致 读 者

本书由国防科技图书出版基金资助出版。

国防科技图书出版工作是国防科技事业的一个重要方面。优秀的国防科技图书既是国防科技成果的一部分，又是国防科技水平的重要标志。为了促进国防科技事业的发展，加强社会主义物质文明和精神文明建设，培养优秀科技人才，确保国防科技优秀图书的出版，国防科工委于1988年初决定每年拨出专款，设立国防科技图书出版基金，成立评审委员会，扶持、审定出版国防科技优秀图书。

国防科技图书出版基金资助的对象是：

1. 学术水平高，内容有创见，在学科上居领先地位的基础科学理论图书；在工程技术理论方面有突破的应用科学专著。

2. 学术思想新颖，内容明确、具体、有突出创见，对国防科技发展具有较大推动作用的专著；密切结合科学技术现代化和国防现代化需要的高科技内容的专著。

3. 有重要发展前景和有重大开拓使用价值，密切结合科学技术现代化和国防现代化需要的新技术、新工艺内容的科技图书。

4. 填补目前我国科学技术领域空白的薄弱学科的科技图书。

国防科技图书出版基金评审委员会在国防科工委的领导下开展评审工作，职责是：负责掌握出版基金的使用方向，评审受理的图书选题，决定资助的图书选题和资助金额，以及决定中断或取消资助等。经评审给予资助的图书，由国防工业出版社列选出版。

国防科技事业已经取得了举世瞩目的成就。国防科技图书承担着记载和弘扬这些成就，积累和传播科技知识的使命。在改革

开放的新形势下，国防科工委率先设立出版基金，扶持出版科技图书，这是一项具有深远意义的创举。此举势必促使国防科技图书的出版随着国防科技事业的发展更加兴旺。

设立出版基金是一件新生事物，是对出版工作的一项改革。因而，评审工作需要不断地摸索、认真地总结和及时地改进，这样，才能使有限的基金发挥出巨大的效能。评审工作更需要国防科技工业战线广大科技工作者、专家、教授，以及社会各界朋友的热情支持。

让我们携起手来，为祖国昌盛、科技腾飞、出版繁荣而共同奋斗！

国防科技图书出版基金
评审委员会

国防科技图书出版基金
第一届评审委员会组成人员

前　言

预测控制是70年代后期产生的一类新型计算机控制算法。它的问世，一方面是受到了计算机技术发展的推动，另一方面也来自复杂工业实践向高层优化控制提出的挑战。10多年来，随着它处理复杂系统控制的策略思想日益为人们所认识，以及它在工业实践中的大量成功应用，这一控制技术的生命力及诱人的应用前景已引起了控制理论界和工业控制界的广泛兴趣。本书的撰写，就是为了适应这方面的需要，通过全面介绍预测控制的机理、算法、设计要点及其在复杂工业系统中的应用技巧，推动预测控制在我国的研究和应用，使这一新型技术对我国国民经济的发展和工业自动化水平的提高产生积极的影响。

本书力图从多方面对预测控制作出全面的描绘。全书内容由三部分组成，一是从概貌上介绍预测控制的产生、机理、研究现状及发展前景（第一、二、九章）；二是介绍预测控制的基础算法及系统的分析与设计（第三、四、五章），这部分一方面对常用的典型算法进行了细致的描述，另一方面对预测控制系统进行了深入的理论分析，为系统设计提供了有益的参考；三是介绍复杂系统中的预测控制（第六、七、八章），即从应用的角度介绍预测控制的实用化算法及其应用实例。这三个部分分别侧重于从总体概念、基础算法及理论分析、实用算法及应用技巧上描绘预测控制，但又相互渗透，有机结合。本书既使读者了解预测控制的方法论，又反映出具体的理论问题及应用技巧。这将有助于读者跳出单一算法框框的束缚，加深对预测控制本质的理解，正确认识预测控制的实际地位和价值，在较高的视野上拓宽研究和应用预测控制的思路。

预测控制发展到今日，已不能仅仅理解为是一种或几种控制

算法，而要从更高的层次上，把它作为一种控制方法来理解。虽然在这种方法框架下，新的算法层出不穷，但由于篇幅所限，本书只能以某些典型算法为例进行介绍。在这里，作者选择了动态矩阵控制算法，这是因为，它是产生最早、影响最大、应用最广的预测控制算法之一。虽然如此，书中所介绍的原理也同样适用于其他算法。基于同样的理由，书中对预测控制的实用化算法及应用实例的介绍，也只能选择有限的典型例子，而不可能将各种策略和实例均包含在内。作者希望这种取材的有限性，不致影响本书对这一领域全貌的反映。

本书是作者近10年来在该领域的研究工作的结晶，同时书中也综合了近年来该领域在国际上的发展动态。在这里，特别要感谢导师施密特（G.Schmidt）教授和张钟俊教授对作者在这一领域研究的引导和支持。几年来，作者的同事和学生们，特别是许晓鸣博士和谢剑英副教授，与作者一起在这一领域中作出了持续有效的努力，华东师范大学袁震东教授，中国石油大学袁璞教授等许多国内同行与作者进行了极为有益的讨论，均为本书的撰写奠定了基础，作者在此表示衷心的感谢。

作者还要感谢妻子和父母对作者工作极为可贵的支持。近10年来，他们为此所作出的巨大牺牲作者将终身难忘。

本书中的研究工作先后受到德国科学研究基金（DFG）、我国国家自然科学基金的资助，作者谨在此一并表示感谢。

预测控制作为一种新型优化控制技术，在我国的工业和其他领域内有着广阔的应用前景。作者希望，本书的出版能为推动预测控制在我国的研究和应用起到一定的作用。由于作者水平所限，书中定有不少不尽人意之处，衷心希望广大读者提出宝贵意见。

席裕庚　于上海交通大学

1991年10月

目 录

第一章 引言 …… 1

第二章 预测控制的基本原理 …… 5

第三章 几种典型的预测控制算法 …… 10

§3.1 动态矩阵控制 …… 10

§3.2 模型算法控制 …… 18

§3.3 广义预测控制 …… 28

第四章 预测控制系统的分析 …… 36

§4.1 动态矩阵控制的状态空间分析 …… 36

§4.2 动态矩阵控制系统的内模控制结构 …… 43

§4.3 动态矩阵控制系统的动态特性分析 …… 52

§4.4 动态矩阵控制系统的稳定性和鲁棒性分析 …… 61

第五章 预测控制系统的参数设计 …… 78

§5.1 动态矩阵控制的参数设计 …… 78

§5.2 典型工业过程动态矩阵控制的解析设计 …… 92

第六章 多变量预测控制算法 …… 107

§6.1 多变量系统的动态矩阵控制 …… 107

§6.2 多变量动态矩阵控制的算法实现及参数整定 …… 115

§6.3 多变量动态矩阵控制的解耦设计 …… 120

§6.4 有约束的多变量动态矩阵控制 …… 128

§6.5 多变量系统的分散预测控制 …… 138

第七章 预测控制在实际应用中的发展 …… 148

§7.1 具有前馈-反馈结构的预测控制 …… 148

§7.2 串级预测控制 …… 152

§7.3 带有自校正的预测控制算法 …… 159

§7.4 高维大系统的递阶预测控制 …… 164

§7.5 利用无穷范数优化的鲁棒预测控制 …… 172

§7.6 非线性系统的预测控制 …… 179

第八章　预测控制的工业应用实例 ……………………………………189
　§8.1　汽轮发电机组蒸汽系统的控制 ……………………………………191
　§8.2　炼油厂加氢裂化单元的动态矩阵控制 ……………………………194
　§8.3　天然气传输网络的在线优化 ………………………………………201
　§8.4　工业机器人的预测函数控制 ………………………………………207
第九章　预测控制的发展前景 ……………………………………………215
参考文献 ……………………………………………………………………220

第一章　引　　言

70年代后期，在美、法等国的工业过程领域内出现了一类新型计算机控制算法，如动态矩阵控制（DMC）、模型算法控制（MAC）。这类算法以对象的阶跃或脉冲响应为模型，采用滚动推移的方式在线地对过程实现优化控制，在复杂的工业过程中显现出良好的控制性能。1978年，理查勒特（Richalet）等在文献[1]中，首次详细阐述了这类算法产生的动因、机理及其在工业过程中的应用效果。从此，预测控制（Predictive Control）作为这类新型控制算法的统一名称，便开始出现在控制领域中。

预测控制的产生，并不是理论发展的需要，而首先是工业实践向控制提出的挑战。众所周知，60年代初形成的现代控制理论，在航天、航空等领域都取得了辉煌的成果。利用状态空间法分析和设计系统，提高了人们对被控对象的洞察能力，提供了在更高层次上设计控制系统的手段。特别是，立足于最优性能指标的设计理论和方法已趋成熟，这对于在工业过程中追求更高控制质量和经济效益的控制工程师来说，无疑具有极大的吸引力。然而，人们不久就发现，在完美的理论与控制实践之间还存在着巨大的鸿沟。这主要表现在以下几个方面：

（1）现代控制理论的基点是对象精确的数学模型，而在工业过程中所涉及的对象往往是多输入、多输出的高维复杂系统，其数学模型很难精确建立，即使建立了模型，从工程实用的角度来说，往往需要简化，从而很难保证得到对象精确的模型。

（2）工业对象的结构、参数和环境都具有很大的不确定性。由于这些不确定性的存在，按照理想模型得到的最优控制在实际上往往不能保持最优，有时甚至会引起控制品质的严重下降。在工业环境中，人们更关注的是控制系统在不确定性影响下保持良

好性能的能力，即所谓鲁棒性，而不能只是追求理想的最优性。

（3）工业控制中必须考虑到控制手段的经济性，对工业控制计算机的要求不能太高。因此，控制算法必须简易以满足实时性的要求。而现代控制理论的许多算法往往过于复杂，难以用低性能的计算机实现。

这些来自实际的原因，阻碍了现代控制理论在复杂工业过程中的有效应用，也向控制理论提出了新的挑战。

为了克服理论与应用之间的不协调，70年代以来，除了加强对系统辨识、模型简化、自适应控制、鲁棒控制等的研究外，人们开始打破传统方法的约束，试图面对工业过程的特点，寻找各种对模型要求低、控制综合质量好、在线计算方便的优化控制新算法。在此期间，数字计算机技术的飞速发展，也为新算法的产生提供了物质基础。预测控制就是在这种背景下发展起来的一类新型计算机优化控制算法。

最早产生于工业过程的预测控制算法，有理查勒特、梅拉（Mehra）等提出的建立在脉冲响应基础上的模型预测启发控制（Model Predictive Heuristic Control，简称为MPHC）[1]，或称模型算法控制（Model Algorithmic Control，简称为MAC）[2]，以及卡特勒（Cutler）等提出的建立在阶跃响应基础上的动态矩阵控制（Dynamic Matrix Control，简称为DMC）[3]。由于这类响应易于从工业现场直接获得，并不要求对模型的结构有先验知识，所以不必通过复杂的辨识过程便可设计控制系统。这些预测控制算法汲取了现代控制理论中的优化思想，但用不断的在线有限优化，即所谓滚动优化，取代了传统的最优控制。由于在优化过程中利用实测信息不断进行反馈校正，所以在一定程度上克服了不确定性的影响，增强了控制的鲁棒性。此外，这些算法的在线计算比较简易。这些特点使它们很适合于工业过程控制的实际要求。70年代后期，模型算法控制和动态矩阵控制分别在锅炉和分馏塔的控制以及石油加工的生产装置上获得了成功的应用，从而引起了工业控制界的广泛兴趣。此后，基于对象脉冲或

阶跃响应的各种预测控制算法相继出现，有些算法已形成了商品化软件包，在石油、化工、电力等领域的过程控制中取得了明显的经济效益。

除了直接来自工业控制的以对象非参数模型为基础的预测控制算法外，还出现了另一类算法。80年代初期，人们在自适应控制的研究中发现，为了克服最小方差控制的弱点，有必要汲取预测控制中的多步预测优化策略，这样可以大大增强算法的适用性和鲁棒性，因此出现了基于辨识模型并带有自校正的预测控制算法，如扩展时域预测自适应控制（Extended Prediction Self-Adaptive Control，简称为EPSAC）[4]、广义预测控制（Generalized Predictive Control，简称为 GPC）[5]等。此外，莫拉里（Morari）等 1982 年在研究一类新型控制结构——内模控制（Internal Model Control，简称为 IMC）[6]时，发现预测控制算法与这类控制结构有着密切的联系，从而从结构的角度对预测控制作了更深入的研究。这些研究和应用，有力地推动了预测控制的进一步发展。现在我们所说的预测控制，就包括了上述来自工业控制、自适应控制及内模控制等多方面的研究成果。它们统称为模型预测控制，或基于模型的控制（Model-based Control），其应用范围也已超出了过程控制领域，而应用到机器人、飞行器、网络系统等更广泛的领域内。

近年来，国内外对预测控制的研究和应用日趋广泛。从1984年起，每年的美国控制年会（ACC）上都有关于预测控制的专题组。1987年召开的第 10 届国际自控联(IFAC)世界大会上，专题讨论了预测控制及其应用，被认为是特别吸引人的两个专题讨论之一。1988 年，IFAC 又组织了以预测控制为主题的“基于模型的过程控制”工作讨论会。关于预测控制及其应用的文献越来越多地出现在各种控制杂志和会议上。特别在过程控制界，已把预测控制作为当前过程控制的发展方向之一。此外，包含有预测控制的多变量优化控制算法已在国外许多大公司得到应用。在我国，近年来也有许多单位开展了预测控制的研究，取得了不少新的成

果，并在工业过程中获得了初步成功的应用。这些事实表明，预测控制已成为当前控制理论界和工业控制界均十分关注的一个热门课题。

由于预测控制对于复杂系统的适应性，它在工业过程和其他领域内有着诱人的应用前景。作为一种有前途的新型控制方法，预测控制在我国的深入研究和推广应用，必将对国民经济的发展产生显著的推动作用。

第二章　预测控制的基本原理

在介绍具体预测控制算法之前，我们首先要对这类算法的一般轮廓作一描绘，以使读者了解什么样的控制算法可称为预测控制算法。

首先应该指出，预测控制是以计算机为实现手段的，因此其算法一般应为采样控制算法而不是连续控制算法。顾名思义，预测控制应包含预测的原理。在传统的采样控制中，有些算法也用到了预测的原理。例如，利用史密斯（Smith）预估器可以把未经纯滞后的对象输出提前反馈给PID控制，其实质就是对输出作了预测。又如在离散最优控制中，性能指标涉及到未来时刻的状态或输出，也需要利用状态方程模型对这些量进行预算。但是这些算法都不能称为预测控制算法。那么，预测控制算法应具备什么特征呢？

就一般的意义来说，预测控制不论其算法形式如何不同，都应建立在下述三项基本原理基础上[7]。

1．预测模型

预测控制是一种基于模型的控制算法，这一模型称为预测模型。预测模型的功能是根据对象的历史信息和未来输入预测其未来输出。这里只强调模型的功能而不强调其结构形式。因此，状态方程、传递函数这类传统的模型都可以作为预测模型。对于线性稳定对象，甚至阶跃响应、脉冲响应这类非参数模型，也可直接作为预测模型使用。此外，非线性系统、分布参数系统的模型，只要具备上述功能，也可在对这类系统进行预测控制时作为预测模型使用。

预测模型具有展示系统未来动态行为的功能，这样，我们就可像在系统仿真时那样，任意地给出未来的控制策略，观察对象

在不同控制策略下的输出变化（见图2-1）。从而为比较这些控制策略的优劣提供了基础。

2. 滚动优化

预测控制是一种优化控制算法，它是通过某一性能指标的最优来确定未来的控制作用的。这一性能指标涉及到系统未来的行为，例如，通常可取对象输出在未来的采样点上跟踪某一期望轨迹的方差为最小；但也可取更广泛的形式，例如要求控制能量为最小而同时保持输出在某一给定范围内等等。性能指标中涉及到的系统未来的行为，是根据预测模型由未来的控制策略决定的。

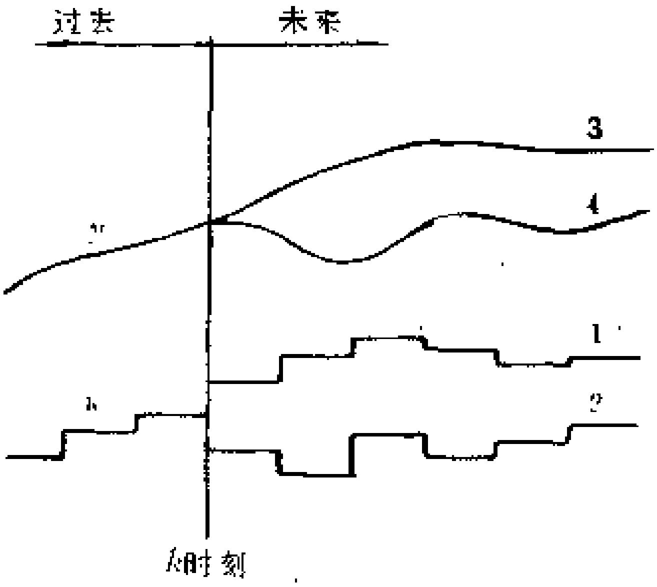


图2-1　基于模型的预测

1—控制策略Ⅰ；2—控制策略Ⅱ；3—对应于Ⅰ的输出；4—对应于Ⅱ的输出。

然而，需要强调的是，预测控制中的优化与传统意义下的离散最优控制有很大的差别。这主要表现在预测控制中的优化是一种有限时段的滚动优化。在每一采样时刻，优化性能指标只涉及到从该时刻起未来有限的时间，而到下一采样时刻，这一优化时段同时向前推移（见图2-2）。因此，预测控制不是用一个对全局相同的优化性能指标，而是在每一时刻有一个相对于该时刻的优化性能指标。不同时刻优化性能指标的相对形式是相同的，但其绝对形式，即所包含的时间区域，则是不同的。因此，在预测控制中，优化不是一次离线进行，而是反复在线进行的，这就是滚动优化的含义，也是预测控制区别于传统最优控制的根本点。

3. 反馈校正

预测控制是一种闭环控制算法。在通过优化确定了一系列未来的控制作用后，为了防止模型失配或环境干扰引起控制对理想状态的偏离，预测控制通常不是把这些控制作用逐一全部实施，

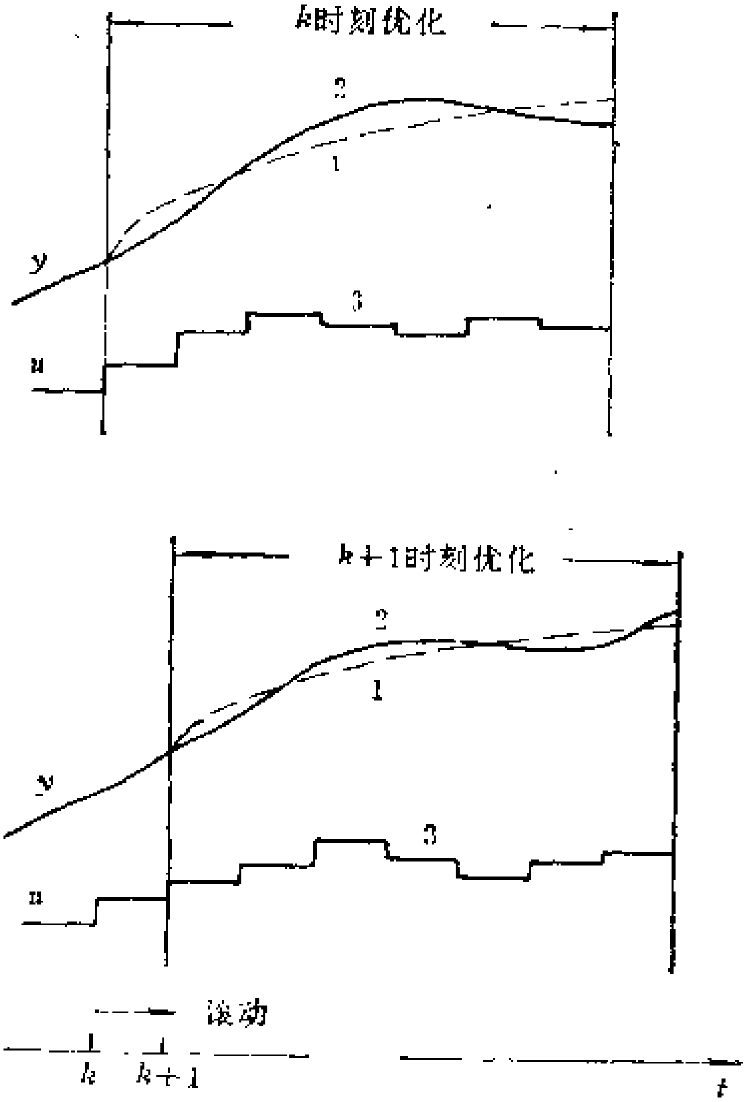


图2-2　滚动优化

1—参考轨迹；2—最优预测输出；3—最优控制作用。

而只是实现本时刻的控制作用。到下一采样时刻，则首先检测对象的实际输出，并利用这一实时信息对基于模型的预测进行修正，然后再进行新的优化（见图2-3）。

反馈校正的形式是多样的，可以在保持预测模型不变的基础上，对未来的误差作出预测并加以补偿，也可以根据在线辨识的原理直接修改预测模型。不论取何种校正形式，预测控制都把优化建立在系统实际的基础上，并力图在优化时对系统未来的动态行为作出较准确的预测。因此，预测控制中的优化不仅基于模型，而且利用了反馈信息，因而构成了闭环优化。

综上所述可以看到，预测控制作为一种新型计算机控制算

法，是有其鲜明特征的，它是一种基于模型、滚动实施并结合反馈校正的优化控制算法。

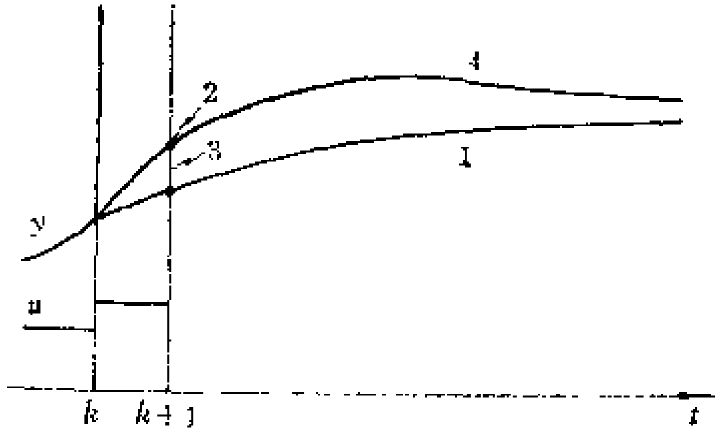


图2-3 误差校正

1—k时刻的预测输出；2—k+1时刻实际输出，3—预测误差；4—k+1时刻校正后的预测输出。

根据以上对预测控制一般原理的介绍，我们不难理解它在复杂的工业环境中受到青睐的原因。首先，对于复杂的工业对象，由于辨识其最小化模型要花费很大的代价，往往给基于传递函数或状态方程的控制算法带来困难。而预测控制所需要的模型只强调其预测功能，不苛求其结构形式，从而为系统建模带来了方便。在许多场合下，我们只需测定对象的阶跃或脉冲响应，便可直接得到预测模型，而不必进一步导出其传递函数或状态方程，这对其工业应用无疑是有吸引力的。更重要的是，预测控制汲取了优化控制的思想，但利用滚动的有限时段优化取代了一成不变的全局优化。这虽然在理想情况下不能导致全局最优，但由于实际上不可避免地存在着模型误差和环境干扰，这种建立在实际反馈信息基础上的反复优化，能不断顾及不确定性的影响并及时加以校正，反而要比只依靠模型的一次优化更能适应实际过程，有更强的鲁棒性。所以，预测控制是针对传统最优控制在工业过程中的不适用性而进行修正的一种新型优化控制算法。

预测控制的这种优化控制原理，实际上反映了人们在处理带有不确定性问题时的一种通用思想方法。例如，人们在穿越马路时，不必去看左右很远处有无车辆，而只需看近几十米处。但还需边走边看，以防近处开出新的车辆或远处车速加快且原来估计不足而发生意外。这里就包含了建立在反馈信息基础上的反复决策过程。又如，当我们作长途旅行时，当然可以凭借已有的车船时

刻表（模型）作一最优计划，以便尽早地到达目的地。然而，由于车船可能更改时刻（模型失配）或因故晚点（干扰），因此，若按原计划班次换乘，则可能会顺延一日或数日，这样不但不会最优，反而会大大延误旅行时间。因此，我们往往只是根据靠近的几个中转地点的车船班次作一规划（有限时段优化），在到达下一地点时，再根据实际到达时间及对下几个中转地点车船班次的了解重新规划。这样行一站，看几站，反复规划，即使时刻表不准确或晚点，仍能较快地到达目的地。实际上，这种对于复杂对象的滚动优化思想早已出现在管理领域中。企业管理中常采用的滚动计划，其思想与上面介绍的基本原理是一致的。预测控制正是汲取了其中包含的方法原理，并把它与控制算法结合起来，从而能有效地应用于复杂系统的控制。

第三章　几种典型的预测控制算法

在预测控制一般原理的基础上，采用不同的模型形式、优化策略和校正措施，可以形成不同的预测控制算法。在本章中，我们将首先介绍三种最有影响的预测控制算法：动态矩阵控制(DMC)[3]、模型算法控制（MAC)[2]和广义预测控制(GPC)[5]。为了使读者透过具体算法更深入地理解其原理，这里只介绍单变量系统的基本算法。至于多变量系统及其他复杂情况，将在以后的章节中讨论。

§3.1　动态矩阵控制

从 1974 年起，动态矩阵控制（DMC）就作为一种有约束的多变量优化控制算法应用在美国壳牌石油公司的生产装置上。1979 年，卡特勒等在美国化工年会上首次介绍了这一算法。10 多年来，它已在石油、化工等部门的过程控制中获得了成功的应用。在本节中，我们将在对原始算法加以推广的基础上，介绍其算法原理。

DMC 算法是一种基于对象阶跃响应的预测控制算法，它适用于渐近稳定的线性对象。对于弱非线性对象，可在工作点处首先线性化；对于不稳定对象，可先用常规 PID 控制使其稳定，然后再使用 DMC 算法。

DMC 控制包括下述三个部分。

1. 预测模型

在 DMC 中，首先需要测定对象单位阶跃响应的采样值 $a_i=a(iT)$，$i=1,2,\cdots$。其中，T为采样周期。对于渐近稳定的对象，阶跃响应在某一时刻 $t_N=NT$ 后将趋于平稳，以致 $a_i(i>N)$与 a_N 的误差和量化误差及测量误差有相同的数量级，因而可认为，a_N 已近似等于阶跃响应的稳态值 $a_s=a(\infty)$。这

样，对象的动态信息就可以近似用有限集合$\{a_1, a_2, \cdots, a_N\}$加以描述。这个集合的参数构成了DMC的模型参数，向量$\boldsymbol{a}=[a_1, \cdots, a_N]^T$称为模型向量，$N$则称为建模时域。

虽然阶跃响应是一种非参数模型，但由于线性系统具有比例和叠加性质，故利用这组模型参数$\{a_i\}$，已足以预测对象在未来的输出值。在k时刻，假定控制作用保持不变时对未来N个时刻的输出有初始预测值$\tilde{y}_0(k+i|k)$，$i=1, \cdots, N$，(例如在稳态起动时便可取$\tilde{y}_0(k+i|k)=y(k)$)，则当k时刻控制有一增量$\Delta u(k)$时，即可算出在其作用下未来时刻的输出值：

$$\tilde{y}_1(k+i|k)=\tilde{y}_0(k+i|k)+a_i\Delta u(k),$$
$$i=1, \cdots, N \tag{3-1}$$

同样，在M个连续的控制增量$\Delta u(k), \cdots, \Delta u(k+M-1)$作用下未来各时刻的输出值为

$$\tilde{y}_M(k+i|k)=\tilde{y}_0(k+i|k)+\sum_{j=1}^{\min(M, i)} a_{i-j+1}$$
$$\times \Delta u(k+j-1), i=1, \cdots, N \tag{3-2}$$

其中，y的下标表示控制量变化的次数，$k+i|k$表示在k时刻对$k+i$时刻的预测。显然，在任一时刻k，只要知道了对象输出的初始预测值$\tilde{y}_0(k+i|k)$，就可根据未来的控制增量由预测模型(3-2)计算未来的对象输出。在这里，式(3-1)只是预测模型(3-2)在$M=1$情况下的特例。

2. 滚动优化

DMC是一种以优化确定控制策略的算法。在每一时刻k，要确定从该时刻起的M个控制增量$\Delta u(k), \cdots, \Delta u(k+M-1)$，使被控对象在其作用下未来$P$个时刻的输出预测值$\tilde{y}_M(k+i|k)$尽可能接近给定的期望值$w(k+i)$，$i=1, \cdots, P$(见图3-1)。这里，$M$、$P$分别称为控制时域与优化时域，它们的意义可从图3-1中直接看出。为了使问题有意义，通常规定$M \leqslant P \leqslant N$。

