

随机迭代学习控制

沈 栋 著



科学出版社

TP273
201723

图 书 号

随机迭代学习控制

沈 栋 著

科 学 出 版 社

北 京

内 容 简 介

本书主要介绍随机迭代学习控制的设计与分析方法,内容包括各种随机信号环境下的算法设计原理与分析技巧,其中随机信号包括随机噪声、随机数据丢包、随机批次长度等。本书第1章综述了随机迭代学习控制的研究现状及主要的原理方法,第2~11章分别针对工程非线性、未知控制方向、数据丢包、批次长度随机变化、量化控制、分散式控制等多个相关问题进行了深入讨论。

本书可供研究迭代学习控制的科研人员 and 研究生参考,亦可供相关工程技术人员研究参考。

图书在版编目(CIP)数据

随机迭代学习控制/沈栋著. —北京:科学出版社,2016.11

ISBN 978-7-03-050335-0

I. ①随… II. ①沈… III. ①学习系统-迭代计算 IV. ① TP273

中国版本图书馆 CIP 数据核字(2016) 第 257025 号

责任编辑: 阚 瑞 / 责任校对: 郭瑞芝

责任印制: 徐晓晨 / 封面设计: 迷底书装

科 学 出 版 社 出 版

北京东黄城根北街16号

邮政编码: 100717

<http://www.sciencep.com>

北京中石油彩色印刷有限责任公司 印刷

科学出版社发行 各地新华书店经销

*

2017年1月第 一 版 开本: 720 × 1000 B5

2017年3月第二次印刷 印张: 13 3/4

字数: 270 000

定价: 75.00 元

(如有印装质量问题, 我社负责调换)

前 言

迭代学习控制是控制理论的一个分支,相较于自适应控制、鲁棒控制等研究方向而言,迭代学习控制是一个相对小众的方向.自20世纪80年代提出以来,经过30多年的发展,迭代学习控制取得了丰硕的成果,发展了一系列独特的研究方法和技巧.在应用方面,迭代学习控制强调系统能够不断重复,从而得以借用“学习”的思想来使得输入信号能够不断地被优化.而现实中的许多实际生产系统都是采用批次生产模式,这也是迭代学习控制提出的工程背景,因此,迭代学习控制在工程上有很多实质性的应用.

随机迭代学习控制是指系统中含有各种随机因素下的迭代学习控制.这里随机因素包括系统噪声、量测噪声、随机数据丢包、通信延迟等各种用随机变量刻画 signals.这些信号同时体现了不确定性与随机性,因此,对这类问题的研究有其独特的理论价值与应用意义.本书对这一问题进行深入的探讨.

全书共11章.第1章全面深入地综述随机迭代学习控制的研究现状及主要原理与方法,并对多个潜力方向进行了介绍,可供读者快速了解随机迭代学习控制的相关内容.第2章介绍系统输入端含有死区、预载、饱和等工程非线性环节下的迭代学习控制.第3章介绍同时含有固定时滞和死区环节的非线性系统的迭代学习控制.第4章介绍 Hammerstein-Wiener 系统的迭代学习控制.第5章介绍线性系统在未知控制方向下的方向调节机制及相应的迭代学习控制.第6章与第7章介绍随机数据丢包环境下的迭代学习控制,分别为控制方向已知与未知两种情形.第8章与第9章介绍系统运行批次长度随机变化下的迭代学习控制,分别为线性系统与非线性系统两种情形.第10章介绍基于量化信息的迭代学习控制.第11章介绍由多个子系统相互关联而成的大规模系统的迭代学习控制问题.全书各章涉及随机迭代学习控制的多个具体问题,研究问题相对独立,算法设计与收敛性分析则成为一体,主要是基于随机逼近方法完成.除第1章综述以外,其余各章均针对具体问题给出迭代学习控制算法,并给出严格的收敛性分析.书中还给出许多仿真算例,用以说明所得结果的有效性与可应用性.

本书的出版得到了国家自然科学基金(基金编号:61304085)与北京市自然科学基金(基金编号:4152040)的资助,在此对资助机构表示感谢.作者自博士起从事迭代学习控制方向的研究,得到了许多前辈的指导和教诲,受益匪浅.感谢恩师陈翰馥院士将我带进迭代学习控制这一富有特色的研究领域,他严谨的学风与敬业精神对我有很深的影响.感谢博士后期间合作导师王飞跃研究员对我的鼓励和帮助,

他提供了许多富有启发性的学术建议. 感谢新加坡国立大学的许建新教授、黎巴嫩美国大学的 Saab 教授、中国台湾华梵大学的简江儒教授的讨论与合作, 在此过程中我获益良多. 感谢迭代学习控制领域的各位前辈和同行一直以来对我的帮助.

