

PID 白话式理解说明及智能车闭环控制详解

By jiahangsonic

编码器专卖 <http://shop58019203.taobao.com>

本文只是技术交流，仅仅是鄙人对一些知识的看法和认识，由于鄙人学疏才浅，必然会在本文中出现定义理解不深刻，原理叙述有误等错误，敬请各位理解，如有错误之处，请大家指出，我将积极学习改进。

其实很早就应该写这么一个东西，由于学习和工作太忙，一直没有时间去写，春节放假，偶尔有了时间，决心一定要写好，本文只是针对初学者，对于那些老鸟和大神们，基本上没有看的必要，所以再您看这篇文章之前，还要对我多多的理解和宽容，写不好，我改进学习，写的好，希望对您有帮助。

（一）PID 的背景和一些原理上理解

PID 控制技术，是最简单的闭环控制技术之一，一般都是利用单反馈或者多反馈来实现对控制对象的调节，实现被控对象的可控性和可预知性的控制。使得设备运行的更加的可靠，合理且平稳。

PID 的全称为比例积分微分控制，P 即为比例，I 即为积分，D 即为微分。PID 往往都是应用于惰性系统，所谓惰性系统就是变化较慢且无法精确控制和调节的对象，其中最最重要的特点就是变化速度慢，调节速度慢，控制周期较长，最经典的控制对象就为温度的温控。

下面就举一个简单的例子进行说明：

比如我们要对一个水箱里面的水进行加热，我们的目标加热温度为 100°C ，首先我们不用闭环对水温进行加热，也就是说我们只是靠人为观察温度计的温度值来对加热器进行人工的干预。

当温度加热到 100°C 以后，我们就停止加热，这个时候，虽然水温已经到达 100 且加热器已经不再通电加热，但是由于加热器的预热和水本身传递温度的惰性，导致水温会继续上升，经过一段时间后，水温会继续升高，并且超过 100°C ，那么该系统就无法达到我们所预期的要求。

这个时候您谁想，停止加热后本身会继续散热继续升温，那等到温度到 90 摄氏度左右以后，我们停止加热，然后利用水的惰性和加热器的散热，让水温继续升温，正好达到 100°C ，这样不就解决问题了吗？这么想是对的，但是水温要达到 90 几度的时候我们停止加热呢？还有就是从停止加热到 100°C 的时间是多少？经过一段时间后，温度没有达到 100°C ，而是小于 100 摄氏度以后温度就达到了顶峰，这样怎么办？

上述所有的办法，可能能够解决水温到达 100°C 的要求，但是其中很多环节很多结果都是无法预测和无法控制的，即便经历了很麻烦的人为干预同时经过了一个较长的时间达到了我们对水温加热到 100°C 的要求，也要经历一个相当复杂和相当漫长的时间才能达到，并且整个过程一直要有人为的干预，实在是属于劳民伤财。

不只是对温度的控制，还有其他很多领域的过程控制，都遇到了这些让人很困惑问题，所以科学家就针对此类问题发明了闭环控制原理，其中最经典最简单最实用的就是 **PID** 闭环控制。该控制原理简单可靠，参数调整简便，实用性强，广泛的受到人们的支持。

利用 **PID** 控制原理对水温进行加热控制，我现在进行举例说明：目标温度

还是我们之前设定的 100°C ，加热前我们首先要把这个温度值输入到加热器控制器中，此时温度计只是作为我们认为观察温度的一个参考，我们在水中放入一个电子温度计，电子温度计测量的温度可以传送到控制器，以使得我们的 PID 控制器知道当前的水温，在刚刚通电加热的时候，水温假定为室温 20°C ，当前的水温会通过电子温度计测量并传输到 PID 控制器中，此温度作为当前温度值 C_TEM ，我们的目标温度值设定的为 100°C 作为 T_TEM ，那么 CPU 通过计算 $T_TEM - C_TEM = 80$ ，二者的差值较大为 80，说明水温远远没有达到我们所需的 100°C ，那么这个时候 CPU 就给加热器通上最高的供电电压，以全速对水箱里面的水进行加热，随着加热的进行，水温会慢慢的增加，某一时刻，水温达到了 80°C ，那么目标温度与当前温度做差，差值仅仅为 20，那么 CPU 就知道了当前的温度跟实际我们所希望的温度差别已经很小了，那么这个时候控制器就减小加热器的供电电压，以一个适当的加热量对水温进行加热。当到了一个时刻，水温达到了 95°C ，目标温度和他的差值仅仅只有 5 了，说明温度已经十分接近目标温度，此时控制器可能已经早已停止对加热器供电或者以一个更加低电压的加热量对水温进行加热，直到水温加热到 100°C 。