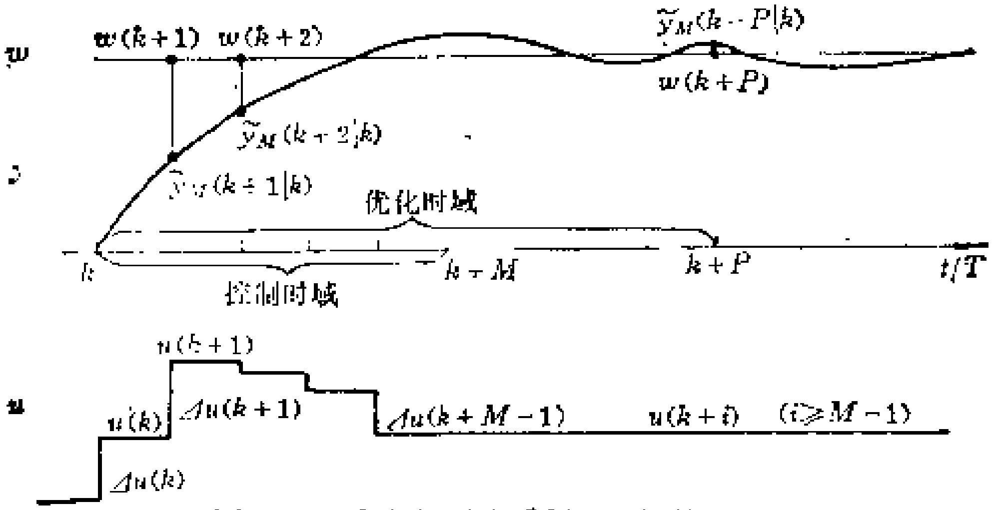


图3-1　动态矩阵控制的优化策略

在控制过程中，往往不希望控制增量 Δu 变化过于剧烈，这一因素可在优化性能指标中加入软约束予以考虑。因此，k 时刻的优化性能指标可取为

$$\min J(k)=\sum_{i=1}^{P} q_i[w(k+i)-\tilde{y}_M(k+i|k)]^2+\sum_{j=1}^{M} r_j\Delta u^2(k+j-1) \tag{3-3}$$

其中 q_i，r_j 是权系数，它们分别表示对跟踪误差及控制量变化的抑制。

在不考虑约束的情况下，上述问题就是以 $\Delta \boldsymbol{u}_M(k)=[\Delta u(k)\cdots\Delta u(k+M-1)]^T$ 为优化变量，在动态模型（3-2）下使性能指标（3-3）最小的优化问题。为了求解这一优化问题，首先可利用预测模型（3-2）导出性能指标中 $\tilde{y}$ 与 Δu 的关系，这一关系可用向量形式写为

$$\tilde{\boldsymbol{y}}_{PM}(k)=\tilde{\boldsymbol{y}}_{P0}(k)+\boldsymbol{A}\Delta \boldsymbol{u}_M(k) \tag{3-4}$$

其中

$$\tilde{\boldsymbol{y}}_{PM}(k)=\begin{bmatrix}\tilde{y}_M(k+1|k)\\ \vdots \\ \tilde{y}_M(k+P|k)\end{bmatrix}\qquad \tilde{\boldsymbol{y}}_{P0}(k)=\begin{bmatrix}\tilde{y}_0(k+1|k)\\ \vdots \\ \tilde{y}_0(k+P|k)\end{bmatrix}$$

$$A=\begin{bmatrix} a_1 & & 0 \\ \vdots & \ddots & \\ a_M & \cdots\cdots & a_1 \\ \vdots & & \\ a_P & \cdots & a_{P-M+1} \end{bmatrix}$$

这里，A是由阶跃响应系数a_i组成的$P\times M$阵，称为动态矩阵。式中向量$\tilde{y}$的前一个下标表示所预测的未来输出的个数，后一个下标则为控制量变化的次数。

同样，性能指标（3-3）也可写成向量形式

$$\min J(k)=\|\boldsymbol{w}_P(k)-\tilde{\boldsymbol{y}}_{PM}(k)\|_Q^2+\|\Delta\boldsymbol{u}_M(k)\|_R^2 \qquad (3\text{-}5)$$

其中

$$\boldsymbol{w}_P(k)=[w(k+1)\cdots w(k+P)]^T$$

$$\boldsymbol{Q}=\mathrm{diag}(q_1\cdots q_P)$$

$$\boldsymbol{R}=\mathrm{diag}(r_1\cdots r_M)$$

由权系数构成的对角阵$\boldsymbol{Q}$、$\boldsymbol{R}$分别称为误差权矩阵和控制权矩阵。

以式（3-4）代入式（3-5），可得

$$\min J(k)=\|\boldsymbol{w}_P(k)-\tilde{\boldsymbol{y}}_{P0}(k)-\boldsymbol{A}\Delta\boldsymbol{u}_M(k)\|_Q^2+\|\Delta\boldsymbol{u}_M(k)\|_R^2$$

在k时刻，$\boldsymbol{w}_P(k)$，$\tilde{\boldsymbol{y}}_{P0}(k)$均为已知，使$J(k)$取极小的$\Delta\boldsymbol{u}_M(k)$可通过极值必要条件$dJ(k)/d\Delta\boldsymbol{u}_M(k)=0$求得

$$\Delta\boldsymbol{u}_M(k)=(\boldsymbol{A}^T\boldsymbol{Q}\boldsymbol{A}+\boldsymbol{R})^{-1}\boldsymbol{A}^T\boldsymbol{Q}[\boldsymbol{w}_P(k)-\tilde{\boldsymbol{y}}_{P0}(k)] \qquad (3\text{-}6)$$

它给出了$\Delta u(k)$，…，$\Delta u(k+M-1)$的最优值。但DMC并不把它们都当作应实现的解，而只是取其中的即时控制增量$\Delta u(k)$构成实际控制$u(k)=u(k-1)+\Delta u(k)$作用于对象。到下一时刻，它又提出类似的优化问题求出$\Delta u(k+1)$。这就是所谓"滚动优化"的策略。

根据式（3-6），可以求出

$$\Delta u(k)=\boldsymbol{c}^T\Delta\boldsymbol{u}_M(k)=\boldsymbol{d}^T[\boldsymbol{w}_P(k)-\tilde{\boldsymbol{y}}_{P0}(k)] \qquad (3\text{-}7)$$

其中，P维行向量

$$\boldsymbol{d}^T=\boldsymbol{c}^T(\boldsymbol{A}^T\boldsymbol{Q}\boldsymbol{A}+\boldsymbol{R})^{-1}\boldsymbol{A}^T\boldsymbol{Q}\triangleq[d_1\cdots d_P] \qquad (3\text{-}8)$$

称为控制向量[●]。M维行向量$c^T=[1\ 0\ \cdots\ 0]$ 表示取首元素的运算。一旦优化策略确定（即P、M、Q、R已定），则d^T可由式（3-8）一次离线算出。这样，若不考虑约束，优化问题的在线求解就简化为直接计算控制律（3-7），它只涉及到向量之差及点积运算，因而是十分简易的。

3. 反馈校正

当k时刻把控制$u(k)$实际加于对象时，相当于在对象输入端加上了一个幅值为$\Delta u(k)$的阶跃，利用预测模型（3-1），可算出在其作用下未来时刻的输出预测值

$$\tilde{y}_{N1}(k)=\tilde{y}_{N0}(k)+a\Delta u(k) \tag{3-9}$$

它实际上就是式（3-1）的向量形式，其中N维向量$\tilde{y}_{N1}(k)$和$\tilde{y}_{N0}(k)$的构成及含义同前述相似。由于$\tilde{y}_{N1}(k)$的元素是未加入$\Delta u(k+1)$，…，$\Delta u(k+M-1)$时的输出预测值，故经移位后，它们可作为$k+1$时刻的初始预测值进行新的优化计算。然而，由于实际存在模型失配、环境干扰等未知因素，由式（3-9）给出的预测值有可能偏离实际值，因此，若不及时利用实时信息进行反馈校正，进一步的优化就会建立在虚假的基础上。为此，在DMC中，到下一采样时刻首先要检测对象的实际输出$y(k+1)$，并把它与由式（3-9）算出的模型预测输出$\tilde{y}_1(k+1|k)$相比较，构成输出误差

$$e(k+1)=y(k+1)-\tilde{y}_1(k+1|k) \tag{3-10}$$

这一误差信息反映了模型中未包括的不确定因素对输出的影响，可用来预测未来的输出误差，以补充基于模型的预测。由于对误差的产生缺乏因果性的描述，故误差预测只能采用时间序列方法，例如，可采用对$e(k+1)$加权的方式修正对未来输出的预测：

$$\tilde{y}_{cor}(k+1)=\tilde{y}_{N1}(k)+he(k+1) \tag{3-11}$$

其中

● 式（3-8）中符号$\triangleq$的含义为“记作”。

$$\tilde{\boldsymbol{y}}_{cor}(k+1)=\begin{bmatrix}\tilde{y}_{cor}(k+1\mid k+1)\\ \vdots\\ \tilde{y}_{cor}(k+N\mid k+1)\end{bmatrix}$$

为校正后的输出预测向量，由权系数组成的N维向量$\boldsymbol{h}=[h_1\cdots h_N]^T$称为校正向量。

在$k+1$时刻，由于时间基点的变动，预测的未来时间点也将移到$k+2,\cdots,k+1+N$，因此，$\tilde{\boldsymbol{y}}_{cor}(k+1)$的元素还需通过移位才能成为$k+1$时刻的初始预测值：

$$\tilde{y}_0(k+1+i\mid k+1)=\tilde{y}_{cor}(k+1+i\mid k+1)$$
$$i=1,\cdots,N-1 \tag{3-12}$$

而$\tilde{y}_0(k+1+N\mid k+1)$由于模型的截断，可由$\tilde{y}_{cor}(k+N\mid k+1)$近似。这一初始预测值的设置可用向量形式表示为

$$\tilde{\boldsymbol{y}}_{N0}(k+1)=\boldsymbol{S}\tilde{\boldsymbol{y}}_{cor}(k+1) \tag{3-13}$$

其中

$$\boldsymbol{S}=\begin{bmatrix}0 & 1 & & 0\\ & \ddots & \ddots & \\ & & & 1\\ 0 & & 0 & 1\end{bmatrix}$$

为移位阵。

有了$\tilde{\boldsymbol{y}}_{N0}(k+1)$，又可像上面那样进行$k+1$时刻的优化计算，求出$\Delta u(k+1)$。整个控制就是以这种结合反馈校正的滚动优化方式反复在线进行的，其算法结构可见图 3-2。

由图 3-2 可见，DMC 算法是由预测、控制、校正三部分构成的。图中粗箭头表示向量流，细箭头表示纯量流。在每一采样时刻，未来P个时刻的期望输出$\boldsymbol{w}_P(k)$与初始预测输出$\tilde{\boldsymbol{y}}_{P0}(k)$构成的偏差向量同动态控制向量$\boldsymbol{d}^T$点乘（见式（3-7)），得到该时刻的控制增量$\Delta u(k)$。这一控制增量一方面通过数字积分（累加）运算求出控制量$u(k)$并作用于对象，另一方面与模型向量$\boldsymbol{a}$相乘并按式（3-9）计算出在其作用后的预测输出$\tilde{\boldsymbol{y}}_{N1}(k)$。到下一采样时刻，首先检测对象的实际输出$y(k+1)$，并与预测值$\tilde{y}_1(k+1\mid k)$相比较后按式(3-10)构成输出误差$e(k+1)$。这一误差与校正向量$\boldsymbol{h}$相乘作为误差预测，再与模型预测一起

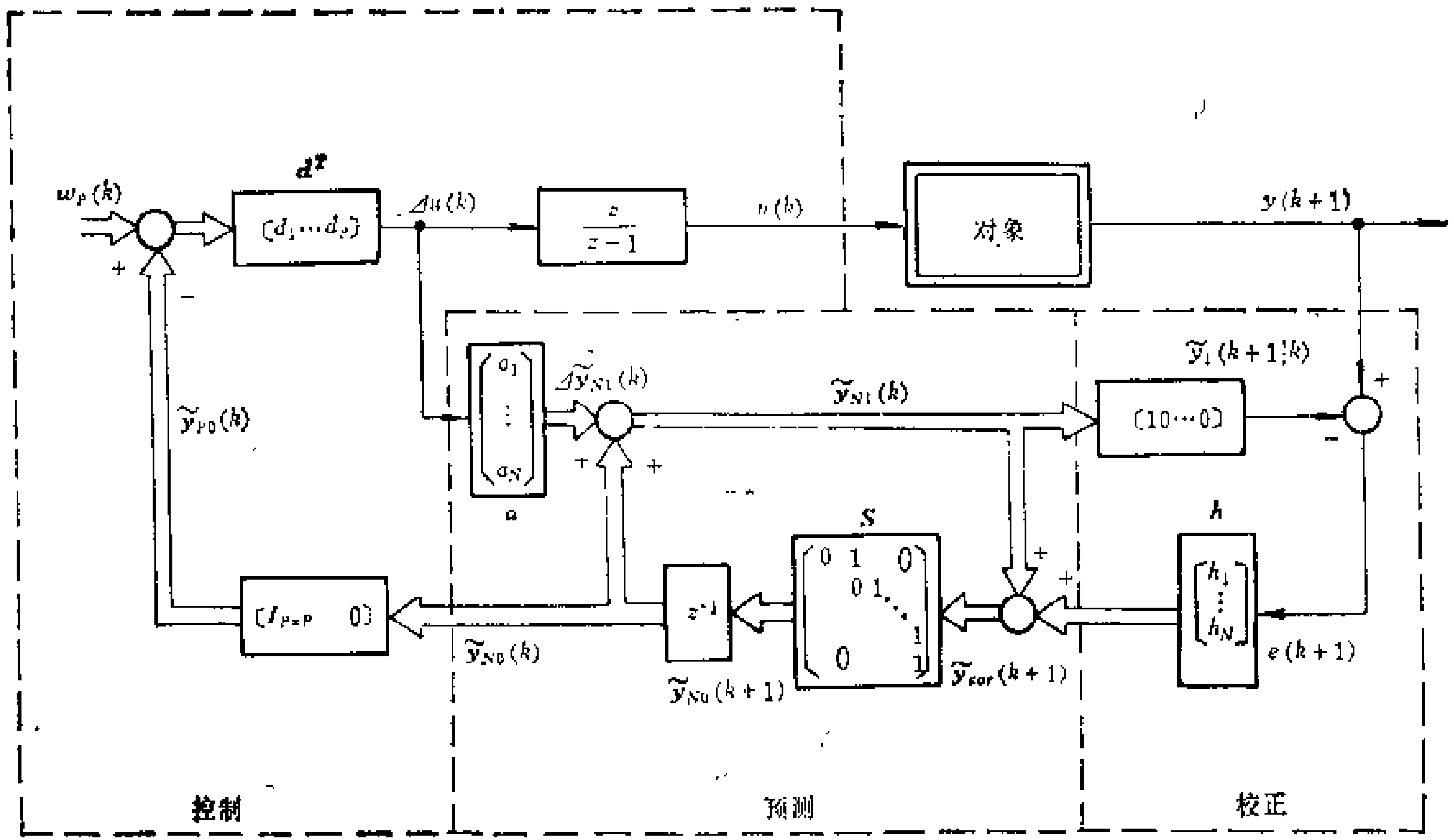

$w_P(k)$
d^T
$[d_1 \cdots d_P]$
$\Delta u(k)$
$\frac{z}{z-1}$
$u(k)$
对象
$y(k+1)$
$\widetilde{y}_{P0}(k)$
a_1
a_N
a
$\Delta\widetilde{y}_{N1}(k)$
$\widetilde{y}_{N1}(k)$
$\widetilde{y}_1(k+1|k)$
$[10\cdots0]$
S
z^{-1}
$[I_{P\times P} \quad 0]$
$\widetilde{y}_{N0}(k)$
$\widetilde{y}_{N0}(k+1)$
$\widetilde{y}_{cor}(k+1)$
h
h_1
h_N
$e(k+1)$
控制
预测
校正

图3-2　动态矩阵控制

按式(3-11)得到校正后的预测输出$\tilde{y}_{cor}(k+1)$，并按式(3-13)移位后作为新的初始预测值$\tilde{y}_{N0}(k+1)$。图3-2中，z^{-1}表示时间基点的记号后推一步，这样等于把新的时刻重新定义为k时刻，整个过程将反复在线进行。

下面具体介绍一下DMC的算法实现。由于DMC是一种基于模型的控制，并且应用了在线优化的原理，与PID算法相比，显然它需要作更多的离线准备工作，这主要包括以下三个方面：

（1）检测对象的阶跃响应并经光滑后得到模型系数$a_1, \cdots, a_N$。在这里，应强调模型的动态响应必须是光滑的，测量噪声和干扰必须滤除，否则会影响控制质量甚至造成不稳定。

（2）利用仿真程序确定优化策略，并根据式（3-8）算出控制系数$d_1, \cdots, d_P$。

（3）选择校正系数$h_1, \cdots, h_N$。

这三组动态系数确定后，应置入固定的内存单元，以便实时调用。

DMC的在线计算由初始化模块与实时控制模块组成。初始化模块是在投入运行的第一步检测对象的实际输出$y(k)$，并把它设定为预测初值$\tilde{y}_0(k+i|k)$，

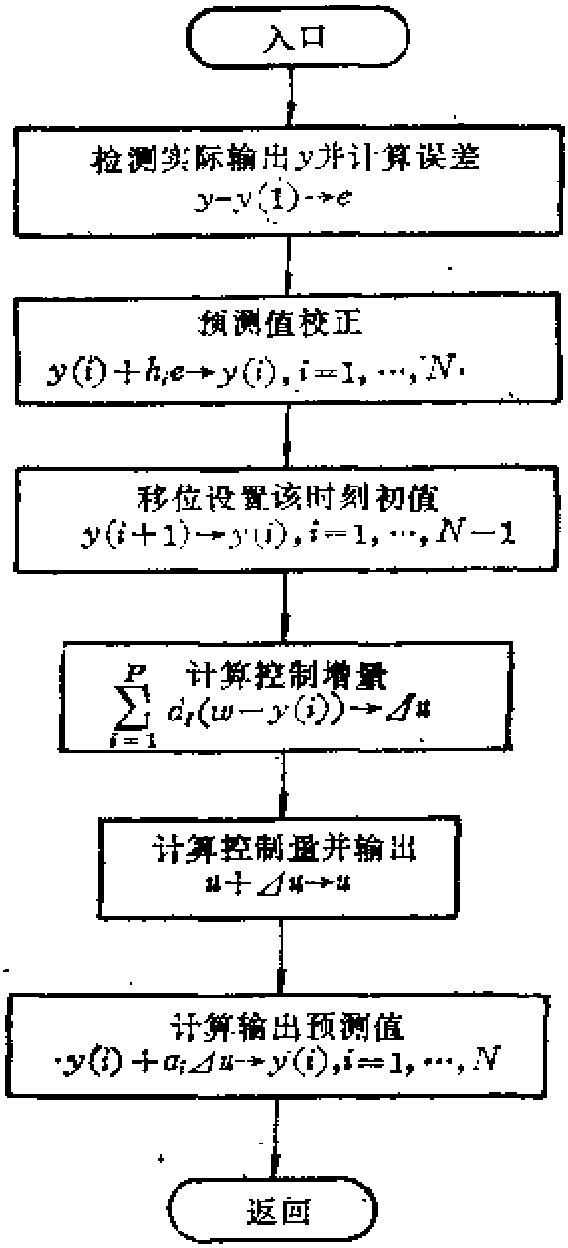


图3-3 动态矩阵控制的在线计算流程

$i=1$，…，N。从第二步起即转入实时控制模块，在每一采样时刻的在线计算流程可见图 3-3，其中对未来输出的预测值只需设置一个N维数组$y(i)$，流程图中的算式依次对应于式(3-10)，式(3-11)，式(3-13)，式(3-7)，由增量算控制量和式(3-9)。

注意，在上图中，设定值w是定值并事先置入内存。若需跟踪时变的轨线，则还应编制一设定值模块，以在线计算每一时刻的期望值$w(i)$，$i=1$，…P，并以此代替流程图中的w。

§ 3.2 模型算法控制

模型算法控制（MAC）又称模型预测启发控制（MPHC），是由梅拉和理查勒特等在 70 年代后期提出的另一类预测控制算法。它已在美、法等国的许多工业过程（如电厂锅炉、化工精馏塔等）的控制中取得了显著的成效，受到了过程控制界的广泛重视。

与 DMC 相同，MAC 也适用于渐近稳定的线性对象，但其设计前提不是对象的阶跃响应，而是其脉冲响应。

一、具有简易性能指标的 MAC 算法

MAC 的控制算法由以下几个部分组成[2]：

1. 预测模型

对于线性对象，如果已知其单位脉冲响应的采样值g_1，g_2，…，则可根据离散卷积公式，写出其输入输出间的关系

$$y(k+i)=\sum_{j=1}^{\infty} g_j u(k+i-j) \qquad (3\text{-}14)$$

这里u、y分别是输入量、输出量相对于稳态工作点的偏移值。对于渐近稳定的对象，由于$\lim\limits_{j\to\infty} g_j=0$，故总能找到一个时刻$t_N=NT$，使得这以后的脉冲响应值$g_j(j>N)$与测量和量化误差有相同的数量级，以致实际可视为 0 而予以忽略。这样，对象

的动态就可近似地用一个有限项卷积表示的预测模型描述

$$y_M(k+i)=\sum_{j=1}^{N} g_j u(k+i-j) \tag{3-15}$$

这一模型可用来预测对象在未来时刻的输出值，其中 y 的下标 M 表示模型输出。由于模型向量 $\boldsymbol{g}=[g_1\ \cdots\ g_N]^T$ 通常存放在计算机的内存中，故文献中有时也称它为内部模型。

2. 参考轨迹

在MAC中，控制系统的期望输出是由从现时实际输出 $y(k)$ 出发且向设定值 c 光滑过渡的一条参考轨迹规定的。在 k 时刻的参考轨迹可由其在未来采样时刻的值 $y_r(k+i)$，$i=1,2,\cdots$ 来描述，它通常可取作一阶指数变化的形式。这时

$$y_r(k+i)=y(k)+[c-y(k)](1-e^{-iT/\tau}),$$
$$i=1,2,\cdots$$

其中，下标 r 表示参考输出；τ 是参考轨迹的时间常数；T 为采样周期。若记 $\alpha=\exp(-T/\tau)$，则上式亦可写作

$$y_r(k+i)=\alpha^i y(k)+(1-\alpha^i)c,\quad i=1,2,\cdots \tag{3-16}$$

如果 $c=y(k)$，则对应着镇定问题，而 $c\neq y(k)$ 则对应着跟踪问题。

显然，τ 越小，则 α 越小，参考轨迹就能越快地到达设定点 c。α 是MAC中的一个重要设计参数，它对闭环系统的动态特性和鲁棒性都有关键作用。

3. 滚动优化

在MAC中，k 时刻的优化准则是要选择未来 P 个控制量，使在未来 P 个时刻的预测输出 y_P（下标 P 表示预测）尽可能接近由参考轨迹所确定的期望输出 y_r（参见图3-4）。这一优化性能指标可写作

$$\min J(k)=\sum_{i=1}^{P}\omega_i[y_P(k+i)-y_r(k+i)]^2 \tag{3-17}$$

式中，P称为优化时域，ω_i为非负权系数，它们决定了各采样时刻的误差在性能指标$J(k)$中所占的比重。

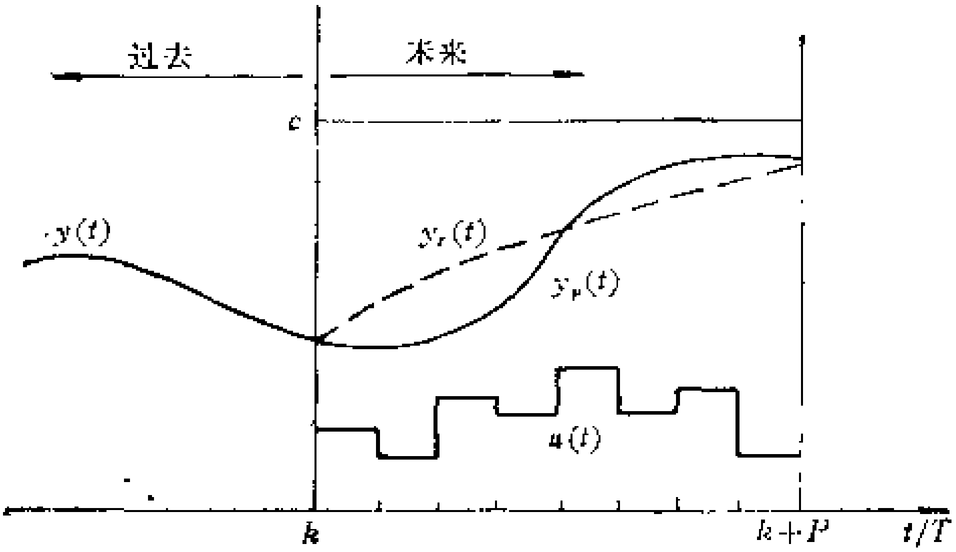


图3-4　模型算法控制的参考轨迹与优化

为了得到式(3-17)中的预测输出值y_P，可采取两种策略：

（1）开环预测

这是直接把由预测模型(3-15)计算的模型输出y_M当作预测输出，即

$$y_P(k+i)=y_M(k+i),\quad i=1,\cdots,P \tag{3-18}$$

根据式(3-15)，可写出它们的详细表达式

$$\left.\begin{aligned}
y_P(k+1)&=g_1u(k)+g_2u(k-1)+\cdots+g_Nu(k+1-N)\\
y_P(k+2)&=g_1u(k+1)+g_2u(k)+\cdots+g_Nu(k+2-N)\\
&\vdots\\
y_P(k+P)&=g_1u(k+P-1)+g_2u(k+P-2)+\cdots\\
&\qquad+g_Nu(k+P-N)
\end{aligned}\right\} \tag{3-19}$$

将上式代入性能指标(3-17)，并且注意到$u(k)$，…，$u(k+P-1)$是待确定的优化变量，在一般情况下，就可通过优化算法求出它们，并将即时控制量$u(k)$作用于实际对象。

这一算法的结构可见图3-5中不带虚线的部分。由于y_P的

计算没有用到实际输出信息y而只依赖于模型输出，故称为开环预测。

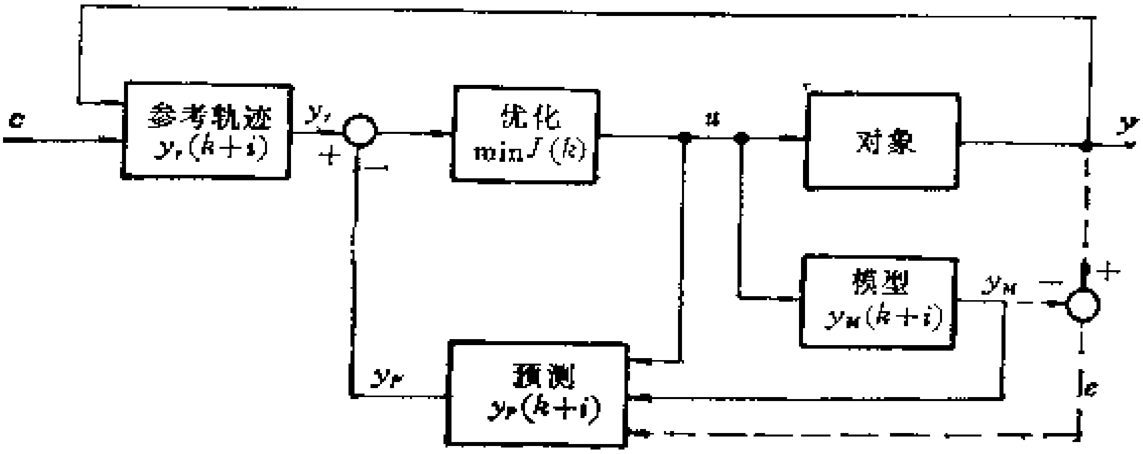


图3-5　模型算法控制的开环预测和闭环预测

如果不考虑约束并且对象无纯滞后或非最小相位特性，则上述优化问题的求解可简化为令性能指标（3-17）中的各项误差为零，并逐项递推求出$u(k)$，$u(k+1)$，…。这时，无论优化时域P取多大，即时最优控制量$u(k)$的计算只取决于

$$y_P(k+1)=y_r(k+1)$$

由此可求得

$$u(k)=\frac{y_r(k+1)-g_2u(k-1)-\cdots-g_Nu(k-N+1)}{g_1} \tag{3-20}$$

在这种情况下，一步优化与P步优化所得的即时控制律是相同的。

上述开环预测的明显缺点是：当存在模型误差时，由于模型预测的不准确，将会产生静差。为了说明这一点，我们把对象的实际脉冲响应系数用向量$\tilde{\boldsymbol{g}}^T=[\tilde{g}_1\cdots\tilde{g}_N]^T$表示，$\tilde{\boldsymbol{g}}\neq\boldsymbol{g}$。考虑最简单的无约束一步优化情况，这时

预测输出

$$\begin{aligned}y_P(k+1)&=y_M(k+1)=\boldsymbol{g}^T\boldsymbol{u}(k)\\&=y_r(k+1)=\alpha y(k)+(1-\alpha)c\end{aligned}$$

实际输出

$$y(k+1)=\tilde{\boldsymbol{g}}^T\boldsymbol{u}(k)$$

其中，$\boldsymbol{u}(k)=[u(k)\cdots u(k-N+1)]^T$。当控制达到稳态时，$y(t)$、$u(t)$均保持为常量不再变化，我们将其分别记作$y_s$、$u_s$，则由上述两式可得

$$\left(\sum_{i=1}^{N} g_i\right)u_s=\alpha y_s+(1-\alpha)c$$

$$y_s=\left(\sum_{i=1}^{N} \tilde{g}_i\right)u_s$$

由此可得

$$y_s=\frac{(1-\alpha)\left(\sum_{i=1}^{N} \tilde{g}_i\right)c}{\left(\sum_{i=1}^{N} g_i\right)-\alpha\left(\sum_{i=1}^{N} \tilde{g}_i\right)}$$

只要$\sum_{i=1}^{N} \tilde{g}_i \neq \sum_{i=1}^{N} g_i$，输出与设定值间就存在静差

$$d_s=c-y_s=\frac{\left(\sum_{i=1}^{N} g_i\right)-\left(\sum_{i=1}^{N} \tilde{g}_i\right)}{\left(\sum_{i=1}^{N} g_i\right)-\alpha\left(\sum_{i=1}^{N} \tilde{g}_i\right)}c$$