谨以此书献给妻子陈腊梅和女儿沈墨琦.

由于水平所限, 本书不足之处在所难免, 请读者不吝批评指正.

沈 栋

2016 年 7 月

目 录

前言

第 1 章	随机迭代学习控制	1
1.1	迭代学习控制	1
1.1.1	迭代学习控制基础	2
1.1.2	随机迭代学习控制	6
1.2	线性随机系统	7
1.2.1	基于卡尔曼滤波的方法	7
1.2.2	基于随机逼近的方法	12
1.2.3	其他方法	15
1.3	非线性随机系统	16
1.3.1	基于卡尔曼滤波的方法	16
1.3.2	基于随机逼近的方法	18
1.3.3	其他方法	22
1.4	针对其他随机信号的迭代学习控制	22
1.4.1	随机丢包	22
1.4.2	随机异步	25
1.4.3	随机时滞	27
1.5	潜力研究方向及展望	27
1.5.1	随机点对点迭代学习控制	27
1.5.2	变跟踪目标的迭代学习控制	29
1.5.3	分散式/分布式协同随机迭代学习控制	30
1.6	本章小结	31
第 2 章	输入端含工程非线性环节的学习控制	33
2.1	引言	33
2.2	问题描述	35
2.3	最优控制	37
2.4	迭代学习控制及其收敛性	39
2.5	本章小结	45
第 3 章	固定时滞非线性系统的学习控制	46
3.1	引言	46

3.2	问题描述	47
3.3	最优控制序列	50
3.4	迭代学习控制及其收敛性	51
3.5	仿真算例	55
3.6	本章小结	58
第 4 章	Hammerstein-Wiener 系统的学习控制	59
4.1	引言	59
4.2	问题描述	59
4.3	最优控制的存在性	62
4.4	迭代学习控制及其收敛性	67
4.5	仿真例子	75
4.6	本章小结	78
第 5 章	未知控制方向下线性系统的学习控制	79
5.1	引言	79
5.2	问题描述	80
5.3	方向切换机制	81
5.4	迭代学习控制及其收敛性	83
5.5	仿真算例	85
5.6	本章定理证明	86
5.7	本章小结	105
第 6 章	存在随机丢包下网络化随机系统的迭代学习控制	106
6.1	引言	106
6.2	问题描述	107
6.3	迭代学习控制及其收敛性	109
6.4	非线性系统情形	113
6.5	仿真算例	116
6.6	本章小结	119
第 7 章	含有损通道且控制方向未知的非线性系统的迭代学习控制	120
7.1	引言	120
7.2	问题描述	121
7.3	迭代学习控制及其收敛性	125
7.4	仿真算例	127
7.5	本章引理证明	133
7.6	本章小结	140

第 8 章 批次长度随机变化下的迭代学习控制	141
8.1 引言	141
8.2 问题描述	142
8.3 迭代学习控制算法	144
8.4 强收敛性	148
8.5 仿真算例	151
8.6 本章小结	153
第 9 章 批次长度随机变化下非线性系统的迭代学习控制	155
9.1 引言	155
9.2 问题描述	156
9.3 迭代学习控制算法	158
9.4 收敛性分析	160
9.5 仿真算例	165
9.6 本章小结	168
第 10 章 量化误差下随机系统的迭代学习控制	169
10.1 引言	169
10.2 问题描述	170
10.3 迭代学习控制及其收敛性	171
10.4 非线性系统情形	176
10.5 仿真算例	179
10.6 本章小结	185
第 11 章 大规模非线性系统的迭代学习控制	186
11.1 引言	186
11.2 问题描述	186
11.3 最优控制	188
11.4 迭代学习控制及其收敛性	189
11.5 仿真算例	194
11.6 本章小结	197
附录 随机逼近算法	198
参考文献	201

第 1 章 随机迭代学习控制

本章综述了随机迭代学习控制方面的相关研究进展. 迭代学习控制适用于可以不断重复完成指定任务的系统. 在过去三十年中, 迭代学习控制在理论和实际两方面都取得了许多重要的进展. 而在迭代学习控制中, 与随机信号有关的研究成果还比较少. 这里的随机信号包括系统噪声、量测噪声、随机数据丢包等各种在实际系统中普遍存在的信号. 本章从关键技巧的角度综述了相关进展, 包括三个方面: 线性随机系统、非线性随机系统和其他随机信号. 进而介绍了几个有潜力的研究方向, 包括点对点迭代学习控制、变轨迹迭代学习控制以及分散式/分布式迭代学习控制.