当然水温不可能一直恒定在 100°C ，水温一定会超过 100 摄氏度，但是由于采用 PID 闭环控制，水温不会超过的太多，比如某一时刻水温到达了 105°C ，做差后，差值为 -5°C ，CPU 就知道此时温度已经超过了目标温度，那么将会在一段时间内停止加热，直到温度降低到 100°C 或者 100°C 以下后再进行进一步控制，所以当水温到达 100 摄氏度附近的时候，CPU 控制器将会频繁的对加热器进行控制，从而导致水温永远维持在 100°C 左右附近，而且水温与 100°C 的目标温度相差不会很大，宏观上通过平均测量，水温就是认为在 100°C 了，其实根

本上，水温是永远不会达到正好的 100°C ，只能是在 100°C 左右进行震荡，但是震荡始终不会太大，从而达到了我们控制目标对象的要求，使得加热系统变得更加科学更加可控。

当然上述所说明的 PID 加热状态是 PID 各项参数调节比较好的情况下的状态，PID 三个参数调整不好，会带来很多问题，比如系统震荡，超调等问题，所以调整 PID 参数也是一个比较重要的环节，这个我将会在下面进行说明。

上述对水温的 PID 控制已经基本上说明了 PID 控制的核心，我们之所以利用 PID 进行闭环控制，就是为了让被控制对象快速稳定的达到目标值，从而实现对被控制对象的可控性，可预知性。

一个人无法控制就是一个疯子，一个车子速度无法控制就是一个疯车。

（二） 智能车电机闭环控制算法

对于智能车的电机闭环控制算法，我之所以标题没有写上“智能车电机 PID 闭环控制算法”是因为 PID 算法根本就不是特别好的适用于智能车这种变化很快的系统，对于智能车，电机的调速可以说是时时刻刻再进行调速控制的，我上面说描述的经典 PID 算法，都是针对一些惰性系统，也就是说变化比较慢的系统的，所以对于智能车的电机调速采用完完整整的 PID 算法，是根本不可取的，及时采用了，你必须要经过一些变换和改进才能使用。以上的简述只是鄙人自己的看法，如有错误，请各位高手指正。

现在估计您会疑问，PID 不适用于智能车的电机控制，那什么才适用呢？鄙人原来做过智能车，从鄙人本身的理解，P 算法控制电机，也就是比例控制是最好的，反应速度快，控制精度高，不存在积分和微分效应，非常适用于适