因此，有必要以实测的输出信息构成闭环预测，以校正对未来输出的预测值。

（2）闭环预测

闭环预测与开环预测（3-18）的差别在于：在构成输出预测值y_P时，除了利用模型预测值y_M外，还附加了一误差项e，其一般形式为

$$y_P(k+i)=y_M(k+i)+e(k),\quad i=1,\cdots,P \tag{3-21}$$

其中，$e(k)$可由k时刻的实际输出$y(k)$与模型输出$y_M(k)$的误差构成：

$$e(k)=y(k)-y_M(k)$$

$$=y(k)-\sum_{j=1}^{N} g_j u(k-j) \tag{3-22}$$

闭环预测的实质就相当于DMC中的误差校正。由于采用了反馈校正原理，它可以在模型失配时有效地消除静差。仍考虑前面讨论的无约束一步优化算法，这时的预测输出改变为

$$y_P(k+1)=y_M(k+1)+e(k)$$

$$=\mathbf{g}^T\mathbf{u}(k)+y(k)-\mathbf{g}^T\mathbf{u}(k-1)$$

$$=y_r(k+1)=\alpha y(k)+(1-\alpha)c$$

当达到稳态时，有

$$\left(\sum_{i=1}^{N} g_i\right)u_s+y_s-\left(\sum_{i=1}^{N} g_i\right)u_s=\alpha y_s+(1-\alpha)c$$

或

$$y_s=c$$

即控制是无静差的。这一带有反馈校正的闭环预测相当于在图3-5中引入了虚线所示的反馈部分。实际的MAC无一例外地采用了闭环预测的策略。

在考虑无约束一步优化时，采用闭环预测的最优控制量可通过

$$y_M(k+1)+e(k)=y_r(k+1)$$

导出，其表达式为

$$u(k)=[\alpha y(k)+(1-\alpha)c-y(k)$$

$$+\sum_{i=1}^{N} g_i u(k-i)-\sum_{i=2}^{N} g_i u(k+1-i)]/g_1$$

$$=[(1-\alpha)(c-y(k))+g_N u(k-N)$$

$$+\sum_{i=1}^{N-1}(g_i-g_{i+1})u(k-i)]/g_1$$

(3-23)

显然，在计算机内存中只需存储固定的参数 g_1，g_1-g_2，…，$g_{N-1}-g_N$，g_N，过去 N 个时刻的控制输入 $u(k-1)$，…，$u(k-N)$ 以及参考轨迹参数 α、c，在每一时刻检测 $y(k)$ 后，即可由式（3-23）算出 $u(k)$。其算法流程在此不再赘述。

上述一步优化的MAC算法虽然简单，但它不适用于有时滞或非最小相位特性的对象，因为前者 $g_1=0$ 将使式（3-23）失效，后者则会引起不稳定的控制。此外，由于控制律（3-23）的计算对 g_1 十分敏感，对于小的 g_1 值，很小的模型误差就会引起 $u(k)$ 大幅度偏离最优值，控制效果将明显变差。所以，这种一步优化算法很难为实际工业控制所接受。为了使MAC能形成实用的工业控制算法，有必要采用多步优化的策略。然而前面已经指出，在取性能指标为式（3-17），在 P 步优化中允许控制量发生 P 次变化，并且不考虑约束时，P 步优化和一步优化所得的即时控制律是相同的，仍不能用于时滞或非最小相位对象。所以，在多步优化时，应考虑采用不同的优化时域 P 和控制时域 M，并把性能指标修改为如下更一般的形式：

$$\min J(k)=\sum_{i=1}^{P} q_i[y_P(k+i)-y_r(k+i)]^2+\sum_{j=1}^{M} r_j u^2(k+j-1) \tag{3-24}$$

下面，我们讨论这种一般形式下的无约束多步优化MAC算法。

二、具有一般性能指标的MAC算法

1. 预测模型

注意到当取 $M<P$ 时，意味着 $u(k+i)$ 在 $i=M-1$ 后保持不变，即

$$u(k+i)=u(k+M-1),\quad i=M,\ \cdots,\ P-1$$

因此，对未来输出的模型预测可写作

$$y_M(k+1)=g_1u(k)+g_2u(k-1)+\cdots+g_Nu(k+1-N)$$
$$\vdots$$
$$y_M(k+M)=g_1u(k+M-1)+g_2u(k+M-2)+\cdots+g_Nu(k+M-N)$$
$$y_M(k+M+1)=(g_1+g_2)u(k+M-1)+g_3u(k+M-2)+\cdots+g_Nu(k+M+1-N)$$
$$\vdots$$
$$y_M(k+P)=(g_1+\cdots+g_{P-M+1})u(k+M-1)+g_{P-M+2}u(k+M-2)+\cdots+g_Nu(k+P-N)$$

上面的式子可用向量形式简记为

$$\mathbf{y}_M(k)=\mathbf{G}_1\mathbf{u}_1(k)+\mathbf{G}_2\mathbf{u}_2(k) \tag{3-25}$$

其中

$$\mathbf{y}_M(k)=[y_M(k+1)\cdots y_M(k+P)]^T$$
$$\mathbf{u}_1(k)=[u(k)\cdots u(k+M-1)]^T$$
$$\mathbf{u}_2(k)=[u(k-1)\cdots u(k+1-N)]^T$$

$$\mathbf{G}_1=\begin{pmatrix} g_1 & & & & \\ g_2 & g_1 & \ddots & & 0 \\ \vdots & \vdots & & g_1 & \\ & & \vdots & & g_1 \\ & & & & \vdots \\ g_P & g_{P-1} & \cdots g_{P-M+2} & & (g_1+\cdots+g_{P-M+1}) \end{pmatrix}_{(P\times M)}$$

$$\mathbf{G}_2=\begin{pmatrix} g_2 & \cdots & & g_N \\ g_3 & \cdots & g_N & 0 \\ \vdots & & \iddots & \\ g_{P+1} & \cdots & g_N & 0 \end{pmatrix}_{(P\times(N-1))}$$

注意，在预测模型（3-25）中，$\mathbf{G}_1$，$\mathbf{G}_2$是由模型参数g_i构成的已知矩阵，$N-1$维向量$\mathbf{u}_2(k)$是由k时刻以前的输入信息组成的已知向量，而M维向量$\mathbf{u}_1(k)$则为所要求的现时和未来的控制输入量。

2．参考轨迹

k时刻的参考轨迹采样值可用向量形式写作

$$\boldsymbol{y}_r(k)=[y_r(k+1)\cdots y_r(k+P)]^T \tag{3-26}$$

其中

$$y_r(k+i)=\alpha^i y(k)+(1-\alpha^i)c\,,\quad i=1,\cdots,P$$

3. 闭环预测

k 时刻对输出的闭环预测可记为

$$\boldsymbol{y}_P(k)=\boldsymbol{y}_M(k)+\boldsymbol{h}\,e(k) \tag{3-27}$$

其中

$$\boldsymbol{y}_P(k)=[y_P(k+1)\cdots y_P(k+P)]^T$$

$$\boldsymbol{h}=[h_1\cdots h_P]^T$$

$$e(k)=y(k)-y_M(k)=y(k)-\sum_{j=1}^{N} g_j u(k-j)$$

这里也采用了加权的误差补偿办法。

4. 最优控制律

根据预测模型（3-25）、参考轨迹(3-26)和闭环预测(3-27)，可求出在性能指标（3-24）下的无约束 MAC 最优控制律

$$\boldsymbol{u}_1(k)=(\boldsymbol{G}_1^T\boldsymbol{Q}\boldsymbol{G}_1+\boldsymbol{R})^{-1}\boldsymbol{G}_1^T\boldsymbol{Q}\cdot[\boldsymbol{y}_r(k)-\boldsymbol{G}_2\boldsymbol{u}_2(k)-\boldsymbol{h}e(k)] \tag{3-28}$$

其中，$\boldsymbol{Q}=\mathrm{diag}(q_1,\cdots,q_P)$，$\boldsymbol{R}=\mathrm{diag}(r_1,\cdots,r_M)$。

最优即时控制量为

$$u(k)=\boldsymbol{d}^T[\boldsymbol{y}_r(k)-\boldsymbol{G}_2\boldsymbol{u}_2(k)-\boldsymbol{h}\,e(k)] \tag{3-29}$$

其中

$$\boldsymbol{d}^T=[1\quad 0\cdots 0]\;(\boldsymbol{G}_1^T\boldsymbol{Q}\boldsymbol{G}_1+\boldsymbol{R})^{-1}\boldsymbol{G}_1^T\boldsymbol{Q}$$

可以看出，多步优化的 MAC 算法与 DMC 算法的推导十分相似，其中有些不同之处，如参考轨迹的引入，误差校正的加权等，两者可相互借用。但下述两个不同点必须加以注意：

第一，MAC 控制律（3-29）中的 $\boldsymbol{d}^T$，其计算与 DMC 中计算 $\boldsymbol{d}^T$ 的式（3-8）十分相似，但矩阵 $\boldsymbol{G}_1$ 中并不是简单地以脉冲响应系数 g_i 取代式(3-8)动态矩阵 $\boldsymbol{A}$ 中的阶跃响应系数 a_i，它的最后一列用到了 g_i 的和，这与 $\boldsymbol{A}$ 中最后一列的形式是不对应的。其原因在于，DMC 以 Δu 为控制输入，在控制时域后的 $\Delta u=0$，不再考虑其阶跃响应的影响，而在 MAC 中则以 u 为控制输入，

在控制时域后 u 不再变化，但 $u=u(k+M-1)\neq 0$，仍需考虑其脉冲响应的叠加。

第二，即使在没有模型误差即 $e(k)=0$ 时，上述多步 MAC 算法一般也存在静差，因为在到达稳态时，根据

$$u(k)=\boldsymbol{d}^T[\boldsymbol{y}_r(k)-\boldsymbol{G}_2\boldsymbol{u}_2(k)]$$

可得

$$\begin{aligned}&u_s[1+d_1(g_2+\cdots+g_N)+\cdots+d_P(g_{P+1}+\cdots+g_N)]\\&=(d_1\alpha+\cdots+d_P\alpha^P)y_s+[d_1(1-\alpha)+\cdots+d_P(1-\alpha^P)]c\end{aligned}$$

而

$$y_s=\left(\sum_{i=1}^{N}g_i\right)u_s$$

故可得

$$y_s=\left\{\left(\sum_{i=1}^{N}g_i\right)[d_1(1-\alpha)+\cdots+d_P(1-\alpha^P)]\right\}\cdot c\Bigg/$$

$$\left\{\left(\sum_{i=1}^{N}g_i\right)[d_1(1-\alpha)+\cdots+d_P(1-\alpha^P)]+[1-d_1g_1-\cdots-d_P(g_1+\cdots+g_P)]\right\}$$

$$\triangleq\mu c$$

由此可见，只要 $1-d_1g_1-\cdots-d_P(g_1+\cdots+g_P)\neq 0$，系统的稳态输出 y_s 便存在静差

$$d_s=c-y_s=(1-\mu)c$$

可以证明，若在优化性能指标（3-24）中选择 $r_j=0$，则上述静差不再出现，因为这时有

$$[d_1\cdots d_P]=[1\ 0\ \cdots\ 0](\boldsymbol{G}_1^T\boldsymbol{Q}\boldsymbol{G}_1)^{-1}\boldsymbol{G}_1^T\boldsymbol{Q}$$

两边右乘 $\boldsymbol{G}_1$ 可得

$$[d_1\cdots d_P]\boldsymbol{G}_1=[1\ 0\ \cdots\ 0]$$

展开后有

$$d_1g_1+d_2g_2+\cdots+d_Pg_P=1$$

$$d_2g_1+\cdots+d_Pg_{P-1}=0$$

$$\vdots$$

$$d_M g_1+\cdots+d_P(g_1+\cdots+g_{P-M+1})=0$$

相加后可得

$$d_1g_1+d_2(g_1+g_2)+\cdots+d_P(g_1+\cdots+g_P)=1$$

因此，在不对控制量抑制时可导致无静差的控制。以上在讨论一步优化时证明了闭环预测可消除静差，正是因为性能指标(3-17)中没有考虑控制的加权。

MAC 算法在一般的性能指标下会出现静差，是由于它以 u 作为控制量，本质上导致了比例性质的控制。而 DMC 算法与此不同，它以 Δu 直接作为控制量，在控制中包含了数字积分环节，因而即使在模型失配的情况下，也能导致无静差的控制，这是 DMC 算法的显著优越之处。

§3.3 广义预测控制

广义预测控制（GPC）是在自适应控制的研究中发展起来的另一类预测控制算法。在过去 10 多年里，自校正控制技术受到了很大重视，并提出了不少新的算法。但它们对数学模型的精度都有一定的要求，有些算法（如最小方差自校正调节器）对于滞后十分灵敏，如果滞后估计不准或是时变的，控制精度将大大降低。另一些算法（如极点配置自校正调节器）则对系统的阶十分敏感，一旦阶数估计不准，算法将不能使用。这种对于模型精度的依赖性，使它们在难以精确建模的复杂工业过程中不能得到广泛有效的应用。而寻找对数学模型要求较低、鲁棒性强的自适应控制算法，自然成为这一领域中富有挑战性的课题。正是在这种背景下，克拉克（Clarke)等人在保持最小方差自校正控制的模型预测、最小方差控制、在线辨识等原理的基础上，汲取了 DMC、MAC 中的多步预测优化策略，提出了广义预测控制算法[5]。

作为一种自校正控制算法，GPC 是针对随机离散系统提出的。与 DMC 算法相比，虽然它们在滚动优化的性能指标方面有非常相似的形式，但 GPC 的模型形式与反馈校正策略同 DMC 都

有很大差别。下面，我们介绍这一算法的基本原理。

1. 预测模型

在GPC中，采用了最小方差控制中所用的受控自回归积分滑动平均（Controlled Auto-Regressive Integrated Moving Average，简称为CARIMA）模型来描述受到随机干扰的对象

$$A(q^{-1})y(t)=B(q^{-1})u(t-1)+\frac{C(q^{-1})\xi(t)}{\varDelta} \tag{3-30}$$

其中

$$A(q^{-1})=1+a_1q^{-1}+\cdots+a_nq^{-n}$$
$$B(q^{-1})=b_0+b_1q^{-1}+\cdots+b_{n_b}q^{-n_b}$$
$$C(q^{-1})=c_0+c_1q^{-1}+\cdots+c_{n_c}q^{-n_c}$$

式中，q^{-1}是后移算子，表示后退一个采样周期的相应的量；$\varDelta=1-q^{-1}$为差分算子；$\xi(t)$是一个不相关的随机序列，表示一类随机噪声的影响。A、B、C都是q^{-1}的多项式，其中多项式$B(q^{-1})$的若干首项元素b_0，b_1，…可以是零，以表示对象相应的时滞数。为了突出方法原理，这里假设$C(q^{-1})=1$。

式（3-30）实际上是用Z传递函数给出了对象的描述，由输入u到输出y间的Z传递函数为

$$G(z^{-1})=\frac{z^{-1}B(z^{-1})}{A(z^{-1})} \tag{3-31}$$

为了利用模型（3-30）导出j步后输出$y(t+j)$的预测值，首先考虑下述丢番图（Diophantine）方程

$$1=E_j(q^{-1})A\varDelta+q^{-j}F_j(q^{-1}) \tag{3-32}$$

其中，E_j、F_j是由$A(q^{-1})$和预测长度j唯一确定的多项式

$$E_j(q^{-1})=e_{j,0}+e_{j,1}q^{-1}+\cdots+e_{j,j-1}q^{-(j-1)}$$
$$F_j(q^{-1})=f_{j,0}+f_{j,1}q^{-1}+\cdots+f_{j,n}q^{-n}$$

这样，在式（3-30）两端乘以$E_j\varDelta q^j$后可得

$$E_jA\varDelta y(t+j)=E_jB\varDelta u(t+j-1)+E_j\xi(t+j)$$

利用式（3-32），可以写出$t+j$时刻的输出量

$$y(t+j)=E_jB\varDelta u(t+j-1)+F_jy(t)+E_j\xi(t+j)$$

注意到E_j、F_j的形式，可以知道$E_jB\Delta u(t+j-1)$与$u(t+j-1)$，$u(t+j-2)$，…有关，$F_jy(t)$只与$y(t)$，$y(t-1)$，…有关，而$E_j\xi(t+j)$与$\xi(t+j)$，…，$\xi(t+1)$有关。由于在t时刻未来的噪声$\xi(t+i)$，$i=1,\cdots,j$都是未知的，所以对$y(t+j)$最合适的预测值可由下式得到

$$\hat{y}(t+j\mid t)=E_jB\Delta u(t+j-1)+F_jy(t) \tag{3-33}$$

这样，根据已知的输入输出信息及未来的输入值，就可以预测对象未来的输出，式（3-33）就是GPC的预测模型。

在式（3-33）中，记$G_j=E_jB$，结合式（3-32）可得

$$G_j=\frac{B[1-q^{-j}F_j]}{A\Delta}$$

因此，多项式$G_j(q^{-1})$中前j项的系数正是对象阶跃响应前j项的采样值，记作$g_1,\cdots,g_j$。若把G_j展开写作

$$G_j(q^{-1})=g_{j,0}+g_{j,1}q^{-1}+\cdots$$

则有$g_{j,i}=g_{i+1}$（$i<j$）。

为了由式（3-33）计算预测值$\hat{y}(t+j\mid t)$，必须首先知道E_j，F_j，对于不同的$j=1,2,\cdots$，这相当于并行地求解一组丢番图方程（3-32），其计算量是很大的。为此，克拉克给出了一个E_j、F_j的递推算法。首先，根据式（3-32）可写出

$$1=E_jA\Delta+q^{-j}F_j$$

$$1=E_{j+1}A\Delta+q^{-(j+1)}F_{j+1}$$

两式相减可得

$$A\Delta(E_{j+1}-E_j)+q^{-j}(q^{-1}F_{j+1}-F_j)=0$$

记

$$\begin{aligned}\tilde{A}=A\Delta&=1+\tilde{a}_1q^{-1}+\cdots+\tilde{a}_{n+1}q^{-(n+1)}\\&=1+(a_1-1)q^{-1}+\cdots+(a_n-a_{n-1})q^{-n}-a_nq^{-(n+1)}\end{aligned}$$

$$E_{j+1}-E_j=\tilde{E}+e_{j+1,j}q^{-j}$$

则可得

$$\tilde{A}\tilde{E}+q^{-j}(q^{-1}F_{j+1}-F_j+\tilde{A}e_{j+1,j})=0$$

显然，必须有

$$\widetilde{E}=0$$

$$F_{j+1}=q\,(F_j-\widetilde{A}e_{j+1,j})$$

并且由于 $\widetilde{A}$ 的首项系数为 1，必然有

$$e_{j+1,j}=f_{j,0}$$

$$f_{j+1,i}=f_{j,i+1}-\tilde{a}_{i+1}e_{j+1,j}=f_{j,i+1}-\tilde{a}_{i+1}f_{j,0},\quad i=0,\cdots,n-1$$

$$f_{j+1,n}=-\tilde{a}_{n+1}e_{j+1,j}=-\tilde{a}_{n+1}f_{j,0}$$

这一 F_j 系数的递推关系亦可用向量形式记为

$$\boldsymbol{f}_{j+1}=\widetilde{\boldsymbol{A}}\boldsymbol{f}_j$$

其中

$$\boldsymbol{f}_{j+1}=[f_{j+1,0}\cdots f_{j+1,n}]^T$$

$$\boldsymbol{f}_j=[f_{j,0}\cdots f_{j,n}]^T$$

$$\widetilde{\boldsymbol{A}}=\begin{bmatrix} 1-a_1 & 1 & & 0 \\ a_1-a_2 & & \ddots & \\ \vdots & & & \\ a_{n-1}-a_n & & & 1 \\ a_n & & 0 & 0 \end{bmatrix}$$

此外还可得 E_j 的递推公式为

$$E_{j+1}=E_j+e_{j+1,j}q^{-j}=E_j+f_{j,0}q^{-j}$$

当 $j=1$ 时，方程 (3-32) 为

$$1=E_1\widetilde{A}+q^{-1}F_1$$

故可取 $E_1=1$，$F_1=q(1-\widetilde{A})$ 为 E_j、F_j 的初值。这样，E_{j+1}，F_{j+1} 便可按下式递推计算：

$$\left.\begin{aligned} &\boldsymbol{f}_{j+1}=\widetilde{\boldsymbol{A}}\boldsymbol{f}_j, \qquad \boldsymbol{f}_0=[1\ \ 0\ \cdots\ 0]^T \\ &E_{j+1}=E_j+f_{j,0}q^{-j},\ E_0=0 \end{aligned}\right\} \tag{3-34}$$

2. 滚动优化

在 GPC 中，t 时刻的优化性能指标具有以下形式：

$$\min J(t)=\mathrm{E}\left\{\sum_{j=N_1}^{N_2}[y(t+j)-w(t+j)]^2+\sum_{j=1}^{NU}\lambda(j)[\Delta u(t+j-1)]^2\right\} \tag{3-35}$$

其中，E为数学期望；w为对象输出的期望值；N_1和N_2分别为优化时域的始值与终值；NU为控制时域，即在NU步后控制量不再变化：

$$u(t+j-1)=u(t+NU-1),\quad j>NU$$

$\lambda(j)$为控制加权系数，为简化计一般常可假设其为常数λ。

性能指标（3-35）采用了长时段预测的概念，把所要优化的方差从一个时间点扩展到一段时域，其中N_1应大于对象的时滞数，而N_2应大到对象动态特性能充分表现出来。由于以多步预测优化代替了一步预测优化，即使对时滞估计不当或时滞发生变化，仍能从整体优化中得到合理的控制，这是GPC对模型不精确性具有鲁棒性的重要原因。这一性能指标的提出，来源于DMC的启发，除去随机系统带来的差别外，它与DMC中的优化性能指标（3-3）非常相似。在式（3-3）中，只需把N_1以前的权系数q_i取为零，以后的q_i取为1，即可得到相同的形式。因此，为了简化记号，在以下的讨论中将假设$N_1=1$，并记N_2为N。如果需要，则可象DMC那样通过加入权系数q_i获得不同的优化初始时域。

在式（3-35）中，对象输出的期望值w可采用MAC中参考轨迹的形式，即

$$\begin{aligned}&w(t)=y(t)\\&w(t+j)=\alpha w(t+j-1)+(1-\alpha)c\\&0<\alpha<1,\quad j=1,2,\cdots,N\end{aligned}\tag{3-36}$$

利用预测模型（3-33），可以写出预测的未来输出取代性能指标（3-35）中的$y(t+j)$：

$$\begin{aligned}\hat{y}(t+1\mid t)&=G_1\Delta u(t)+F_1y(t)\\&=g_{1,0}\Delta u(t)+f_1(t)\\\hat{y}(t+2\mid t)&=G_2\Delta u(t+1)+F_2y(t)\\&=g_{2,0}\Delta u(t+1)+g_{2,1}\Delta u(t)+f_2(t)\\&\ \ \vdots\\\hat{y}(t+N\mid t)&=G_N\Delta u(t+N-1)+F_Ny(t)\end{aligned}$$

$$
\begin{aligned}
&= g_{N,0}\Delta u(t+N-1)+\cdots+g_{N,N-NU}\\
&\quad \times\Delta u(t+NU-1)+\cdots\\
&\quad +g_{N,N-1}\Delta u(t)+f_N(t)\\
&= g_{N,N-NU}\Delta u(t+NU-1)+\cdots\\
&\quad +g_{N,N-1}\Delta u(t)+f_N(t)
\end{aligned}
$$

其中

$$
\left.
\begin{aligned}
&f_1(t)=[G_1(q^{-1})-g_{1,0}]\Delta u(t)+F_1y(t)\\
&f_2(t)=q[G_2(q^{-1})-q^{-1}g_{2,1}-g_{2,0}]\Delta u(t)+F_2y(t)\\
&\quad\vdots \qquad\qquad\qquad \vdots\\
&f_N(t)=q^{N-1}[G_N(q^{-1})-q^{-(N-1)}g_{N,N-1}-\cdots-g_{N,0}]\\
&\qquad\qquad\qquad\qquad \times\Delta u(t)+F_Ny(t)
\end{aligned}
\right\}
\tag{3-37}
$$

均可由 t 时刻已知的信息 $\{y(\tau),\ \tau\leqslant t\}$ 以及 $\{u(\tau),\ \tau<t\}$ 计算。如果记

$$
\begin{aligned}
&\hat{\boldsymbol{y}}=[\hat{y}(t+1\mid t)\cdots\hat{y}(t+N\mid t)]^T\\
&\tilde{\boldsymbol{u}}=[\Delta u(t)\cdots\Delta u(t+NU-1)]^T\\
&\boldsymbol{f}=[f_1(t)\cdots f_N(t)]^T
\end{aligned}
$$

并且注意到 $g_{j,i}=g_{i+1}$（$i<j$）是阶跃响应系数，则可得

$$\hat{\boldsymbol{y}}=\boldsymbol{G}\tilde{\boldsymbol{u}}+\boldsymbol{f} \tag{3-38}$$

其中

$$
\boldsymbol{G}=\begin{bmatrix} g_1 & & & \\ \vdots & \ddots & & 0 \\ & & g_1 & \\ \vdots & & \vdots & \\ g_N & \cdots & g_{N-NU+1} & \end{bmatrix}_{(N\times NU)}
$$

因此，使性能指标（3-35）最优的解为

$$\tilde{\boldsymbol{u}}=(\boldsymbol{G}^T\boldsymbol{G}+\lambda\boldsymbol{I})^{-1}\boldsymbol{G}^T(\boldsymbol{w}-\boldsymbol{f}) \tag{3-39}$$

其中

$$\boldsymbol{w}=[w(t+1)\cdots w(t+N)]^T$$

即时最优控制量则可由下式给出：

$$u(t)=u(t-1)+\boldsymbol{g}^T(\boldsymbol{w}-\boldsymbol{f}) \tag{3-40}$$

其中，$\boldsymbol{g}^T$ 是矩阵 $(\boldsymbol{G}^T\boldsymbol{G}+\lambda\boldsymbol{I})^{-1}\boldsymbol{G}^T$ 的第 1 行。

3. 在线辨识与校正

GPC是从自校正控制发展起来的，因此保持了自校正的方法

原理，即在控制过程中，不断通过实际输入输出信息在线估计模型参数，并以此修正控制律。这是一种广义的反馈校正。与DMC相比，DMC相当于用一个不变的预测模型，并附加一个误差预测模型共同保证对未来输出作出较准确的预测，而GPC则只用一个模型，通过对其在线修正来给出较准确的预测。

考虑将对象模型（3-30）改写为

$$A(q^{-1})\Delta y(t)=B(q^{-1})\Delta u(t-1)+\xi(t)$$

可得

$$\Delta y(t)=-A_1(q^{-1})\Delta y(t)+B(q^{-1})\Delta u(t-1)+\xi(t)$$

其中，$A_1(q^{-1})=A(q^{-1})-1$。把模型参数与数据参数分别用向量形式记为

$$\boldsymbol{\theta}=[a_1\cdots a_n \vdots b_0\cdots b_{n_b}]^T$$

$$\boldsymbol{\varphi}(t)=[-\Delta y(t-1)\cdots-\Delta y(t-n)\vdots \Delta u(t-1)\cdots\Delta u(t-n_b-1)]^T$$

则可将上式写作

$$\Delta y(t)=\boldsymbol{\varphi}^T(t)\cdot\boldsymbol{\theta}+\xi(t)$$

在此，可用渐消记忆的递推最小二乘法估计参数向量：

$$\left.\begin{aligned}
\hat{\boldsymbol{\theta}}(t)&=\hat{\boldsymbol{\theta}}(t-1)+\boldsymbol{K}(t)[\Delta y(t)-\boldsymbol{\varphi}^T(t)\hat{\boldsymbol{\theta}}(t-1)]\\
\boldsymbol{K}(t)&=\boldsymbol{P}(t-1)\boldsymbol{\varphi}(t)[\boldsymbol{\varphi}^T(t)\boldsymbol{P}(t-1)\boldsymbol{\varphi}(t)+\mu]^{-1}\\
\boldsymbol{P}(t)&=\frac{1}{\mu}[\boldsymbol{I}-\boldsymbol{K}(t)\boldsymbol{\varphi}^T(t)]\boldsymbol{P}(t-1)
\end{aligned}\right\} \tag{3-41}$$

其中，$0<\mu<1$为遗忘因子，常可选$0.95<\mu<1$。$\boldsymbol{K}(t)$为权因子，$\boldsymbol{P}(t)$为正定的协方差阵。在控制起动时，需要设置参数向量$\boldsymbol{\theta}$和协方差阵$\boldsymbol{P}$的初值，通常可令$\hat{\boldsymbol{\theta}}(0)=\underline{0}$，$\boldsymbol{P}(0)=\alpha^2\boldsymbol{I}$，$\alpha$是一个足够大的正数。在控制的每一步，首先要组成数据向量，然后就可由式（3-41）先后求出$\boldsymbol{K}(t)$、$\hat{\boldsymbol{\theta}}(t)$和$\boldsymbol{P}(t)$。

在通过辨识得到多项式$A(q^{-1})$、$B(q^{-1})$的参数后，就可重新计算控制律（3-40）中的$\boldsymbol{g}^T$和$\boldsymbol{f}$，并求出最优控制量。

综上所述，GPC的在线控制可归结为以下步骤：

（1）根据最新的输入输出数据，用递推公式（3-41）估计模型参数，得到$A(q^{-1})$、$B(q^{-1})$。

（2）根据所得的$A(q^{-1})$，按式（3-34）递推计算$E_j(q^{-1})$、$F_j(q^{-1})$。

（3）根据$B(q^{-1})$、$E_j(q^{-1})$、$F_j(q^{-1})$，计算G的元素g_i，并依式（3-37）计算出$f_i(t)$。

（4）重新计算出$\mathbf{g}^T$，并按式（3-40）计算出$u(t)$，将其作用于对象。这一步涉及到$NU \times NU$维矩阵的求逆，在线的计算量必须在选择NU时加以考虑。

以上对于三种典型预测控制算法的介绍，都是针对单变量系统的，并且是在不考虑输入输出有约束时的基本算法。它们虽然在模型形式、优化性能指标及校正方法上各有特色，但却有共同的方法机理，即包含预测模型、滚动优化、反馈校正三项要素，这正是预测控制算法最本质的特征。为了使讨论明确具体，在以下的章节中，我们将以 DMC 为典型算法，研究预测控制系统的分析和设计问题。

第四章　预测控制系统的分析

在本章中，将以3.1节中介绍的DMC算法为例，从时域和频域两方面分析预测控制系统的结构特点。在此基础上，将导出闭环系统中各传递环节的定量表示，并以此分析闭环系统的动态特性、稳定性及鲁棒性。