1.1 迭代学习控制

在生活中, 有这样一种认识, 一件事情重复去做, 通常会做得越来越好. 例如定点投篮, 当一个人进行定点投篮时, 一开始他可能投不中, 但随着不断练习, 他很有可能很快就会命中. 这里一个很重要的原因就是, 投篮者可以不断地从已有经验中学习. 具体地说, 投篮者可以根据之前投篮的偏差来调整下一次投篮的角度和力度, 从而使得篮球逐渐接近篮筐乃至命中. 迭代学习控制便是基于这样一种朴素的想法发展起来的控制分支. 粗略地说, 迭代学习控制是通过对之前过程信息的学习来改善系统的表现性能的.

因为要不断重复学习, 迭代学习控制适用于运行过程能够不断重复的系统, 通常用于跟踪某一给定目标且跟踪目标往往保持不变. 对这样的系统, 迭代学习控制能够根据之前的输入和输出数据, 结合跟踪目标来构造应用到下一次运行过程的输入信号, 从而达到逐步改善系统跟踪性能的目标. 由此可知, 迭代学习控制具有如下几个特点: ① 系统能够在有限的时间长度运行完毕并不断重复; ② 系统能够重复回到同一初始值; ③ 系统重复跟踪同一目标轨迹等. 迭代学习控制的基本思想如图 1.1 所示.

在图 1.1 中, 对跟踪目标 y_d , 在第 $k+1$ 次运行过程中, 根据第 k 次过程的输入 u_k 及跟踪情况 $e_k = y_d - y_k$ 构造新的输入 u_{k+1} . 这个输入 u_{k+1} 即应用到第 $k+1$ 次运行过程的输入, 同时 u_{k+1} 也存储到记忆器中用于下一次运行过程输入的构造, 系统在输入 u_{k+1} 作用下输出信号为 y_{k+1} , 其与跟踪目标 y_d 的差值也被用于构造下一次过程的输入. 至此, 系统沿迭代过程形成闭环反馈.

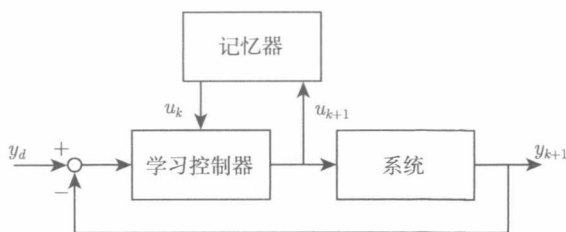


图 1.1 迭代学习控制示意图

将迭代学习控制与现实生活类比来看, 系统当前及以前的输入/输出信息相当于现实生活中的经验. 在现实生活中, 我们就是根据这些经验制定出下一次执行相同任务时的策略, 而这种新的策略就相当于在迭代学习控制中应用到下一运行过程中的输入信号. 在现实中, 以往的经验会改善我们做事的表现, 那么在系统控制之前的运行情况一定程度上也会帮助改善我们对系统的控制性能.

迭代学习控制的主要优势在于其控制律的设计仅要求系统的跟踪目标及输入/输出数据. 换言之, 关于系统自身的信息要求很少, 甚至可以完全未知. 迭代学习控制算法形式简单且行之有效.

注意到迭代学习控制是一种沿着运行过程调整而非沿运行时间调整的控制方法, 这恰是迭代学习控制不同于非学习型控制的本质所在. 例如, 反馈控制是一种典型的非学习型控制, 迭代学习控制与反馈控制的不同之处在于, 反馈控制没有利用之前运行过程的情况, 仅根据当前运行过程的输入/输出数据进行调整, 在每个运行过程可能产生相同的跟踪误差, 而迭代学习控制则充分学习之前运行过程的跟踪情况, 在其控制中包含了对已有信息的学习与利用. 此外, 迭代学习控制与具有一定学习能力的自适应控制也有区别, 自适应控制通常是针对相对固定的控制器来不断调整其控制器参数, 而迭代学习控制则是构造新的输入信号.

迭代学习控制的概念可以追溯到 Uchiyama 于 1978 年发表的文献 [1], 但该文为日文文献, 未获得广泛传播. 而在 1984 年的文献 [2]~[4] 开启了对迭代学习控制的研究. 自此以后, 大量关于迭代学习控制的文献涌现出来, 尤其是在 20 世纪末 21 世纪初研究活跃. 相关的专著有文献 [5]~[12], 综述文章包括文献 [13]~[15] 等. 部分期刊也组织了一些期刊专号, 如文献 [16]~[18]. 时至今日, 迭代学习控制已经成为一个重要的控制分支, 并在多种实际应用中取得了良好的控制效果^[9]. 迭代学习控制在实际中的应用包括机器人^[19-22]、硬盘驱动^[23, 24]、工业/化工过程^[25, 26] 等.

