408
- 【运动控制】Apollo6.0的leadlag_controller解析leadlag_controller解析1 leadlag_controller.h2 leadlag_contr
- apollo学习笔记（10） 横向控制代码详读

qq_24649627的博客

789
- status LatController::ComputeControlCommand(const Localization::LocalizationEstimate *localization, cons
- IXF解析成运动控制程序

lj312302482的博客

3524
- IXF解析成运动控制程序。支持图形显示，鼠标可放大缩小、平移 支持打印尺寸按比例缩放 支持按图层解析
- 【运动控制】连续时间状态方程的离散化

weixin_44041199的博客

238
- 连续时间状态方程的离散化1 连续时间状态方程2 离散化2.1 前向差分2.2 后向差分2.3 双线性变换3 结语 在众
- 【运动控制】Apollo6.0的lon_controller解析

weixin_44041199的博客

590
- 【运动控制】Apollo6.0的lon_controller解析lon_controller解析1 创建纵向控制器1.1 构造函数1.2 析构函数1.3
- 【运动控制】Apollo6.0的pid_controller解析

weixin_44041199的博客

310
- 【运动控制】Apollo6.0的pid_controller解析pid_controller解析1 pid_controller.h2 pid_controller.cc2.1 Init2.2
- 【滤波器学习笔记】一阶RC低通滤波

热门推荐

叮勉!的博客

19万+
- 一阶RC低通滤波从模拟到数字 本文整理自网络、《匠人手记》等书籍文章 模拟电路低通滤波时域、频域 软件
- DSP] Butterworth （巴特沃斯）数字滤波器设计参考

Water's Space

1万+
- Butterworth （巴特沃斯）滤波器设计参考 在嵌入式音频产品开发过程中经常会到LPF（Low Pass Filter 低通



yanghq13

码龄3年

暂无认证

18

原创

10万+

周排名

9万+

总排名

1万+

访问

等级

等级

246

积分

41

粉丝

24

获赞

41

评论

167

收藏



私信

已关注

搜博主文章



热门文章



LOAM学习-安装与运行 1610

LOAM学习-代码解析（二）点云数据配准 scanRegistration 807

LOAM学习-LM方法 684

LOAM学习-代码解析（一）点云数据配准 scanRegistration 676

【ROS】动态链接库(.so文件)的生成和调用 611

分类专栏



Apollo 6.0

6篇



SLAM

9篇



loam

9篇



ros

1篇



C++

1篇



MPC

1篇



最新评论



【自动驾驶】MPC控制器 + 驾驶模型器
兴涛: 大佬你好，进入下载好的 MPC-Project 文件夹内，输入mkdir build &&...

【自动驾驶】MPC控制器 + 驾驶模型器
yanghq13: 你或者私信我一下...我不太理解你这个错误...发个图片给我。

【ROS】动态链接库(.so文件)的生成...
yanghq13: 不客气喔

【自动驾驶】MPC控制器 + 驾驶模型器
现代自闭控制: 就是执行完./term2_sim.x86_64后弹出的那个unity窗口一直...

【ROS】动态链接库(.so文件)的生成...
wechat-920086481: 博主写得很详细，
可以运行成功，之前参考了很多博客...

您愿意向朋友推荐“博客详情页”吗？



强烈不推荐



不推荐



一般般



推荐



强烈推荐

最新文章



【运动控制】Apollo6.0的mpc_controller
解析

【运动控制】连续时间状态方程的离散化

【运动控制】Apollo6.0的
leadlag_controller解析

2021年 8篇

2020年 10篇

目录



【运动控制】Apollo6.0的LpfCoeffici...

LpfCoefficients解析

1 digital_filter_coefficients.h

2 digital_filter_coefficients.cc

2.1 LpfCoefficients

2.2 LpFirstOrderCoefficients

3 感谢