§4.1　动态矩阵控制的状态空间分析

3.1节所介绍的DMC算法，是建立在预测控制三要素的基础上推出的，其算法原理和具体步骤清楚，但缺乏与其他传统算法在结构上的联系比较。本节中，我们提出一种用状态空间法的理论框架独立设计预测控制器的方法[8,9]，它可以导出与DMC算法相同的结果，因而可看作是在时域中对DMC的一种结构解释。由于采用了这种新的理论框架，DMC与传统最优控制的区别变得十分明了，其简洁的符号和鲜明的结构也为下面分析推导DMC系统在频域中的定量描述提供了基础。

对于单输入单输出的渐近稳定对象，从3.1节相同的设计前提出发，假定已测定了其阶跃响应的采样值 a_1，a_2，…。根据传统的控制理论，可以由 $\{a_i\}$ 得到对象的状态空间描述，它与 $\{a_i\}$ 之间必须满足一定的实现条件。特别对于以控制增量 Δu 为输入量的非常规的状态空间描述

$$\left.\begin{aligned} \boldsymbol{x}(k+1)&=\boldsymbol{A}\boldsymbol{x}(k)+\boldsymbol{b}\Delta u(k)\\ y(k)&=\boldsymbol{c}^T\boldsymbol{x}(k)\end{aligned}\right\}\qquad(4-1)$$

文献〔10〕已推导了其实现条件为

$$\boldsymbol{c}^T\boldsymbol{A}^{i-1}\boldsymbol{b}=a_i,\quad i=1,2,\cdots\qquad(4-2)$$

为了得到一个满足上述条件的最小化实现式(4-1)，通常需要由 $\{a_i\}$ 辨识出对象的阶数，然后再按某些典范型确定 $\boldsymbol{A}$，$\boldsymbol{b}$，$\boldsymbol{c}^T$

的元素，这就是传统的辨识建模过程。然而，如果只考虑实现条件（4-2），而不管式（4-1）的描述是否最小化的，则可取

$$S=\begin{bmatrix} 0 & 1 & & 0 \\ & \ddots & \ddots & \\ & & 0 & 1 \\ 0 & & & 1 \end{bmatrix}_{(N\times N)} \qquad a=\begin{bmatrix} a_1 \\ \vdots \\ a_N \end{bmatrix}_{(N\times 1)} \qquad c^T=[1\ \ 0\ \cdots\ 0]_{(1\times N)}$$

使所得到的状态方程描述

$$\left.\begin{aligned} x(k+1)&=Sx(k)+a\Delta u(k) \\ y(k)&=c^Tx(k) \end{aligned}\right\} \tag{4-3}$$

成为$\{a_i\}$的一个近似实现。可以验证，只要N取得充分大以致$a_i\approx a_N$（$i>N$），则实现条件（4-2）近似满足，有

$$c^TS^{i-1}a=\begin{cases} a_i, & i\leqslant N \\ a_N\approx a_i, & i>N \end{cases}$$

因此，式（4-3）可近似作为具有阶跃响应$\{a_i\}$的对象的状态空间表达式。由于其中S、c^T均为常数阵，而a的元素正是阶跃响应系数，所以这一建模过程无须作进一步的辨识即可直接进行，并且表达式（4-3）不依赖于对象的结构。这一状态空间描述可用图4-1的结构表示。

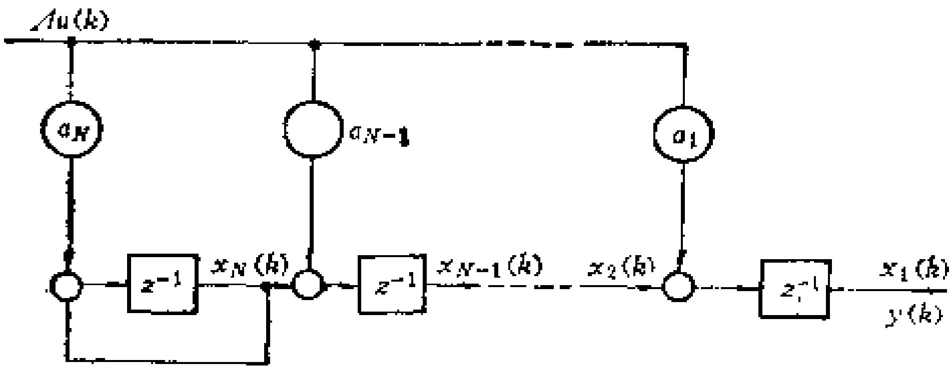


图4-1　阶跃响应在状态空间的实现

由该图及式（4-3）均可看出，N维状态向量$x(k)$的各分量$x_i(k)$可令

$$x(k+1)=Sx(k)$$
$$y(k)=c^Tx(k)$$

逐项递推导出：

$$x_j(k)=c^T S^{j-1}x(k)=y(k+j-1),\quad j=1,\cdots,N$$

这表明 $x_j(k)$ 正是在 u 保持不变时 $k+j-1$ 时刻的未来输出。在这个意义上，可把 $x(k)$ 称为“预测状态”，而式（4-3)则称为是阶跃响应在“预测状态空间”的实现。

式（4-3）是在 $a_i\approx a_N$（$i>N$）条件下导出的状态空间描述，要求阶跃响应到 a_N 时已几乎稳定不变。而恰恰在这一条件下，式（4-3）却几乎是状态不完全可控的，这是因为它的可控性矩阵的行列式

$$\det Q_c=\det[a\ \ Sa\cdots S^{N-1}a]$$

$$=\det\begin{bmatrix} a_1 & a_2 & \cdots & a_N \\ \vdots & & \iddots & \vdots \\ a_N & & \cdots & a_N\end{bmatrix}=(-1)^{\frac{(N-1)N}{2}}a_N(a_N-a_{N-1})^{N-1}$$

由于 a_{N-1} 十分接近于 a_N，而N又很大，所以它充分接近于零。模型越是精确，即N越大，a_{N-1} 与 a_N 的差别越小，其可控性就越弱。这里所残留的一点可控性实际上是由模型截断引起的假象。因此，对于这样一种状态空间模型，不能采用基于系统可控的设计方法，如极点配置、状态反馈最优控制等。这种预测状态的不完全可控性，是由于在实现时忽略了对象的结构而引起的。

式（4-3）的状态不完全可控性，并不妨碍我们用其他方法设计控制器。注意到只要 $a_N\neq 0$，该系统是输入输出可控的，因而就可考虑一个以输出优化形式出现的离散有限时域优化问题。与3.1节相同，k时刻的优化是要寻找一组控制增量 $\Delta u(k),\cdots,\Delta u(k+M-1)$，使下述性能指标最优：

$$\min J(k)=\|\boldsymbol{w}(k)-\boldsymbol{y}(k)\|_Q^2+\|\Delta\boldsymbol{u}(k)\|_R^2 \qquad (4\text{-}4)$$

式中各符号的意义与3.1节相同，只是用$\boldsymbol{y}(k)=[y(k+1)\cdots y(k+P)]^T$代替了原来的$\tilde{\boldsymbol{y}}_{PM}(k)$，以表示在 Δu 作用下未来P个时刻的输出。

根据预测模型（4-3)，可求出在 $\Delta\boldsymbol{u}(k)$作用下对象的未来

输出，这可用向量形式表为

$$\boldsymbol{y}(k)=\boldsymbol{GSx}(k)+\boldsymbol{A}\Delta\boldsymbol{u}(k) \tag{4-5}$$

其中，$\boldsymbol{G}=[\boldsymbol{I}_{P\times P} \vdots \boldsymbol{0}_{P\times(N-P)}]$表示从$N$维向量中取前$P$个的运算，而动态矩阵为

$$\boldsymbol{A}=\begin{bmatrix} a_1 & & 0 \\ \vdots & \ddots & \\ a_M & \cdots & a_1 \\ \vdots & & \vdots \\ a_P & \cdots & a_{P-M+1} \end{bmatrix}$$

以式（4-5）代入优化性能指标（4-4），可导出最优控制增量系列

$$\Delta\boldsymbol{u}(k)=(\boldsymbol{A}^T\boldsymbol{Q}\boldsymbol{A}+\boldsymbol{R})^{-1}\boldsymbol{A}^T\boldsymbol{Q}[\boldsymbol{w}(k)-\boldsymbol{GSx}(k)]$$

而即时最优控制增量则可取其首元素得到

$$\begin{aligned}\Delta u(k)&=[1\ 0\cdots 0]\Delta\boldsymbol{u}(k)\\&=\boldsymbol{d}^T[\boldsymbol{w}(k)-\boldsymbol{GSx}(k)]\end{aligned} \tag{4-6}$$

其中

$$\boldsymbol{d}^T=[1\ 0\cdots 0]\ (\boldsymbol{A}^T\boldsymbol{Q}\boldsymbol{A}+\boldsymbol{R})^{-1}\boldsymbol{A}^T\boldsymbol{Q}$$

式（4-6）的控制律包含了预测状态$\boldsymbol{x}(k)$，可看作是一种状态反馈形式的控制律。但由于其元素$x_j(k)$（$j=2,\cdots,N$）在k时刻无法直接测量，因而是不能实现的。为此，有必要通过状态观测器重构预测状态$\boldsymbol{x}(k)$。由于式（4-3）的可观测性矩阵为

$$\boldsymbol{Q}_o=\begin{bmatrix}\boldsymbol{c}^T\\ \boldsymbol{c}^T\boldsymbol{S}\\ \vdots \\ \boldsymbol{c}^T\boldsymbol{S}^{N-1}\end{bmatrix}=\boldsymbol{I}$$

故其状态是完全可观测的，因此可构造观测器

$$\left.\begin{aligned}\hat{\boldsymbol{x}}(k+1)&=\boldsymbol{S}\hat{\boldsymbol{x}}(k)+\boldsymbol{a}\Delta u(k)+\boldsymbol{h}[y(k+1)-\hat{y}(k+1)]\\ \hat{y}(k+1)&=\boldsymbol{c}^T\boldsymbol{S}\hat{\boldsymbol{x}}(k)+\boldsymbol{c}^T\boldsymbol{a}\Delta u(k)\end{aligned}\right\} \tag{4-7}$$

其中，$\hat{\boldsymbol{x}}$，$\hat{y}$分别为重构的状态与输出，$\boldsymbol{h}$为N维观测器反馈向量。

以由此重构的状态$\hat{x}(k)$取代式（4-6）中的真实状态$x(k)$，就可得到可行的闭环控制律

$$\Delta u(k)=d^T[w(k)-GS\hat{x}(k)] \qquad (4-8)$$

这样，整个控制系统具有带观测器的状态反馈形式，其结构见图 4-2。

对比图 4-2 与图 3-2 可以看到，本节的设计虽然采用了状态空间的框架，但其结果与 3.1 节的 DMC 算法是相同的。事实上，只要通过下述对应关系

$$\hat{y}_{N0}(k)\longleftrightarrow S\hat{x}(k),\quad \hat{y}_{P0}(k)\longleftrightarrow GS\hat{x}(k)$$

$$\tilde{y}_{N1}(k)\longleftrightarrow S\hat{x}(k)+a\Delta u(k),\quad \tilde{y}_1(k+1\mid k)\longleftrightarrow \hat{y}(k+1)$$

$$\hat{y}_{cor}(k+1)\longleftrightarrow \hat{x}(k+1)$$

图 4-2 就等价于图 3-2，并且 3.1 节中的式（3-7）、式(3-9)、式(3-10)、式（3-11）与式（3-13）可转变为本节中的式（4-7）、式（4-8)。这表明，本节的设计方法可看作是 DMC 算法在状态空间实现的一种理论解释。两者的计算步骤是完全相同的，但3.1节侧重于从运算过程描述算法，本节则强调了算法的结构性质。后者因其记号紧凑，结构清楚，有利于作进一步的变换和定量推导，并适用于对系统进行分析。

在原始的 DMC 算法中，对未来的误差补偿都是采用现时误差这一个量进行的（参见文献[3]）。3.1 节中我们引入了校正向量h，把它推广为加权修正的形式，是带有启发式的。而在这里，由式（4-7）可以看到，h 是作为观测器反馈向量自然引入的，因而使 h 的引入具有理论的意义。根据观测器理论，还可进一步得到以下的结论：

（1）在模型参数 a_i 准确的情况下，h 的设计独立于优化策略，并且只要使

$$\det[zI-(I-hc^T)S]=0$$

的全部根都在单位圆内，就可使观测过程收敛。这是因为，由式(4-3)、式（4-7)、式（4-8)，这种带观测器的状态反馈系统具有闭环形式

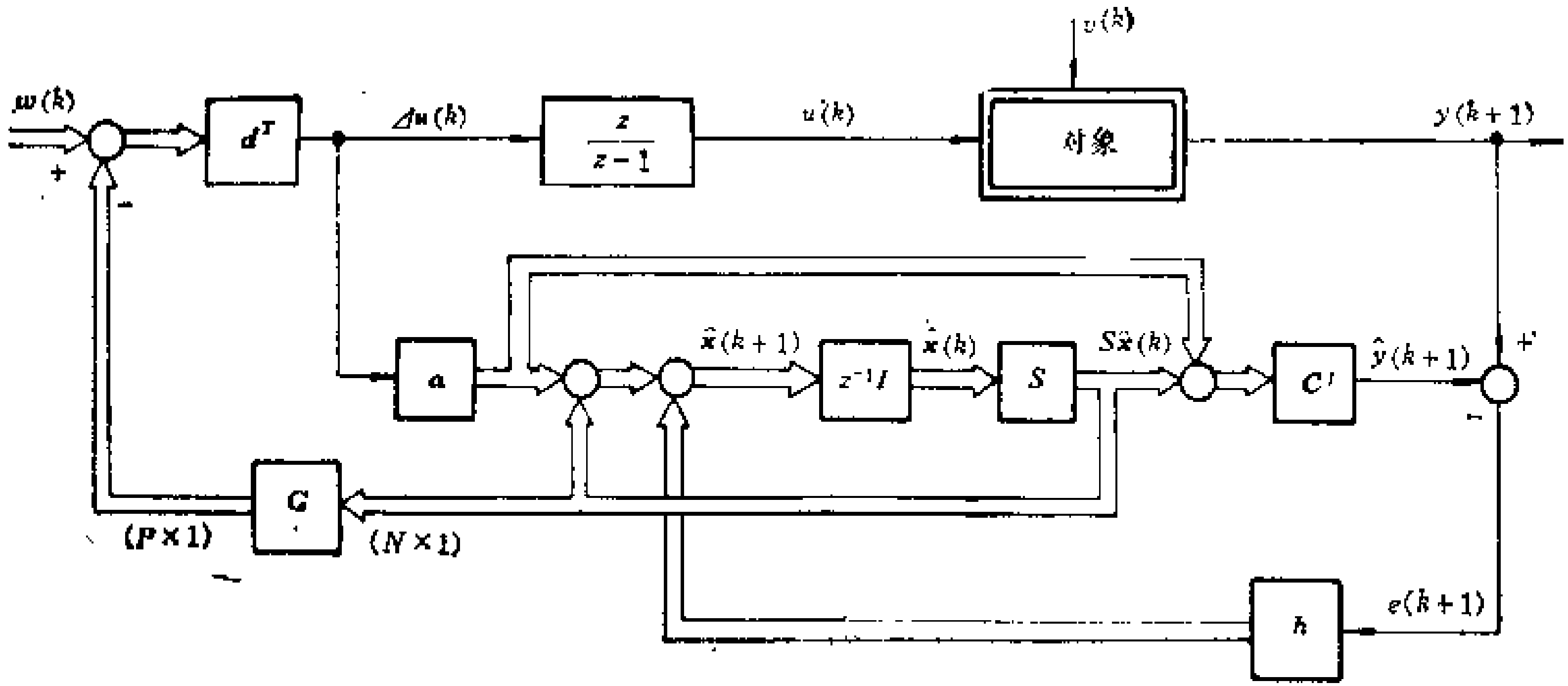


图4-2　带有观测器的预测状态反馈

$$\begin{bmatrix} x(k+1) \\ \tilde{x}(k+1) \end{bmatrix} = \begin{bmatrix} (I-ad^TG)S & ad^TGS \\ 0 & (I-hc^T)S \end{bmatrix}$$

$$\times \begin{bmatrix} x(k) \\ \tilde{x}(k) \end{bmatrix} + \begin{bmatrix} ad^T \\ 0 \end{bmatrix} w(k) \qquad (4\text{-}9)$$

其中，$\tilde{x}(k)=x(k)-\hat{x}(k)$ 为观测器误差。根据分离原理，观测器的设计可独立于控制进行。

（2）只要 h 的选择使观测过程收敛，则不管初始预测即观测器初始状态 $\hat{x}(0)$ 是否准确，观测值 $\hat{x}(k)$ 都能渐近趋于真值 $x(k)$，这就保证了用式（4-8）代替式(4-6)时优化的可靠性，同时也使算法在非稳态起动或初始预测不准时仍能有效工作。

这种带有校正向量 h 的 DMC 算法是对原始 DMC 算法的推广，称为广义动态矩阵控制（Generalized Dynamic Matrix Control，简称为 GDMC)。本书中讨论的 DMC 算法（包括 3.1 节）均是指这种带有校正自由度的 GDMC 算法。

本节提出的 DMC 的状态空间设计方法，还为 DMC 与带观测器的状态反馈最优控制的传统方法进行比较提供了基础。在这里，虽然两者具有相似的结构形式，但存在着重要的差别：

（1）DMC 的观测器是建立在状态空间描述式（4-3）的基础上的，这是一种不考虑最小化的近似实现，其优点是可利用阶跃响应系数 a_i 直接建模，但同时却丧失了可控性。而传统的观测器通常是用对象的最小化模型描述的，在阶跃响应的基础上还须作进一步的辨识。

（2）DMC 采用了有限时域滚动优化的策略，着眼于输出量的最优控制，易导出简易的控制律，而最优性的不足或不确定因素的影响可通过在线反复优化予以补偿。通常的最优控制则是以一个不变的状态最优性能指标导出闭环控制律的，在计算上涉及到复杂的黎卡提（Riccati）方程求解，在线实施时除了反馈外，缺乏对于不确定性的其他补偿手段。

在这里，（1）所涉及的模型问题毕竟只是为了建模的方便，

并不是本质的，事实上也有采用最小化模型的预测控制。但(2)中所述的 DMC 以在线滚动输出优化取代传统的一次性状态优化，却是 DMC 与传统的观测器加状态反馈最优控制的本质差别。正是在这个意义上，DMC 呈现出区别于传统最优控制的特色，并在减少计算量和增强鲁棒性方面表现出明显的优点。

§4.2 动态矩阵控制系统的内模控制结构

通过上节对 DMC 算法的时域分析，它的各部分结构都以熟悉的状态空间形式表现在图 4-2 中，在此基础上作 Z 变换，就可将其转化为一类具有模型、控制、反馈环节的频域结构——内模控制（IMC）结构。

IMC 是莫拉里等在 1982 年提出的一类新型控制结构[8]，其典型框图如图 4-3 所示。图中，Z 传递函数 $G_P(z)$、$G_M(z)$、$G_C(z)$、$G_F(z)$、$G_W(z)$ 表示的环节分别称为对象、模型、控制器、滤波器和参考模型，它们都由 z^{-1} 的有理式组成。

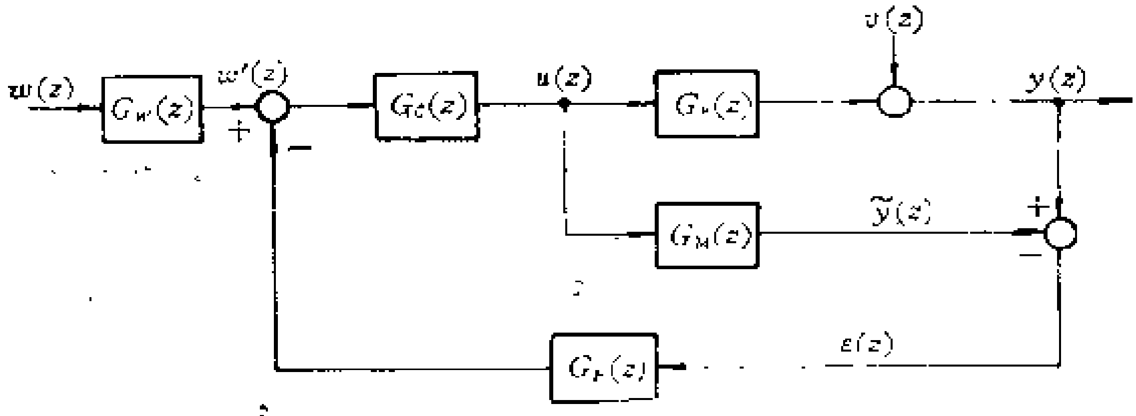


图4-3 内模控制结构

为了讨论 IMC 系统的闭环性质，首先假设 $G_W(z)=1$，在此情况下，由图 4-3 可得闭环系统的传递关系

$$
\begin{aligned}
y(z)=&\frac{G_C(z)G_P(z)}{1+G_C(z)G_F(z)[G_P(z)-G_M(z)]}w(z)\\
&+\frac{1-G_C(z)G_F(z)G_M(z)}{1+G_C(z)G_F(z)[G_P(z)-G_M(z)]}v(z)
\end{aligned}
\tag{4-10}
$$

可以看出，当模型与对象间没有失配，即 $G_M(z)=G_P(z)$ 时，闭环系统的输入传递函数只取决于 IMC 结构中的前向通道

$$F_o(z)=G_C(z)G_P(z) \tag{4-11}$$

因此，模型准确时的稳定性分析实际上只涉及到 IMC 结构中的开环（前向通道）稳定性，而 IMC 中的闭环稳定性则已是模型失配时整个系统的鲁棒性问题。

在式（4-10）的基础上，文献[6]分析了 IMC 结构的下述重要性质：

性质 1（对偶稳定准则） 在模型精确即 $G_M(z)=G_P(z)$ 时，整个系统稳定的条件是对象 $G_P(z)$和控制器 $G_C(z)$ 同时稳定。

这一性质可由式（4-11）直接看出。它表明，对于开环稳定的对象，只要控制器稳定并且模型准确，就可保证闭环稳定。即使控制器是非线性的，只要输入输出稳定，也能使闭环系统稳定。

性质 2（完全控制器） 在对象稳定且模型准确的前提下，若取控制器为

$$G_C(z)=\frac{1}{G_-(z)} \tag{4-12}$$

则控制系统对镇定或跟踪控制都具有最小输出方差。式中$G_-(z)$是对模型进行下述分解后得到的

$$G_M(z)=G_+(z)G_-(z) \tag{4-13}$$

其中

$$G_+(z)=z^{-(l+1)}\prod_{i=1}^{p}\left(\frac{z-z_i}{z-\hat{z}_i}\right)\left(\frac{1-\hat{z}_i}{1-z_i}\right)$$

式中，l 为对象的纯滞后数，$l+1$ 则计入了采样保持所附加的一拍滞后，p 为对象在单位圆外的零点数，z_i 为单位圆外的零点，$\hat{z}_i=1/z_i$ 为其在单位圆内的映射。控制器（4-12）称为完全控制器（perfect controller）。

上述完全控制的概念产生于使控制器完全补偿对象动态的愿望。由式（4-11）可见，理想的完全控制器应具有形式

$$G_C(z)=\frac{1}{G_P(z)}$$

然而，由于经采样保持后对象的 Z 传递函数$G_P(z)$至少具有一拍纯滞后，按上式得到的 $G_C(z)$ 中将出现超前项 z，这种理想的完全控制是不能实现的。此外，对于非最小相位的 $G_P(z)$，因其有单位圆外的零点，按上式得到的控制器 $G_C(z)$ 将有不稳定的极点，根据性质 1，控制系统将是不稳定的。

式（4-12）的控制器正是针对上述情况对理想完全控制所作的修正。在模型精确时，$G_P(z)=G_M(z)$，经过式（4-13）的分解，$G_P(z)$中所有的纯滞后（它们的逆需要超前预测）和单位圆外零点（它们的逆会引起控制器不稳定）都被包含在$G_+(z)$中，这实质上是对象中因物理性质限制而用任何控制手段无法改变的部分。而所谓的完全控制器（4-12），则是对其经控制可改变部分 $G_-(z)$ 的精确补偿，是可实现的稳定控制器。

在用非参数模型描述对象时，显然不能用因式分解（4-13）来得到完全控制器（4-12）。但根据同样的道理，可以在分离出模型的纯滞后因子后，寻找出其剩余部分的一个近似逆作为控制器$G_C(z)$，它应该是可实现的，稳定的，并且至少在稳态时是模型的逆，即 $G_C(1)=1/G_M(1)$。

性质 3（无静差性质） 不管模型与对象是否失配，只要控制器满足$G_C(1)=1/G_M(1)$，滤波器满足$G_F(1)=1$，且闭环系统稳定，则系统对于阶跃输入 w 及常值扰动 v 均不存在输出静差。

上述性质可由式（4-10）直接推出，注意到

$$G_C(1)=\frac{1}{G_M(1)}$$

及

$$G_F(1)=1$$

对于单位阶跃输入$w(z)=1/(1-z^{-1})$及常值扰动$v(z)=v_0/(1-z^{-1})$，若闭环系统稳定，由式（4-10）及终值定理可得

$$\lim_{t\to\infty} y(t)=\lim_{z\to 1}(1-z^{-1})y(z)$$
$$=\frac{G_C(1)G_P(1)}{1+G_C(1)G_F(1)[G_P(1)-G_M(1)]}$$
$$+\frac{1-G_M(1)G_C(1)G_F(1)}{1+G_C(1)G_F(1)[G_P(1)-G_M(1)]}\cdot v_0$$
$$=1$$

故系统输出是无静差的。

IMC结构的上述基本性质，在预测控制系统的分析中有十分重要的作用。

在IMC结构中，实际输出与模型输出的误差不是直接反馈而是通过滤波器 $G_F(z)$ 反馈的。由图4-3可以看出，滤波器只有在模型失配或有干扰引起输出误差时才起作用，因此它对闭环系统的鲁棒性和抗干扰性有着至关重要的影响。对于稳定的对象，根据式（4-10），可把闭环特征方程写作

$$G_C^{-1}(z)+G_F(z)[G_P(z)-G_M(z)]=0 \qquad (4\text{-}14)$$

对于给定的模型失配，可通过适当选择 $G_F(z)$ 使闭环系统仍然保持稳定。例如，设对象为

$$G_P(z)=z^{-2}H(z)$$

其中，$H(z)$ 是一个无滞后、最小相位的稳定传递函数，而模型有一拍纯滞后误差

$$G_M(z)=z^{-1}H(z)$$

若取完全控制器 $G_C(z)=1/H(z)$，则式（4-14）变为

$$H(z)[1+G_F(z)(z^{-2}-z^{-1})]=0$$

如果选择滤波器 $G_F(z)=1$，则上述方程为

$$H(z)(1+z^{-2}-z^{-1})=0$$

它有两个根在单位圆上，闭环系统不是渐近稳定的。但若选择滤波器具有指数滤波形式

$$G_F(z)=\frac{1-\alpha}{1-\alpha z^{-1}},\quad 0<\alpha<1$$

则这两个根变为

$$z_{1,2}=0.5(1\pm\sqrt{4\alpha-3})$$

它们在 $0<\alpha<1$ 时均落在单位圆内，从而可在模型失配时保持闭环系统稳定。

带有滤波器的反馈通道还可起到抑制干扰的作用。例如在模型精确的情况下，由干扰所引起输出的 Z 变换为

$$y(z)=[1-G_C(z)G_F(z)G_M(z)]\,v(z)$$

对于某些类型的干扰 $v(z)$，可以通过设计 $G_F(z)$ 使之补偿 $v(z)$ 的影响从而较快地抑制干扰。例如，当

$$G_P(z)=G_M(z)=z^{-1}H(z)$$

$$G_C(z)=\frac{1}{G_-(z)}=\frac{1}{H(z)}$$

时，其中 $H(z)$ 的含义同上，由上式可得

$$y(z)=[1-z^{-1}G_F(z)]\,v(z)$$

如果已知输出端的干扰为指数上升形式

$$v(z)=\frac{(1-\alpha)z^{-1}}{(1-\alpha z^{-1})(1-z^{-1})},\quad 0<\alpha<1$$

则当无反馈通道即 $G_F(z)=0$ 时，y 将按 $1-\alpha$，$1-\alpha^2$，… 的过程逐渐上升，干扰 v 将无衰减地完全反映在输出端并产生静差。若取 $G_F(z)=1$，则有

$$y(z)=\frac{(1-\alpha)z^{-1}}{1-\alpha z^{-1}}$$

输出将按 $1-\alpha$，$\alpha(1-\alpha)$，$\alpha^2(1-\alpha)$…的过程逐渐衰减并完全抑制干扰。但若取滤波器为

$$G_F(z)=(\alpha+1)-\alpha z^{-1}$$

则

$$y(z)=(1-\alpha)z^{-1}$$

干扰对输出的影响只出现一拍，幅值为 $1-\alpha$，而后即被完全抑制。

由此可见，在 IMC 结构中，滤波器 $G_F(z)$ 的选择必须考虑无静差、抗干扰性、模型失配时的鲁棒性等多方面的要求。这些要求有时是矛盾的，例如取 $G_F(z)=0$ 时对鲁棒性最有利，但