1.1.1 迭代学习控制基础

本节给出迭代学习控制的基本模型设定, 并给出一些传统的收敛结果. 考虑如

下离散时间线性时不变系统

$$\begin{aligned} x(t+1, k) &= Ax(t, k) + Bu(t, k) \\ y(t, k) &= Cx(t, k) \end{aligned} \quad (1.1)$$

其中, $x \in \mathbb{R}^n$, $u \in \mathbb{R}^p$ 和 $y \in \mathbb{R}^q$ 分别表示系统的状态、输入和输出; 矩阵 A 、 B 和 C 是具有适当维数的系统矩阵; t 代表一个运行批次内的任意时刻, $t = 0, 1, \dots, N$, 这里 N 表示系统运行的时间长度; 而 k 表示不同的运行过程, 取值为 $k = 0, 1, \dots$. 为方便, 对离散时间系统的 t 取值 $0 \sim N$, 用记号 $t \in [0, N]$ 表示.

由于迭代学习控制要求系统能够不断重复运行, 所以系统需要不断重置. 在基本模型及众多研究中, 需要下述初始值重置条件

$$x(0, k) = x_0, \quad \forall k \quad (1.2)$$

系统的跟踪目标为事先给定的目标轨迹 $y(t, d)$, $t \in [0, N]$. 注意到初始值重置条件, 一般要求 $y(0, d) = y_0 \triangleq Cx_0$. 迭代学习控制的设计目标即通过不断更新系统的输入 $u(t, k)$, 使得相应输出 $y(t, k)$ 尽可能地接近于 $y(t, d)$. 为此, $\forall t \in [0, N]$, 记跟踪误差为

$$e(t, k) = y(t, d) - y(t, k) \quad (1.3)$$

因此, 迭代学习控制的设计实际上是基于之前运行过程的输入 $u(t, k)$ 与跟踪误差 $e(t, k)$ 来构造控制更新律, 以得到下一次运行过程的系统输入信号. 这一关系在广义上可以写为

$$u(t, k+1) = h(u(\cdot, k), \dots, u(\cdot, 0), e(\cdot, k), \dots, e(\cdot, 0)) \quad (1.4)$$

当上述函数关系仅依赖于上一过程时, 称为一阶迭代学习控制律; 其余的则称为高阶迭代学习控制律. 一般而言, 要同时兼顾算法的简单性与有效性, 因此很多迭代学习控制都设计为一阶形式, 即

$$u(t, k+1) = h(u(\cdot, k), e(\cdot, k)) \quad (1.5)$$

特别地, 控制律多设计为线性形式. 其中, 最简单的迭代学习控制律为

$$u(t, k+1) = u(t, k) + Ke(t+1, k) \quad (1.6)$$

在这一更新律中, K 为学习增益矩阵, 它是这一算法的设计目标; $u(t, k)$ 表示上一运行过程的输入; 而 $Ke(t+1, k)$ 表示基于上一过程跟踪误差的修正项. 注意到在系统 (1.1) 中 t 时刻的输入直接影响到 $t+1$ 时刻的状态, 进而影响到 $k+1$ 时刻的

输出, 因此在算法 (1.6) 中用 $t+1$ 时刻的跟踪误差来修正 t 时刻的输入信号. 通常, 称算法 (1.6) 为 P 型迭代学习控制律. 当修正项变为 $K(e(t+1, k) - e(t, k))$ 时, 相应的控制律称为 D 型迭代学习控制律.

对系统 (1.1) 及更新算法 (1.6), 一个基本的收敛结果为: 若 K 满足

$$\|I - CBK\| < 1 \quad (1.7)$$

则有 $\|e(t, k)\| \xrightarrow[k \rightarrow \infty]{} 0$. 其中, $\|\cdot\|$ 为算子范数.

从上述收敛结果可以看出, K 的设计并不需要已知关于系统矩阵 A 的任何信息, 而仅依赖于矩阵乘积 CB . 这反映了迭代学习控制律的设计优势, 它对系统自身的信息依赖性很小. 因此可以用于处理未知性较多的系统跟踪问题.

注记 1.1 从迭代学习控制的表示形式可以看出, 其模型具有典型的 2D 系统特征. 很多学者便从这一点入手, 发展起了一套基于 2D 系统的研究方法, 成为研究迭代学习控制的主要方法之一.

