

# 自动控制的故事（全集）

自动控制的科普综述文章少之又少，这一篇是难得好文章。甚至高于科普了。自动化专业的学生，能够把自己学的道理理解到这种程度的，又能占到多少比例呢？第一次看到此文的时候，简直惊为天人，短短数十页的文章，就把自动控制的发展过程及精髓用非常通俗的语言描述出来，其中不发精彩的比喻（如舞龙、李亚普诺夫 大锤等等），对于自动控制人员来说实在是不可多得的宝贵资料。

注：文章写于 2005 年，作者晨枫，是老三届的大学生，具有扎实的理论基础和丰富的现场经验，目前旅居加拿大，一直从事化工控制方面的工作和研究。文章虽然长一些，但非常值得静下心来好好读一读。

写完聚乙烯的故事，就有写自动控制的故事的念头，但一直没有动笔。这个题目太大了，大得都不知道从何说起。既然大家看聚乙烯的故事还有一点意思，四一在催，马鹿要“一”什么的，那就接着写自动控制的故事吧。反正是故事，别太当真，看着好玩就接着看，看着枯燥就拍桌子，看着有错就提个醒，看着糊涂的尽管问，看着不耐烦了呢，也言语一声，我好闭上嘴不再烦人。哈哈。

小时候喜欢看杂书，没什么东西看，不正在文化\_大革命嘛？不过看进去了两个“化”：机械化和自动化。打小就没有弄明白，这机械化和自动化到底有什么差别，机器不是自己就会动的吗？长大了，总算稍微明白了一点，这机械化是力气活，用机器代替人的体力劳动，但还是要人管着的，不然机器是不知道该干什么不该干什么的；这自动化嘛，就是代替人的重复脑力劳动，是用来管机器的。也就是说，自动化是管着机械化的，或者说学自动化的是管着学机械的……啊，不对，不对，哪是哪啊！

有人考证古代就有自动化的实例，但现代意义上的自动控制开始于瓦特的蒸汽机。据说纽考门比瓦特先发明蒸汽机，但是蒸汽机的转速控制问题没有解决，

弄不好转速飞升，机器损坏不说，还可能说大事故。瓦特在蒸汽机的转轴上安了一个小棍，棍的一端和放汽阀连着，放气阀松开来就关闭，转速增加；按下去阀就打开，转速降低；棍的另一端是一个小重锤，棍中间某个地方通过支点和转轴连接。转轴转起来的时候，小棍由于离心力的缘故挥起来。转速太高了，小棍挥会挥得很高，放汽阀就被按下去打开，转速下降；转速太低了，小棍挥不起来，放汽阀就被松开来关闭，转速回升。这样，蒸汽机可以自动保持稳定的转速，即保证安全，又方便使用。也就是因为这个小小的转速调节器，瓦特的名字和工业革命\_命连在一起，而纽考门的名字就要到历史书里去找了。

类似的例子在机械系统里很多，家居必备的抽水马桶是另一个例子。放水冲刷后，水箱里水位降低，浮子随水面下降，进水阀打开。随着水位的升高，进水阀逐渐关闭，直到水位达到规定高度，进水阀完全关闭，水箱的水正好准备下一次使用。这是一个非常简单但非常巧妙的水位控制系统，是一个经典的设计，但不容易用经典的控制理论来分析，不过这是题外话了。

这些机械系统设计巧妙，工作可靠，实在是巧夺天工。但是在实用中，如果每次都需要这样的创造性思维，那太累，最好有一个系统的方法，可以解决“所有”的自动控制问题，这就是控制理论的由来。

从小大人就教我们，走路要看路。为什么呢？要是不看着路，走路走歪了也不知道，结果就是东撞西撞的。要是看着路呢？走歪了，马上就看到，赶紧调整脚步，走回到正道上。这里有自动控制里的第一个重要概念：反馈（feedback）。

反馈是一个过程：

- 1、设定目标，对小朋友走路的例子来说，就是前进的路线。
- 2、测量状态，小朋友的眼睛看着路，就是在测量自己的前进方向。
- 3、将测量到的状态和设定的目标比较，把眼睛看到的前进方向和心里想的前进方向作比较，判断前进方向是否正确；如果不正确，相差有多少。
- 4、调整行动，在心里根据实际前进方向和设定目标的偏差，决定调整的量。
- 5、实际执行，也就是实际挪动脚步，重回正确的前进方向。

在整个走路的过程中，这个反馈过程周而复始，不断进行，这样，小朋友就

不会走得东倒西歪了。但是，这里有一个问题：如果所有的事情都是在瞬时里同时发生的，那这个反馈过程就无法工作。要使反馈工作，一定要有一定的反应时间。还好，世上之事，都有一个过程，这就为反馈赢得了所需要的时间。

小时候，妈妈在锅里蒸东西，蒸好了，从锅里拿出来总是一个麻烦，需要抹布什么的垫着，免得烫手。但是碗和锅的间隙不大，连手带抹布伸进去颇麻烦，我常常不知天高地厚，自告奋勇地徒手把热的碗拿出来。只要动作快，手起碗落，可以不烫手。当然喽，要是捧着热碗再东晃晃，西荡荡，那手上感觉的温度最终会和热碗一样，肯定要把手心、手指烫熟不可的。在从接触碗到皮肤温度和碗表面一样，这里面有一个逐渐升温的过程，这就是动态过程(dynamic process)。这里面有两个东西要注意：一个是升温的过程有多快，另一个是最终的温度可以升到多少。要是知道了这两个参数，同时知道自己的手可以耐受多少温度，理论上可以计算出热的碗在手里可以停留多少时间而不至于烫手。

反馈过程也叫闭环(closed loop)过程。既然有闭环，那就有开环(open loop)。开环就是没有反馈的控制过程，设定一个控制作用，然后就执行，不根据实际测量值进行校正。开环控制只有对简单的过程有效，比如洗衣机和烘干机 按定时控制，到底衣服洗得怎么样，烘得干不干，完全取决于开始时的设定。对于洗衣机、烘干机这样的问题，多设一点时间就是了，稍微浪费一点，但可以保证效果。对于空调机，就不能不顾房间温度，简单地设一个开 10 分钟、关 5 分钟的循环，而应该根据实际温度作闭环控制，否则房间里的温度天知道到底会达到多少。记得 80 年代时，报告文学很流行。徐迟写了一个《哥德巴赫猜想》，于是全国人民都争当科学家。小说家也争着写科学家，成就太小不行，所以来一个语不惊人死不休，某大家写了一个《无反馈快速跟踪》。那时正在大学啃砖头，对这个科学新发现大感兴趣，从头看到尾，也没有看明白到底是怎么无反馈快速跟踪的。现在想想，小说就是小说，不过这无良作家也太扯，无反馈还要跟踪，不看着目标，不看着自己跑哪了，这跟的什么踪啊，这和永动机差不多了，怎么不挑一个好一点的题目，冷聚变什么的，至少在理论上还是可能的。题外话了。

在数学上，动态过程用微分方程描述，反馈过程就是在描述动态过程的微分方程的输入项和输出项之间建立一个关联，这样改变了微分方程本来的性质。自动控制就是在这个反馈和动态过程里做文章的。

房间内的空调是一个简单的控制问题。不过这只是指单一房间，整个高层大楼所有房间的中央空调问题实际上是一个相当复杂的问题，不在这里讨论的范围。夏天了，室内温度设在 28 度，实际温度高于 28 度了，空调机启动致冷，把房间的温度降下来；实际温度低于 28 度了，空调机关闭，让房间温度受环境气温自然升上去。通过这样简单的开关控制，室内温度应该就控制在 28 度。不过这里有一个问题，如果温度高于 28 度一点点，空调机就启动；低于 28 度一点点，空调机就关闭；那如果温度传感器和空调机的开关足够灵敏的话，空调机的开关频率可以无穷高，空调机不断地开开关关，要发神经病了，这对机器不好，在实际上也没有必要。解决的办法是设立一个“死区”（dead band），温度高于 29 度时开机，低于 27 度时关机。注意不要搞反了，否则控制单元要发神经了。

有了一个死区后，室内温度不再可能严格控制在 28 度，而是在 27 到 29 度之间“晃动”。如果环境温度一定，空调机的制冷量一定，室内的升温/降温动态模型已知，可以计算温度“晃动”的周期。不过既然是讲故事，我们就不去费那个事了。

这种开关控制看起来“土”，其实好处不少。对于大部分过程来说，开关控制的精度不高但可以保证稳定，或者说系统输出是“有界”的，也就是说实际测量值一定会被限制在一定的范围，而不可能无限制地发散出去。这种稳定性和一般控制理论里强调的所谓渐进稳定性不同，而是所谓 BIBO 稳定性，前者要求输出最终趋向设定值，后者只要求在有界的输入作用下输出是有界的，BIBO 指 bounded input bounded output。

对于简单的精度要求不高的过程，这种开关控制（或者称继电器控制，relay control，因为最早这种控制方式是用继电器或电磁开关来实现的）就足够了。但是很多时候，这种“毛估估”的控制满足不了要求。汽车在高速公路上行驶，速度设在定速巡航控制，速度飘下去几公里，心里觉得吃亏了，但要是飘上去几公里，被警察抓下来吃一个罚单，这算谁的？

开关控制是不连续控制，控制作用一加就是“全剂量”的，一减也是“全剂量”的，没有中间的过渡。如果空调机的制冷量有三个设定：小、中、大，根据室温和设定的差别来决定到底是用小还是中还是大，那室温的控制精度就可以大大提高，换句话说，温度的“晃动”幅度将大幅度减小。那么，如果空调机有更多的设定，从小小到小中到.....到大大，那控制精度是不是更高呢？是的。既然如此，何不用无级可调的空调机呢？那岂不可以更精确地控制室温了吗？是的。

无级可调或连续可调的空调机可以精确控制温度，但开关控制不能再用了。家用空调机中，连续可调的不占多数，但冲热水淋浴是一个典型的连续控制问题，因为水龙头可以连续调节水的流量。冲淋浴时，假定冷水龙头不变，只调节热水。那温度高了，热水关小一点；温度低了，热水开打一点。换句话说，控制作用应该向减少控制偏差的方向变化，也就是所谓负反馈。控制方向对了，还有一个控制量的问题。温度高了1度，热水该关小多少呢？

经验告诉我们，根据具体的龙头和水压，温度高1度，热水需要关小一定的量，比如说，关小一格。换句话说，控制量和控制偏差成比例关系，这就是经典的比例控制规律：控制量=比例控制增益\*控制偏差，偏差越大，控制量越大。控制偏差就是实际测量值和设定值或目标值之差。在比例控制规律下，偏差反向，控制量也反向。也就是说，如果淋浴水温要求为40度，实际水温高于40度时，热水龙头向关闭的方向变化；实际水温低于40度时，热水龙头向开启的方向变化。

但是比例控制规律并不能保证水温能够精确达到40度。在实际生活中，人们这时对热水龙头作微调，只要水温还不合适，就一点一点地调节，直到水温合适为止。这种只要控制偏差不消失就渐进微调的控制规律，在控制里叫积分控制规律，因为控制量和控制偏差在时间上的累积成正比，其比例因子就称为积分控制增益。工业上常用积分控制增益的倒数，称其为积分时间常数，其物理意义是偏差恒定时，控制量加倍所需的时间。这里要注意的是，控制偏差有正有负，全看实际测量值是大于还是小于设定值，所以只要控制系统是稳定的，也就是实际测量值最终会稳定在设定值上，控制偏差的累积不会是无穷大的。这里