却会引起静差并失去抗干扰能力。因此，如何根据具体情况综合设计滤波器，是 IMC 结构中一个十分关键的问题。

为了用以上介绍的 IMC 结构的性质分析预测控制系统，下面我们就在 4.1 节的基础上，把时域中的 DMC 算法转化为频域中的IMC 结构。

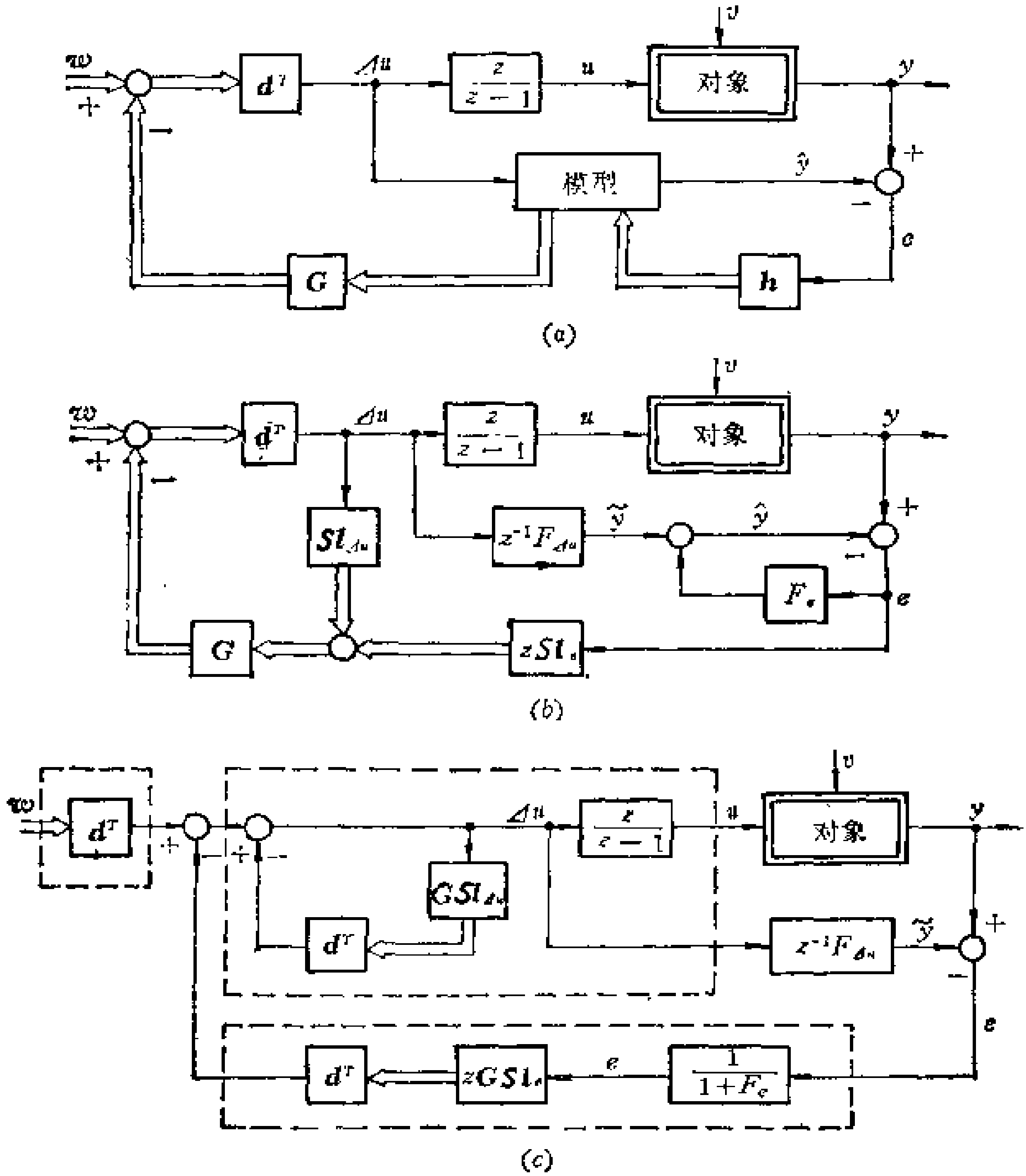


图4-4 动态矩阵控制变换为内模控制结构

图 4-2 所示的带观测器的状态反馈 DMC 结构具有图 4-4（a）的形式。现在我们参照图 4-4 作如下的变换与计算。

1. 图4-4(a)变为图4-4(b)

在图 4-2 中，$e(k+1)=y(k+1)-\hat{y}(k+1)$，对观测器方程（4-7）作 Z 变换，可得

$$z\hat{\boldsymbol{x}}(z)=\boldsymbol{S}\hat{\boldsymbol{x}}(z)+\boldsymbol{a}\Delta u(z)+\boldsymbol{h}ze(z)$$

$$z\hat{y}(z)=\boldsymbol{c}^T\boldsymbol{S}\hat{\boldsymbol{x}}(z)+\boldsymbol{c}^T\boldsymbol{a}\Delta u(z)$$

故有

$$\hat{\boldsymbol{x}}(z)=\boldsymbol{l}_{\Delta u}\Delta u(z)+\boldsymbol{l}_e(ze(z))$$

$$z\hat{y}(z)=F_{\Delta u}\Delta u(z)+F_e(ze(z))$$

其中

$$\boldsymbol{l}_{\Delta u}=(z\boldsymbol{I}-\boldsymbol{S})^{-1}\boldsymbol{a},\quad \boldsymbol{l}_e=(z\boldsymbol{I}-\boldsymbol{S})^{-1}\boldsymbol{h}$$

$$F_{\Delta u}=\boldsymbol{c}^T\boldsymbol{S}\boldsymbol{l}_{\Delta u}+\boldsymbol{c}^T\boldsymbol{a},\quad F_e=\boldsymbol{c}^T\boldsymbol{S}\boldsymbol{l}_e$$

从而可得图 4-4(b)。根据$\boldsymbol{S}$，$\boldsymbol{a}$，$\boldsymbol{h}$的表达式可算得

$$\boldsymbol{l}_{\Delta u}=\begin{bmatrix} a_1z^{-1}+a_2z^{-2}+\cdots+a_{N-1}z^{-(N-1)}+\dfrac{a_N}{z-1}z^{-(N-1)} \\ a_2z^{-1}+a_3z^{-2}+\cdots+a_{N-1}z^{-(N-2)}+\dfrac{a_N}{z-1}z^{-(N-2)} \\ \vdots \\ a_{N-1}z^{-1}+\dfrac{a_N}{z-1}z^{-1} \\ \dfrac{a_N}{z-1} \end{bmatrix}$$

$$\boldsymbol{l}_e=\begin{bmatrix} h_1z^{-1}+h_2z^{-2}+\cdots+h_{N-1}z^{-(N-1)}+\dfrac{h_N}{z-1}z^{-(N-1)} \\ h_2z^{-1}+h_3z^{-2}+\cdots+h_{N-1}z^{-(N-2)}+\dfrac{h_N}{z-1}z^{-(N-2)} \\ \vdots \\ h_{N-1}z^{-1}+\dfrac{h_N}{z-1}z^{-1} \\ \dfrac{h_N}{z-1} \end{bmatrix}$$

$$F_{\Delta u}=a_1+a_2z^{-1}+\cdots+a_{N-1}z^{-(N-2)}+\frac{a_Nz^{-(N-2)}}{z-1}$$

$$F_e = h_2 z^{-1} + \cdots + h_{N-1} z^{-(N-2)} + \frac{h_N z^{-(N-2)}}{z-1}$$

2. 图4-4(b)变为图4-4(c)

注意到图 4-4(b) 中

$$e(z)=y(z)-\hat{y}(z),\quad \tilde{y}(z)+F_e e(z)=\hat{y}(z)$$

记 $\varepsilon(z)=y(z)-\tilde{y}(z)$，则 $e(z)=\varepsilon(z)/(1+F_e)$。

此外，把反馈部分进行分解，即可得到图 4-4(c) 所示的结构。

3. 图4-4(c)变为图4-3

把图 4-4(c) 中模型输入由 Δu 移至 u，在控制通道中加入 $d_s=\sum_{i=1}^{P} d_i$ 并同时将所有 $\boldsymbol{d}^T$ 除以 d_s，整理计算各环节后，即可得到图 4-3 所示的 IMC 结构，其中各传递函数的计算如下：

(1) 模型

$$\begin{aligned} G_M(z) &= \frac{z-1}{z} z^{-1} F_{\Delta u} \\ &= a_1 z^{-1} + (a_2-a_1)z^{-2} + \cdots + (a_N - a_{N-1})z^{-N} \qquad (4\text{-}15) \end{aligned}$$

(2) 控制器

$$G_C(z) = \frac{z}{z-1} \cdot \frac{d_s}{1+\boldsymbol{d}^T \boldsymbol{G} \boldsymbol{S} \boldsymbol{l}_{\Delta u}}$$

记 $\boldsymbol{d}^T\boldsymbol{G}=[d_1 \cdots d_P\ 0\ \cdots\ 0] \triangleq [d_1 \cdots d_N]$，其中 $d_{P+1}=d_{P+2}=\cdots=d_N=0$，注意到 $\boldsymbol{S}\boldsymbol{l}_{\Delta u}$ 相当于把 $\boldsymbol{l}_{\Delta u}$ 的元素上推一行而末行不变，则有

$$\boldsymbol{d}^T\boldsymbol{G}\boldsymbol{S}\boldsymbol{l}_{\Delta u} = [d_1 \cdots d_N]\boldsymbol{S}\boldsymbol{l}_{\Delta u}$$

$$= [d_1 \cdots d_N] \cdot \begin{bmatrix} a_2 z^{-1} + a_3 z^{-2} + \cdots + a_{N-1} z^{-(N-2)} + \dfrac{a_N}{z-1} z^{-(N-2)} \\ a_3 z^{-1} + a_4 z^{-2} + \cdots + a_{N-1} z^{-(N-3)} + \dfrac{a_N}{z-1} z^{-(N-3)} \\ \vdots \\ \dfrac{a_N}{z-1} \\ \dfrac{a_N}{z-1} \end{bmatrix}$$

$$=\frac{1}{z-1}\{[d_1a_2+d_2a_3+\cdots+d_{N-1}a_N+d_Na_N]+[d_1(a_3-a_2)+d_2(a_4-a_3)+\cdots+d_{N-2}(a_N-a_{N-1})]z^{-1}+\cdots+[d_1(a_{N-1}-a_{N-2})+d_2(a_N-a_{N-1})]z^{-(N-3)}+d_1(a_N-a_{N-1})z^{-(N-2)}\}$$

$$=\frac{1}{z-1}\{b_2+(b_3-b_2)z^{-1}+\cdots+(b_{N-1}-b_{N-2})z^{-(N-3)}+(b_N-b_{N-1})z^{-(N-2)}\}$$

式中

$$b_i=\sum_{j=1}^{N}d_ja_{i+j-1}=\sum_{j=1}^{P}d_ja_{i+j-1},\quad i=1,\ \cdots,\ N,$$

且当 $l>N$ 时 $\qquad a_l=a_N$

由此可得

$$G_C(z)=\frac{d_sz}{z-1+b_2+(b_3-b_2)z^{-1}+\cdots+(b_N-b_{N-1})z^{-(N-2)}}$$

$$=\frac{d_s}{b(z)}\tag{4-16}$$

其中

$$b(z)=1+(b_2-1)z^{-1}+(b_3-b_2)z^{-2}+\cdots+(b_N-b_{N-1})z^{-(N-1)}$$

（3）滤波器

$$G_F(z)=\frac{\boldsymbol{d}^T\boldsymbol{GS}\boldsymbol{l}_ez}{d_s}\cdot\frac{1}{1+F_e}$$

式中，$\boldsymbol{d}^T\boldsymbol{GS}\boldsymbol{l}_e$ 可类似于 $\boldsymbol{d}^T\boldsymbol{GS}\boldsymbol{l}_{\Delta u}$ 算得为

$$\boldsymbol{d}^T\boldsymbol{GS}\boldsymbol{l}_e=\frac{1}{z-1}\{c_2+(c_3-c_2)z^{-1}+\cdots+(c_N-c_{N-1})z^{-(N-2)}\}$$

其中

$$c_i=\sum_{j=1}^{N}d_jh_{i+j-1}=\sum_{j=1}^{P}d_jh_{i+j-1},\quad i=1,\ \cdots,\ N,$$

且当 $l>N$ 时 $\qquad h_l=h_N$

此外可根据 F_e 的表达式算得

$$\frac{1}{1+F_e}=\frac{1}{1+h_2z^{-1}+\cdots+h_{N-1}z^{-(N-2)}+\dfrac{h_N}{z-1}z^{-(N-2)}}$$

由此可得

$$G_p(z)=\frac{c(z)}{d_s h(z)} \tag{4-17}$$

其中

$$c(z)=c_2+(c_3-c_2)z^{-1}+\cdots+(c_N-c_{N-1})z^{-(N-2)}$$

$$h(z)=1+(h_2-1)z^{-1}+(h_3-h_2)z^{-2}+\cdots+(h_N-h_{N-1})z^{-(N-1)}$$

（4）参考模型

在经过上述变换后，图 4-3 中的 w' 为

$$w'=\frac{1}{d_s}(\mathbf{d}^T\mathbf{w})$$

对于定值控制

$$\mathbf{w}=[w\cdots w]^T$$

而对于跟踪控制

$$\mathbf{w}=[w(k+1)\cdots w(k+P)]^T$$

由于 $w'(z)=G_W(z)w(z)$，最终可得

$$G_W(z)=\begin{cases}1, & \text{定值控制}\\ \dfrac{1}{d_s}(d_1z+\cdots+d_Pz^P), & \text{跟踪控制}\end{cases} \tag{4-18}$$

式（4-15）～（4-18）给出了 DMC 的 IMC 结构中各环节的 Z 传递函数。这些定量表达式和 IMC 的结构性质，为我们在频域中分析 DMC 闭环系统的各项性质奠定了基础。

§4.3　动态矩阵控制系统的动态特性分析

本节中，我们将根据上节所导出的 DMC 在 IMC 结构下各传递环节的定量表达式，来研究 DMC 系统在模型无失配情况下的动态特性。

1. 预测控制器的最小化形式

在模型无失配即 $G_P(z)=G_M(z)$ 且无干扰时，闭环系统的输入输出传递特性可由 Z 传递函数（4-11）描述，即

$$F_o(z)=G_C(z)G_P(z)$$

如果就 $G_C(z)$ 的表达式（4-16）来看，是很难由上式理解DMC的控制机理的。因为根据传统的控制理论，对象的极点是反映其动态特性的，要使其动态发生改变，必须通过控制器的零点消去其一部分极点，并设置新的极点。但由式（4-16）却可看到，$G_C(z)$ 虽然引入了 $N-1$ 个新的极点，但其零点却全部位于 Z 平面的原点处，它们显然是无法补偿 $G_P(z)$ 的极点的。产生以上困惑的原因在于DMC采用了非最小化的模型，从而导致其控制器（4-16）也具有非最小化的形式。因此，我们将首先研究 $G_C(z)$ 的最小化形式[11]。

设阶跃响应为 $\{a_i\}$ 的对象的最小化描述为

$$G_P(z)=\frac{m(z)}{p(z)}=\frac{m_1z^{-1}+\cdots+m_nz^{-n}}{1+p_1z^{-1}+\cdots+p_nz^{-n}} \tag{4-19}$$

其中，n 为对象的阶数。根据阶跃响应与对象传递函数的关系可写出

$$\frac{m_1z^{-1}+\cdots+m_nz^{-n}}{1+p_1z^{-1}+\cdots+p_nz^{-n}}=a_1z^{-1}+(a_2-a_1)z^{-2}+(a_3-a_2)z^{-3}+\cdots$$

由此可得递推关系

$$\left.\begin{aligned}
&a_1=m_1\\
&(a_2-a_1)+a_1p_1=m_2\\
&\qquad\vdots\\
&(a_n-a_{n-1})+(a_{n-1}-a_{n-2})p_1+\cdots+a_1p_{n-1}=m_n\\
&(a_{n+1}-a_n)+(a_n-a_{n-1})p_1+\cdots+a_1p_n=0\\
&(a_{i+2}-a_{i+1})+(a_{i+1}-a_i)p_1+\cdots+(a_{i-n+2}-a_{i-n+1})p_n=0\text{，}\quad i\geqslant n
\end{aligned}\right\} \tag{4-20}$$

现在考察无模型失配时的闭环传递函数

$$F_o(z)=G_C(z)G_P(z)=\frac{d_sm(z)}{p(z)b(z)}\triangleq\frac{q(z)}{n(z)}$$

其中

$$q(z)=d_sm(z)$$

$$n(z)=p(z)b(z)=(1+p_1z^{-1}+\cdots+p_nz^{-n})$$

$$\times[1+(b_2-1)z^{-1}+\cdots+(b_N-b_{N-1})z^{-(N-1)}]$$
$$\triangleq 1+p_1^*z^{-1}+\cdots+p_{N+n-1}^*z^{-(N+n-1)}$$

式中

$$\begin{bmatrix} 1 \\ p_1^* \\ \vdots \\ p_{n+1}^* \\ p_{n+2}^* \\ \vdots \\ p_{N-1}^* \\ p_N^* \\ \vdots \\ p_{N+n-1}^* \end{bmatrix} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ b_2-1 & 1 & \cdots & 0 \\ \vdots & \vdots & & \\ b_{n+2}-b_{n+1} & b_{n+1}-b_n & \cdots & b_2-1 \\ b_{n+3}-b_{n+2} & b_{n+2}-b_{n+1} & \cdots & b_3-b_2 \\ \vdots & \vdots & & \vdots \\ b_N-b_{N-1} & b_{N-1}-b_{N-2} & \cdots & b_{N-n}-b_{N-n-1} \\ 0 & b_N-b_{N-1} & \cdots & b_{N-n+1}-b_{N-n} \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & b_N-b_{N-1} \end{bmatrix} \begin{bmatrix} 1 \\ p_1 \\ \vdots \\ \vdots \\ p_{n-1} \\ p_n \end{bmatrix} \tag{4-21}$$

如果N取得足够大以使$a_i \approx a_N$($i>N$)，则可导出

$$b_{N+i}=d_1a_{N+i}+\cdots+d_pa_{N+p+i-1}\approx d_ia_N,\quad i=0,\ 1,\ \cdots$$

从而有

$$b_{N+i}-b_{N+i-1}\approx 0,\quad i=1,\ 2,\ \cdots$$

因此，可把式（4-21）中下面的行均近似写作

$$p_{i+2}^*\approx[b_{i+3}-b_{i+2}\cdots b_{i-n+3}-b_{i-n+2}]p,$$
$$i=n,\ n+1,\ \cdots,\ N+n-3 \tag{4-22}$$

其中，$p=[1\ p_1\cdots p_n]^T$。

根据b_i的定义及递推关系式（4-20），进一步可得

$$p^*{}_{i+2}\approx[d_1\cdots d_p]\begin{bmatrix} a_{i+3}-a_{i+2} & \cdots & a_{i-n+3}-a_{i-n+2} \\ \vdots & & \vdots \\ a_{i+p+2}-a_{i+p+1} & \cdots & a_{i+p-n+2}-a_{i+p-n+1} \end{bmatrix}p=0,\quad i\geqslant n$$

因此，可把分母多项式$n(z)$近似写作

$$n(z)=1+p_1^*z^{-1}+\cdots+p_{n+1}^*z^{-(n+1)}\triangleq p^*(z) \tag{4-23}$$

其中

$$\begin{bmatrix} 1 \\ p_1^* \\ \vdots \\ p_{n+1}^* \end{bmatrix} = \begin{bmatrix} 1 & & 0 \\ b_2-1 & 1 & \ddots \\ \vdots & & 1 \\ b_{n+2}-b_{n+1} & \cdots & b_2-1 \end{bmatrix} \begin{bmatrix} 1 \\ p_1 \\ \vdots \\ p_n \end{bmatrix} \tag{4-24}$$

这样，闭环传递函数便具有最小化形式

$$F_o(z)=\frac{d_s m(z)}{p^*(z)} \tag{4-25}$$

这表明，预测控制器 $G_C(z)$ 实际上有最小化形式

$$G_C(z)=\frac{d_s p(z)}{p^*(z)} \tag{4-26}$$

由此可见，以式(4-16)出现的 $G_C(z)$ 是一种非最小化形式，在 N 充分大的前提下，它可近似地用最小化形式（4-26）表达。这一最小化形式把对象的极点多项式作为控制器的零点多项式，同时又通过极点多项式系数空间的变换（4-24）为控制器设置了新的极点多项式，这一变换取决于对象特性（a_i）和控制策略的选择（d_i）。

在一般情况下，$G_C(z)$ 的最小化形式（4-26）是 $n+1$ 阶的，这里 n 是对象的最小化阶数。但若在优化策略中选择 $\boldsymbol{R}=\mathbf{0}$，则由

$$\mathbf{d}^{\mathrm{T}}=[1\quad 0\cdots 0](\mathbf{A}^{\mathrm{T}}\mathbf{Q}\mathbf{A})^{-1}\mathbf{A}^{\mathrm{T}}\mathbf{Q}$$

可得

$$\mathbf{d}^{\mathrm{T}}\mathbf{A}=[1\ 0\ \cdots\ 0]$$

即

$$b_1=d_1a_1+\cdots+d_pa_p=1$$

由此可知

$$\begin{aligned}p_{n+1}^*&=[b_{n+2}-b_{n-1}\cdots b_2-1\ ]\boldsymbol{p}\\&=[d_1\cdots d_p]\begin{bmatrix}a_{n+2}-a_{n+1} & \cdots & a_2-a_1\\ \vdots & & \vdots\\ a_{n+p+1}-a_{n+p} & \cdots & a_{p+1}-a_p\end{bmatrix}\boldsymbol{p}=0\end{aligned}$$

故在 $\boldsymbol{R}=\mathbf{0}$ 的特殊选择下，最小化控制器是 n 阶的，有

$$p^*(z)=1+p_1^*z^{-1}+\cdots+p_n^*z^{-n} \tag{4-27}$$

其中

$$\begin{bmatrix}1\\ p_1^*\\ \vdots\\ p_n^*\end{bmatrix}=\begin{bmatrix}1 & & 0\\ b_2-1 & 1 & \\ \vdots & & \ddots\\ b_{n+1}-b_n & \cdots & 1\end{bmatrix}\begin{bmatrix}1\\ p_1\\ \vdots\\ p_n\end{bmatrix}$$

2. 预测控制器的补偿性质

预测控制器的最小化形式明显地解释了 DMC 对于对象动态的补偿性质，并指出了 DMC 控制器是一种补偿控制器，它消除了对象原有的极点，同时设置了依赖于控制策略的新的极点，从而可以达到改善系统动态性能的目的。显然，控制策略的选择对于被控系统的稳定性和动态特性都有关键的影响，而式（4-24）正是对它们关系的定量描述。它也可看作是在控制策略作用下，从对象特征多项式系数空间向被控系统特征多项式系数空间的变换。

图 4-5 画出了这种补偿控制的零、极点分布。图 4-5(a)相应于 DMC 的实际控制式（4-11），其中被控系统的零点就是对象的零点（用 ∘ 表示），而其极点则是对象极点（用 × 表示）与控制器的 $N-1$ 个极点（用□表示）的总和，它们均匀分布在围绕 Z 平面原点的一条封闭曲线上，此外在实轴上还有一个单独的极点。根据这样的零极点分布，是很难对被控系统的动态特性作出判断的。

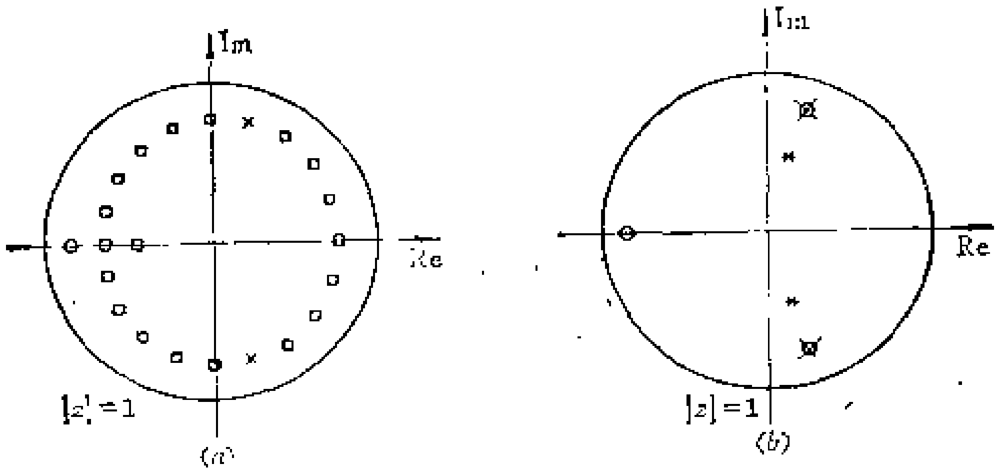


图4-5　动态矩阵控制的补偿性质

(a) 非最小化形式；(b) 最小化形式。

根据预测控制器的最小化形式可以知道，在 N 充分大时，图 4-5(a) 的控制实际上蕴含着图 4-5(b)的控制机理。由于以控制器的最小化形式（4-26）取代了其实际形式（4-16），它的零点正好补偿了对象的极点，而其极点（用 * 表示）则成为被控系

统的极点。控制的补偿性质在图 4-5(b)中变得一目了然。因此，DMC 虽然在实际上是以图 4-5(a)的非最小化方式实现的，但它却等价地蕴含着图 4-5(b)所示的最小化补偿机理。

DMC 的这种非传统补偿性质，在分析被控系统的动态特性时导致了若干非平凡的结论：

（1）尽管模型长度N的选取直接影响到控制器（4-16）的阶数，因而影响到被控系统（4-11）和图 4-5(a)中极点的数目和分布，但只要N取得足够大以致$a_i \approx a_N(i > N)$，则它对被控系统的动态几乎不产生作用。一般情况下，系统经控制后本质上是$n+1$阶的，并不取决于控制器$G_C(z)$的阶数，并且其动态特性只取决于对象特性及控制策略，而与模型长度N几乎无关。

（2）虽然被控系统的极点是对象极点与非最小化控制器极点的总和，但由于控制器的非最小化性质，这些极点对系统动态的影响不能简单地用传统的概念去理解。例如，可以看出，被控系统的动态收敛速度不能用图 4-5(a)中各极点与原点间的最大距离来度量。因此，通过非最小控制器（4-16）的极点来分析控制系统的动态收敛性将会导致错误的结论。特别在N较大时，$G_C(z)$的极点可能会充分接近单位圆，然而图 4-5(b)中所示的闭环极点，却可使被控系统有充分快的响应。

预测控制器的最小化形式不但有助于从概念上澄清 DMC 的补偿机理，而且在对象的最小化传递函数（4-19）已知时，可大大简化系统的分析与设计。这时，我们不必通过高阶控制器(4-16)分析系统的稳定性与动态特性，而只需对最小化特征多项式$p^*(z)$进行分析，它的维数远远低于模型长度N。

3．一步预测优化时的完全控制

对于式（4-19）给出的被控对象，假设它没有纯滞后和非最小相位特性，现考虑它在采用特殊的控制策略——一步预测优化时的控制性能。

所谓一步预测优化，就是指在滚动优化时取优化时域$P=1$并且对控制增量Δu不加软约束的情况。这时在性能指标（3-5）

中有

$$P=M=1,\quad Q=1,\quad R=0$$

由式（3-8）计算的控制向量 d^T 退化为数

$$d_1=\frac{1}{a_1}$$

由此可求出

$$b_i=d_1a_i=\frac{a_i}{a_1},\quad i=1,\cdots,N$$

以此代入式（4-27）中 $p^*(z)$ 的系数变换公式，可得

$$\begin{bmatrix}1\\p_1^*\\\vdots\\p_n^*\end{bmatrix}=\begin{bmatrix}1 & & & 0\\ \frac{a_2}{a_1}-1 & 1 & & \\ \vdots & & \ddots & \\ \frac{a_{n+1}-a_n}{a_1} & & & 1\end{bmatrix}\begin{bmatrix}1\\p_1\\\vdots\\p_n\end{bmatrix}$$

$$=\frac{1}{a_1}\begin{bmatrix}a_1 & & 0\\ a_2-a_1 & a_1 & \\ \vdots & & \ddots \\ a_{n+1}-a_n & \cdots & a_1\end{bmatrix}\begin{bmatrix}1\\p_1\\\vdots\\p_n\end{bmatrix}$$

$$=\frac{1}{a_1}\begin{bmatrix}a_1\\(a_2-a_1)+a_1p_1\\\vdots\\(a_{n+1}-a_n)+\cdots+a_1p_n\end{bmatrix}$$

根据递推关系（4-20），可得

$$p^*(z)=\frac{1}{a_1}(m_1+m_2z^{-1}+\cdots+m_nz^{-(n-1)})=\frac{1}{a_1}zm(z)$$

注意到 $d_s=d_1=1/a_1$，由式（4-26)可得到控制器的最小化形式

$$G_C(z)=\frac{d_sp(z)}{p^*(z)}=z^{-1}\frac{p(z)}{m(z)}=z^{-1}G_P^{-1}(z)\qquad(4-28)$$

由此可见，在一步预测优化策略下，其控制器具有式（4-12）所示的完全控制器的形式。这时，受控系统的 Z 传递函数为

$$F_o(z)=G_C(z)G_P(z)=z^{-1}$$

它仅残留了因采样保持而出现在对象Z传递函数中的一拍固有滞后。在控制起动后，系统输出将在一拍后立即达到设定值，相应于通常的最少拍控制。由式（4-28）可以看出，这种具有完全控制性质的一步预测优化策略只适用于无纯滞后的非最小相位对象，并且它对模型的失配和干扰均十分敏感，因此它在实际过程中很少采用。

在采用一般的优化策略时，即使对象没有纯滞后和非最小相位特性，DMC控制器（4-26）也不具有完全控制器（4-12）的形式。这是因为，$p^*(z)$ 在一般情况下与 $m(z)$ 没有直接的关系。但由式（4-23）和式（4-24）可得

$$p^*(1)=1+p_1^*+\cdots+p_{n+1}^*=b_{n+2}+b_{n+1}p_1+\cdots+b_2p_n$$

$$=\sum_{j=1}^{P}d_j[a_{j+n+1}+a_{j+n}p_1+\cdots+a_{j+1}p_n]$$

由于方括号中的数可由式（4-20）各项相加求出，即

$$a_{j+n+1}+a_{j+n}p_1+\cdots+a_{j+1}p_n=m_1+\cdots+m_n,\quad j\geqslant 1$$

故可知

$$p^*(1)=\sum_{j=1}^{P}d_j(m_1+\cdots+m_n)=d_sm(1)$$

因此有

$$G_C(1)=\frac{d_sp(1)}{p^*(1)}=\frac{p(1)}{m(1)}=G_P^{-1}(1)$$

这表明，控制器 $G_C(z)$ 是对象 $G_P(z)$ 的一个近似逆，它至少保证了实现完全的稳态控制。因此，在只知道对象的非参数模型而使IMC中控制器的设计发生困难时，采用DMC的优化控制设计方法，可以导出以模型近似逆形式出现的IMC控制器。正是因为这个原因，IMC的研究往往紧密结合着预测控制的算法研究。