注意到系统总是运行有限时间长度 N 后便重置并再次重复, 因此可以把这有限个时刻的输入与输出信号通过堆积的办法构造造成如下形式, 称为超向量 (super-vector)

$$U_k = [u^T(0, k), u^T(1, k), \dots, u^T(N-1, k)]^T \quad (1.8)$$

$$Y_k = [y^T(1, k), u^T(2, k), \dots, y^T(N, k)]^T \quad (1.9)$$

记

$$G = \begin{bmatrix} CB & 0 & 0 & \cdots & 0 \\ CAB & CB & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ CA^{N-1}B & CA^{N-2}B & \cdots & \cdots & CB \end{bmatrix} \quad (1.10)$$

则易知

$$Y_k = GU_k + d \quad (1.11)$$

其中

$$d = [(Cx_0)^T, (CAx_0)^T, \dots, (CA^N x_0)^T]^T \quad (1.12)$$

类似于式 (1.8) 和式 (1.9), 可记

$$Y_d = [y^T(1, d), u^T(2, d), \dots, y^T(N, d)]^T$$

$$E_k = [e^T(1, k), e^T(2, k), \dots, e^T(N, k)]^T$$

则有

$$U_{k+1} = U_k + \mathbf{K}E_k \quad (1.13)$$

其中, $\mathbf{K} = \text{diag}\{K, K, \dots, K\}$. 此时

$$\begin{aligned} E_{k+1} &= Y_d - Y_{k+1} = Y_d - GU_{k+1} - d \\ &= Y_d - GU_k - GKE_k - d \\ &= E_k - GKE_k \\ &= (I - GK)E_k \end{aligned}$$

由此不难得出前面所述迭代学习控制收敛的充分条件, 即式 (1.7). 不仅如此, 从上式还可以看出, 保证跟踪误差渐近收敛所需条件为 GK 的谱半径小于 1, 换言之, 即 CBK 的谱半径小于 1. 实际上, 堆积模型的意义并不仅仅在于帮助我们得到收敛条件, 更重要的是, 它使得我们可以更为本质地看待迭代学习控制. 在堆积模型 (1.11) 中, 同一迭代过程内的演化过程被集成到各个记号中, 从而突出了迭代过程的意义所在. 换言之, 堆积模型 (1.11) 实际上是一个沿 k 变化的过程, 而 t 此时已经没有任何影响.

注记 1.2 由于迭代学习控制的关注点集中在如何沿迭代过程逐步改善系统的跟踪性能. 从其控制律的设计模式或上述基于超向量的模型 (1.11) 都可以看出这一点. 因此粗略地说, 系统由线性时不变模型拓展到线性时变模型一般不会产生新的难点. 这是因为, 当固定一个时刻后, 沿迭代过程观察在这一点上的更新情况, 本质上是一个时不变模型.

跟踪目标 $y(t, d)$ 通常为可实现的, 即对合适的初始值 x_0 , 存在输入 $u(t, d)$, 使得式 (1.1) 中的 k 替换为 d 仍成立. 换言之, $Y_d = GU_d + d$, 其中 U_d 的定义类似于式 (1.8). 那么研究输出收敛到目标轨迹的问题 $\lim_{k \rightarrow \infty} Y_k = Y_d$, 便可转化为研究输入收敛到目标输入的问题 $\lim_{k \rightarrow \infty} U_k = U_d$. 对输出含有随机噪声的系统, 这种方法更为方便.

注记 1.3 若跟踪目标不可实现, 这意味着不存在控制输入能够产生目标轨迹, 即无论如何是不能实现完全跟踪的. 此时迭代学习控制律的设计目标也不再是实现渐近精确跟踪目标, 而是渐近收敛到一个距离目标轨迹最近的轨迹上去. 因此, 跟踪问题变成一个优化问题. 另外, 从实际应用的角度来看, 跟踪目标通常都是可实现的, 这一假定并不苛刻.

上述讨论都是围绕线性系统情形给出的, 而针对非线性系统的迭代学习控制问题也引起了众多研究^[27-29]. 因其自身内在非线性特性的制约, 针对非线性系统的迭代学习控制难度要明显高于线性系统. 一个明显的佐证是, 大多数研究非线性系统的文献都要求非线性函数满足全局 Lipschitz 条件 (global Lipschitz condition,

GLC), 而这一条件限制了相关研究结果的适用范围. 因此, 如何进一步放宽这一条件是非线性系统迭代学习控制研究的重要方向.

已有文献的研究主题涵盖了迭代学习控制的众多方面, 包括更新律的设计、初始重置条件的放宽、鲁棒性、最优性、瞬态性能表现、变化轨道的跟踪, 以及实际应用等. 这里不再一一展开细述, 有兴趣的读者可以通过相关综述文献了解研究脉络. 例如, 文献 [14] 中, 作者对 1998~2004 年的迭代学习控制文献从理论和应用两个方面进行了详细分类. 文献 [30] 从范数优化的角度综述了相关系列研究.