再啰嗦一遍，积分控制的基本作用是消除控制偏差的余差（也叫残差）。

比例和积分控制规律可以应付很大一类控制问题，但不是没有改进余地的。如果水管水温快速变化，人们会根据水温的变化调节热水龙头：水温升高，热水龙头 向关闭方向变化，升温越快，开启越多；水温降低，热水龙头向开启方向变化，降温越快，关闭越多。这就是所谓的微分控制规律，因为控制量和实际测量值的变化 率成正比，其比例因子就称为比例控制增益，工业上也称微分时间常数。微分时间常数没有太特定的物理意义，只是积分叫时间常数，微分也跟着叫了。微分控制的 重点不在实际测量值的具体数值，而在其变化方向和变化速度。微分控制在理论上和实用中有很多优越性，但局限也是明显的。如果测量信号不是很“干净”，时不 时有那么一点不大不小的“毛刺”或扰动，微分控制就会被这些风吹草动搞得方寸大乱，产生很多不必要甚至错误的控制信号。所以工业上对微分控制的使用是很谨慎的。

比例-积分-微分控制规律是工业上最常用的控制规律。人们一般根据比例-积分-微分的英文缩写，将其简称为 **PID** 控制。即使在更为先进的控制规律广泛应用的今天，各种形式的 **PID** 控制仍然在所有控制回路中占 85% 以上。

在 **PID** 控制中，积分控制的特点是：只要还有余差（即残余的控制偏差）存在，积分控制就按部就班地逐渐增加控制作用，直到余差消失。所以积分的效果比较缓慢，除特 殊情况外，作为基本控制作用，缓不救急。微分控制的特点是：尽管实际测量值还比设定值低，但其快速上扬的冲势需要及早加以抑制，否则，等到实际值超过设定 值再作反应就晚了，这就是微分控制施展身手的地方了。作为基本控制使用，微分控制只看趋势，不看具体数值所在，所以最理想的情况也就是把实际值稳定下来， 但稳定在什么地方就要看你的运气了，所以微分控制也不能作为基本控制作用。比例控制没有这些问题，比例控制的反应快，稳定性好，是最基本的控制作用，是“皮”，积分、微分控制是对比例控制起增强作用的，极少单独使用，所以是“毛”。在实际使用中比例和积分一般一起使用，比例承担主要的控制作用，积分帮助 消除余差。微分只有在被控对象反应迟缓，需要在开始有所反应时，及早补偿，才予以采用。只用比例和微分的情况很少见。

连续控制的精度是开关控制所不可比拟的，但连续控制的高精度也是有代价的，这就是稳定性问题。控制增益决定了控制作用对偏差的灵敏度。既然增益决定了控制的灵敏度，那么越灵敏岂不越好？非也。还是用汽车的定速巡航控制做例子。速度低一点，油门加一点，速度低更多，油门加更多，速度高上去当然就反过来。但是 如果速度低一点，油门就加很多，速度更低，油门狂加，这样速度不但不能稳定在要求的设定值上，还可能失控。这就是不稳定。所以控制增益的设定是有讲究的。 在生活中也有类似的例子。国民经济过热，需要经济调整，但调整过火，就要造成“硬着陆”，引起衰退；衰退时需要刺激，同样，刺激过火，会造成“虚假繁荣”。要达成“软着陆”，经济调整的措施需要恰到好处。这也是一个经济动态系统的稳定性问题。

实际中到底多少增益才是最合适的，理论上有很多计算方法，但实用中一般是靠经验和调试来摸索最佳增益，业内行话叫参数整定。如果系统响应在控制作用后面拖拖沓沓，大幅度振荡的话，那一般是积分太过；如果系统响应非常神经质，动不动就打摆子，呈现高频小幅度振荡的话，那一般是微分有点过分。中频振荡当然 就是比例的问题了。不过各个系统的频率都是不一样的，到底什么算高频，什么算低频，这个几句话说不清楚，应了毛主席那句话：“具体情况具体分析”，所以就 打一个哈哈了。

再具体说起来，参数整定有两个路子。一是首先调试比例增益以保证基本的稳定性，然后加必要的积分以消除余差，只有在最必要的情况下，比如反映迟缓的温度过程或容量极大的液位过程，测量噪声很低，才加一点微分。这是“学院派”的路子，在大部分情况下很有效。但是工业界有一个“歪路子”：用非常小的比例作用，但大大强化积分作用。这个方法是完全违背控制理论的分析的，但在实际中却是行之有效，原因在于测量噪声严重，或系统反应过敏时，积分为主的控制规律动作比较缓和，不易激励出不稳定的因素，尤其是不确定性比较高的高频部分，这也是(敏感词 1068)“稳定压倒一切”的初衷吧。

在很多情况下，在初始 PID 参数整定之后，只要系统没有出现不稳定或性能显著退化，一般不会去重新整定。但是要是系统不稳定了怎么办呢？由于大部

分实际系统都是开环稳定的，也就是说，只要控制作用恒定不变，系统响应最终应该稳定在一个数值，尽管可能不是设定值，所以对付不稳定的第一个动作都是把比例增益减小，根据实际情况，减小 1/3、1/2 甚至更多，同时加大积分时间常数，常常成倍地加，再就是减小甚至取消微分控制作用。如果有前馈控制，适当减小前馈增益也是有用的。在实际中，系统性能不会莫名其妙地突然变坏，上述“救火”式重新整定常常是临时性的，等生产过程中的机械或原料问题消除后，参数还是要设回原来的数值，否则系统性能会太过“懒散”。

对于新工厂，系统还没有投运，没法根据实际响应来整定，一般先估计一个初始参数，在系统投运的过程中，对控制回路逐个整定。我自己的经验是，对于一般的流量回路，比例定在 0.5 左右，积分大约 1 分钟，微分为 0，这个组合一般不致于一上来就出大问题。温度回路可以从 2、5、0.05 开始，液位回路从 5、10、0 开始，气相压力回路从 10、20、0 开始。既然这些都是凭经验的估计，那当然要具体情况具体分析，不可能“放之四海而皆准”。

微分一般用于反应迟缓的系统，但是事情总有一些例外。我就遇到过一个小小的冷凝液罐，直径才两英尺，长不过 5 英尺，但是流量倒要 8-12 吨/小时，一有风吹草动，液位变化非常迅速，不管比例、积分怎么调，液位很难稳定下来，常常是控制阀刚开始反应，液位已经到顶或到底了。最后加了 0.05 的微分，液位一开始变化，控制阀就开始抑制，反而稳定下来了。这和常规的参数整定的路子背道而驰，但在这个情况下，反而是“唯一”的选择，因为测量值和控制阀的饱和变成稳定性主要的问题了。

对工业界以积分为主导控制作用的做法再啰嗦几句。学术上，控制的稳定性基本就是渐近稳定性，BIBO 稳定性是没有办法证明渐近稳定性时的“退而求其次”的东西，不怎么上台面的。但是工业界里的稳定性有两个看起来相似、实质上不尽相同的方面：一个当然是渐近稳定性，另一个则是稳定性，但不一定向设定值收敛，或者说稳定性比收敛性优先这样一个情况。具体来说，就是需要系统稳定在一个值上，不要动来动去，但是不是在设定值并不是太重要，只要不是太离谱就行。例子有很多，比如反应器的压力是一个重要参数，反应器不稳定，



原料进料比例就乱套，催化剂进料也不稳定，反应就不稳定，但是反应器的压力到底是 10 个大气压还是 12 个大气压，并没有太大的关系，只要慢慢地但是稳定地向设定值移动就足够了。这是控制理论里比较少涉及的一个情况，这也是工业上时常采用积分主导的控制的一个重要原因

前面说到系统的频率，本来也就是系统响应持续振荡时的频率，但是控制领域里有三拨人在捣腾：一拨是以机电类动力学系统为特色的电工出身，包括航空、机器人等，一拨是以连续过程为特色的化工出身的，包冶金、造纸等，还有一拨是以微分方程稳定性为特色的应用数学出身的。在瓦特和抽水马桶的年代里，各打各的山头，井水不犯河水，倒也太平。但控制从艺术上升为理论后，总有人喜欢“统一”，电工帮抢了先，好端端的控制理论里被塞进了电工里的频率。童子们哪，那哪是频率啊，那是……复频率。既然那些变态的电工帮（啊耶，这下鹿蹄真的要来了）能折腾出虚功率，那他们也能折腾出复频率来，他们自虐倒也算了，只是苦了我等无辜之众，被迫受此精神折磨。

事情的缘由是系统的稳定性。前面提到，PID 的参数如果设得不好，系统可能不稳定。除了摸索，有没有办法从理论上计算出合适的 PID 参数呢？前面也提到，动态过程可以用微分方程描述，其实在 PID 的阶段，这只是微分方程中很狭窄的一支：单变量线性常微分方程。要是还记得大一高数，一定还记得线形常微的解，除了分离变量法什么的，如果自变量时间用  $t$  表示的话，最常用的求解还是把  $\exp(\lambda t)$  代入微分方程，然后解已经变成  $\lambda$  的代数方程的特征方程，解出来的解可以是实数，也可以是复数，是复数的话，就要用三角函数展开了（怎么样，大一噩梦的感觉找回来一点没有？）。只要实根为负，那微分方程就是稳定的，因为负的指数项最终向零收敛，复根到底多少就无所谓了，对稳定性没有影响。但是，这么求解分析起来还是不容易，还是超不出“具体情况具体分析”，难以得出一般的结论。

法国人以好色、好吃出名，但是他们食色性也之后，还不老实，其中一个叫拉普拉斯的家伙，捣鼓出什么拉普拉斯变换，把常微分方程变成  $s$  的多项式。然

后那帮电工的家伙们，喜欢自虐，往  $s$  里塞  $j\omega$ ，就是那个复频率，整出一个变态的频率分析，用来分析系统的稳定性。不过说变态，也不完全公平，在没有计算机的年代，各种图表是最有效的分析方法，还美其名曰“几何分析”。频率分析也不例外。美国佬 Evans 搞出一个根轨迹（root locus），思路倒是满有意思的。他用增益作自变量，将系统的根（不管实的虚的）在复平面上画出轨迹来，要是轨迹在左半平面打转转，那就是实根为负，就是稳定的。再深究下去，系统响应的临界频率之类也可以计算出来。最大的好处是，对于常见的系统，可以给出一套作图规则来，熟练的大牛、小牛、公牛、母牛们，眼睛一瞄，随手就可以画出根轨迹来，然后就可以告诉你，增益变化多多少，系统开始振荡，再增加多少，系统会不稳定，云云。

根轨迹还是比较客气的，还有更变态的奈奎斯特、伯德和尼科尔斯法，想想脑子都大。都是叫那帮电工分子害的。时至今日，计算机分析已经很普及了，但是古典的图示分析还是有经久不衰的魅力，就是因为图示分析不光告诉你系统是稳定还是不稳定，以及其他一些动态响应的参数，图示分析还可以定性地告诉你增益变化甚至系统参数变化引起的闭环性能变化。咦，刚才还不是在说人家变态吗？呃，变态也有变态的魅力不是？哈哈。

以频率分析（也称频域分析）为特色的控制理论称为经典控制理论。经典控制理论可以把系统的稳定性分析得天花乱坠，但有两个前提：一、要已知被控对象的数学模型，这在实际中不容易得到；二、被控对象的数学模型不会改变或漂移，这在实际中更难做到。对简单过程建立微分方程是可能的，但简单过程的控制不麻烦，经验法参数整定就搞定了，不需要费那个麻烦，而真正需要理论计算帮忙的回路，建立模型太困难，或者模型本身的不确定性很高，使得理论分析失去意义。经典控制理论在机械、航空、电机中还是有成功的应用，毕竟从  $F=ma$  出发，可以建立“所有”的机械系统的动力学模型，铁疙瘩的重量又不会莫名其妙地改变，主要环境参数都可以测量，但是经典控制理论至少在化工控制中实用成功的例子实在是凤毛麟角，给你一个 50 块塔板的精馏塔，一个气相进料，一个液相进料，塔顶、塔底出料加一个侧线出料，塔顶风冷冷凝器，塔