4．纯滞后对象预测控制的性能

考虑带有 l 拍纯滞后的对象

$$\bar{G}_P(z)=z^{-l}\cdot\frac{m(z)}{p(z)}=z^{-l}G_P(z)$$

其中 $G_P(z)$ 是 $\overline{G}_P(z)$ 中去除了附加纯滞后的剩余部分，它和 $m(z)$、$p(z)$ 的表达式可见式（4-19），其中$m_1 \neq 0$。以下，我们将讨论 DMC 用于纯滞后对象 $\overline{G}_P(z)$ 时的控制性能，并与无滞后对象 $G_P(z)$ 的 DMC 控制进行比较。

由于有 l 拍纯滞后，$\overline{G}_P(z)$ 与 $G_P(z)$ 的阶跃响应系数间存在关系

$$\bar{a}_i=\begin{cases}0, & i\leqslant l\\ a_{i-l}, & i>l\end{cases}$$

为了与无滞后时的 DMC 控制相对应，取模型长度 $\bar{N}=N+l$，并在优化性能指标（3-5）中取

$$\bar{P}=P+l,\ \bar{M}=M$$

$$\bar{\boldsymbol{Q}}=\begin{bmatrix}\boldsymbol{0}_{l\times l} & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{Q}\end{bmatrix},\ \bar{\boldsymbol{R}}=\boldsymbol{R}$$

在此优化性能指标下，由式（3-8）可算得

$$\bar{\boldsymbol{d}}^T=\boldsymbol{c}^T(\bar{\boldsymbol{A}}^T\bar{\boldsymbol{Q}}\bar{\boldsymbol{A}}+\bar{\boldsymbol{R}})^{-1}\bar{\boldsymbol{A}}^T\bar{\boldsymbol{Q}}$$

注意到 $\bar{a}_i$ 与 a_i 的关系，有

$$\bar{\boldsymbol{A}}=\begin{bmatrix}\bar{a}_1 & & \boldsymbol{0}\\ \vdots & \ddots & \\ & & \bar{a}_1\\ & & \vdots\\ \bar{a}_{\bar{P}} & \cdots & \bar{a}_{\bar{P}+\bar{M}-1}\end{bmatrix}=\begin{bmatrix}0 & & \boldsymbol{0}\\ \vdots & & \\ a_1 & & \\ & \ddots & \\ \vdots & & a_1\\ & & \vdots\\ a_P & \cdots & a_{P+M-1}\end{bmatrix}=\begin{bmatrix}\boldsymbol{0}_{l\times M}\\ \boldsymbol{A}_{P\times M}\end{bmatrix}$$

从而可得

$$\bar{\boldsymbol{d}}^T=\boldsymbol{c}^T\left([\boldsymbol{0}\ \boldsymbol{A}^T]\begin{bmatrix}\boldsymbol{0} & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{Q}\end{bmatrix}\begin{bmatrix}\boldsymbol{0}\\ \boldsymbol{A}\end{bmatrix}+\boldsymbol{R}\right)^{-1}[\boldsymbol{0}\ \boldsymbol{A}^T]\begin{bmatrix}\boldsymbol{0} & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{Q}\end{bmatrix}$$

$$=\boldsymbol{C}^T(\boldsymbol{A}^T\boldsymbol{Q}\boldsymbol{A}+\boldsymbol{R})^{-1}[\boldsymbol{0}\quad \boldsymbol{A}^T\boldsymbol{Q}]=[\boldsymbol{0}\quad \boldsymbol{d}^T]$$

其中，$\boldsymbol{d}^T$ 是对 $G_P(z)$ 采用优化性能指标（3-5）所得的 DMC 控制向量，见式（3-8）。由此可得两者的对应关系

$$\bar{d}_i=\begin{cases}0, & i\leqslant l\\ d_{i-l}, & i=l+1,\cdots,\bar{P}\end{cases}$$

这表明，对于纯滞后对象，只要在DMC控制中把优化时域比无滞后时延伸 l 拍，就相当于对其无滞后部分的DMC控制，只是控制律的计算要对未来 l 拍以后的 P 个时间点进行而已。

这一控制律的相似性还可通过IMC结构中的最小化控制器形式看出。由 $\bar{a}_i$ 和 $\bar{d}_i$ 的表达式可知

$$\bar{b}_i=\sum_{j=1}^{\bar{P}}\bar{d}_j\bar{a}_{j+i-1}=\sum_{j=l+1}^{P+l}d_{j-l}a_{j+i-l-1}$$

$$=\sum_{j=1}^{P}d_ja_{j+i-1}=b_i,\quad i=1,\ \cdots,\ \bar{N}$$

由于在式（4-24）中有相同的变换阵，故 $\bar{p}^*(z)=p^*(z)$。此外显然有 $\bar{d}_s=d_s$，故两者的最小化控制器完全相同，即

$$\bar{G}_C(z)=G_C(z)$$

这时，纯滞后对象DMC控制的 Z 传递函数为

$$\bar{F}_o(z)=\bar{G}_C(z)\bar{G}_P(z)=z^{-l}\frac{d_sm(z)}{p^*(z)}=z^{-l}F_o(z)$$

它与无滞后对象的DMC控制有完全相同的动态特性，只是附加了对象的 l 拍纯滞后，从而使其输出与无滞后相比推迟了 l 拍。

以上分析表明，DMC算法在处理纯滞后对象时有其独特的优点。由于算法的预测性质，它可把纯滞后自然考虑在内而无须增加附加的控制结构。而其控制效果，则相当于对无滞后部分的控制再附加一输出延迟。由于时滞对象DMC控制律与无时滞控制律存在着对应关系，故可使时滞对象预测控制的分析与设计得到一定程度的简化。

§4.4 动态矩阵控制系统的稳定性和鲁棒性分析

控制系统的稳定性一般可分两种情况研究，一是在模型准确时的稳定性，二是在模型失配时的稳定性即所谓鲁棒性。在IMC结构中，这两种情况恰好对应着IMC系统的开环稳定性和闭环稳定性。在本节中，我们将结合DMC系统在IMC结构下的具

体形式，来分析 DMC 系统的稳定性和鲁棒性[10]。

1. DMC系统在模型无失配时的稳定性

当模型准确即 $G_M(z)=G_P(z)$ 时，由式（4-10）可知，IMC 系统的传递关系为

$$y(z)=G_C(z)G_P(z)w(z)+[1-G_C(z)G_M(z)G_P(z)]v(z)$$

由于在 DMC 控制中被控对象 $G_P(z)$ 总是稳定的，要使系统对输入和扰动都有稳定的响应，显然只需控制器 $G_C(z)$ 和滤波器 $G_F(z)$ 都是稳定的即可。下面我们首先分析控制器的稳定性。

在 4.3 节中，我们已导出了 DMC 控制器的最小化形式（4-26），在对象的最小化描述（4-19）已知时，可通过式（4-24）的变换，得到 $G_C(z)$ 的分母多项式（4-23），并依此分析 $G_C(z)$ 的稳定性。然而，在许多情况下，DMC 是在对象的最小化描述未知时直接从其阶跃响应出发的。这时，对 $G_C(z)$ 的稳定性分析就不能立足于其最小化形式，而只能针对其实际形式(4-16)进行。显然，由于阶跃响应系数 a_i 已取定，$G_C(z)$ 的分母多项式 $b(z)$ 将只取决于控制参数 d_i，即优化性能指标（3-5）中的优化、控制时域和权矩阵。下面，我们将给出若干稳定性定理来揭示这些参数与控制器稳定性的关系。为此，首先引入几个稳定性的引理。

引理4.1 考虑实系数多项式

$$P(x)=x^n+\alpha_1x^{n-1}+\cdots+\alpha_{n-1}x+\alpha_n \tag{4-29}$$

如果

$$1>\alpha_1>\cdots>\alpha_{n-1}>\alpha_n>0 \tag{4-30}$$

则 $P(x)$ 的全部根都在单位圆内。

引理4.2 对于实系数多项式（4-29），如果

$$|\alpha_1|+\cdots+|\alpha_{n-1}|+|\alpha_n|<1 \tag{4-31}$$

则 $P(x)$ 的全部根都在单位圆内。

以上两个引理是离散系统稳定性的充分条件，其证明可参见一般的离散控制系统书籍。

引理4.3 设多项式

$$\varphi(z)=z^n+a_1z^{n-1}+\cdots+a_n$$

的全部根都在单位圆内。构成多项式

$$F(z)=(z-1)\varphi(z)+\sigma\psi(z),\quad \sigma>0$$

其中

$$\psi(z)=b_0z^m+b_1z^{m-1}+\cdots+b_m,\quad m\leqslant n$$

则当 $\sigma\to 0$ 时，$F(z)$ 的全部根 λ_i 都在单位圆内的充要条件是

$$\varphi(1)\psi(1)>0$$

证明 当 $\sigma=0$ 时，有

$$F(z)=(z-1)\varphi(z)$$

它的全部根为

$$\lambda_1=z_1,\ \cdots,\ \lambda_n=z_n,\ \lambda_{n+1}=1$$

其中，z_1，…，z_n 为 $\varphi(z)$ 的根，它们均在单位圆内。由于 $F(z)$ 的根是随 σ 连续变化的，故存在 σ 的一个有限值 σ_0，使在 $0<\sigma<\sigma_0$ 时，$F(z)$ 的前 n 个根仍在单位圆内，而另一个对应于 $\sigma=0$ 时在 1 处的根，由于对称性的要求，必定落在实轴上近 1 处，即

$$\lambda_{n+1}=1+\varepsilon$$

为了确定 ε 的符号，将上式代入 $F(z)$ 的表达式

$$\begin{aligned}F(\lambda_{n+1})&=\varepsilon[(1+\varepsilon)^n+a_1(1+\varepsilon)^{n-1}+\cdots+a_n]\\&\quad+\sigma[b_0(1+\varepsilon)^m+b_1(1+\varepsilon)^{m-1}+\cdots+b_m]\\&=\varepsilon[(1+a_1+\cdots+a_n)+A(\varepsilon)]\\&\quad+\sigma[(b_0+b_1+\cdots+b_m)+B(\varepsilon)]\\&=0\end{aligned}$$

式中 $A(\varepsilon)$，$B(\varepsilon)$ 均为 ε 的无穷小量，即

$A(\varepsilon)\to 0$，$B(\varepsilon)\to 0$，当 $\varepsilon\to 0$ 时

当 $\sigma\to 0$ 时，$\varepsilon\to 0$，忽略与有界量相比无穷小的 $A(\varepsilon)$、$B(\varepsilon)$，可得

$$\varepsilon=-\sigma\frac{b_0+b_1+\cdots+b_m}{1+a_1+\cdots+a_n}=-\sigma\frac{\psi(1)}{\varphi(1)}$$

由于$\sigma>0$，可见当且仅当

$$\psi(1)\varphi(1)>0$$

时，$\varepsilon<0$，即$\lambda_{n+1}=1+\varepsilon$在单位圆内。证毕。

推论4.1 设有多项式

$$N(z)=z^n+(a_1-1)z^{n-1}+a_2z^{n-2}+\cdots+a_n$$

其中，$a_i(i=1,\cdots,n)$是参数$r>0$的函数，且当$r\to+\infty$时，有

$$a_i\to 0,\quad ra_i\to s_i,\quad i=1,\cdots,n$$

式中，$s_i(i=1,\cdots,n)$均为实常数。则当$r\to+\infty$时，$N(z)$的全部根都在单位圆内的充要条件是

$$a_1+\cdots+a_n>0,\quad r\to+\infty$$

上述结果可在引理4.3中令

$$\varphi(z)=z^{n-1},\quad \psi(z)=r(a_1z^{n-1}+\cdots+a_n)$$

$$\sigma=\frac{1}{r},\quad F(z)=(z-1)\varphi(z)+\sigma\psi(z)=N(z)$$

直接推出。

以下，我们给出关于控制器$G_C(z)$稳定性的几个定理。根据式(4-16)，可得$G_C(z)$的特征多项式为

$$B(z)=z^{N-1}+(b_2-1)z^{N-2}+(b_3-b_2)z^{N-3}+\cdots$$
$$+(b_N-b_{N-1})\tag{4-32}$$

其中

$$b_i=\sum_{j=1}^{P}d_ja_{j+i-1},\quad i=1,\cdots,N$$

式中

$$a_l=a_N\quad(l>N)$$

$$[d_1\cdots d_P]=[1\ 0\cdots 0](\boldsymbol{A}^T\boldsymbol{Q}\boldsymbol{A}+\boldsymbol{R})^{-1}\boldsymbol{A}^T\boldsymbol{Q}$$

分析$B(z)$的稳定性，可得到下面的定理[10]。

定理4.1 选择优化策略$\boldsymbol{Q}=\boldsymbol{I}$，$\boldsymbol{R}=0$，$M=1$，如果

（1）阶跃响应序列$\{a_i\}$单调递增且$P=N$，或者

（2）阶跃响应序列$\{a_i\}$单调递增且为凸的

则控制器 $G_C(z)$ 是稳定的。

证明 在取 $\boldsymbol{Q}=\boldsymbol{I}$，$\boldsymbol{R}=0$，$M=1$ 时，有

$$[d_1\cdots d_P]=s[a_1\cdots a_P]$$

其中，$s=1\Big/\sum_{i=1}^{P}a_i^2$，此外还有

$$b_1=d_1a_1+\cdots+d_Pa_P=1$$

$$b_i=d_1a_i+\cdots+d_Pa_{P+i-1}=s\sum_{j=1}^{P}a_ja_{j+i-1},\quad i=1,\ \cdots,\ N$$

（1）对于 $P=N$ 的情况，可得

$$\begin{aligned}
b_2-1&=s[a_1(a_2-a_1)+\cdots+a_{N-1}(a_N-a_{N-1})]\\
b_3-b_2&=s[a_1(a_3-a_2)+\cdots+a_{N-2}(a_N-a_{N-1})]\\
&\vdots\\
b_N-b_{N-1}&=s[a_1(a_N-a_{N-1})]
\end{aligned}$$

由于

$$a_N>a_{N-1}>\cdots>a_1>0$$

显然有
$$b_2-1>b_3-b_2>\cdots>b_N-b_{N-1}>0$$

且因

$$\begin{aligned}
b_2-1&=s[-a_1^2-a_2(a_2-a_1)-\cdots-a_N(a_N-a_{N-1})+a_N^2]\\
&<sa_N^2<1
\end{aligned}$$

故由引理4.1知，$B(z)$ 是稳定多项式。

（2）对于 $\{a_i\}$ 是凸的情况，进一步有

$$a_1>a_2-a_1>\cdots>a_N-a_{N-1}>0$$

当取任意 P 满足 $M\leqslant P\leqslant N$ 时，有

$$\begin{aligned}
b_2-1&=s[a_1(a_2-a_1)+\cdots+a_P(a_{P+1}-a_P)]\\
b_3-b_2&=s[a_1(a_3-a_2)+\cdots+a_P(a_{P+2}-a_{P+1})]\\
&\vdots\\
b_N-b_{N-1}&=s[a_1(a_N-a_{N-1})]
\end{aligned}$$

由于 $a_i>0$，很显然有

$$b_2-1>b_3-b_2>\cdots>b_N-b_{N-1}>0$$

此外

$$b_2-1=s\left[a_1(a_2-a_1)+\cdots+a_P(a_{P+1}-a_P)\right]$$
$$<s\left[a_1a_1+\cdots+a_P(a_P-a_{P-1})\right]$$
$$=s\left[\sum_{j=1}^{P}a_j^2-(a_1a_2+\cdots+a_{P-1}a_P)\right]<1$$

所以$B(z)$也是稳定的多项式。证毕。

定理4.2 选择优化策略$Q=I$，$R=0$，$M=1$，则不论阶跃响应$\{a_i\}$有何种形式，通过选择充分大的P总可得到稳定的控制器。

证明 P充分大是指优化时域P已跨越了模型时域N而延伸到模型截断后的稳态部分。如果把模型时域形式地延伸为$N'=P$，则有

$$a_i=a_N,\quad i=N+1,\ \cdots,\ N'(P)$$

在定理假设的优化策略下可得

$$b_1=1$$
$$d_i=sa_i,\quad i=1,\ \cdots,\ P$$

其中

$$s=\frac{1}{\sum\limits_{j=1}^{P}a_j^2}=\frac{1}{\sum\limits_{j=1}^{N}a_j^2+(P-N)a_N^2}$$

注意到$a_i=a_N(i>N)$，可得

$$b_{i+1}-b_i=\sum_{j=1}^{P}d_j(a_{j+i}-a_{j+i-1})=\begin{cases}0, & i=N,\ \cdots,\ N'-1\\ sD_i, & i=1,\ \cdots,\ N-1\end{cases}$$

其中，$D_i=\sum\limits_{j=1}^{N-i}a_j(a_{j+i}-a_{j+i-1})$是与$P$无关的常数。因此有

$$|b_2-1|+|b_3-b_2|+\cdots+|b_{N'}-b_{N'-1}|=sD$$

其中，$D=|D_1|+\cdots+|D_{N-1}|$与P无关。注意到s是$P\to+\infty$时的无穷小量，故必存在充分大的P_0，使得在$P>P_0$时，有$s<1/D$，从而

$$|b_2-1|+|b_3-b_2|+\cdots+|b_{N'}-b_{N'-1}|<1$$

根据引理 4.2，可得到稳定的控制器。证毕。

定理 4.2 中 P 充分大的条件在物理上意味着优化强调了对象响应的稳态部分，它虽然可得到稳定的控制，但不是好的动态控制器。在 $P\to+\infty$ 的极端情况下，由于 $s\to 0$，可得

$$b_{i+1}-b_i\to 0$$

此外

$$d_s=\sum_{i=1}^{P}d_i=\frac{\sum_{i=1}^{P}a_i}{\sum_{i=1}^{P}a_i^2}$$

$$=\frac{\sum_{i=1}^{N}a_i+(P-N)a_N}{\sum_{i=1}^{N}a_i^2+(P-N)a_N^2}\longrightarrow\frac{1}{a_N}$$

由式（4-16）可得

$$G_C(z)\longrightarrow\frac{1}{a_N}$$

控制器 $G_C(z)$ 将退化为一比例环节，只能起到稳态控制的作用，而不能改变对象的动态。

值得注意的是，定理 4.1 和定理 4.2 都是稳定性的充分条件。定理的意义并不在于其表述的极端情况，而在于其反映的参数对稳定性的影响趋势。在大多数情况下，两个定理中的控制时域 M 取得足够小但不为 1，也能得到稳定的控制，而在定理 4.2 中，优化时域取得足够大但不超过模型时域，仍能使控制器稳定。这些都为控制系统的整定提供了参考，但其直接的定量关系是很难通过解析手段导出的。

在以上两个定理中，都没有考虑权矩阵 $\boldsymbol{Q}$、$\boldsymbol{R}$ 的影响。下面，将进一步讨论它们对于稳定性的作用。

定理4.3 取权矩阵 $R=rI(r\geqslant 0)$，则控制器可通过增大 r 稳定的充要条件是：

$$\left(\sum_{i=1}^{P} a_i q_i\right) a_N > 0 \tag{4-33}$$

证明 在取 $R=rI$ 且当 $r\to+\infty$时，有

$$[d_1\cdots d_P]=\frac{1}{r}[1\ 0\ \cdots\ 0]\left(\frac{1}{r}A^TQA+I\right)^{-1}A^TQ$$

$$\longrightarrow\left[\frac{a_1q_1}{r}\cdots\frac{a_Pq_P}{r}\right]$$

由此可得

$$b_i=\sum_{j=1}^{P} d_j a_{j+i-1}\longrightarrow\frac{1}{r}\sum_{j=1}^{P} a_j a_{j+i-1} q_j\to 0$$

$$r\to+\infty,\quad i=2,\cdots,N$$

$$d_s=\sum_{i=1}^{P} d_i\longrightarrow\frac{1}{r}\sum_{i=1}^{P} a_i q_i,\quad r\to+\infty$$

注意到在$B(z)$ 的表达式（4-32）中，当 $r\to+\infty$时

$$b_2\to 0,\ rb_2\to\sum_{j=1}^{P} a_j a_{j+1} q_j$$

$$b_{i+1}-b_i\to 0,\quad r(b_{i+1}-b_i)\to\sum_{j=1}^{P} a_j q_j(a_{j+i}-a_{j+i-1})$$

$$i=2,\ \cdots,\ N-1$$

此外还有

$$b_2+(b_3-b_2)+\cdots+(b_N-b_{N-1})=b_N=d_s a_N\to\frac{1}{r}a_N\left(\sum_{i=1}^{P} a_i q_i\right)$$

对照推论 4.1 的条件可知，当 $r\to+\infty$时，$B(z)$ 为稳定多项式的充要条件为

$$\left(\sum_{i=1}^{P} a_i q_i\right) a_N > 0$$

证毕。

定理 4.3 给出了 r 充分大时控制器稳定的条件。它表明，即使充分抑制控制增量，也未必能得到稳定的控制。只有在满足条件（4-33）的情况下，才可能通过加大 r 得到稳定的控制。式（4-33）中的和式可理解为阶跃响应在优化时域中的加权重心，条件（4-33）则意味着优化的加权重心必须和阶跃响应的稳态值在同一方向。

定理 4.3 只讨论了 r 充分大时的稳定性，下面的定理将对 r 在〔0，$+\infty$）全局变化时控制器的稳定性提供若干结论。

定理4.4 如果下述两个条件同时满足：

（1）
$$\left(\sum_{i=1}^{P} a_i q_i\right) a_N < 0$$

（2）$(M-1)\times(M-1)$ 维矩阵

$$L=\begin{pmatrix} l_{22}-l_{21}\dfrac{f_{12}}{f_{11}} & \cdots & l_{2M}-l_{21}\dfrac{f_{1M}}{f_{11}} \\ \vdots & & \vdots \\ l_{M2}-l_{M1}\dfrac{f_{12}}{f_{11}} & \cdots & l_{MM}-l_{M1}\dfrac{f_{1M}}{f_{11}} \end{pmatrix}$$

的全部实根都是正的，其中 l_{ij} 是矩阵（A^TQA）的第（i，j）个元素

$$f_{1j}=\sum_{i=1}^{P-j+1} a_i q_{i+j-1},\quad j=1,\ \cdots,\ M$$

则对于一切 $0\leqslant r<+\infty$，$B(z)$ 都是不稳定的。

证明 首先注意到，多项式 $B(z)$ 的系数都是 r 的连续函数，虽然在一般情况下 r 不能从这些系数中线性分离出来，但 $B(z)$ 的根轨迹随 r 连续变化且对称于实轴的特性是保持不变的。

由定理 4.3 及引理 4.3 可知，在本定理的条件（1）下，当 r 充分大时，$B(z)$ 的 $N-2$ 条根轨迹在 Z 平面原点附近，另有一条根轨迹则在 $z=1$ 右方的实轴上，因此 $B(z)$ 是不稳定的。当 r 减小时要使 $B(z)$ 稳定，必要的条件是这一根轨迹必须返

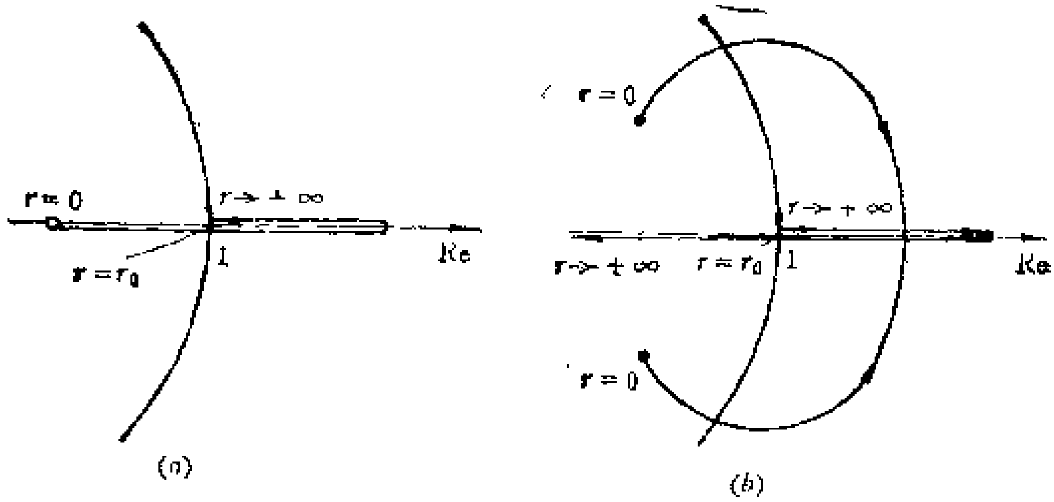


图4-6　$B(z)$ 根轨迹的变化

（a）根轨迹沿实轴进入单位圆，（b）根轨迹在复平面上进入单位圆。

回单位圆内，这时只可能有图 4-6 所示的两种情况。

对于图 4-6(a)，当 r 减小时这一根轨迹折回，并向左方沿实轴经 $z=1$ 后进入单位圆；

对于图 4-6(b)，当 r 减小时这一根轨迹在 $z=1$ 右方某一点脱离实轴进入复平面，并在复平面上穿越单位圆。由于根轨迹的对称性，这时必定有另一条根轨迹沿实轴向左与之汇合，然后再分叉进入复平面形成对称的根轨迹。由于这另一条根轨迹在 r 充分大时位于单位圆内，当 r 减小时它必须首先沿实轴穿越单位圆，才能到达分叉点。

因此，这一根轨迹要返回单位圆内，至少要有一条根轨迹经过 $z=1$ 处，即至少存在一个有限的 r_0 值，使得相应的 $B(1)=0$。

由于

$$B(1)=b_N=a_Nd_s$$

上述条件意味着存在某一 r_0 值，使

$$d_s(r_0)=0 \tag{4-34}$$

注意到

$$\begin{aligned} d_s &= [d_1\cdots d_P][1\ \cdots\ 1]^T \\ &= [1\ 0\ \cdots\ 0](A^TQA+R)^{-1}A^TQ \\ &\quad \cdot [1\ \cdots\ 1]^T \end{aligned}$$

由于

$$[1\ \ 0\cdots 0](\boldsymbol{A}^T\boldsymbol{Q}\boldsymbol{A}+\boldsymbol{R})^{-1}=\frac{1}{\det(\boldsymbol{A}^T\boldsymbol{Q}\boldsymbol{A}+\boldsymbol{R})}[A_{11}\cdots A_{M1}]$$

其中，A_{i1}为矩阵（$\boldsymbol{A}^T\boldsymbol{Q}\boldsymbol{A}+\boldsymbol{R}$）第1列元素的代数余子式，并且

$$\boldsymbol{A}^T\boldsymbol{Q}\cdot\begin{bmatrix}1\\ \vdots\\ 1\end{bmatrix}=\begin{bmatrix}\sum\limits_{i=1}^{P}a_iq_i\\ \vdots\\ \sum\limits_{i=1}^{P-M+1}a_iq_{i+M-1}\end{bmatrix}=\begin{bmatrix}f_{11}\\ \vdots\\ f_{1M}\end{bmatrix}$$

注意到（$\boldsymbol{A}^T\boldsymbol{Q}\boldsymbol{A}+\boldsymbol{R}$）为对称阵，$A_{1i}=A_{i1}$，$i=1,\cdots,M$，故有

$$d_r=\frac{1}{\det(\boldsymbol{A}^T\boldsymbol{Q}\boldsymbol{A}+\boldsymbol{R})}(f_{11}A_{11}+\cdots+f_{1M}A_{1M})$$

$$=\frac{1}{\det(\boldsymbol{A}^T\boldsymbol{Q}\boldsymbol{A}+\boldsymbol{R})}\begin{vmatrix}f_{11} & f_{12} & \cdots & f_{1M}\\ l_{21} & r+l_{22} & \cdots & l_{2M}\\ \vdots & \vdots & & \vdots\\ l_{M1} & l_{M2} & \cdots & r+l_{MM}\end{vmatrix}$$

$$=\frac{f_{11}}{\det(\boldsymbol{A}^T\boldsymbol{Q}\boldsymbol{A}+\boldsymbol{R})}\begin{vmatrix}r+l_{22}-l_{21}\dfrac{f_{12}}{f_{11}} & \cdots & l_{2M}-l_{21}\dfrac{f_{1M}}{f_{11}}\\ \vdots & & \vdots\\ l_{M2}-l_{M1}\dfrac{f_{12}}{f_{11}} & \cdots & r+l_{MM}-l_{M1}\dfrac{f_{1M}}{f_{11}}\end{vmatrix}$$

$$=\frac{f_{11}\det(r\boldsymbol{I}+\boldsymbol{L})}{\det(\boldsymbol{A}^T\boldsymbol{Q}\boldsymbol{A}+\boldsymbol{R})}$$

根据定理条件（1）和（2），显然$d_r\neq 0$，这表明稳定的必要条件（4-34）不能满足，所讨论的根轨迹将始终在单位圆外的实轴上，因此$B(z)$对于一切$0\leqslant r<+\infty$均是不稳定的。证毕。

注意在定理4.4中，我们只讨论了r充分大时趋于$z=1$的那条根轨迹对稳定性的影响，而没有注意其他根轨迹在r减小时是否会穿出单位圆外的问题，所以该定理的不稳定条件是充分的。

即使定理条件不满足，$B(z)$ 仍可能对一切 r 均不稳定。但从稳定性来考虑，则上述条件的反面提供了稳定的必要条件。

由定理 4.4，可以推出M较小时$B(z)$ 对一切 r 不稳定的充分条件。

推论4.2 取$M=1$时，若

$$\left(\sum_{i=1}^{P} a_i q_i\right)a_N < 0 \tag{4-35}$$

则$B(z)$ 对一切 $0\leqslant r<+\infty$均是不稳定的。

推论4.3 取$M=2$时，若下述两条件同时满足：

（1）
$$\left(\sum_{i=1}^{P} a_i q_i\right)a_N < 0$$

（2）
$$\left[\left(\sum_{i=1}^{P} a_i q_i\right)\left(\sum_{i=1}^{P-1} a_i^2 q_{i+1}\right) - \left(\sum_{i=1}^{P-1} a_i q_{i+1}\right)\left(\sum_{i=1}^{P-1} a_i a_{i+1} q_{i+1}\right)\right]a_N < 0$$