1.1.2 随机迭代学习控制

本节将给出本书中随机迭代学习控制的含义. 在本书中, 随机迭代学习控制指对模型中含有随机信号的系统的迭代学习控制. 这里随机信号指用概率论中的随机变量描述的信号, 如系统运行过程的噪声、输出量测噪声、随机延迟信号等. 用随机变量描述的信号同时具有不确定性与随机性, 其值往往没有一个明确的上界.

具体而言, 本书所述的随机迭代学习研究主要分为如下几类.

(1) 含有系统噪声及/或量测噪声的线性系统.

(2) 含有系统噪声及/或量测噪声的非线性系统.

(3) 含有其他随机信号的系统, 如带随机丢包的网络控制系统、带随机异步信号的大规模系统等.

目前, 关于确定性系统已经有较为深入的研究, 得到了许多有意义的结果, 并已应用于实际工程. 然而, 在实际系统中总是不可避免地存在各种各样的扰动或噪声. 因此, 考虑扰动抑制的迭代学习控制是一个重要的研究方向. 不过, 这类研究目前多是假定系统中的扰动有界, 进而证明跟踪误差是被一个与扰动上界有关的函数界住. 具体而言, 记系统扰动为 w_k , 假定 $\|w_k\| \leq \epsilon$, 其中 ϵ 为适当正数, 通过各种迭代学习控制器可以使跟踪误差 e_k 满足 $\|e_k\| \leq L(\epsilon)$, $L(\epsilon) \xrightarrow{\epsilon \rightarrow 0} 0$. 这方面的文献如文献 [31] 和文献 [32]. 也有部分文献考虑通过学习的方法来消除扰动. 然而这些方法都不能用于处理含有随机信号的系统.

因此, 要处理随机迭代学习控制问题, 需要在确定性的迭代学习控制研究基础上, 更多地考虑对随机变量的处理方法. 现在研究中已经得到了部分处理方法及收敛结果, 本章将在其余节中进行回顾, 但仍有大量的问题有待深入研究.

值得注意的是, 在随机迭代学习控制中, 由于有随机信号的存在, 相应的收敛结果表述往往有别于传统的收敛表述. 换言之, 在随机迭代学习控制中, 关于收敛性的表述往往与某种概率收敛结合在一起, 例如, 算法以概率 1 收敛到最优值, 算法依概率收敛到目标值, 或者算法均方收敛到最优值等. 此外, 若系统输出端含有随机噪声, 直观来看, 很难得到实际输出渐近收敛到 0 的收敛结果, 此时得到的最好的收敛结果应为除去输出噪声外, 实际输出与跟踪目标的差值渐近趋于 0. 换言

之,从长远来看,系统输出的跟踪误差应逐渐地仅由输出噪声导致.

1.2 线性随机系统

1.2.1 基于卡尔曼滤波的方法

考虑如下随机线性系统

$$\begin{aligned} x(t+1, k) &= A(t)x(t, k) + B(t)u(t, k) + w(t, k) \\ y(t, k) &= C(t)x(t, k) + v(t, k) \end{aligned} \quad (1.14)$$

其中, $A(t)$ 、 $B(t)$ 与 $C(t)$ 为时变系统矩阵; $w(t, k)$ 与 $v(t, k)$ 分别为系统噪声与量测噪声. 注意到系统 (1.14) 与系统 (1.1) 的区别之处除了系统矩阵变为时变外, 就是增加了噪声项, 因此, 传统的基于范数压缩的分析方法及条件 (1.7) 不再适用.

Saab 针对式 (1.14) 的迭代学习控制问题给出了一系列优秀的结果. 他在文献 [33] 中首次考虑了离散时间线性系统的随机迭代学习控制, 并在之后的一系列文献中不断完善相应的研究^[34-36]. 文献 [33] 中, 更新律设计为

$$u(t, k+1) = u(t, k) + K(t, k)[e(t+1, k) - e(t, k)] \quad (1.15)$$

其中, $K(t, k)$ 为学习增益矩阵.

记 1.4 实际上, 文献 [33] 中考虑的系统模型与式 (1.14) 略有差别, 不同之处在于系统量测方程除了噪声 $v(t, k)$ 外, 还有一项与时间无关的噪声项 $v_b(k)$. 但设计输入更新律时, 采用了如式 (1.15) 的 D 型迭代学习控制律, 噪声项 $v_b(k)$ 被直接消掉, 并在后续收敛性分析中无任何体现, 故可以不考虑这一项, 而直接考虑式 (1.14) 所示模型.