底再沸器加一个中间再沸器，你就慢慢建模去吧，等九牛二虎把模型建立起来了，风冷冷凝器受风霜雨雪的影响，再沸器的高压蒸汽的压力受友邻装置的影响，气相进料的温度和饱和度受上游装置的影响而改变，液相进料的混合组分受上游装置的影响而改变，但组分无法及时测量（在线气相色谱分析结果要 45 分钟才能出来），动态特性全变了。

老家伙歌德两百年前就说了，理论是灰色的，生命之树常青。我们知道马鹿喜欢金光的或者银光的，至少也要红的，不过只好将就啦，青绿地干活。在实用中，PID 有很多表兄弟，帮着大表哥一块打天下。

比例控制的特点是：偏差大，控制作用就大。但在实际中有时还嫌不够，最好偏差大的时候，比例增益也大，进一步加强对大偏差的矫正作用，及早把系统拉回到设定值附近；偏差小的时候，当然就不用那么急吼吼，慢慢来就行，所以增益小一点，加强稳定性。这就是双增益 PID（也叫双模式 PID）的起源。想想也对，高射炮瞄准敌机是一个控制问题。如果炮管还指向离目标很远的角度，那应该先尽快地把炮管转到目标角度附近，动作猛一点才好；但炮管指向已经目标很近了，就要再慢慢地精细瞄准。工业上也有很多类似的问题。双增益 PID 的一个特例是死区 PID（PID with dead band），小偏差时的增益为零，也就是说，测量值和设定值相差不大的时候，就随他去，不用控制。这在大型缓冲容器的液位控制里用得很多。本来缓冲容器就是缓冲流量变化的，液位到底控制在什么地方并不紧要，只要不是太高或太低就行。但是，从缓冲容器流向下游装置的流量要尽可能稳定，否则下游装置会受到不必要的扰动。死区 PID 对这样的控制问题是最合适的。但是天下没有免费的午餐。死区 PID 的前提是液位在一般情况下会“自动”稳定在死区内，如果死区设置不当，或系统经常受到大幅度的扰动，死区内的“无控”状态会导致液位不受限制地向死区边界“挺进”，最后进入“受控”区时，控制作用过火，液位向相反方向不受限制地“挺进”，最后的结果是液位永远在死区的两端振荡，而永远不会稳定下来，业内叫 hunting（打猎？打什么？打鹿？）。双增益 PID 也有同样的问题，只是比死区 PID 好一些，毕竟只有“强控制”和“弱控制”的差别，而没有“无控区”。在实用中，双增益的内外增益差别小于 2:1 没有多大意义，大于 5:1 就要注意上述的持续振荡或 hunting 的

问题。

双增益或死区 PID 的问题在于增益的变化是不连续的，控制作用在死区边界上有一个突然的变化，容易诱发系统的不利响应，平方误差 PID 就没有这个问题。误差一经平方，控制量对误差的曲线就成了抛物线，同样达到“小偏差小增益、大偏差大增益”的效果，还没有和突然的不连续的增益变化。但是误差平方有两个问题：一是误差接近于零的时候，增益也接近于零，回到上面死区 PID 的问题；二是很难控制抛物线的具体形状，或者说，很难制定增益在什么地方拐弯。对于第一个问题，可以在误差平方 PID 上加一个基本的线性 PID，是零误差是增益不为零；对于后一个问题，就要用另外的模块计算一个连续变化的增益了。具体细节比较琐碎，将偏差送入一个分段线性化（也就是折线啦）的计算单元，然后将计算结果作为比例增益输出到 PID 控制器，折线的水平段就对应于不同的增益，而连接不同的水平段的斜线就对应于增益的连续变化。通过设置水平段和斜线段的折点，可以任意调整变增益的曲线。要是“野心”大一点，再加几个计算单元，可以做出不对称的增益，也就是升温时增益低一点，降温时增益高一点，以处理加热过程中常见的升温快、降温慢的问题。

双增益或误差平方都是在比例增益上作文章，同样的勾当也可以用在积分和微分上。更极端的一种 PID 规律叫积分分离 PID，其思路是这样的：比例控制的稳定性好，响应快，所以偏差大的时候，把 PID 中的积分关闭掉；偏差小的时候，精细调整、消除余差是主要问题，所以减弱甚至关闭比例作用，而积分作用切入控制。概念是好的，但具体实施的时候，有很多无扰动切换的问题。

这些变态的 PID 在理论上很难分析系统的稳定性，但在实用中解决了很多困难的问题。大言不惭一句，这些 PID 本人在实际中都用过。

复杂结构 PID 打仗时，如果敌人太顽固，要么换更大的炮，把敌人轰倒；要么采用更巧妙的战术，把敌人晕倒。控制也是一样，单回路 PID 难以解决的问题，常常可以通过更巧妙的回路结构来解决。

单一的 PID 回路当然可以实现扰动抑制，但要是主要扰动在回路中，而且是明确的，加一个内回路作帮手是一个很不错的主意。还记得洗热水澡的例子吗？

要是热水压力不稳定，老是要为这个而调整 热水龙头，那很麻烦。要是有一个专门负责根据热水压力调节热水流量，把热水压力稳定下来，而且稳定在标定值，那洗澡的时候，水温就容易控制多了，只要告诉那个人现在需要多少热水流量，而不必烦心热水压力对热水流量的影响。这个负责热水流量的控制回路就是内回路，也叫副回路，而洗澡的温度就是外回路，也叫主回路，当然是主回路指挥副回路，就像自动化指挥机械化、学自控的人指挥学机电的人……打住打住，再扯远了要挨鹿蹄了，或者马蹄、牛蹄、驴蹄……。这种主回路套副回路的结构叫串级控制（cascade control），曾经是单回路 PID 后工业上第一种“先进过程控制”，现在串级已经用得很多了，也不再有人叫它“先进过程控制”了。串级控制最主要的功用是抑制回路内的扰动，增强总体控制性能。不过串级也不能乱用。如果主回路和副回路的相应速度差不多，或者主回路的相应速度甚至慢于副回路（通过变态的调试是可以做到的），这样的串级要出问题。理论上可以用共振频率什么的分析，但是不用费那个事，用膝盖想想就知道，一个急性子的头儿把一个温吞水的下属指挥得团团转，结果只能是大家都精疲力竭，事情还办砸了。相反，一个镇定自若的头儿指挥一个手脚麻利的下属，那事情肯定办得好。

如果主要扰动在回路以外，但是可以预知，那就要用另一个办法，就是马鹿前面说到的前馈了。还是用洗热水澡的例子。如果冷水管和同一个水房的抽水马桶功用，你在洗澡，别人一抽水，那你就变煮熟的龙虾了（本想说猴子 PP 的，但是那个不雅，我们要五讲四美不是？）。这个时候，要使那个人在抽水的同时告诉你一声，你算好时间，算好量，猛减热水，那温度还是可以大体不变的。这就是所谓前馈控制（feed-forward control）。前馈控制有两个要紧的东西：一是定量的扰动对被控变量的影响，也就是所谓前馈增益；二是扰动的动态，别人抽水到洗澡龙头的水温变热，这里面有一个过程，不是立时三刻的。如果可以精确知道这两样东西，那前馈补偿可以把可测扰动完全补偿掉。但实际上没有精确知道的事情，要是指望前馈来完全补偿，弄巧成拙是肯定的。所以前馈通常和反馈一起用，也就是在 PID 回路上再加一个前馈。一般也只用静态前馈，也就是只补偿扰动对被控变量的静态影响，而忽略扰动的动态因素，主要是为

了静态前馈已经把前馈 80% 的好处发掘出来了，动态前馈既复杂又不可靠，在 PID 回路里很少有人用。理论上的前馈都是在 PID 的控制作用上再加一个前馈作用，实际上也可以乘一个控制作用。乘法前馈的作用太猛，我从来没有用过，一般都是用加法。在实施中，前馈是和扰动的变化（也就是增量）成比例的，所以扰动变量不变了，前馈作用就消失，否则，整定前馈控制增益会对 PID 主回路造成扰动。前馈增益可以根据粗略计算得到，比如说，抽水的量会造成温度下降多少、需要调整多少热水流量才能维持温度，这不难从热量平衡算出来。不想费这个事的话，也可以从历史数据中推算。一般算出来一个前馈增益后，打上 7 折甚至 5 折再用，保险一点，不要矫枉过正。

前馈作用一般是用作辅助控制作用的，但是在特殊情况下，前馈也可以作为“预加载”（pre-loading）作为基准控制作用。比如说，在一个高压系统的启动过程中，压力可以从静止状态的常压很快地升到很高的压力。高压系统不容许阀门大幅度运动，所以控制增益都比较低，但是这样一来，启动升压过程中，压力控制的反应就十分迟缓，容易造成压力过高。这时用压缩机的转速或高压进料的流量作前馈，将压力控制阀“预先”放到大概的位置，然后再用反馈慢慢调节，就可以解决这个问题。

有点搞笑兄在前面提到用单个阀门有时难以控制大范围变化的流量，这是一个很实际的问题。工业阀门一般 turn down 只有 10: 1，也就是说，如果这个阀门的最大流量是 100 吨/小时的话，低于 10 吨/小时就难以控制了，当然，高于 90 吨/小时也几近失去控制。所以，要真的保证 0-100 的精确控制，需要将一个大阀和一个小阀并列，小阀负责小流量时的精确控制，大阀负责大流量时的精确控制，这就是所谓的分程控制（split range control）。分程控制时，小阀首先打开，超过小阀最大流量是小阀就固定在全开位置，大阀开始打开，接过控制。这是开-开型分程控制。也有关-开型分程控制，比如反应器夹套温度控制，随温度逐渐上升，冷却水逐渐关闭，直到冷却水全关，加热蒸气开始打开。分程控制当然不一定只有两截，三截甚至更多都是可以的，道理都一样。分程控制的问题在于不同阀门的交接点。阀门在特别小的开度时，控制非常不灵敏，前面说到

的 10: 1 也是这个道理。所以实用中，开-开型 分程控制常常在交接点附近有一段重叠，也就是小阀快要全开但还没有全开时，大阀已经开始动作，这样，到小阀全开、不能再动弹时，大阀已经进入有效控制范围。关-开型分程控制常常在交接点设置一个死区，避免出现两个阀都有一点点开度的情况。分程控制的交接点的设置有一点讲究，应该根据阀的大小。比如 A 阀比 B 阀大一倍，那分程点应该设在 1/3 先开 B 阀，而不是懒汉做法的 1/2。

很多过程参数都是可以测量的，但也有很多参数是没法直接测量的，这时，如果能够通过别的可以测量的过程参数来间接计算真正需要控制的参数，这就是所谓的推断控制（inferential control）。比如精馏塔顶的产品纯度可以用气相色谱（gas chromatograph, GC）来测量，但结果要等 40 分钟才能出来，用来做实时控制，黄花菜都凉了。推断控制是和“软传感器”（soft sensor）的概念紧密相连的。对精馏塔塔顶纯度这个例子来说，可以用纯度和塔顶温度、压力作一个数学模型，用可以测量的温度和压力，间接计算出纯度。在计算机控制普及的今天，这是很容易实现的，但是在很多地方，推断控制仍然被看成很神秘的东西，悲哀。