则$B(z)$ 对一切 $0\leqslant r<+\infty$均是不稳定的。

以上四个定理讨论了控制器 $G_c(z)$ 的稳定性与设计参数的关系。由于在优化性能指标中的时域和权矩阵与特征多项式 $B(z)$ 的系数间缺乏直接的解析关系，这些定理只能在比较特殊的情况下或比较充分地给出稳定性的某些结论，要导出一般的解析关系是很困难的。然而，对于滤波器 $G_F(z)$来说，由式(4-17)可知其特征多项式为

$$H(z)=z^{N-1}+(h_2-1)z^{N-2}+(h_3-h_2)z^{N-3}+\cdots +(h_N-h_{N-1}) \tag{4-36}$$

由于 h_i 是直接可选的设计参数，对其稳定性的讨论则要简单得多。下面我们考虑 h_i 的几种常用选法。

（1） $h_1=1$，$h_i=\alpha$，$i=2$，…，N

在此情况下，有

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

只要选择$0<\alpha<2$，滤波器$G_F(z)$就是稳定的，特别当$0<\alpha<1$时，它还是无振荡的。相应于这一选择的滤波器形式可由式（4-17）算得为

$$G_F(z)=\frac{\alpha}{1+(\alpha-1)z^{-1}} \tag{4-37}$$

它相当于一阶滤波形式。

（2）$h_1=1$，$h_{i+1}=h_i+\alpha^i$，$i=1$，…，$N-1$

在此情况下，式（4-36）为

$$H(z)=z^{N-1}+\alpha z^{N-2}+\cdots+\alpha^{N-1}$$

只要$-1<\alpha<1$，$G_F(z)$就是稳定的，并且当N充分大时

$$H(z)=\frac{z^N-\alpha^N}{z-\alpha}\approx\frac{z^N}{z-\alpha}$$

式（4-17）可近似地化为

$$G_F(z)=\frac{1}{d_s}(1-\alpha z^{-1})c(z) \tag{4-38}$$

这一滤波器相当于引入了一个零点，起到了某种类似微分的作用。

2. DMC系统在模型失配时的鲁棒性

模型失配是指实际对象$G_P(z)$与我们依以设计DMC系统的模型$G_M(z)$不相吻合的情况。它是由于在辨识对象阶跃响应时不够准确，或者对象参数发生时变，或因存在非线性因素等原因引起的。在这种情况下，DMC设计是针对已知模型$G_M(z)$而非实际对象$G_P(z)$进行的。对象的实际阶跃响应$\{\bar{a}_i\}$往往是未知的。

当$G_M(z)\neq G_P(z)$时，由DMC的IMC结构及式（4-10）可知，只要$G_M(z)$、$G_P(z)$，$G_C(z)$、$G_F(z)$四个环节都是稳定的，闭环系统的稳定性将取决于

$$1+G_C(z)G_F(z)[G_P(z)-G_M(z)]$$

的零点多项式。为便于分析起见，可把稳定对象的Z传递函数也写成脉冲响应级数的形式

$$G_P(z)=\bar{a}_1z^{-1}+(\bar{a}_2-\bar{a}_1)z^{-2}+\cdots+(\bar{a}_N-\bar{a}_{N-1})z^{-N}$$

这样，根据式（4-15）～（4-17），就可写出决定闭环系统稳定性的特征多项式

$$P(z)=B(z)H(z)+C(z)[\bar{A}(z)-A(z)] \tag{4-39}$$

其中$B(z)$、$H(z)$的表达式分别见式（4-32）、式（4-36），

$$C(z)=c_2z^{N-2}+(c_3-c_2)z^{N-3}+\cdots+(c_N-c_{N-1})$$

$$\bar{A}(z)=\bar{a}_1z^{N-1}+(\bar{a}_2-\bar{a}_1)z^{N-2}+\cdots+(\bar{a}_N-\bar{a}_{N-1})$$

$$A(z)=a_1z^{N-1}+(a_2-a_1)z^{N-2}+\cdots+(a_N-a_{N-1})$$

在模型参数a_i已知时，式（4-39）中的$B(z)$和$H(z)$的系数分别只取决于控制参数d_i和校正参数h_i，而$C(z)$的系数则同时取决于d_i和h_i。这表明，闭环系统的鲁棒性将同时受到控制策略和校正方式的影响。但4.3节已经指出，控制器$G_c(z)$对模型无失配时的动态特性有决定性的影响，而滤波器$G_F(z)$则与此无关。因此，在DMC系统的设计中，往往通过控制策略的选择确定$G_C(z)$以满足无模型失配时稳定性和动态特性的要求，而模型失配时的鲁棒性则通过校正策略即滤波器$G_F(z)$的选择加以改善。所以下面我们着重讨论滤波器$G_F(z)$对于闭环系统鲁棒性的影响。

在对模型失配缺乏定量描述的情况下，要由式（4-39）对DMC系统作一般的鲁棒性分析是十分困难的。但通过下面的定理，可以看出滤波器$G_F(z)$对于闭环稳定性的影响趋势。

定理4.5 设滤波器$G_F(z)$取式（4-37）的形式，其中$0<\alpha<1$，对于稳定的$G_C(z)$、$G_P(z)$、$G_M(z)$，不论$G_M(z)\neq G_P(z)$如何失配，只要$a_N\cdot\bar{a}_N>0$，则总存在一个充分小的α，使闭环系统在模型失配时既是稳定的，又是无静差的。

证明 上述滤波器相当于取校正参数

$$h_1=1,\ h_i=\alpha,\ i=2,\ \cdots,\ N$$

由此可得

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

$$C(z)=d_s\alpha z^{N-2}$$

故特征多项式（4-39）为

$$
\begin{aligned}
P(z)=&B(z)(z+\alpha-1)z^{N-2}\\
&+d_r\alpha z^{N-2}[\overline{A}(z)-A(z)]\\
=&z^{N-2}\cdot P_1(z)
\end{aligned}
$$

其中

$$
\begin{aligned}
P_1(z)=&(z-1)B(z)+\alpha[B(z)\\
&+d_r(\overline{A}(z)-A(z))]
\end{aligned}
$$

注意到$B(z)$为稳定的多项式，$\alpha>0$，由引理4.3可知，当α充分小时，$P_1(z)$的全部根在单位圆内的充要条件是

$$B(1)[B(1)+d_r(\overline{A}(1)-A(1))]>0$$

由于$B(z)$是稳定的，$B(1)\neq 0$，由$B(1)=d_r a_N$，$\overline{A}(1)=\bar{a}_N$，$A(1)=a_N$可知，上述条件等价于

$$a_N\bar{a}_N>0$$

在这一条件满足时，总存在一个充分小的$\alpha>0$，使闭环系统稳定。

在闭环系统稳定的前提下，可算得输入到输出的稳态增益为

$$
\begin{aligned}
K_{wy}=&\frac{G_C(1)G_P(1)}{1+G_C(1)G_F(1)[G_P(1)-G_M(1)]}\\
=&\frac{\dfrac{1}{a_N}\cdot\bar{a}_N}{1+\dfrac{1}{a_N}\cdot(\bar{a}_N-a_N)}=1
\end{aligned}
$$

而由于扰到输出的稳态增益为

$$
\begin{aligned}
K_{vy}=&\frac{1-G_C(1)G_M(1)G_F(1)}{1+G_C(1)G_F(1)[G_P(1)-G_M(1)]}\\
=&\frac{1-\dfrac{1}{a_N}\cdot a_N}{1+\dfrac{1}{a_N}(\bar{a}_N-a_N)}=0
\end{aligned}
$$

所以闭环系统是无静差的。证毕。

定理4.5揭示了滤波器$G_F(z)$在控制系统中的作用。一方

面，它的存在使 DMC 系统具有反馈校正的闭环机制，从而在模型失配时也具有无静差的性质；另一方面，反馈校正的强弱必须适当选择，以保证模型失配时的鲁棒性。定理中的条件$a_N\bar{a}_N>0$是十分重要的。它表明：并不是对于任意的模型失配，只要充分减弱反馈α使之接近于开环控制，就可得到稳定的响应，这一结论只适用于对象与模型的稳态响应在同一方向的情况。如果对象与模型的稳态响应是反向的，在α充分小时，闭环系统将有一个在实轴上充分接近于$z=1$但在单位圆外的极点，其响应是缓慢发散的。

最后，我们对特定的控制策略和模型失配，较为定量地讨论一下闭环系统的鲁棒性。在这里，考虑一步预测优化策略即$P=M=1$，$Q=1$，$R=0$的情况，由式（4-28）可知，这时

$$G_C(z)=z^{-1}G_M^{-1}(z)$$

注意，由于存在模型失配，式中的$G_P(z)$已为设计时所依据的$G_M(z)$代替。现在，只考虑增益失配

$$G_P(z)=\mu G_M(z)$$

的情况。这时，闭环稳定性将取决于

$$\begin{aligned}&1+G_C(z)G_F(z)[G_P(z)-G_M(z)]\\=&1+(\mu-1)z^{-1}G_F(z)\end{aligned}$$

的零点多项式。考虑前述滤波器的两种取法，即式（4-37）和式（4-38）。

（1）$h_1=1$，$h_i=\alpha$，$i=2$，…，N，$0<\alpha<1$

这时由式（4-37）可得闭环特征多项式

$$P_1(z)=z+(\alpha-1)+\alpha(\mu-1)$$

闭环系统稳定的充要条件为

$$-1<(\alpha-1)+\alpha(\mu-1)<1$$

由此可得

$$0<\mu<\frac{2}{\alpha}$$

可见当实际增益比模型增益小时，均能使闭环系统稳定。并且由

于 $0<\alpha<1$，实际增益大到模型的两倍，也能使系统稳定。α 越小，允许 μ 的失配范围就越大。

（2）$h_1=1, h_{i+1}=h_i+\alpha^i, i=1,\cdots,N-1, \quad 0<\alpha<1$

由于采用一步预测优化，式(4-38) 中 $c(z)$ 的各项系数为

$$c_i=d_1h_i=d_s(1+\alpha+\cdots+\alpha^{i-1})$$

故可知 $\quad c(z)=d_s(1+\alpha)+d_s\alpha^2z^{-1}+\cdots+d_s\alpha^{N-1}z^{-(N-2)}$

当N充分大时近似有

$$c(z)\approx d_s\frac{(1+\alpha)-\alpha z^{-1}}{1-\alpha z^{-1}}$$

结合式（4-38）可得

$$G_F(z)\approx(1+\alpha)-\alpha z^{-1}$$

由此可得闭环特征多项式

$$P_1(z)=z^2+(\mu-1)(1+\alpha)z-\alpha(\mu-1)$$

其稳定的充要条件为

$$-1<-\alpha(\mu-1)<1$$

$$\mu>0$$

$$1-(\mu-1)(1+\alpha)-\alpha(\mu-1)>0$$

由此可得 $$0<\mu<1+\frac{1}{1+2\alpha}$$

当实际增益小于模型增益时，闭环系统均稳定，但当实际增益大于模型增益时，增益失配至多只能大到模型的两倍。可见滤波器的这种取法，其闭环鲁棒性要比上一种取法差。但在 4.2 节中我们看到，它对干扰的抑制要比前者好。

以上的讨论，均是针对对象的最小化模型未知的情况进行的。由于采用了阶跃响应的非最小形式，使闭环鲁棒性涉及到高阶多项式的稳定性判别，除在特殊情况下，很难得到较为定量的结果。但是，如果有可能用最小化形式描述对象，则利用控制器的最小化形式，可大大降低特征多项式的阶数，在这种情况下，有可能更加定量地讨论闭环系统的鲁棒性。在下一章设计典型工业对象的 DMC 系统时，将会遇到这种情况。

第五章　预测控制系统的参数设计

§5.1　动态矩阵控制的参数设计

预测控制与传统的最优控制有很大不同，它采用了启发式优化的概念，允许设计者自由地选择优化性能指标的形式。因此，对于同一被控对象，若在性能指标（3-3）P12中取不同的时域参数和权矩阵，就可导致完全不同的控制效果。这一方面为设计控制系统增加了自由度，但同时却又使缺乏设计经验的人面对一大堆设计参数不知从何做起。为了减少设计的盲目性，本节将以单变量DMC算法为例，讨论设计参数与控制性能的关系，并给出其一般的设计步骤。

由 3.1节给出的DMC算法及其算法流程图 3-3 可知，当DMC算法在线实施时，只涉及到模型参数a_i、控制参数d_i和校正参数h_i。但其中除了h_i可由设计者直接自由选择外，a_i取决于对象阶跃响应特性及采样周期的选择，d_i取决于a_i及优化性能指标(3-3)，它们都是设计的结果而非直接可调参数。从算法的形成来看，在设计中真正要确定的原始参数应该是：

（1）采样周期T；

（2）优化性能指标（3-3）中的优化时域P、控制时域M、误差权矩阵$\boldsymbol{Q}$及控制权矩阵$\boldsymbol{R}$；

（3）校正参数h_i。

由第四章的分析可知，在一般情况下，设计者不可能通过解析方法唯一确定上述参数以满足设计要求。这是因为，这些设计参数是通过a_i、d_i、h_i间接影响控制系统的性能的，它们与控制的快速性、稳定性、鲁棒性、抗干扰性等并没有直接的解析关系可作为设计的定量依据。所以，对于一般的被控对象，DMC通

常要用凑试结合仿真的方法，对设计参数进行整定。

为了便于参数整定，下面首先讨论一下参数的选择原则以及它们对控制系统性能的定性影响。

1．采样周期T与模型长度N

在DMC中，采样周期T的选择应遵循一般采样控制中对T的选择原则。它必须满足香农（Shannon）采样定理，并取决于被控对象的类型及其动态特性。根据被控物理量是电气、机械还是过程量，T可从毫秒数量级取到几十秒数量级。在许多文献中，已对采样周期的选择提出了各种建议，例如：

对于单容对象，可取$T \leqslant 0.1T_a$，这里T_a是对象的惯性时间常数；

对于振荡对象，可取$T \leqslant 0.1T_c$，这里T_c是振荡周期；

对于滞后对象，可取$T \leqslant 0.25T_t$，这里T_t是对象的纯滞后时间。

这些规则自然也可作为DMC选择采样周期的参考。但是，DMC作为一种建立在非最小化模型基础上的算法，采样周期的选择还与模型长度N有密切关系。为了使模型参数a_i（$i=1,\cdots,N$）尽可能完整地包含对象的动态信息，通常要求在NT后的阶跃响应已近似接近稳态值。因此，T的减小将会导致模型维数N的增高。如果T取得过于小，不但加大了计算频率，而且在很短的采样间隔内，计算量因N的增大也将增大，因而影响控制的实时性。所以，从计算机内存和实时计算的需要出发，应合适地选择采样周期，以使模型维数N保持在20～50。

然而，从抗干扰的要求出发，通常希望采用小的采样周期，以便快速及时地抑制干扰的影响。这样，在采样周期的选择上显然存在着抗干扰性与计算实时性的矛盾，这对过渡时间延续较长的动态对象尤为突出。这一困难产生的主要原因，在于模型的非最小化形式，以及在这种形式下的准确预测是通过把模型取得足够长以包含全部动态信息来保证的。当对象过渡时间很长时，为了维持较低的模型维数，采样周期必须取得很大，有时会达到不合

理的地步。

由于工业控制计算机的成本不宜过高，要通过改善硬件环境来解决上述矛盾是不可取的。但通过对其产生原因的分析，可以采用下述措施来解决选择采样周期时所面临的困难。

（1）对于电气、机械等动态较快的对象，最好采用以参数模型为基础的预测控制算法。这样，在线递推计算可以较低的维数进行，采样周期亦可选得较小以适应过程动态的要求。

（2）对于过程量（如温度、液位、流量等）控制，取模型维数为20～50，相应的采样周期可符合采样控制中的一般选取规则。但若对抗干扰性有较高的要求而需进一步减小采样周期，则会遇到模型维数过高的困难。这时，可采用下面的办法较早地截断模型：

若对象有一时间常数较大的主特征运动，其阶跃响应经一段复杂的动态变化后将呈指数上升形式。这时，模型长度N的选择不须延伸到阶跃响应的稳态，而只需覆盖其复杂动态部分后即可截断。对于截断后因信息丢失而引起的预测误差，可将第N个输出预测，由平推形式

$$\tilde{y}_0(k+N+1\,|\,k+1)=\tilde{y}_{cor}(k+N\,|\,k+1)$$

改为指数式递推形式

$$\begin{aligned}\tilde{y}_0(k+N+1\,|\,k+1)=&\ \sigma\tilde{y}_{cor}(k+N\,|\,k+1)\\&+(1-\sigma)\tilde{y}_{cor}(k+N-1\,|\,k+1)\end{aligned}$$

其中$0.9<\sigma<1$。这相当于把模型截断后的指数上升信息都集中在参数σ中反映出来。而我们所用的预测模型，实际上是前$N-1$个数据形成的阶跃响应模型和最后一步的参数化模型的混合。

如果对象有一对时间常数较大的共轭复极点，则其阶跃响应将较长时间地在其稳态值附近振荡。这时，模型也可较早地截断，但N的选择要使a_N近似为阶跃响应的稳态值。

采取了上述策略后，就可在减小采样周期的同时仍将模型维数保持在20～50。这时，虽然模型被过早截断而引起了误差，但由于在截断的同时考虑了其后动态响应的总趋势，故由此引起的预

测误差仍能得到有效的补偿。

（3）DMC是一种优化控制算法。对于过渡时间长的对象，也可先用PID控制加速其动态后再用DMC进行优化控制，这就是所谓"透明控制"(transparent control) 结构[1]。在这种结构下，抗干扰完全可由内部采样频繁的PID控制处理，而在此基础上的DMC，则可采用较大的采样周期和较低的模型维数。

2．优化时域P和误差权矩阵Q

优化时域P和误差权矩阵Q对应着优化性能指标（3-3)中的下述项：

$$\sum_{i=1}^{P} q_i[w(k+i)-\tilde{y}_M(k+i\mid k)]^2$$

它们的物理意义是显而易见的，P表示我们对k时刻起未来多少步的输出逼近期望值感兴趣，而q_i作为权系数，则反映了对不同时刻逼近的重视程度。

为了使动态优化真正有意义，首先要求优化的范围应该是对象的真实动态部分。因此，优化时域P必须超过对象阶跃响应的时滞部分，或由非最小相位特性引起的反向部分，并覆盖对象动态响应的主要部分。此外，由于时滞和非最小相位特性是不可改变的动态，在这一阶段不可能指望对象输出跟踪期望值，所以对应于时滞或反向特性部分的权系数q_i可取为0。这与4.4节中定理4.3和定理4.4给出的理论分析是一致的。为了使控制系统稳定，通常应选择P和q_i满足条件（4-33)，即

$$\left(\sum_{i=1}^{P} a_i q_i\right) a_N > 0$$

这样，在优化时域中阶跃响应的重心与其稳态值同号，至少在控制的方向上是正确的，从而可能得到稳定的控制。

优化时域P的大小对于控制的稳定性和快速性有较大的影响。这里可考虑两种极端情况，一是P足够小，例如只取1，这时优化问题退化为选择$\Delta u(k)$使$y(k+1)=w(k+1)$的

控制问题。如果预测模型足够准确，则它可使对象输出在各采样点紧密跟踪期望值，即实现一步最少拍控制。但这种快速的一步优化控制也有很大的局限性：一是在采样点之间可能有纹波；二是对模型失配及干扰的鲁棒性极差；三是不适用于时滞或非最小相位对象，在非最小相位时甚至会导致不稳定的控制。另一种极端情况是在保持有限控制时域M时把P取得充分大，这相当于定理4.2所指的情况。这时，优化性能指标中稍后时刻的输出预测值几乎只取决于这M个控制增量的稳态响应，并且它们在优化性能指标中占有主要的比重。因此，形为动态的优化实际上接近于稳态优化，很容易导出稳定的控制律，但系统的动态响应将接近于对象的自然响应，在快速性方面不会有明显的改善。上述两种极端情况中，前者虽然快速但稳定性和鲁棒性极差，后者虽然稳定性好但动态响应过于缓慢，且需要把模型长度N人为地加大，因此在实际上都是不可取的。而在这两者之间的P的选择，则涉及到稳定性（鲁棒性）与动态快速性之间的权衡。对于动态复杂的对象，这些性质的好坏并不是随P单调变化的，但在接近极端情况下，上述分析的结论仍然成立。而对于工业过程中大量出现的纯滞后加一阶惯性对象，可以证明稳定性（鲁棒性）和快速性的好坏是随P单调变化的，因此很容易找到适中的P兼顾这两方面的要求。

根据上述分析的结果，一般情况下可令

$$q_i=\begin{cases}0\text{，对应 } a_i \text{为时滞或反向部分}\\1\text{，其他情况}\end{cases}$$

然后选择P，使优化时域包含对象阶跃响应的主要动态部分。以此初选结果进行仿真。若快速性不够，则可适当减小P；若稳定性较差，则可加大P。

由于在DMC的优化性能指标（3-3）中加入了自由选择的权系数q_i，故它的不同选择，可形成目前各种预测控制算法中的不同策略。例如，取

$$q_i=\begin{cases}0, & i<N_1\\ 1, & i\geqslant N_1\end{cases}$$

并记$P=N_2$，则可得到3.3节中GPC的优化性能指标（3-35）。又如，取

$$q_i=\begin{cases}0, & i<P\\ 1, & i=P\end{cases}$$

则可得到文献〔12〕中单步预测控制的性能指标。它的优化实际上只是针对未来某一个时刻进行的。

3. 控制时域M

控制时域M在优化性能指标（3-3）中表示了所要确定的未来控制量改变的数目。由于优化主要是针对未来P个时刻的输出误差进行的，它们至多只受到P个控制增量的影响，所以从物理上应该规定$M\leqslant P$。

由于M是优化变量的个数，在P已确定的情况下，M越小，越难保证输出在各采样点紧密跟踪期望值，所得到的性能指标也就越差。例如$M=1$意味着只用一个控制增量$\Delta u(k)$就要使系统输出在$k+1$，…，$k+P$时刻跟踪期望值，这对动态略为复杂的对象是不可能的，因而得到的动态响应难以满意。为了改善跟踪性能，就要增加M（即控制变量的个数）来提高控制的能力，使各点输出误差的最小化得到更大程度的满足。用M个优化变量实现P个点的输出优化，从物理上讲，就是把P个点优化的要求分担到M个优化变量上。M越小，控制的机动性越弱，这些要求只能在总体上得到平均的兼顾，这样一来，虽然动态响应未必满意，但容易导致稳定的控制，并对模型失配也有较好的鲁棒性。在$M=1$的条件下推出的定理4.1和定理4.2正说明了这一点。而M越大，控制的机动性越强，这样一来，就有可能改善动态响应，但因提高了控制的灵敏度，其稳定性和鲁棒性都会变差。

M对应于矩阵$(\boldsymbol{A}^T\boldsymbol{Q}\boldsymbol{A}+\boldsymbol{R})$的维数，在计算动态控制系数$d_i$时，必须对该矩阵求逆，因此，减小$M$有利于控制系数的计算。

特别当$M=1$时，矩阵求逆退化为求倒数，计算量将显著减小。所以，在一些带有自校正需在线重算d_i的预测控制算法（如GPC）中，常建议取$M=1$。然而正如前面已指出的，如对象的动态较为复杂，则取$M=1$不能得到好的动态响应。在这种情况下，必须在兼顾计算量的同时适当加大M。

上述的物理分析和仿真经验均表明，在许多情况下，增大（减小）P与减小（增大）M有着类似的效果。因此，从简化整定出发，通常可根据对象的动态特性首先选定M，然后只需对P进行整定即可。

4．控制权矩阵$\boldsymbol{R}$

控制权矩阵$\boldsymbol{R}=\text{diag}\ (r_1\cdots r_M)$中，$r_j$常取作同一值，记作$r$。由性能指标（3-3）可知，$q_i$与$r_j$的取值是相对的，因此一旦$q_i$按上述讨论取为0或1，$r$便成为一个可调参数。

权矩阵$\boldsymbol{R}$的作用是对Δu的剧烈变化加以适度的限制，它是作为一种软约束加入到性能指标（3-3）中的。但$\boldsymbol{R}$的加入并不意味着改善控制系统的稳定性。理论分析可以证明[10]，对于一阶对象，充分小和充分大的r均可导致无振荡的控制，但对r的某一中间区域，被控系统是以振荡形式收敛的。而对于二阶对象，虽然在r充分小和充分大时都能得到稳定的控制，但r的某一中间区域却会使控制系统振荡发散。这表明r对稳定性的影响不是单调的，不能简单地通过加大r来改善控制系统的稳定性。

定理4.3已经指出，只要式（4-33）满足，任何系统均可通过增大r得到稳定的控制。但从该定理的证明过程可知，当r充分大时，闭环系统虽然稳定，但有一个接近于$z=1$的极点，因此，这时的闭环响应相当缓慢，会出现类似于积分饱和的现象。这种极端情况显然不能提供满意的控制。

上述分析表明，在调整参数r时，着眼点不应放在控制系统的稳定性上，这部分要求可通过调整P和M得到满足，而引入r的主要作用，则在于防止控制量过于剧烈的变化。因此，在整定时，可先置$r=0$，若相应的控制系统稳定而控制量变化太大，

则可略为加大 r。事实上，只要取一个很小的 r 值，就足以使控制量的变化趋于平缓。

5. 校正参数 h_i

误差校正向量 $\boldsymbol{h}$ 中各元素 h_i 的选择独立于其他设计参数，是DMC算法中唯一直接可调的运算参数。由 4.4 节的分析可知，它仅在对象受到未知干扰或存在模型失配造成预测输出与实际输出不一致时才起作用，而对控制的动态响应没有明显的影响。

4.4 节已经介绍了校正参数 h_i 的两种典型取法，它们对于系统的抗干扰性和鲁棒性有着不同的作用。对相应于滤波器形式(4-37)的选择

$$h_1=1,\ h_i=\alpha,\ i=2,\ \cdots,\ N,\ 0<\alpha\leqslant 1 \tag{5-1}$$

控制系统的鲁棒性将随 α 的减小而增强。α 越接近于 0，反馈校正越弱，鲁棒性有所加强，但对扰动的敏感程度下降，抗干扰性差。α 接近于 1，则情况正好相反。对于相应于滤波器形式(4-38)的另一种取法

$$h_1=1,\ h_{i+1}=h_i+\alpha^i,\ i=1,\ \cdots,\ N-1,\ 0<\alpha<1 \tag{5-2}$$

由于其滤波器中近似引入了一个零点，有助于部分抵消扰动响应的极点，故有较好的抗干扰性，但对模型失配的鲁棒性将会变差。4.4 节中最后的特例讨论，正反映了这两种选择的不同适应性。

由此可以看到，在选择校正系数 h_i 时，必须遵循两个原则：

（1）h_i 应归结为某一参数 α 的有规则的简易表达式，使 h_i 的整定简易可行。

（2）h_i 的类型可根据控制要求的侧重选择式(5-1)或式(5-2)，但其中参数 α 的选择应兼顾到抗干扰性和鲁棒性的要求。

h_i 区别于其他设计参数的最有利之处在于，由于它是直接可选的，因而可在算法中在线设置和改变。如果在DMC结构中再附加一辅助单元，用以分析预测误差的产生是由模型失配还是干扰引起的，则可通过在线改变 h_i 的类型，很好地兼顾抗干扰性和鲁棒性。

根据以上对各设计参数作用的分析，可以看到它们对系统动态性能、稳定性、鲁棒性和抗干扰性的影响趋势。由于这些设计参数的物理意义明确、直观，在结合仿真或实验进行调试时，不难根据实际结果调整相应的参数。应该指出，DMC的设计参数有很大的冗余性，例如改变P、M或q_i都可得到相似的效果，因此对于一般的被控对象，结合上述规律进行整定，不难达到期望的要求。为了增加参数整定的规律性，我们提出下面的适用于一般对象DMC控制的参数整定步骤：

（1）根据对象的类型和动态特性确定采样周期T，获得相应的经光滑的阶跃响应系数a_i。在这里，模型系数应尽可能平滑变化，以消除测量噪声和干扰的影响，否则将严重影响控制系统的稳定性和动态性能。在必要情况下，甚至可放弃模型与实际响应的完全匹配而构造一个光滑的响应模型。模型维数N一般应取20～50。若根据抗干扰的要求有必要进一步减小采样周期，则可依本节前面所述的策略对模型进行截断，以保持较低的模型维数。

（2）取优化时域P覆盖阶跃响应的主要动态部分，这意味着在优化时域中阶跃响应的主要动态已有完全的表现，而不是要把P取到阶跃响应的动态变化全部结束。P的取值可按1，2，4，8，…的序列挑选。初选P后，取

$$q_i=\begin{cases}0\text{，对应 }a_i\text{为时滞或反向部分}\\1\text{，其他情况}\end{cases}$$

（3）初选$r=0$，并取定控制时域

$$M=\begin{cases}1\sim2\text{，对于S形动态简单的对象}\\4\sim8\text{，对于包括振荡的动态复杂的对象}\end{cases}$$

（4）计算控制系数d_i，仿真检验控制系统的动态响应，若不稳定或动态过于缓慢，可调整P直至满意为止。

（5）若对应于上述满意控制的控制量变化幅度偏大，可略为加大r值。

（6）在上述基础上，根据控制要求的侧重点，选择校正参

数h_i的类型，并通过仿真选择参数α，使之兼顾鲁棒性和抗干扰性的要求。

以下，我们给出用上述步骤进行DMC设计的两个例子，并由此观察设计参数对控制性能的影响。

〔例 1〕 最小相位对象

$$G(s)=\frac{8611.77}{[(s+0.55)^2+6^2][(s+0.25)^2+15.4^2]}$$

其阶跃响应为

$$a(t)=1-1.1835e^{-0.55t}\sin(6t+1.4973)$$
$$-0.18038e^{-0.25t}\sin(15.4t-1.541)$$