为分析这一迭代学习控制算法的收敛性, 需要几个假设条件. 首先跟踪目标 $y(t, d)$ 可实现, 需要此假设的原因及相关解释已经在注记 1.3 中给出. 换言之, 存在目标输入 $u(t, d)$ 及初始状态 $x(0, d) = x_0$ 使得

$$\begin{aligned} x(t+1, d) &= A(t)x(t, d) + B(t)u(t, d) \\ y(t, d) &= C(t)x(t, d) \end{aligned} \quad (1.16)$$

此外, 假设输入/输出耦合矩阵 $C(t+1)B(t)$ 为列满秩. 为在不引起混淆的前提下保持符号简洁, 在本章其余部分简记符号 $C^+B = C(t+1)B(t)$. 同时用前置符号 δ 的方式来表示目标值与实际值的差距, 即 $\delta x(t, k) \triangleq x(t, d) - x(t, k)$, $\delta u(t, k) \triangleq u(t, d) - u(t, k)$. 以 E 表示期望. 然后, 关于噪声、初始状态及初始批次输入的假设可以明确如下^[33]. $w(t, k)$ 与 $v(t, k)$ 均假定为零均值的高斯白噪声, 且

协方差阵 $Q_t = E[w(t, k)w(t, k)^T]$ 为半正定阵, 而 $R_t = E[v(t, k)v(t, k)^T]$ 为正定阵. 同时假定 $w(t, k)$ 与 $v(s, l)$ 不相关, $\forall t, s \in [0, N], k, l \in \mathbb{R}$. 初始状态假定为随机变量, 满足 $\delta x(0, k)$ 为零均值高斯白噪声且协方差阵 $P_{x,0} = E[\delta x(0, k)x(0, k)^T]$ 为半正定阵. 此外, $\delta x(0, k)$ 与各种噪声均不相关. 初始批次输入需要满足初始输入偏差 $\delta u(t, 0)$ 为零均值白噪声且其协方差阵 $E[\delta u(t, 0)\delta u(t, 0)^T] = P_{u,0}$ 为对称正定矩阵.

注记 1.5 上述关于噪声、初始状态及初始批次输入的假设主要来源于卡尔曼滤波方法的要求. 在这里, 我们对初始批次输入假设给出进一步的评论. 文献 [33] 关于初始输入的假设实际上相当于要求初始输入需要在目标输入 $u(t, d)$ 附近按照正态分布产生 (这里按正态分布产生可从重复试验的角度出发理解). 作者还指出, 一种简单的满足上述要求的情形是 $\delta u(t, 0) = 0, \forall t$. 事实上, 在系统信息未知的情形下, 满足这一要求非常困难. 迭代学习控制的本质思想更希望能够对任意初始输入, 通过不断地迭代学习来得到足够良好的输入信号. 从另一方面而言, $\delta u(t, 0) = 0, \forall t$ 相当于说明初始输入就已经选择目标输入 $u(t, d)$, 此时显然没有必要再继续更新, 因为从某种意义上而言 (例如, 从 $Ee(t, k) = 0$ 且 $E[e(t, k)e(t, k)^T] = \min$ 角度来看), $u(t, d)$ 已经是足够好的输入信号. Saab 的后续研究文献 [34]~文献 [38] 中同样存在这一限制. 这是一个值得研究的开放性问题.

针对系统 (1.14)、迭代学习控制律 (1.15) 及上述假设条件, 文献 [33] 通过随机最小化轨道误差得到了学习增益矩阵 $K(t, k)$ 在最小二乘意义下的计算算法, 并进而证明了算法的均方收敛性. 为得到这一结果, 作者首先推导得到如下 2D-Roessor 模型

$$X^+ = \Phi X + \Gamma Z \quad (1.17)$$

其中

$$\begin{aligned} X^+ &= \begin{bmatrix} \delta u(t, k+1) \\ \delta x(t+1, k) \end{bmatrix}, \quad X = \begin{bmatrix} \delta u(t, k) \\ \delta x(t, k) \end{bmatrix} \\ Z &= \begin{bmatrix} w(t, k) \\ v(t+1, k) - v(t, k) \end{bmatrix} \\ \Phi &= \begin{bmatrix} I - K(t, k)C^+B & K(t, k)[C(t) - C(t+1)A(t)] \\ B(t) & A(t) \end{bmatrix} \\ \Gamma &= \begin{bmatrix} K(t, k)C(t+1) & K(t, k) \\ -I & 0 \end{bmatrix} \end{aligned}$$