有的时候，对同一个变量有不只一个控制手段。比如说，风冷器有风扇的转速可以调节，也有百叶窗的开度可以调节。风扇转速的效果快，控制精确；百叶窗开度的效果猛，不容易掌握，但有利于节能。所以，可以用风扇的快速响应来控制温度，但是用百叶窗开度来通过温度间接地缓慢地影响风扇的转速，使风扇转速回到最经济的设定。当然百叶窗开度的控制回路必须要比风扇转速的控制回路整定得慢得多，一般是缓慢的纯积分控制，否则两人要打架。由于这相当于控制风扇转速的“阀位”，工业上称其为阀位控制（valve position control）。这个阀位控制也可以变一变，风扇转速高于某一数值（比如 80% 的最大转速）时，把百叶窗开大一格，还是高就继续开大；风扇转速低于某一数值（比如低于 20% 最大转速）时，把百叶窗关小一格。这实际上是一个单向的积分作用，不同的地方有两点：

一、有两个设定值，由风扇转速是高还是低而定

二、积分作用只有在风扇转速在这两个“极限”的外面起作用，在里面时，百叶窗的开度不变。

这样，风扇转速不必回到一个特定值，而是可以在一个范围内浮动。

另外一个两个控制器“竞争”一个控制阀的情况是选择性控制（**override control** 或 **selective control**）。举个例子，锅炉的温度由燃料流量控制，温度高了，燃料流量就减下来，但是燃料流量低到燃料管路压力低于炉膛压力，那要出现危险的回火，所以，这时，燃料管路压力就要接管控制，而牺牲炉膛温度。换句话说，正常时候，炉膛温度控制起作用，燃料管路压力低于一定数值时，燃料管路压力控制器作用。在实施时，就是炉膛温度控制器和燃料管路压力控制器的输出都接到一个高选器，然后搞选器的输出接到实际的燃料阀。这个概念很清楚，但是初次接触选择性控制的人，常常容易被高选还是低选搞糊涂，明明是压力太低，怎么是高选呢？其实，只要记住高选还是低选是从阀门这一头看的，和温度、压力的高度没有关系。如果“非常”变量超过界限了，你要阀门打开，那就是高选；你要阀门关闭，那就是低选。

PID 从二、三十年到开始在工业界广泛应用，戏法变了几十年，也该换换花样了。PID 说一千道一万，还是经典控制理论的产物。50-60 年代时，什么 都要现代派，建筑从经典的柱式、比例、细节的象征意义，变到“形式服从功能”的钢架玻璃盒子；汽车从用机器牵引的马车，变到流线型的钢铁的艺术；控制理论 也要紧跟形势，要现代化。这不，美国佬卡尔曼隆重推出……现代控制理论。

都看过舞龙吧？一个张牙舞爪的龙头气咻咻地追逐着一个大绣球，龙身子扭来扭去，还时不时跳跃那么一两下。中国春节没有舞龙，就和洋人的圣诞节没有圣诞老人一样不可思议。想象一下，如果这是一条看不见的盲龙，只能通过一个人在龙尾巴后面指挥龙尾巴，然后再通过龙身体里的人一个接一个地传递控制指令，最后使龙头咬住绣球。这 显然是一个动态系统，龙身越长，人越多，动态响应越迟缓。如果只看龙头的位置，只操控龙尾巴，而忽略龙身子的动态，那就是所谓的输入-输出系统。经典控制 理论就是建立在输入-输出系统的基础上的。对于很多常见的应用，这就足够了。



但是卡尔曼不满足于“足够”。龙头 当然要看住，龙尾巴当然要捏住，但龙身体为什么就要忽略呢？要是能够看住龙身体，甚至操纵龙身体，也就是说，不光要控制龙尾巴，控制指令还要直接传到龙身体里的那些人，那岂不更好？这就是状态空间的概念：将一个系统分解为输入、输出和状态。输出本身也是一个状态，或者是状态的一个组合。在数学上，卡尔曼的状态空间方法就是将一个高阶微分方程分解成一个联立的一阶微分方程组，这样可以使用很多线性代数的工具，在表述上也比较简洁、明了。

卡尔曼是一个数学家。数学家的想法就是和工程师不一样。工程师脑子里转的第一个念头就是“我怎么控制这劳什子？增益多少？控制器结构是什么样的？”数学家想的却是什么解的存在性、唯一性之类虚头八脑的东西。不过呢，这么说数学家也不公平。好多时候，工程师凭想象和“实干”，辛苦了半天，发现得出的结果完全不合情理，这时才想起那些“性”（不要想歪了啊，嘿嘿），原来那些存在性、唯一性什么的还是有用的。

还是回过来看这条龙。现在，龙头、龙尾巴、龙身体都要看，不光要看，还要直接操控龙头到龙尾的每一个人。但是，这龙不是想看就看得的，不是想舞就舞得的。说到“看”，直接能够测量/观测的状态在实际上是不多的，所谓看，实际上是估算。要是知道龙身体有多少节（就是有多少个人在下面撑着啦），龙身体的弹性/韧性有多少，那么捏住龙尾巴抖一抖，再看看龙头在哪里，是可以估算出龙身体每一节的位置的，这叫状态观测。那么，要是这龙中间有几位童子开小差，手不好好拉住，那再捏住龙尾巴乱抖也没用，这时系统中的部分状态就是不可观测的。如果你一声令下，部分童子充耳不闻，那这些状态就是不可控制的。卡尔曼从数学上推导出不可控和不可观的条件，在根本上解决了什么时候才不是瞎耽误工夫的问题。这是控制理论的一个重要里程碑。

再来看这条龙。如果要看这条龙整齐不整齐，排成纵列的容易看清楚；如果要清点人数，看每一个人的动作，排成横列的容易看清楚。但是不管怎么排，这条龙还是这条龙，只是看的角度不同。那时候中国人的春节舞龙还没有在美国的中国城里闹腾起来，不知道卡尔曼有没有看到过舞龙，反正他把数学上的线性

变换和线性空间 的理论搬到控制里面，从此，搞控制的人有了工具，一个系统横着看不顺眼的话，可以竖着看，因为不管怎么看，系统的本质是一样的。但是不同的角度有不同的用处，有的角度设计控制器容易一点，有的角度分析系统的稳定性容易一点，诸如此类，在控制理论里就叫这个那个“标准型”。这是控制理论的又一个里程碑。

观测状态的目的最终还是控制。只用输出的反馈叫输出反馈，经典控制理论里的反馈都可以归到输出反馈里，但是用状态进行反馈的就叫状态反馈了。输出反馈 对常见系统已经很有效了，但状态反馈要猛得多。你想想，一个系统的所有状态都被牢牢地瞄住，所有状态都乖乖地听从调遣，那是何等的威风？台商的大妈们的最高境界呀。

尽管学控制的人都要学现代控制理论，但大多数人记得卡尔曼还是因为那个卡尔曼滤波器（Kalman Filter）。说它是滤波器，其实是一个状态观测器（state observer），用来从输入和输出“重构”系统的状态。这重构听着玄妙，其实不复杂。不是有系统的数学模型吗？只要模型精确，给它和真实系统一样的输入，它不就乖乖地把系统状态给计算出来了吗？且慢：微分方程的解不光由微分方程本身决定，还有一个初始条件，要是初始条件不对，微分方程的解的形式是正确的，但是数值永远差一拍。卡尔曼在系统模型的微分方程后再加了一个尾巴，把实际系统输出和模型计算的理论输出相比较，再乘上一个比例因子，形成一个实际上的状态反馈，把状态重构的偏差渐进地消除，解决了初始条件和其他的系统误差问题。卡尔曼滤波器最精妙之处，在于卡尔曼推导出一个系统的方法，可以考虑进测量噪声和系统本身的随机噪声，根据信噪比来决定上述比例因子的大小。这个构型其实不是卡尔曼的独创，隆伯格（Luenburg）也得出了类似的结构，但是从系统稳定性角度出发，来决定比例因子。同样的结构大量用于各种“预测-校正”模型结构，在工业上也得到很多应用，比如聚合反应器的分子重量分布可以用反应器的温度、进料配比、催化剂等来间接计算，但不够精确，也无法把林林总总的无法测量的干扰因素统统包括进数学模型里，这时用实验室测定的真实值来定期校正，就可以结合数学模型及时的特点和实验室结果精确的特点，

满足实时控制的要求，这或许可以算静态的卡尔曼滤波器吧。卡尔曼滤波器最早的应用还是在雷达上。所谓边扫描边跟踪，就是用卡尔曼滤波器估计敌机的位置，再由雷达的间隙扫描结果来实际校正。实际应用中还有一个典型的问题：有时候，对同一个变量可以有好几个测量值可用，比如有的比较直接但不精确，有的是间接的估算，有很大的滞后但精确度高，这时可以用卡尔曼滤波器把不同来源的数据按不同的信噪比加权“整合”起来，也算是民用版的“传感器融合”（sensor fusion）吧。

除了卡尔曼滤波器外，卡尔曼的理论在实际中用得不多，但是卡尔曼的理论在理论上建立了一个出色的框架，对理解和研究控制问题有极大的作用。

顺便说一句，卡尔曼的理论基本局限于线形系统，也就是说，十块大洋买一袋米，二十块大洋就买两袋米，都是成比例的。实际系统中有很多非线性的，两千块大洋还能买两百袋米，但两千万大洋就要看米仓有没有货了，市场涨不涨价了，不是钱越多，买的米越多，有一个非线性的问题。非线性的问题研究起来要复杂得多。实际系统还有其他特性，有的是所谓时变系统，像宇宙火箭，其质量随时间和燃料的消耗而变，系统特性当然也就变了。很多问题都是多变量的，像汽车转弯，不光方向盘是一个输入，油门和刹车也是输入变量。但是，状态空间的理论在数学表述上为线性、非线性、单变量、多变量、时变、时不变系统提供了一个统一的框架，这是卡尔曼最大的贡献。

=====

前面说到，搞控制有三拨人：电工出身的，化工出身的，和应用数学出身的。在卡尔曼之前，电工出身的占主导地位，数学家们好在象牙塔里打转转，化工出身则还对控制理论懵里懵懂，还在“实干”呢。卡尔曼之后，一大批数学出身的人，利用对数学工具的熟悉，转攻控制理论。一时间，控制理论的数学化似乎成了“天下大势，顺我者昌，逆我者亡”了。在状态空间的框架下，多变量没有太多的问题好研究，于是最优化成为控制理论的新时尚。

对于一根给定的曲线，求一阶导数为零的点，就是这个曲线的极点；在对这一极点球二阶导数，大于零就是最小点，小于零就是最大点。这时牛顿老爷子

就整明白的东东，现在高中或大一人人都学 过的东西。但是动态系统是一个微分方程，对微分方程求一阶导数为零，就导致变分法和所谓欧拉方程。但这个东西用起来不方便。实际的最优控制不大直接使用变 分。

俄罗斯是一个奇怪的地方。老\_毛子们要么蔫蔫的，要么疯狂的。俄罗斯的悲剧电影看得你也郁闷得想去自杀。但是老\_毛子要是搭错筋整出一个喜剧呢？那你要么跟着疯狂，要么被逼疯狂。就是这么一个地方，除了无数托尔斯泰、柴可夫斯基、普希金、屠格涅夫等文艺巨璧外，俄罗斯也盛产数学家，其中两个是庞特 里亚京和哈里学控制的人老惦记着的李亚普诺夫。