用于控制周期短的系统，当然，对于一些特殊的逻辑控制算法，可能要采用 PD 算法，用微分来做补偿，防止震荡和超调。

下面来说下电机控制算法从开始的加入到最终的确定的方法：

当然这一切的前提就是安装了编码器，车速有反馈，只有加上编码器，有了反馈，才能组成一个闭环系统。当然您也可以加上码盘，或者霍尔开关等一切可以返回车速的东西都可以。

- (1) 首先建议在车速比较慢的时候，采用 PID 算法来控制电机，为什么开始要建议您采用 PID 呢？主要是为了让您更加深刻理解 PID 算法的精髓和调试步骤方法等，有助于以后对控制算法更加深入的研究和书写。调试 PID 三个参数的方法，很多地方都提供了，我在这里简单的说下：首先将 ID 参数都变为 0，先调整 P 比例参数，调整到速度基本上跟您给定的速度差不多，也就是说基本稳定在您给定的脉冲数，当然这个时候会非常的震荡，不要担心，接下来调整 I，调整 I 的结果就是震荡会消除很多，但是车速会变化缓慢，也就是说会有一些延迟，然后再调整 D，调整 D 的结果就是增强调节的灵活性和预见性，在给定速度变化的过程中，能够以一个平稳过渡来变换，而且速度可以长时间稳定在给定速度附近，然后 PID 三个参数的基本范围就确定了，然后再根据实际的跑车来微调这些参数，当然在您调试 PID 之前，请仔细阅读 PID 理论知识，这样有助于您的调试和理解，当您把 PID 调整好以后，您就基本上领会了 PID 的精髓，对于以后的调试作用非常大。这里的给定速度，就相当于上面对温

度例子中说的设定温度。意思是一样一样的。

- (2) 当您对 PID 掌握以后，您能做到给定多少速度，就能让车速稳定在这个给定速度左右，可以适当提高车速，慢慢观察效果，你会发现当速度增加的时候，车速的控制会变得越来越糟糕，调节缓慢，灵活性差。这个时候您就可以试着去掉 I 积分环节，然后再重新调整参数，按照上面说的，然后继续观察效果，试着提高速度，继续观察，因为车速毕竟很慢，很多细节我们都能看到，所以您就可以看到 PID 到 PD 变换以后，实际的效果是增加多少，也算是一种进步，这个时候，经典的 PID 公式已经被我们简化，不要担心，您可以多多实验多多观察，可能在这个变换之间，您能受到很多启发，会产生很多思路，谁以后的调试车子都是有好处的。 所以搞智能车，自己一步一步的进步学习，远远比去某些淘宝店铺购买现成的程序学到的东西更多，体会的也更多，思维开阔的也更多。
- (3) 在您的车速在 3M 内的时候，我强烈建议您采用 PD 算法，因为 PD 算法是最好调整，最容易控制的。当然这个时候我也强烈建议您加上 P 控制，P 控制的话，不是说就单纯的 $P \times \text{偏差}$ 然后得到的值给定 PWM，而是一定要加上一些逻辑控制，不加上逻辑控制而是直接运用比例公式来给定 PWM 值，将会带来很多问题，具体的逻辑控制讲解，在以后我会写出相关的文档，谢谢。

采用增量式编码器为反馈的智能车电机闭环调速系统：

其实借助于我上面举得例子，到这里，您可能已经明白对于智能车来说闭环系统是一个必须的，不可脱离的。当然，您可能已经完全明白智能车电机闭环调速系统怎么做，并且您可能也做的更好，要比我在这里所说的更加的先进更加的有创意更加的有效果。所以在这里我只是简单的说说，谢谢。

首先，我们需要时刻反馈当前的智能车速度给单片机或者 ARM，让系统知道当前速度，有助于控制。

我们先说说为什么我们要反馈车子速度，要知道车子当前的速度？因为我们的 CPU 利用传感器采集道路信息，通过 CPU 收集和处理变化，系统就能知道车子当前的状态，比如是在直到上？还是在进入弯道？还是在弯道内？还是正在出弯道？还是在上坡？还是穿过了起跑线？有了这些信息，我们首先就是对舵机的控制，控制车子根据什么道路情况来进行转弯，转弯的方向，大小，都是靠这些数据来实现，当然当前的速度值也是控制舵机转向大小的一个附加因素，这样更加的有利于我们及时处理不同弯道。知道了这些信息，就跟我们开车一样，我们要转弯了，得感知我们车的速度，怎么感知啊，就靠汽车自带的速度表，这样我们就可以通过控制刹车或者油门还有方向盘来控制车子平稳的转过弯道，不同的弯道，不同的速度，转弯的控制和刹车的控制都是不同的。如果我们不知道当前的速度，那么很可能因为拐弯速度过快，导致翻车，或者侧滑，造成危险。智能车情况跟真车是一样的，我们必须知道车子当前速度，比如我们智能车拐弯安全速度是 3M/S,所谓安全速度就是车子顺利安全且没有较大侧滑过弯时的速度，在进入弯道前我们车子的速度是 4M/S，当传感器感知要进入弯道