进而, 对 X^+ 的协方差阵 $P^+ \triangleq E(X^+X^{+T})$ 的迹关于 $K(t, k)$ 求导, 并令导数为 0,

即可得到关于 $K(t, k)$ 的递推算法如下

$$K(t, k) = P_{u,k} \Xi^T (\Xi P_{u,k} \Xi^T + \Lambda_{D,k})^{-1} \quad (1.18)$$

$$P_{u,k+1} = (I - K(t, k) \Xi) P_{u,k} \quad (1.19)$$

其中, $\Xi \triangleq C(t+1)B(t)$, $\Lambda_{D,k} \triangleq (C(t) - C(t+1)A(t))P_{x,k}(C(t) - C(t+1)A(t))^T + C(t+1)Q_t C(t+1)^T + R_t + R_{t+1}$, $P_{x,t} = E[\delta x(t, k)\delta x(t, k)^T]$, $P_{u,k} = E[\delta u(t, k)\delta u(t, k)^T]$.

注记 1.6 实际上, 在文献 [33] 中, 分析 $K(t, k)$ 对 P^+ 的影响时, 注意到这样一个事实: 学习增益矩阵 $K(t, k)$ 实际上是用来更新得到 $(t, k+1)$ 时的输入, 因此 $K(t, k)$ 对 $(t+1, k)$ 位置的状态 $x(t+1, k)$ 并没有影响, 因此将 P^+ 的迹关于 $K(t, k)$ 求导相当于将 $P_{u,k+1}$ 的迹关于 $K(t, k)$ 求导. 事实上, 在后续研究中, 如文献 [35]~文献 [39] 中, 作者在推导 $K(t, k)$ 的递推公式时, 便使用了后一种思路.

下述定理为文献 [33] 的主要收敛定理.

定理 1.1 [33] 对系统 (1.14) 应用学习律 (1.15) 及递推算法 (式 (1.18)~式 (1.19)), 若 $C(t+1)B(t)$ 列满秩, 则 $\forall k, t$, 存在合适的范数使得 $\|I - K(t, h)C(t+1)B(t)\| < 1$, 从而 $\|P_{u,k+1}\| < \|P_{u,k}\|$. 进而, 当 $k \rightarrow \infty$ 时, 在 $[0, N]$ 上一致地有 $P_{u,k} \rightarrow 0$, $K(t, h) \rightarrow 0$.

Saab 的一系列研究文献 [34]~文献 [37] 均延续了这一研究方法: 首先建立输入差值 $\delta u(t, k)$ 及状态差值 $\delta x(t, k)$ 的 2D-Roessor 模型, 然后对待优化的目标协方差阵 (一般是输入误差协方差阵 $P_{u,k}$) 关于学习增益矩阵 $K(t, k)$ 求导, 并令导数为 0, 得到关于学习增益矩阵的递推算法. 最后, 基于噪声的互不相关性及其协方差阵的 (半) 正定性证明均方收敛性.

定理 1.1 建立起了基于卡尔曼滤波的随机迭代学习控制方法框架. 由定理 1.1 可以看出, 在 $C(t+1)B(t)$ 列满秩及其他合适条件下, 作者得到了关于学习增益矩阵的递推算法, 并且该增益矩阵满足类似于式 (1.7) 的压缩映射条件. 此外, 虽然学习更新律 (1.15) 为差分形式, 但定理 1.1 的结果表明算法对量测噪声并不敏感. 不过应当注意到, 为使得学习律 (1.15) 及递推算法 (式 (1.18)~式 (1.19)) 能够良好运行, 需要已知较多的系统信息, 包括系统矩阵 $A(t)$ 、 $B(t)$ 、 $C(t)$, 噪声协方差阵 Q_t 、 R_t , 以及状态误差协方差阵 $P_{x,t}$ 等 (见 $\Lambda_{D,k}$ 的表达式). 文献 [34]~文献 [36] 在放宽其中的部分信息已知要求上作出了一些改进结果.

针对算法 (1.18) 中需要已知 $P_{x,t}$ 的限制, 文献 [34] 研究了去掉这一项之后系统的收敛性. 具体而言, 将算法改写成

$$\tilde{K}(t, k) = \tilde{P}_{u,k} \tilde{\Xi}^T (\tilde{\Xi} \tilde{P}_{u,k} \tilde{\Xi}^T + \tilde{\Lambda}_D)^{-1} \quad (1.20)$$

$$\tilde{P}_{u,k+1} = (I - \tilde{K}(t, k) \tilde{\Xi}) \tilde{P}_{u,k} \quad (1.21)$$