庞特里亚京的极大值原理听起来吓人，其实说白了很简单。看见那山吗？山顶就是最高点（切，这还用你 说吗？）；看见那山坡吗？要是在山腰划一道线，从山下往上爬，尽管山坡还在继续往上延伸，但是到线为止，不得逾越，那山腰上那道三八线就是最高点（切，这 还用说？）。这就是庞特里亚京的极大值原理。当然啦，庞特里亚京是用精巧、深奥的数学语言表述的，要不然他在数学界里也别混了。不过呢，意思就是这么一个 意思。

庞特里亚京极大值原理的一个典型应用就是所谓最速控制问题，或者叫时间最优控制（time optimal control）问题，简单 地说，就是给定最大马力和最大刹车功率，怎么开汽车能够最快地从 A 点开到 B 点（什么转弯、上下坡、红绿灯，这种琐碎的事情也要拿来烦人？一点品味都没有！）。你可以用优美但繁琐的数学求证，或者用膝盖想想，最快的方法，就是一上来就加足马力，全速前进；然后在不到终点的某一地点，全力刹车，使慢下来的 汽车在到达终点时正好停下来。这时最快的方法，不可能比这更快了。稍微发挥一点想象力，可以想象“梆”的一下，控制量的油门板一脚到底，再是“梆”的一 下，刹车板一脚到底，控制任务就完成了。所以最速控制也叫“梆-梆”控制（Bang-Bang control）。

最速控制在理论上是一个很有趣 的问题，解法也是简洁、优美，但在实际中直接使用的例子实在是凤毛麟角，一般都是开始时用“梆-梆”，或者匀速上升到最大控制，以缓和控制的冲击力；到终 点附近时，改用 PID 作闭环微调，以

克服“梆-梆”的系统模型误差十分敏感的缺点。电梯控制就是这样一个例子。从一楼到四楼，电动机很快匀速上升到最高转速，一过三楼，电动机就匀速下降到较低的转速，然后根据电梯实际位置和楼面之差，有控制地减速，直至停下来。要是控制参数调得好的话，一下子就稳稳当当地停下来；要是调的不够好，会在停下来之前上下晃动几下。

最速控制问题是较早的最优控制问题，它提供了一个很有趣的思路，但这颗树上开花结果不多。相比之下，最优控制的另外一支枝繁叶茂，有生气得多了。这一支就是线型二次型最优控制（linear quadratic control）。数学是有趣的，但数学也是盲目的。在数学上，最优化问题就是一个在曲面上寻找凸点的问题，只要你能把一个物理问题表述成一个曲面，数学是不理会姓无姓资的。既然如此，控制偏差的平方在时间上的累积就是很自然的选择，二次型就是平方在线性代数里的说法。线型系统的偏差平方有很好的性质，这山峰是一个馒头山，没有悬崖峭壁，没有沟坎，容易爬；一山只有一峰，不用担心找错地方。不过这山峰不能只包含控制偏差，还要包含控制量，原因有三个：

1、如果不包括控制量，那最优控制的解是没有意义的，因为无穷大的控制量可以使累计平方偏差为最小，但无穷大的控制量是不现实的。

2、控制量的大小通常和能量、物料的消耗连在一起，实际控制问题一般是“在最小能量、物料消耗小达到最高的控制精度”，所以在“山峰”中同时包含控制偏差和控制量是很自然的

3、系统模型总是有误差的，误差“总是”在高频、大幅度控制作用下最突出，所以为了减低系统对模型误差的敏感性，也有必要限制控制量的大小。

所以线性二次型最优控制的“目标函数”（也就是定义山峰形状的数学表述）是一个控制偏差和控制量各自平方的加权求和的积分。积分当然就是“在时间上的累积”了，加权求和其实就是在控制偏差的平方项和控制量的平方项前分别乘以比例因子，然后再相加。两个比例因子的相对大小决定了谁更重要。运用矩阵微分和线型代数工具，不难导出线性二次型控制律——一个基本的状态反馈控制律！只是反馈增益矩阵是按最优化的要求计算出来的。

线型二次型最优控制开创了一整个新的控制领域，很快从状态空间走出来，

进入其他领域，子孙繁衍，人丁兴旺。这一支是当今最优控制在应用中的主体。

线性二次型控制具有各种各样的优点，但是，线性二次型没有回答一个最基本的控制问题：这个闭环系统是不是稳定。这里，我们的饱受牵记的李亚普诺夫同志出场了。李亚普诺夫也是一个脑子搭错筋的人，一百多年前，玩微分方程玩出了瘾，整出两个稳定性（或者叫收敛性）的定理，前一个没有什么了不起的，把非线性系统线性化，就是把一根曲线用很多一小段、一小段的直线近似，然后按直线来分析。后一个就有点邪门了。老李琢磨出一个定理，说是对于任意一个系统，如果能找到一个自我耗散的能量函数（数学说法是正定函数），也就是其数值永远为正，但随时间渐进地趋向零，或者说这个能量函数对时间的导数永远为负，那这个系统就是稳定的。据说定理的证明是一个天才的杰作，我等凡人只有频频点头的份。不过想想也对，系统的能量耗散没了，系统不也就安分下来了吗？当然就稳定喽。

李亚普诺夫比卡尔曼还要数学家，他的定理只给出“如果存在……就……”，怎么找这个自我耗散的能量函数他没说明，这个函数一般是什么样他也没说。这难道不倒搞自动控制的广大革命群众。不是要正定函数吗？不是没有限制什么形式的正定函数吗？那就用控制偏差的平方吧。说干就干，但是干着干着，好玩的事情出现了，对偏差平方（或二次型）的求导，导出了和线性二次型最优控制推导过程中同样出现的一个所谓黎卡蒂方程(Riccati equation)，感情这是殊途同归呀。换句话说，线性二次型控制总是稳定的。这是线性二次型控制的一个重要贡献：把最优性和稳定性连到一起。

再扯一句李亚普诺夫，他的第二个定理非常威猛，但是有点像一个奇形怪状的大锤，到现在人们还在找合适的钉子，好用这把大锤砸几下。线性二次型控制是已知的仅有的几个钉子之一，另一个是变结构控制，也可以用李亚普诺夫方法，这是题外话了。

都说瓦特的蒸汽机后，计算机是影响人类进程最大的发明，计算机当然也对自动控制带来深刻的影响。如前所述，控制理论基本上都是围绕微分方程转的，所以在“本质”上是连续的。但是数字计算机是离散的，也就是说，数字控制器的眼睛不是一直盯着被控对象看的，而是一眨一眨的。数字控制器的“手脚”也不

是一刻不停地连续动作的，而是一顿一顿的。这是数字计算机的天性使然。于是，传统的控制理论全部“翻译”到离散时间领域，微分方程变成了差分方程，所有方法、结论都有了连续、离散两套，不尽相同，但是大同小异。

要是数字控制就是简单的连续系统离散化，计算机控制也就没有什么了不起。离散控制带来了一些连续控制所不可能具备的新特点，这就是：差分方程用清晰界定的时刻之间的关系来描述动态过程。回到洗热水澡的例子，如果热水龙头不在跟前，而是在村外一里地的小锅炉房里，你只能用电话遥控，那水温可以表示为

下一分钟水温 =  $0.7 \times \text{现在水温} + 0.2 \times \text{上一分钟水温} + 0.1 \times \text{再上一分钟水温} + 0.4 \times$   
(5 分钟前锅炉房龙头开度 - 6 分钟前锅炉房龙头开度)

显然，下一分钟的水温受现在水温的影响比上一分钟和再上一分钟的水温的影响要大，但锅炉房龙头开度要是不变，现在、上一分钟、再上一分钟水温都一样的话，下一分钟的水温也应该和现在的水温一样。为什么用 5 分钟前锅炉房的龙头开度呢？那是因为热水从村外流到洗澡房要有一定的时间，这个时间就是滞后。要是把时间向前推，那现在的龙头开度就会影响 5 分钟后的水温。这说明了离散模型的一个重要特质：预估能力。所有预报模型都是建立在离散模型的这个预估能力上，不管是天气预报，还是经济预测，还是自动控制里有滞后的过程的控制。

数字控制的另一特质是可以实施一些不可能在连续时间实现的控制规律。工业上常有控制量的变化需要和当前的实际值有关的情况。比如对于不同的产品，反应器的转化率总是大体在 88-92% 之间，没有太大的变化，但是催化剂可以在 0.5 到 35ppm 之间变化，采用常规的 PID 的话，增益就非常难设，对一个情况合适了，对另一个情况就不合适。所以催化剂需要按百分比变化率调整，而不是简单地按偏差比例调整。比如说，转化率偏离 1% 时，催化剂要是在 0.5ppm，应该调整 0.05ppm；但是在 15ppm 的时候，就应该是 1.5ppm。这样，控制律就可以表示为：

当前的控制量=上一步的控制量\*（设定值/当前的测量值）

也就是说，在被控变量高于设定值 10%的情况下，控制量也增加 10%；测量值和设定值一样时，控制量不再变化。实际使用时，谁除以谁要根据测量值上升你是要控制量上升还是下降来决定，控制律也要稍微修改一下，成为

当前的控制量=上一步的控制量\*（当前的测量值/设定值）<sup>k</sup>

k 次方是用来调整控制律对“偏差”（这是已经不是差值，而是比值了，严格地说，应该叫“偏比”？）的灵敏度，相当于比例增益。这个控制律实际上相当于对数空间的纯积分控制，要是有兴趣，对很多常见的非线性过程有相当不错的效果，实现也简单。然而，这是一个本质离散的控制律，在连续时间里无法实现。

离散控制可以“看一步、走一步”的特性，是连续控制很难模仿的，也是在实际中极其有用的。

形形色色的控制理论再牛，没有被控过程的数学模型，照样抓瞎。前面的洗澡水温就是一个数学模型。这个模型是杜撰的，当然可以很容易地给它所有模型参数。但在实际中，模型参数不会从天上掉下来。多少科学家毕生致力于建立某一特定的物理、生物、化学或别的学科的数学模型，基本机制已经清楚的模型都不容易建立，更不用说很多过程的基本机制或深层机制并不清楚。所以靠机理推导被控过程的数学模型是可能的，但对日常的控制问题来说，并不实际。这就是控制理论的另一个分支——辨识——显身手的地方了。

如果给定一个模型，也就是一个数学公式，给它一组输入数据，模型就可以计算出对应的输出数据。比如说，给定模型  $y=2*x+1$ ，再给出  $x=1, 2, 3, 4$ ，那  $y$  就等于 3, 5, 7, 9，就这么很简单。辨识的问题反过来，先给定一个模型结构，在这里就是  $y=a*x+b$ ，已知输入-输出数据是  $x=1, 2$  时  $y=3, 5$ ，要求计算出  $a$  和  $b$ 。显然，这是一个二元一次方程，谁都会解。在实际中，输入-输出的观察数据含有测量噪声，这对参数估计的精度不利；但通常积累观察的数据量



远远超过未知参数的个数，不说数学，感觉上这就应该对克服测量噪声有利，关键是怎么利用这“多余”的数据。一个办法是把数据组两两配对，借众多的二元一次方程，然后对解出来的  $a$  和  $b$  作平均。还有一个办法就是有名的最小二乘法了，说穿了，就是以  $a$  和  $b$  为最优化的“控制量”，使模型输出和实际观测值之间的累积平方误差为最小。

实际工业过程大多有多年的运行经验，大量的数据不成问题。对于大多数常见过程，模型的基本结构和定性性质也可以猜一个八九不离十，有了如此有力的数学“大锤”，那么应该可以砸开一切建模的硬核桃啦。且慢，世上没有真正的“神奇子弹”，一个问题解决了，另一个同等难度的问题又会出现。对于辨识来说，问题有好几个。