了，通过跟安全速度对比，我们发现智能车的速度远大于安全弯道速度，这个时候我们就要刹车，把车子速度快速降低到 3M/S，这样保证我们顺利快速的过弯。在这里您可能问 3M/S 的安全速度怎么来？其实很简单这个速度是靠实验来的，不用什么计算，完全属于经验值，根据跑道的材料、车子机械性能和弯道的角度来决定，所以为什么我们要比赛的时候要试车，试车中一个最重要的环节就是熟悉比赛跑道所用的材料，用于我们实验车子的安全速度等。

对于智能车的速度反馈，在这里我就针对增量式编码器来说，其实码盘啊，霍尔啊，他们原理都一样。光电增量式编码器不同于绝对值编码器，绝对值编码器在编码器转动一周的每个特定位置都有一个确定的值输出，增量式编码器转动一圈，固定输出一定的脉冲，无法确定转动到哪个位置，对外输出一般为方波，这样有助于 MCU 进行数据采集。

有了编码器，我们通过齿轮、皮带等将编码器跟电机或者传动轮进行连接，这样电机转动的时候，就会带动编码器转动，然后编码器就能向外输出方波，接到单片机上，我们就能进行采集编码器脉冲了。有了脉冲，可能您又要问这个脉冲没有规律性，随着转动不停的增加，到底怎么来衡量速度？您这个时候啊，动动手，观察观察数据特点，您会发现，当车速度比较快的时候啊，脉冲数自增的速度比较快，也就是说，转速越快，频率越高，单位时间内的脉冲数就越多，这样立刻有了灵感，我们用定时器设置一个中断，这个中断的时间为 5MS，那么每 5MS 我们就采集编码器一次脉冲值，当采集完毕后，我们就对脉冲数寄存器清零，这样在 5MS 内，得到的脉冲数越多，说明车子速度越快，到这里车速就解决了。到这

里您可能会说：还没完呢，还没转换为速度呢？要把这个脉冲根据距离和时间转换为具体的 XX 米/秒的速度。 您的这个想法很好，但是真的是没有必要，我们根本没有必要浪费时间去计算这个，因为我们本来就是想知道速度，脉冲数的多少已经就能衡量速度大小了，我们何必要很麻烦的转换为具体速度？ 一定时间内的脉冲数已经够了，够我们底层硬件进行判断计算了。 既然我们反馈的数据类型为脉冲数，所以我们所给定的速度也对应的到脉冲数，比如 50 个脉冲代表速度 3M/S，80 个代表 4M/S 的速度，这样通过道路类型给定脉冲数，就能实现电机的闭环控制了，是不是很简单？ 到这里我知道又有人会疑问，50 个脉冲对应 3M/S，80 个脉冲对应 4M/S 是怎么得到的？ 其实这个更简单，很多人做智能车容易把简单问题复杂化，其实很多问题都可以用几行程序解决，比如那个坡道，很多人用了很多复杂方法解决，比如加一个开关啊，加个加速度啊，加个陀螺仪啊，根本没有必要，程序几行就搞定了。具体的脉冲对应速度，您固定给定一个脉冲让车子跑跑到一圈，然后记下时间，跑道实际长度除以时间，就是对应速度，你只是需要大概知道对应就行了，不用一个一个实验，具体的 X 米/秒是宏观上我们感知的，跟单片机没有任何关系。

迷迷糊糊的写了这么多，不知道会不会对初学者有什么帮助，鄙人在这里献丑了，如有不对的地方，您大可批评，但请您理解我知识的欠缺，我会更加努力，为广大客户提供更加周到的服务！如果您在使用我产品上有什么问题，请随时找我，我会第一时间帮您解决，再次感谢支持！祝福比赛成功！