第一个问题是工业数据的闭环性。大多数重要参数都有闭环回路控制。如果没有闭环回路控制，那要么就是过程特性实在太复杂，简单回路控制不了；要么就是这个参数其实不重要，飘移一点没人在乎。然而，一旦闭环，系统地输入和输出就是相关的了。这一相关不要紧，输入-输出数据之间的因果性就全乱了：输出通过被控过程本身和输入相关（这是好的，辨识就是要测算出这个相关关系，输出要是和输入不相关，也没有控制或辨识什么事了），输入通过反馈和输出相关；输入-输出成为一个闭合系统，你可以用任意多条定理或方法证明同样的事：由于因果不分，闭环辨识是不可能的，除非另外加入“新鲜”的激励，比如使劲变设定值，或者在闭环回路里额外施加独立于输入、输出的激励信号，比如“莫名其妙”地把阀门动几下。弄到最后，工业数据到底能用多少，就不是一个简单的回答。有的过程常年稳定操作，像乙烯装置，只有小范围的微调。这倒不是人家懒或者不求上进，而是这些装置早已高度优化，常年操作的极其接近极限的位置，但原料和产品单一，所以工艺状况不怎么大变。这种系统的闭环数据用起来很吃力，常常必须做一定的开环试验。有的过程经常在不同的状态之间转换（transition），或者由于不同的原料，如“吃”得很杂的炼油厂，或者由于不同的产品，如聚乙烯装置，这实际上就是“使劲变设定值”，是新鲜的激励。这种系统的闭环数据比较好用，但又别的问题，下面要谈到。

第二个问题是动态和稳态。动态模型的作用有两个：一是描述需要多少时间输出才能达到某一数值；二是输出最终能够达到什么数值。用股票市场举一个例子，你需要知道两件事：一是这支股票最后会升到多少，二是需要多少时间才能升到那里，只知道其中一个对你并没有太大的用处。当然为了简化，这里假定这支股票一路飙升，不来忽升忽降或跌买涨卖的名堂。这就要求输入-输出数据必须包含充分的动态和稳态信息，过于偏颇其中一方面对另一方面会不利。所以，长期稳定运行的过程中可能包含足够的稳态数据，但动态不足；常年不怎么稳定的过程可能包含足够的动态数据，但稳态不足。用 PID 控制打比方，精确的稳态数据有助于计算正确的比例控制增益，精确的动态数据有助于计算正确的积分和微分增益，显然，把比例增益整对了更为重要。为了获得精确的稳态，在辨识中常常需要等过程开环稳定下来才进行下一步，但是问题是，实际过程有时时间常数很长，几个精馏塔一串联，时间常数几个小时是客气的，一、两天都是可能的。这样一来，一个不太大的模型，十来个变量，开环试验一做就是一、两个星期。要是有一个装置能够两个星期开环，那也不需要什么控制了。

第三个问题是激励的信噪比。都说人类活动是二氧化碳和温室效应的主要原因，但要是你去生一堆篝火，再去高空大气层去测一测二氧化碳和温室效应，肯定什么也测不出来，本来多少，现在还是多少。为什么呢？不是因为这堆篝火没有效果，而是环境中的自然的变化远远超过了篝火的作用，换句话说，就是噪声远远超过了信号。工业测试也是一样，信号一定要有一定的强度，否则是白耽误工夫。信号强度应该使过程达到严重失稳的边缘，这样才好获得在大范围内都精确的模型，以便控制器不光在“风平浪静”的情况下可以正常工作，在“惊涛骇浪”的情况下也能使系统恢复稳定。然而，工厂以生产为主，在一切都“斤斤计较”的今天，如此大范围的测试所带来的产品损失甚至对设备的可能的危害，都是工厂极不愿意见到的。理论家们设计了一个伪随机信号，用一连串宽窄不等的方波信号，作为激励过程的输入，在理论上可以是过程参数的平均值不致偏离设定值太多，但 ISO9000 不仅要求产品质量的平均值要保证，产品质量的一致性也要保证。再说，伪随机信号的脉宽不好确定，太窄了，稳态数据不够；太宽了，和常规的阶跃信号也没有什么两样。所以伪随机信号在实际上用得很少。

第四个问题是输入的相关性。实际工业过程到了要用辨识来确定模型的时候，都是单回路对付不了了，所以都是多变量过程。在理论上，多个输入变量可以同时变化，只要输入变量的变化是相互独立的，数学上容许多个输入变量同时变化，而辨识可以正确地辨别模型。然而，在使用实际过程的历史数据时，常常遇到多个输入变量并不相互独立的问题。比如说，在制作巧克力的过程中，香草巧克力比较“苦”，或者说不太甜，而牛奶巧克力比较甜。问题是做牛奶巧克力时，不光加糖，还要加牛奶（废话，不加牛奶那还是牛奶巧克力吗？）由于两者总是同时出现，在甜度模型里，就难以辨别甜度是由于加糖的关系，还是由于加牛奶的关系。有的时候可以根据对具体过程的认识，人工地限制辨识的过程，来消除这种影响，有的时候，就不太容易了，只好不用历史数据，专门做试验，用各自独立的输入，辨识模型。

第五个问题是模型结构。模型结构包括两个方面，一是模型的阶数，二是剔除在物理上不可能的模型。辨识的模型归根结底还是差分方程，这就有一个如何预设阶数的问题。数学上有很多验前和验后的检验方法，在工业上，人们偷一个懒，改用费参数模型，也就是用一条响应曲线而不是一个方程来表述一个模型，这样就可以绕过阶数的问题。但是剔出不现实的模型还是一个手工活，需要对每一个模型仔细研究，以确定模型所描述的动态关系是否合理。数学方法还是不够可靠。

在搞模型的人中间，常常会听到黑箱、白箱和灰箱的说法。黑箱模型就是不理睬实际过程的物理、化学等性质，纯粹从数学出发，假设一个模型结构，然后用种种数学方法找出一个最好的模型。白箱反其道而行之，从物理、化学等性质出发，建立机理模型。黑箱模型的好处是“放之四海而皆准”，不需要对具体过程有深入的了解。黑箱模型是一种削足适履的作法，但是如果履本身就做得比较好，具有相当的灵活性和适应性，就并不需要削足。由于黑箱模型可以自由假设模型结构，黑箱模型的处理和使用都比较方便。黑箱模型是经验主义的，数据里没有包含的情况，黑箱模型无法预测。白箱模型则是“量身度造”的，反映了过程的物理、化学等性质，对实际过程的数据没有太大的依赖，对数据中不包含的情况

也能可靠地预测。但是白箱模型的结构有具体问题决定，得出的模型不一定容易使用。在实际中，人们经常在假设一个模型结构的时候考虑进大大简化的过程机理，所以模型结构不是凭空拍脑袋出来，而是粗略地抓住了过程的基本特质，然后再用黑箱方法的“数据绞肉机”，将简化模型没有能够捕捉的细微末节一网打尽。这种模型结合了黑箱和白箱的特点，所以称为灰箱。实际建模中，纯粹黑箱或白箱的成功例子很少，灰箱的成功机会就要多得多。

不管什么箱，最后还是有一个如何辨识实际过程的问题。闭环辨识的好处不用多说了，问题是如何从闭环辨识中获得有用的模型。工业上有一个办法，没有一个“官名”，但实际上是一个开环-反馈过程。具体做法是这样的：先用粗略的过程知识构造一个简单的多变量控制器，其任务不是精确控制被控过程，而是将被控变量为此在极限之内，一旦逼近或超过极限，就采取动作将其“赶”回极限内；但只要在极限内，就按部就班地做阶跃扰动，测试过程特性。测试的结果用来改进控制器的模型，然后再来一遍。几遍（一般两遍就够了）之后，模型精度应该很不错了。这个方法比较好地解决了辨识精度和过程稳定性的要求。

西游记里最好看的打斗是孙悟空大战二郎神那一段。孙悟空打不过就变，二郎神则是“敌变我变”，紧追不舍，最后把个无法无天的顽皮猴子擒拿归案。用控制理论的观点看，这“敌变我变”的本事就是自适应控制控制器结构根据被控过程的变化自动调整、自动优化。

自适应控制有两个基本思路，一是所谓模型跟踪控制，二是所谓之校正控制。模型跟踪控制也叫模型参考控制，在概念上对人们并不陌生。毛主席那阵子，经常树立各种榜样，目的就是要在党发出号召时，我们比照榜样的行为，尽量调整自己的行为，使我们的行为了榜样的行为接近。这就是模型跟踪控制的基本思路。模型跟踪控制在航空和机电上用得比较多，在过程控制中很少使用。

自校正控制的思路更接近人们对自适应的理解。自校正控制是一个两步走的过程，首先对被控过程作实时辨识，然后再辨识出来的模型的基础，实时地重新构造控制器。思路简单明了，实施也不算复杂，但自校正控制在一开始的欢呼后，

并没有在工业上取得大范围的成功，原因何在呢？

原因之一是闭环辨识。虽然自校正控制不断改变控制器的参数，在一定程度上打破了固定增益反馈控制对输入、输出带来的因果关系，但是因果关系还是存在，还是相当强烈，对辨识模型的质量带来影响。

原因之二是所谓“协方差爆炸”。数学上当然有严格的说法，但简单地说，就是自校正控制器的目的当然还是系统稳定下来，但是在系统越来越稳定的过程中，自校正控制器对偏差和扰动的敏感度越来越高，最后到“万籁俱静”的时候，敏感度在理论上可以达到无穷大，然而，这时如果真的扰动来了，控制器一下子就手足无措了。

原因之三是实际过程的复杂性。在辨识实际过程时，最重要的步骤不是后面的“数学绞肉机”，而是对数据的筛选，必须把各种异常数据剔除出去，否则就是“垃圾进来，垃圾出去”。但是，要实时、自动地剔除异常数据，这个要求非同小可，比设计、投运一个自校正控制器费事多了。这是自校正控制在实际中成功例子有限的最大原因。

自动控制从一开始就是以机电控制为主导的。60 年代数学派主导了一段时间后，70 年代化工派开始“小荷才露尖尖角”。自校正控制已经有很多化工的影子，但化工派的正式入场之作是模型预估控制（model predictive control, MPC）。这是一个总称，其代表作是动态矩阵控制（dynamic matrix control, DMC）。DMC 是 Charlie Cuttler 的 PhD 论文，最先在壳牌石油公司获得应用，以后 Cuttler 自立门户，创办 DMC 公司，现在是 Aspen Technology 公司的一部分。

数学控制理论非常优美，放之四海而皆准，但是像老虎一样，看起来威猛，却是干不得活的，干活毕竟靠老牛。DMC 的成功之处在于应用伪理论，将一些本来不相干的数学工具一锅煮，给一头老老实实的老牛披上一张绚烂的老虎皮，在把普罗大众唬得一愣一愣的时候，悄悄地把活干了。

DMC 基本就是把非参数模型（在这里是截断的阶跃曲线）放入线性二次型

最优控制的架构下，成功地解决了解决了多变量、滞后补偿和约束控制问题。多变量的含义不言自明，滞后放在离散动态模型下也很容易实现预测，这也没有什么稀奇。稀奇的是，DMC 用“土办法”解决了约束控制问题。所有实际控制问题的控制量都有极限。加速时，油门踩到底了，那就是极限，再要多一马力也多了不出来了。庞特里亚金的最大值原理在理论上可以处理约束控制问题，在实际上很难求出有用的解来，最速控制是一个特例。那 DMC 是怎么解决约束控制问题的呢？当某个控制量达到极限时，这个控制量就固定在极限值上了，这就不再是变量，而是已知量，把已知量代进去，将控制矩阵中相关的行和列抽掉，重新排列矩阵，剩余的接着求解。这也没有什么稀奇。令人头疼的是如何处理输出约束的问题。DMC 把线性规划和控制问题结合起来，用线性规划解决输出约束的问题，同时解决了静态最优的问题，一石两鸟，在工业界取得了极大的成功。自卡尔曼始，这是第一个大规模产品化的“现代控制技术”，Cutler 在 DMC 上赚了大钱了，在“高技术泡沫”破碎之前把公司卖给 Aspen Technology，更是赚得钵满盆溢。他女婿是一个医生，也不行医了，改行搞过程控制，跟着 Cutler 干了。

DMC 的英明之初在于从实际需要入手，不拘泥于理论上的严格性、完整性，人参、麻黄、红药水、狗皮膏药统统上，只要管用就行。在很长一段时间内，DMC 的稳定性根本没有办法分析，但是它管用。搞实际的人容易理解 DMC 的歪道理，但搞理论的人对 DMC 很头疼。

DMC 打开局面后，一时群雄蜂起，但尘埃落定之后，如今只有三家还在舞台上。Honeywell 的 RMPCT (Robust Multivariable Predictive Control Technology) 是一个中国同胞开创的，他的独特之处在于引入“漏斗”概念。大部分控制问题都有一个特点：如果扰动当前，有一点控制偏差是可以容忍的；但时间一长，控制偏差应该消除。换句话说，这就像一个时间轴上对偏差的横放的漏斗。这个概念对复杂过程的 MPC 参数整定非常有用，已经在别的公司的产品上也出现了。

第三家就是方兴正艾的 Pavilion Technology 的 Perfecter。美国公司有一个坏毛病，喜欢对好好的产品取一个不伦不类的名字。Perfecter 的特色是将神经元

技术（neural net）和 MPC 结合起来，所以可以有效地处理非线性过程。神经元模型没有什么神秘的，说穿了，就是具有某些特定复杂形式的回归模型，但是比回归模型更不适宜内插和外推。DMC 也号称可以处理非线性，因为即使阶跃响应曲线拐上几拐，DMC 照样囫圇吞枣，可以计算控制输出来，这就是非参数模型的好处。但是问题在于 DMC 的结构框架毕竟还是线性的，阶跃响应的概念根本不适合非线性过程，因为非线性响应和输入的绝对数值、相对变化 甚至变化方向有关，甚至可以更复杂，所以所谓 DMC 可以处理非线性是放空炮。如果实际过程的非线性不强，根本可以忽略它；如果实际过程有很强的非线性，DMC 肯定抓瞎。那么， Perfecter 用了神经元，是不是就所向披靡了呢？也不尽然。Perfecter 继承了 DMC 不问理论、唯实用是问的好 传统，但是 Perfecter 的基本骨架还是线性的 MPC，只是用静态的神经元模型时不时地作一个线性化。Perfecter 在理论上乏善可陈，在实用上还是管用的。

前面说到 PID 在当今过程控制中占至少 85%，那 MPC 就要占 14.5%了。

计算机对自动控制的影响要是只局限在离散控制理论上，那也就不是计算机控制了。事实上，80 年代以后新建的化工厂，基本都采用计算机控制。说是可以采用比 PID 更先进的技术，实际上，绝大多数还是在用 PID，加上顺序控制，按部就班地执行一系列动作。那计算机控制的好处到底在什么地方呢？

过程控制的实际装置最初全是直接安装在现场的，后来出现气动单元仪表，可以把压缩空气的信号管线从现场拉到中心控制室，操作工可以在中控观察、控制全厂了。电动单元仪表防爆问题解决后，中控的使用更加广泛。操作工坐在仪表板前，对所辖工段的情况一目了然。但是随着工厂的增大和过程的复杂，仪表板越来越长，一个大型化工厂随随便便就可以有上千个基本控制回路和上万个各种监控、报警点，仪表板非有几百米长不可，这显然是不可能的。生产过程的高度整合，使一两个人控制整个工厂不光满足削减人工的需要，也对减少 通讯环节、综合掌控全局有利。所以，计算机显示屏就不光是酷，而是必须的了。另外，计算机控制使现场仪表（阀门、测量变送器等）的自检成为可能，大大提高了系统的可靠性。于是，计算机控制就是不花没人性了。

计算机控制从一开始的集中控制（用 IBM 的大型机）到现在的分散控制（所谓 Distributed Control System, DCS）走过一个螺旋形上升的过程。集中控制的要害在于风险集中，要是大型机挂了，全厂都要失控。分散控制将全厂划分为若干条条块块，用以微处理器为基础的一个控制用局部网来分散控制，主要子系统都是实时冗余的，故障时在第一时间切换切换到备用系统，主系统和备用系统在平时定期互相自检、切换，以保证可靠。分散控制显然大大提高由于计算机本身引起的可靠性。但是现场仪表和接线终端（field terminal assembly, FTA）不是冗余的，整个可靠性链还是有漏洞。另外，控制局部网的同轴电缆长度有物理限制，FTA 到 DCS 的长度也有物理限制，所以最后分散控制还是不怎么分散，全是集中在中控室附近或地下室里。不过 DCS 在地理上的集中，并不妨碍其在逻辑上的分散，只要不是一把火把 DCS 的机房烧掉，部件可靠性的问题还是可以很好地隔离在小范围。

既然 DCS 是一个局部网，那就有一个通信协议的问题。DCS 基本上用两大类型的通信协议：轮询（polling）和中断。轮询由中心控制单元轮流查询所有子系统，不管有没有数据更新，到时候就来问一遍，所以不管什么时候，系统地通信流量都很高，但是恒定。中断方式正好相反，子系统自己先检查一下，如果数据没有变化，就不上网更新；直到数据有变化，再上网“打一个招呼”。这个方式的平时通信流量较低，所以网路带宽要求较低。但是生产过程发生异常时，大量警报数据蜂拥而来，如果带宽不够，就会发生通信阻塞的问题。所以，中断和轮询到最后对带宽的要求是一样的，因为谁也不能承担生产过程异常时通信阻塞的后果。

二十年前，Honeywell 是第一个吃 DCS 这个螃蟹的公司，今天 Honeywell 仍然是行业里的老大，尽管其设备昂贵，被戏称为 Moneywell。当年的 DCS 全是量身度造的硬件、软件。今天在“开放系统”（open architecture）的大潮里，DCS 的制造厂家都纷纷将控制台和计算、网络控制单元转向通用的 WIntel 或 UNIX 平台，自己专注于工控专用装置（如基本控制装置，包括 I/O）和系统的软件整合。但是这带来了新的问题。通用/商用硬件、软件的可靠性常常不能满



足 24 小时、365 天的连续运转要求。对于大多数 IT 来说，机子坏了，两小时内换上就是很快的了。但是对于生产过程来说，这是不可容忍的。开放结构容许将 DCS 和经营、管理、办公网络相连接，极大地提高了信息交流速度和深度、广度，但也带来了网络安全问题，紧接着就是 DCS 前面竖起一道又一道防火墙，把数据分享和远程操控压缩到最低。另外就是 WINTEL 夜以继日的不断更新换代，使硬件、软件的稳定性十分糟糕，没有过多少时间，又要升级，又是头疼。这是 DCS 的第二个螺旋形上升，只是现在还是盘旋多于上升。

计算机控制的领地也在扩大，类似 USB 那样的技术也开始用于数字化的仪表。过去的仪表都必须把信号线拉到接线板（marshalling panel）上，然后再连到 FTA 上，这样同样远在百把米外的 10 台仪表，需要并行拉线，很浪费。用了类似 USB 的现场总线（field bus），各个仪表可以“挂”在总线上，然后一根总线连到 DCS 就可以了，大大节约拉线费用和时间，对系统（如加一个测量用的变送器或控制阀）的扩展也极为方便。

DCS 的最大优越性是可编程。这不是简单的像 PLC（programmable logic controller，可编程序逻辑控制器，多用于机电控制）的梯形逻辑那样编程，而是可以像 C、FORTRAN 那样“正规”的编程。没有在 IT 干过，只能和学校里计算机语言课程和大作业的程序相比。DCS 编程和平常的编程相比，还是有一些特点的。首先，DCS 的程序属于“再入”式，也就是定时反复运行的，而不是一次从头到底运行就完事的。所以 DCS 程序可以在运行完毕时在内存里存放数据，到下次运行时再调用，形成所谓“递归”运算。这既是优点，也是缺点，要是别人在你两次运算中间把那个中间数据更改了，你就惨了，找债主都不容易。

DCS 程序的特色是实时，所以其执行非常取决于一系列事件在时间上的顺序。时序上要是搞岔了，老母鸡也就变鸭了。问题是，分散控制要求越分散越好，不光是可靠性，在系统资源的调度上，分散了也容易使系统的计算负荷均匀。这样一来，一个应用程序包常常将一个巨大的程序打散成很多小程序，各自的时序和衔接就要非常小心。

和学术型控制计算程序最大的不同，或许还在于对异常情况的处理。一个多

变量控制问题在实际上常常会有部分变量处于手动控制，而其余变量处于自动控制的情况。这在理论上是一个麻烦，在实际上是一个噩梦。不光要考虑所有的排列、组合，还要考虑所有情况平顺的切入、切出，不同模式之间的切换。还有就是要考虑异常情况下如何安全、自动地退出自动控制，交还手动控制。有时操作规程上的一句话，程序写写就是一页。如果操作规程上来一句“视情处理”，那就更惨了。在所有控制程序中，控制计算通常不超过 30%，20%为人机接口功能，而 50%为异常情况处理。

计算机控制不是因为更先进、更有效的人机界面才开始的。从一开始，人机界面就面临一个管中窥豹的问题。计算机的 CRT 显示屏只有这么大，不可能“一言以蔽之”，在一瞥之中把所有的过程信息尽收眼底。计算机可以不断地换屏，分段显示其他装置、工段的信息，但是把所有的工段、装置分别用各自的画幅表示，如果没有有效的组织，找都很难找到，就像在同一个目录里杂乱无章地放上百把个文件一样。分级的菜单是传统的解决办法，但是要逐级上去再逐级下来，很费时间，情急之中，往往来不及更换。大键盘上 short cut 键可以“一键调出”，但需要死记硬背，这可不是几个、十几个画幅，而是上百个甚至更多。很长时间以来，如何有效地在画幅之间导航，可以在最短时间和最少点击内，不需要死记硬背，就可以直观地找到所需要的画幅，一直是一个令人头疼的问题。

人机界面设计的另一个问题是色彩。还记得 DOS 2.0 时代的 WordStar 吗？那是黑底绿字的。那时候，CRT 亮度不足，寿命也糟糕，黑底可以延长寿命，绿字可以增加反差，帮助阅读，反正机房是暗暗的，黑底并不伤眼睛。到了 WordPerfect 5.0 的时候，就是蓝底白字了，字和背景之间的反差大大减小，蓝底也比较适宜于在明亮的房间内使用。到了 Word 的时代，没有昏暗的机房了，基本上都用像纸上写字一样的白底黑字了，再用黑底绿字，太伤眼睛。

中控室计算机显示也经历了类似的旅程。早期 DCS 的显示都是黑底绿字的，到了用 Wintel 或 UNIX 的时代，很多人出于习惯，仍然采用黑底绿字，但是现代人机工程研究表明，浅色背景大大减低眼睛的疲劳，在明亮室内的灯光对

屏幕的反光也小，所以控制室的显示开始向浅灰背景进化了。人机工程研究同时发现，色彩可以作为过程信息的一部分，天下太平的时候，应该用最不显眼的灰色，所有的图形、数据都用不同深浅的灰色来表示，只有在过程参数越限或报警时，才采用彩色显示，这样可以一下子就把操作工的注意力吸引到需要的地方。但是，出于习惯思维，很多地方还是大量采用各种色彩表示不同的设备状态和参数，即使是正常状态也是一样。这样在平日里色彩缤纷很好看，但在异常情况时，不容易在万马军中找到上将的首级，实际上是舍本逐末。

显示器的布置也很有讲究，少了当然不行，也不是越多越好，一个操作工的视界的上下左右有一定的范围，控制台的色彩、构造、照明都不能想当然的。这不是助长修正主义，而是保持操作工最有效地控制生产过程的要求。

传统上，如果操作工不抱怨，控制回路的性能就是可以接受的，除非你想精益求精，一般不会去没事找事，重新整定参数。在对经济效益斤斤计较的今天，生产过程的工艺条件被推到极端，对控制性能提出极大的挑战，控制回路必需时时、处处都在最优状态。随着控制回路数的迅速增长，单靠人工观察，已经难于随时掌握所有控制回路的性能状况了。控制回路性能评估技术应运而生。

理论上，对一个过程可以设计一个最优控制，其中一种就叫最小方差控制。这其实是线性二次型最优控制的一种，控制作用比较猛，但是这是理论上的极限，控制方差不可能再小了。90 年代时，理论界提出一个方法，可以用闭环辨识的方法，不辨识模型，而是直接确定理论上的最小方差，然后将实际方差和理论上的最小方差相比，判别控制回路是否需要重新整定。这个方法开创了控制回路性能评估的先河，但是在实用上不容易排除不利影响，应用不多。

然而，不和理论上的最优值比较，而是和实际上的理想值比较，就可以绕过很多麻烦的理论问题。比如说，流量回路应该在 1 分钟内安定下来，那理想值就是 1 分钟。通过快速富利叶变换和频域分析，可以将理论性能和实际性能相比较，迅速确定回路的当前性能状况。最要紧的是，这可以用计算机自动采集数据，自动计算，每天早上（或随便什么时候）给出报表，控制工程师可以一目了然，哪些回路需要重新整定，哪些没有问题，可以有的放矢。实时频域分析还可以将所

有以相近频率振荡的回路罗列出来，接下来控制工程师就可以按图索骥，找出害群之马了。

控制回路性能评估的下一步当然就是自动整定。这实际上是一个简化的、断续运行的自校正 PID 控制器，在理论上已经没有问题，但实用上还有很多可靠性问题没有完全解决，现在产品不少，但实用的还是不多。

对控制回路性能评估的更进一步，当然就是对生产过程的故障诊断了。故障就是异常情况，异常就是和正常不一样。所以故障诊断的核心在于如何探测这“不一样”。

故障总是有蛛丝马迹的，问题在于工业过程的数据量太大，在大海里捞针，等捞到的时候，常常已经时过境迁了。在数据分析中，PLS（其实是 Projection to Latent Structure，而不是一般所认为的 Partial Least Square）和主元分析（Principal Component Analysis, PCA）是很流行的方法。PLS 和 PCA 将众多相关的变量归拢到少数几个“合成”的变量，这样一个有大量变量的复杂大系统就可以简化为一个小系统，就从大海捞针变为碗里捞针了。捞出来的针不再是单个的变量，而是变量的组合。这和实际是相符的，故障的早期征兆常常是若干变量的组合，而不能单从一两个变量上看出来。

PLS 和 PCA 还可以和图形方法结合起来使用。比如说，将那些合成变量标称化，就是除以正常值，那所有合成变量的标称值就是 1。把所有变量画成“蜘蛛图”（spider chart），每一个蜘蛛脚代表一个合成变量，由于合成变量的标称值都是 1，蜘蛛图就是大体为圆的。如果哪一个脚出现变化，蜘蛛就不圆了，非常容易看出异常来，接下来就可以有的放矢地寻找故障的早期迹象了。

图形数据分析的另一个路子是所谓 co-linear 分析。这是 IBM 早年琢磨出来的一个东西，理论上简直没有东西，但要求换一个思路，正所谓退一步海阔天空。平常的数据点，三维以上就没法画了。但是如果把三维空间的所有数轴画成平行线，而不是常见的直角坐标，那三维空间里的一个点，就是连接三根平行线的一根折线。如果仅此而已，那也就是一个简单但愚蠢的数学游戏。平行坐标系的妙处在于，平行线可以尽着画，所以 5 维、20 维、3 千维，只要纸足够大，都可以

画，而且可以看见，而不是只能想像。平行坐标只有一个缺点，就是只能表述离散点，而难以表述连续的线或面，但这对于计算机采集的数据来说，不是问题，计算机采集的数据本来就是离散点。这样，用平行坐标把大量的数据点画成折线簇，可以很直观地看出数据中的模式来。

故障诊断的另一个思路是对整个过程进行辨识。辨识出来的模型表述系统的行为，故障当然就是行为的改变，所以将实时辨识出来的模型和正常模型相比较，就可以判断系统是否出现异常或故障。

计算机和模型的另一个用处就是仿真。仿真（simulation）也叫模拟，但是模拟容易和模拟电路（analog circuit）搞混，所以现在叫仿真多了。只要对实际过程有一个足够精确的模型，计算机是可以相当精确地模仿实际系统的行为的。

仿真有静态仿真和动态仿真。静态仿真基本上就是解一个巨大的非线性联立方程组，描述空间分布的微分方程也被有限元方法分解了。现代静态仿真已经可以做得相当精确，但这也是在多年结合实际过程数据“磨合”模型的基础上才能做到的。静态仿真大量用于工艺设备设计计算，但是对研究实际过程的真实行为的作用有限，因为对整个生产过程和工艺的仿真要考虑进各个设备动作的时间和控制回路的影响，这些静态仿真是无法体现的。动态仿真要解同样巨大的联立微分方程组，由于要达到实时或更快，一般只能大大简化，否则计算速度跟不上。希望有朝一日，动态仿真可以达到静态仿真同等的精度，而不必担心损失计算速度。

仿真在工业上十分有用。现代化工厂越来越稳定，越来越安全，很多操作工一辈子也没有遇到过真正危险的情况。但没有遇到过不等于不会遇到，操作工必须接受足够的训练，只有这样，才能当遇到危险情况时，首先能及时、正确地识别故障，然后才能及时、正确地作出反应。这就要靠仿真训练了。现代化工厂也在不断地拓展工艺参数的极限，经常需要做各种各样的试验。有了仿真，就可以预先验证试验的构思，和验证对紧急情况的处理。

仿真更是控制工程师的好帮手，新的控制回路先放到仿真上试一下，得出初始整定参数，验证异常情况的处理能力，然后再放到真家伙上，可以避免很多不必要的惊讶。

仿真的一个远亲是实时最优化（real time optimization, RTO）。对于斤斤计较的现代制造业，实时最优化当然是求之不得的。实时最优化就是把整个生产过程当作一个大的实时仿真来运算，实时（实际上是每小时）计算出最优工况。想法是好的，困难是多的。首先，那么大一个方程组收敛不容易，要划成很多条条块块，分别求解，然后拼起来。问题就出在“拼”上，边界条件碰不拢怎么办？模型总是有相当的简化，其中有些参数必须和实际测量值符合，有些就没有实际测量值对应，就是“经验系数”（fudge factor）了。这些经验系数就是承担收拾烂账的，边界碰不拢，就调整经验系数，使他们对齐。问题是，好多时候，这一招也不灵，所以实时最优化的喇叭吹得很响，真正用起来的很少，花了大钱最后放弃的也不在少数。

和打仗一样，赢得战斗的是武士，不是武器。控制工程师是控制回路成功与否的关键，而不是价值千金的计算机，或者“放之四海而皆准”的数学控制理论。

在加拿大，化工系的控制“专业”要选满所有化工学分，然后再加选控制学分，所以要求比一般的化学工程师还要高一点。选满化工学分是很重要的一点，如果没有对化工的话语权，那化工控制也别混了，这一点是国内（至少是二十多年前我读大学的时候）所欠缺的。这就像医生一样，只有对生理、病理有深刻的了解，对病人的具体情况有深刻的了解，才有可能可靠地判断病情，才能可靠地开方治病。只会看单抓药，这就不是医生，而是药剂师了。在实际中，控制工程师对工艺过程的动态行为的理解至少应该和工艺工程师同等，和操作工相当。事实上，很多时候，控制工程师的使命就是将工艺工程师和操作工的经验和知识具体化、自动化了，如果你不能深刻理解，那如何实现呢？一个优秀的控制工程师可以在操作工不在的时候，当班操作；可以在工艺工程师不在的时候，做出工艺决定。

但是控制工程师毕竟不是工艺工程师，也不是操作工。控制工程师应该掌握前面说到的所有领域，从数学控制理论，到计算机网络，到人机工程，到工艺和仪表知识。这个要求很高，但不是不切实际的。这些是揽这个瓷器活所必需的金钢钻。这也是为什么现在工业界热衷于招雇具有硕士学位的控制毕业生，因为本科的几年已经很难学习必须的知识了。至于博士，那还是有眼高手低的嫌疑，呵

呵。

专业知识只是成功的一面，控制工程师必须善于与人打交道。工艺工程师比较好说，毕竟有类似的背景，但操作工是控制系统成败的关键，如果无法取得操作工对你个人和你的控制系统的信任和合作，那控制系统很可能就是永久性地被关闭，操作工宁愿手工控制，出了问题还是因为控制系统不可靠，你就等着里外不是人吧。但是取得操作工的信任和合作后，事情会向相反的方向发展。操作工主动向你提出改进建议，或新的想法，主动找机会帮你试验新的功能，主动拓展控制系统的性能极限。如果说顾客是上帝的话，操作工而不是部门主管才是控制工程师的上帝。

控制工程师也要善于和头儿打交道，毕竟搞项目、要钱的时候，还是要找头儿的。打报告、作报告、项目控制和管理、和供应商打交道，这些都是必备的技能。

工艺工程师也是工程师，但用军队的比方来说，他们人多势众，更像常规部队，习惯大兵团协同作战。控制工程师则像特种部队，人数少，行止怪癖（至少对工艺的人来说，他们永远弄不明白控制的人到底在做什么，怎么做出来的），从规划到实施到维修，全一手包办。

控制理论的发展历程就是一个寻找“放之四海而皆准”的“神奇子弹”的历程，终极目标是可以用一个统一的数学控制工具去“套”任何一个具体的控制问题，而不必对具体过程的物理、化学等特性有深入的理解。控制理论的每一次重大进展，都给人们带来希望，“这一次终于找到了”。但每一次希望都带来了新的失望，新方法、新工具解决了老问题，但带来了新的局限，有的时候甚至转了一圈兜回去了。新的局限往往比老问题更棘手，需要对过程的理解是更多而不是更少。矛和盾就是这么着在螺旋形上升中斗法。

但是现实常常和人们的认识背道而驰。在商业化的大潮中，推销先进控制算法的公司拍胸脯担保可以如何如何用“万能”的数学控制工具解决一切控制问题，那些绚烂的老虎皮也确实照得不明就里的人眼花缭乱，心旌飘荡；公司的头儿也一口吃进，毕竟“技术万能论”不仅在美军中盛行，在北美的公司文化中也是大行

其道。直到有一天，人们发现永动机依然是神话，人还是不能在水上步行，方才想起来，原来世上是没有这等好事的。不过这是题外话了。

本来就知道这个系列会又臭又长，但是既然写了，就写完吧，希望没有占用太多的带宽，希望没有浪费大家的时间，希望给对自动控制有兴趣的朋友提供一点入门的知识，希望给同好提供一些从实际中得到的经验，错误的地方肯定很多，敬请原谅，同时谢谢阅读。