图 5-1 画出了这一对象的阶跃响应。这是一个弱阻尼振荡的最小相位对象，$a(t)$的稳态值为$a_s=1$，最大超调量$c_{max}=0.93$，过渡时间为$T_{95\%}=6.4$s。

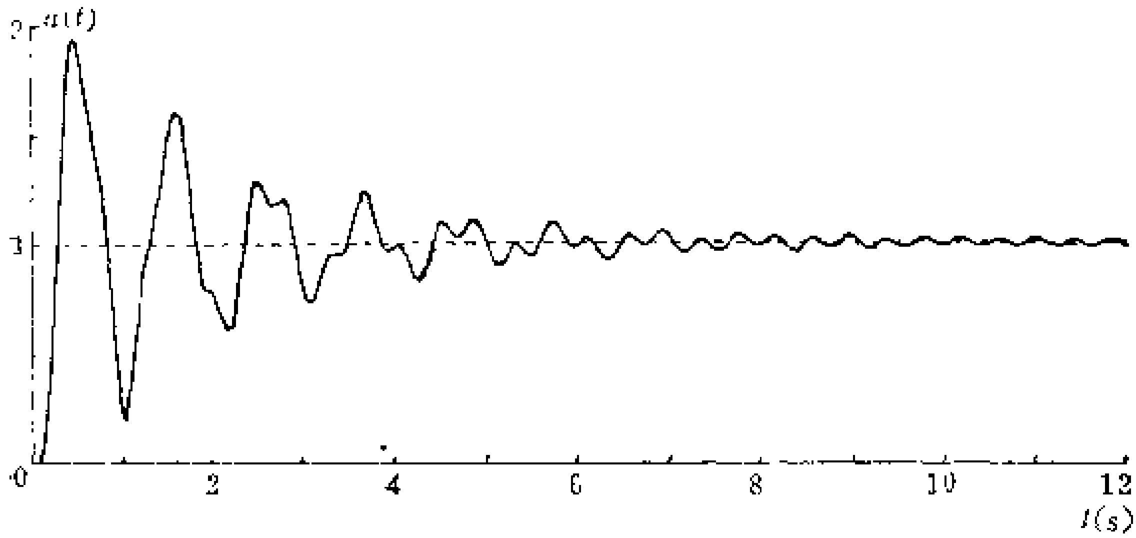


图5-1 〔例 1〕对象的阶跃响应

在DMC设计中，并不要求知道对象的传递函数$G(s)$，而是直接从图 5-1 所示的阶跃响应出发设计控制系统。首先，可根据上述分析取采样周期$T=0.2$s，模型长度$N=40$，显然，这已基本满足了$a_i\approx a_s(i>N)$的要求。由图 5-1 可知，优化时域P的选择至少要使$a(t)$的振荡经历一个周期，以将其主要动态包含在内，故可取$P=6$。此外，由于对象是最小相位的且无

时滞，可令$Q=I$，$r=0$。这时，如取控制时域$M=1$，则可得到图 5-2（a）的曲线。显然，由于对象的动态比较复杂，$M=1$不能得到良好的动态响应。因此，可加大M（取$M=4$），图 5-2（b）给出了这时的动态响应曲线。被控系统阶跃响应的稳态值$a'_s=1$，最大超调量$c'_{max}=0.165$，比控制前减少了5.5倍，过渡时间$T'_{95\%}=0.72$ s，加快为原来的 9 倍。这一结果基本上是满意的。

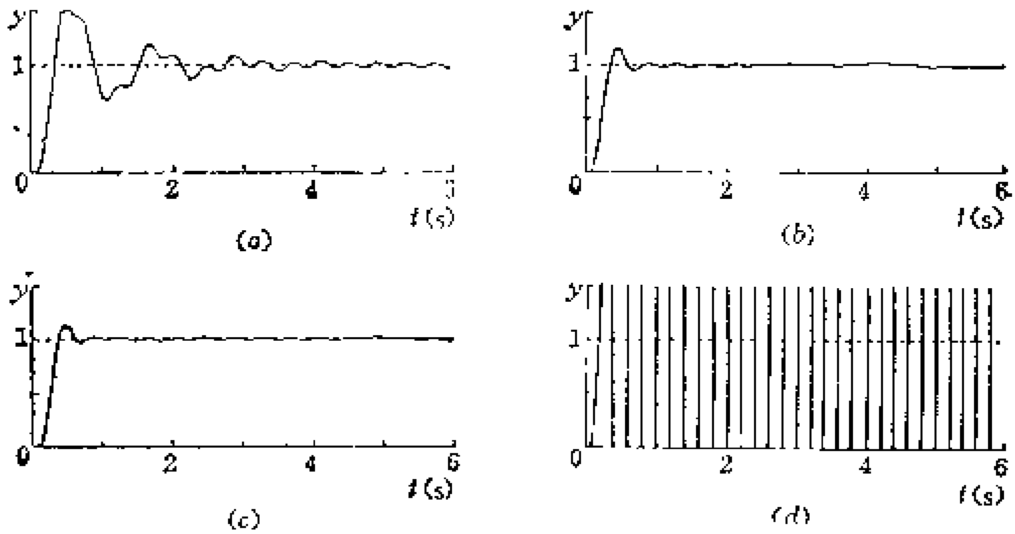


图5-2　控制系统在不同设计参数下的动态响应

（全部情况下均有$Q=I$，$R=0$）

（a）$P=6$，$M=1$；（b）$P=6$，$M=4$；

（c）$P=20$，$M=4$；（d）$P=4$，$M=4$。

优化时域$P=6$（相当于1.2 s）已覆盖了阶跃响应一个周期的变化，对象的主要动态信息已包含在内。由于阶跃响应是以振荡形式重复出现的，所以继续增大P，控制的效果并不会明显变化，图 5-2（c）给出了$P=20$（相当于 4 s）时的响应曲线，它与图 5-2（b）十分接近。这说明，对于振荡型的对象，P对动态快速性的影响不具有单调性。但若把P取得很大，则会出现接近于稳态控制的极端情况。

这一对象尽管是无时滞且为最小相位的，但一步预测优化仍不能得到稳定的控制。图 5-2（d）给出了$P=M=4$时的控制结果（注意，在$Q=I$，$R=0$的条件下，取$P=M$为任意值都

与$P=M=1$的一步控制效果相同)，这时在各采样点上都有$y=1$，但系统是不稳定的。这是因为，一步预测优化将导致完全控制器（4-28)，而这一最小相位对象在采样保持后，其Z传递函数却出现了单位圆外的零点$z=-3.946$，引起了控制器的不稳定。

DMC在形成闭环控制时，对模型失配有很好的鲁棒性。若把实际对象的阶跃响应记作

$$\bar{a}(t)=1-A_1e^{-D_1t}\sin(\omega_1t+1.4973)$$
$$-A_2e^{-D_2t}\sin(\omega_2t-1.541)$$

考虑以下频率、阻尼和幅值失配的情况

	A_1	A_2	D_1	D_2	ω_1	ω_2
标称值	1.1835	0.18038	0.55	0.25	6	15.4
(a)	*	*	*	*	*	12.0
(b)	*	*	*	*	*	18.0
(c)	*	*	0.3	*	*	*
(d)	*	*	1.0	*	*	*
(e)	1.5	0.5	*	*	*	*
(f)	1.0	0	*	*	*	*

其中，*表示取标称值，在取反馈系数$h_i=1$时，图5-3给出了对应情况下的闭环响应，它们基本上都是稳定的。

〔例2〕 非最小相位对象

$$G(s)=\frac{-28705(s-0.3)}{[(s+0.55)^2+6^2][(s+0.25)^2+15.4^2]}$$

它是在〔例1〕对象的传递函数中增加了一个右半平面的零点$s=0.3$。这使对象表现出非最小相位特性并呈现出大幅度的振荡，其过渡时间也较〔例1〕长得多，见图5-4。

这一非最小相位对象的阶跃响应为

$$a(t)=1-23.9e^{-0.55t}\sin(6t+0.067)$$
$$+9.265e^{-0.25t}\sin(15.4t+0.065)$$

阶跃响应的稳态值$a_s=1$，但振荡幅值却达到了$a_{max}=20$和$a_{min}=-27$，过渡时间$T_{95\%}=20$ s。

从图5-4所示的阶跃响应出发，取$T=0.16$ s，$N=65$，

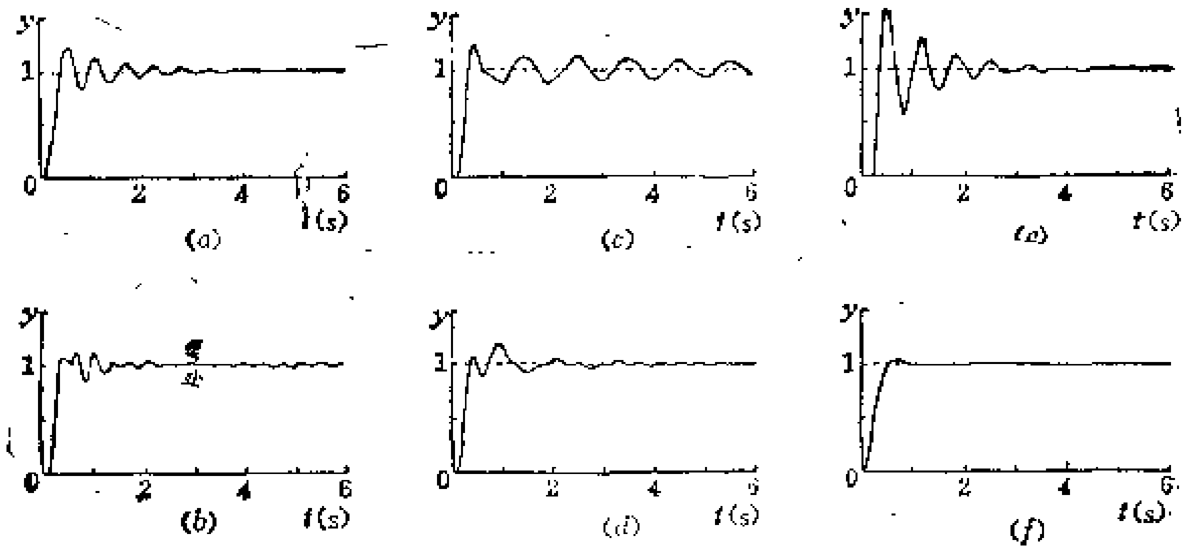


图5-3 模型失配时的动态响应

（a）频率失配$\omega_2=12.0$；（b）频率失配$\omega_2=18.0$；

（c）阻尼失配$D_1=0.3$；（d）阻尼失配$D_1=1.0$；

（e）幅值失配$A_1=1.5$，$A_2=0.5$；（f）幅值失配$A_1=1.0$，$A_2=0$。

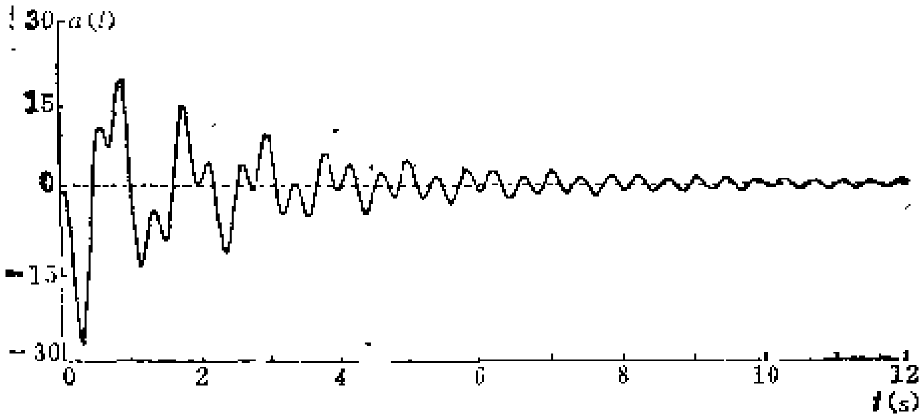


图5-4 〔例 2〕对象的阶跃响应

这时 $a(t)$ 在 $NT=10.4$ s 处虽远未到达稳态，但因 $a_N=a(10.4)=1.0254$ 十分接近 a_s，故可在此处截断模型。考虑到对象动态的复杂性，取 $M=4$，为了覆盖至少一个周期的动态，可取 $P=8$（相当于 1.28 s）。若此时令所有 $q_i=1$，则可算出 $\sum_{i=1}^{8} a_i q_i<0$，由图 5-5 可见，不论如何加大控制权系数 r，都不能使系统稳定。这里的不稳定表现为发散而不是振荡。

为了改变这一情况，根据本节中的分析，可取

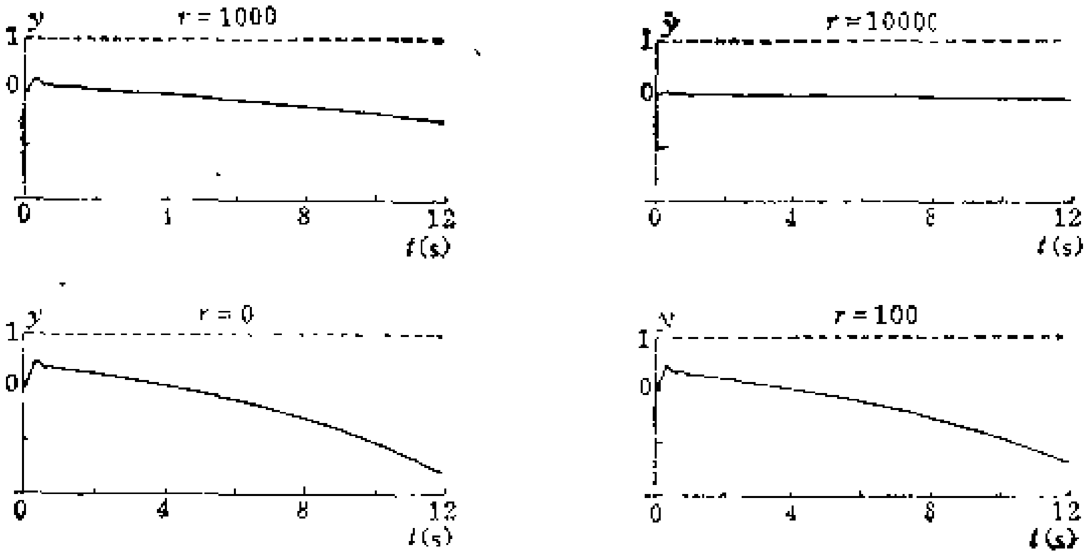


图5-5 加大 r 不能稳定控制系统

$$q_i = \begin{cases} 0 & i = 1, \cdots, 4 \\ 1 & i = 5, \cdots, 8 \end{cases}$$

这时 $\sum_{i=1}^{8} a_i q_i > 0$，即使在 $r = 0$ 时也可得到图 5-6 中曲线 1 所示的稳定响应。被控系统阶跃响应的稳态值仍为 $a_s' = 1$，但无正向超调，由非最小相位特性引起的最大反向幅值为 -1，过渡时间也缩短到10 s 左右。

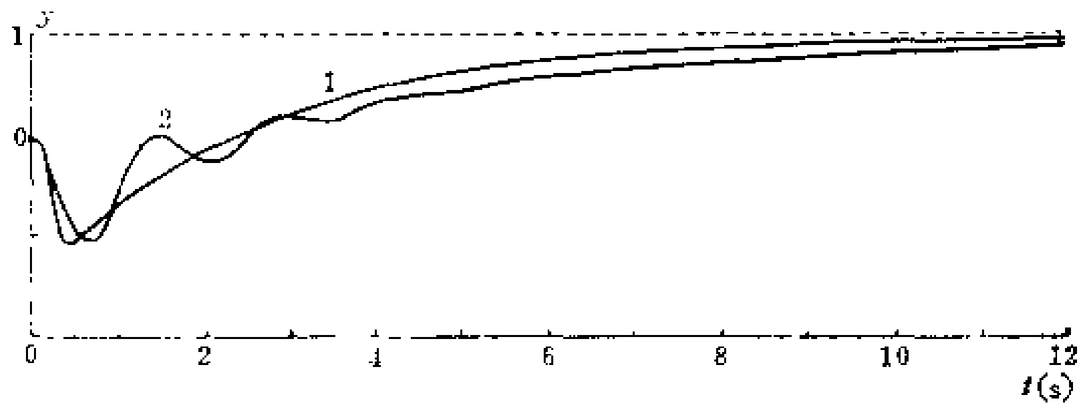


图5-6 DMC 和 PID对〔例 2〕对象的控制结果比较

作为比较，图 5-6 中还画出了用 PID 算法控制这一系统的结果，见曲线 2 。这一用来比较的 PID 控制器是精心选择的，它是在已知对象传递函数的基础上，通过消去对象阻尼较小的部

分$((s+0.25)^2+15.4^2)$并利用误差最小化准则来求出PID参数，然后再以较为频繁的采样周期（$T=0.04\,s$）将其离散化实现的。尽管如此，由图可见，其效果还是不如DMC控制。在同样的反向幅值下，DMC控制曲线较快且平稳地趋于稳态值，而PID作为低阶控制器，至多只能对消对象的一对极点，而由另一对复极点引起的振荡仍将保留在其响应中。

§5.2 典型工业过程动态矩阵控制的解析设计

上节给出的DMC参数设计方法适用于一般的对象。由于缺乏设计参数和控制性能之间的定量关系，故设计参数只能根据它们对控制性能的影响趋势通过仿真或实验加以整定。但对于某些特殊对象，其传递函数易于辨识并具有简单的形式，则有可能在理论分析的基础上导出更为定量的结果，直接用于参数设计。

在工业生产过程中，有很大一类对象可近似地用下述一阶惯性加纯滞后的环节表示

$$G_0(s)=\frac{Ke^{-\tau_0 s}}{1+T_0 s} \tag{5-3}$$

其中，K为过程增益，τ_0为纯滞后，T_0是惯性环节的时间常数，它们很容易根据对象的阶跃响应曲线直接辨识。为了简化讨论，可将时间标度改变为$t'=t/T_0$，输入量改变为$y'=y/K$，首先考虑规范化的传递函数

$$G(s)=\frac{e^{-\tau s}}{1+s} \tag{5-4}$$

其中，$\tau=\tau_0/T_0$。以下，我们就在式(5-4)的基础上讨论这类典型工业过程的 DMC 设计问题。其结果可通过还原变换 $t=T_0t'$，$y=Ky'$ 直接应用于原系统（5-3）。

取采样周期为T，且设τ为T的整数倍即$\tau=lT$（l为非负整数)。规范化对象（5-4）在前接一零阶保持器后，其Z传递函数为

$$G(z)=z^{-(l+1)}\frac{1-\sigma}{1-\sigma z^{-1}} \tag{5-5}$$

其中，$\sigma = \exp(-T)$。相应的阶跃响应系数为

$$a_i = \begin{cases} 0, & i \leqslant l \\ 1 - \sigma^{i-l}, & i > l \end{cases} \tag{5-6}$$

下面我们就从阶跃响应系数$\{a_i\}$出发，来分析DMC系统的闭环性质及其与设计参数的关系[13]。

1. 典型环节DMC控制的策略和IMC形式

典型环节（5-4）有比较简单的动态响应，为了减少设计参数的数目，可首先将其中一部分固定。在这里，除根据采样周期T适当选定模型维数N外，还可设定

控制时域　　　　$M = 1$

误差权矩阵　　　　$\mathbf{Q} = \begin{bmatrix} \mathbf{0}_{l \times l} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_{P \times P} \end{bmatrix}$

控制权矩阵　　　　$\mathbf{R} = \mathbf{0}$

此外，为了记号上的方便，规定优化时域取作$P + l$，并选择式（5-1）形式的校正系数，即

$$h_i = h, \quad i = 2, \cdots, N, \quad 0 < h \leqslant 1$$

这样，所需设计的参数就只剩下优化时域中的P及校正系数h。

在这种控制策略下，可由式（3-8）及阶跃响应表达式(5-6)算出控制系数

$$d_i = \frac{a_i}{s}, \quad i = 1, \cdots, P + l \tag{5-7}$$

其中

$$s = \sum_{i=1}^{P+l} a_i^2 = \sum_{i=l+1}^{P+l} (1 - \sigma^{i-l})^2 = \sum_{i=1}^{P} (1 - \sigma^i)^2$$

$$= \frac{(1 - \sigma^2)P - \sigma(1 - \sigma^P)(2 + \sigma - \sigma^{P+1})}{1 - \sigma^2} \tag{5-8}$$

在4.3节中已经指出，对于时滞对象，其DMC控制只要选择了适当的策略，则它在IMC结构中控制器的最小化形式与无时滞对象采用对应控制策略时的结果完全相同。由于本节采用的控制策略是4.3节中的特例，故可根据对象的无滞后部分求出控制

器的最小化形式

$$G_C(z)=\frac{d_s p(z)}{p^*(z)}$$

其中，$p(z)=1-\sigma z^{-1}$为模型（5-5）的分母多项式，d_s可根据式（5-7）算出为

$$d_s=\sum_{i=1}^{P+l} d_i=\frac{1}{s}\sum_{i=1}^{P}(1-\sigma^i)$$

$$=\frac{(1+\sigma)[(1-\sigma)P-\sigma(1-\sigma^P)]}{(1-\sigma^2)P-\sigma(1-\sigma^P)(2+\sigma-\sigma^{P+1})} \tag{5-9}$$

$p^*(z)$ 可由式（4-27）给出为

$$p^*(z)=1+[(b_2-1)-\sigma]z^{-1}\triangleq 1-(\sigma-\lambda)z^{-1}$$

其中

$$\lambda=b_2-1=\sum_{i=1}^{P+l} d_i a_{i+1}-1$$

$$=\frac{1}{s}\left[\sum_{i=1}^{P+l} a_i(a_{i+1}-a_i)\right]=\frac{1}{s}\left[\sum_{i=1}^{P}(1-\sigma^i)(\sigma^i-\sigma^{i+1})\right]$$

$$=\frac{\sigma(1-\sigma)(1-\sigma^P)(1-\sigma^{P+1})}{(1-\sigma^2)P-\sigma(1-\sigma^P)(2+\sigma-\sigma^{P+1})} \tag{5-10}$$

由式（5-9）及式（5-10）不难验证

$$1-(\sigma-\lambda)$$

$$=\frac{(1-\sigma)[(1-\sigma^2)P-\sigma(1+\sigma)(1-\sigma^P)]}{(1-\sigma^2)P-\sigma(1-\sigma^P)(2+\sigma-\sigma^{P+1})}$$

$$=(1-\sigma)d_s$$

故控制器的最小化形式为

$$G_C(z)=\frac{1-(\sigma-\lambda)}{1-\sigma}\cdot\frac{1-\sigma z^{-1}}{1-(\sigma-\lambda)z^{-1}} \tag{5-11}$$

此外，还可写出模型和滤波器的表达式

$$G_M(z)=z^{-(l+1)}\frac{1-\sigma}{1-\sigma z^{-1}} \tag{5-12}$$

$$G_F(z)=\frac{h}{1-(1-h)z^{-1}} \tag{5-13}$$

其中，式（5-13）即式(4-37)。这样，我们就得到了 DMC 系统在 IMC 结构下各环节的 Z 传递函数，它们为定量分析和设计 DMC 系统提供了依据。在这里，模型和控制器的 Z 传递函数按其算法本来都应是非最小化的高阶形式，见式（4-15）和式（4-16）。但是，只要模型长度 N 取得充分大，以使 $a_i \approx a_s(i > N)$，它们就可近似地用上述最小化形式式（5-12）和式（5-11）表示。

2．典型环节DMC控制的闭环分析

（1）无模型失配时的稳定性和动态特性

当无模型失配即 $G_M(z) = G_P(z)$ 时，根据第四章的分析以及式（5-5）和式（5-11），可得DMC系统的 Z 传递函数为

$$F_o(z) = G_C(z)G_P(z) = z^{-(l+1)}\frac{1-(\sigma-\lambda)}{1-(\sigma-\lambda)z^{-1}} \tag{5-14}$$

由 λ 的表达式（5-10）可知，它取决于参数 P、σ，即 $\lambda = \lambda(P,\ \sigma)$。对于 $0 < \sigma < 1$，可以用归纳法证明 λ 是随 P 的增大单调减小的，因此有

$$\lambda(+\infty,\ \sigma) < \lambda \leqslant \lambda(1,\ \sigma)$$

或

$$0 < \lambda \leqslant \sigma$$

由此可导出关系式

$$0 \leqslant \sigma - \lambda < \sigma < 1 \tag{5-15}$$

由上式及闭环 Z 传递函数表达式（5-14）可知，被控系统是稳定的，其动态特性仍呈现为一阶惯性加纯滞后，它与原对象有相同的增益和纯滞后，只是惯性环节的时间常数由原来的 1 改变为 T_0^*，且由 $\sigma - \lambda = \exp(-T/T_0^*)$ 可导出新的时间常数为

$$T_0^* = \frac{-T}{\ln(\sigma-\lambda)} = \frac{\ln\sigma}{\ln(\sigma-\lambda)} \tag{5-16}$$

由式（5-15）可知

$$T_0^* < 1$$

这表明经 DMC 控制后系统的动态响应将得到加速。此外，对于固定的采样周期，λ 随 P 增加而单调减小，因此 T_0^* 也将随 P 增加而单调增大。P 越大，DMC 系统的动态响应越缓慢。特别当

$P \to +\infty$时，$\lambda \to 0$，$T_0^* \to 1$，即接近于对象的自然响应，而在另一种极端情况$P=1$即一步预测优化时，$\lambda=\sigma$，$T_0^* \to 0^+$，DMC系统除采样系统固有的纯滞后外，可以无过渡地立即到达期望状态。

式（5-14）、式（5-16）结合式（5-10）及$\sigma=\exp(-T)$给出了无模型失配时DMC系统的动态响应与设计参数间的定量关系。被控系统新的时间常数T_0^*将取决于$\sigma(T)$和$\lambda(P, T)$。除了解析式（5-16）外，还可用图5-7所示的曲线形象地反映T_0^*随P、T的变化。可以看出，T_0^*除了随P增大而单调增大外，还随采样周期T的增大而单调增大。特别在T取得较小或P取得较小时，T_0^*随T的变化几乎是线性的。

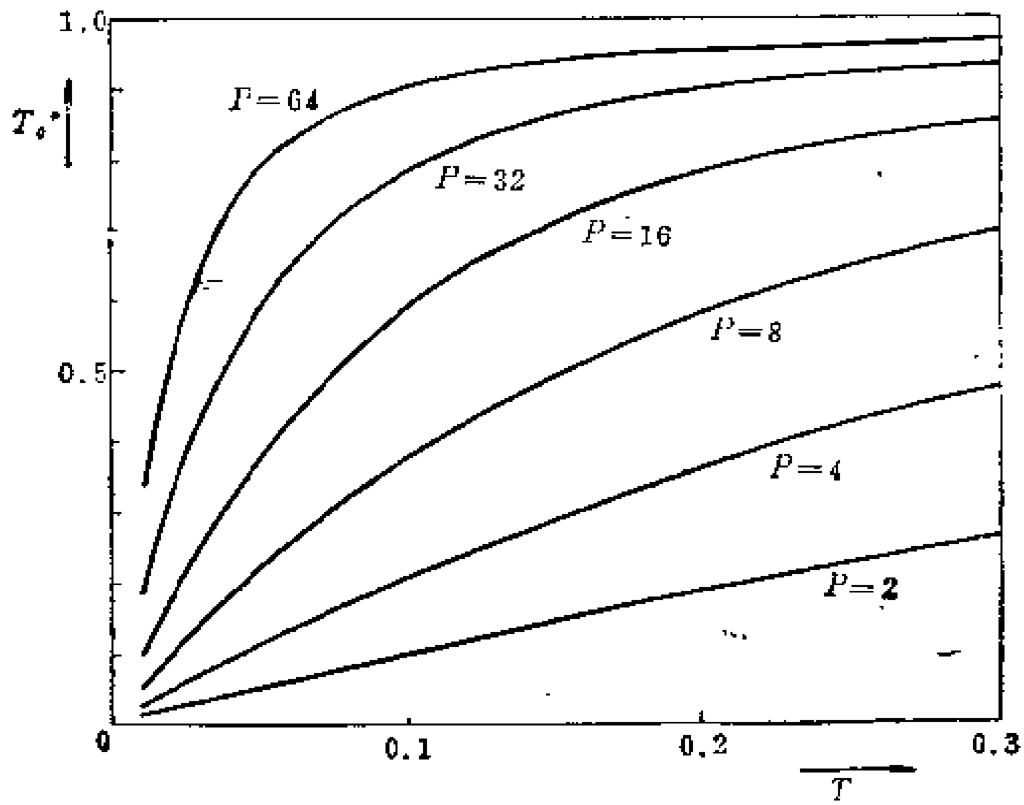


图5-7　时间常数T_0^*与P、T的关系

（2）模型失配时的鲁棒性

模型失配是指我们用式（5-12）作为模型$G_M(z)$进行DMC设计，但实际对象由于非线性、参数时变等原因不再符合式（5-12），从而$G_P(z) \neq G_M(z)$。这里可考虑两种典型情况。

1）增益失配

设实际对象的 Z 传递函数为

$$G_p(z)=\mu G_M(z),\quad \mu>0$$

即它与模型间存在增益失配。这时，由闭环传递关系式（4-10）及这里各环节的具体表达式可得到闭环系统的特征方程

$$\begin{aligned}f(z)=&z^{l+1}+(h-1-\sigma+\lambda)z^{l}+(1-h)(\sigma-\lambda)z^{l-1}\\&+(\mu-1)h(1-\sigma+\lambda)=0\end{aligned} \tag{5-17}$$

当 $\mu=1$ 即对象与模型间无失配时，上述方程的根均在单位圆内，闭环系统稳定。我们关心的则是使闭环系统稳定的 μ 的范围。在 l 较小时，这一界域可通过稳定性分析直接推出。例如，

$l=0$ 时

$$\begin{aligned}f(z)=&\frac{1}{z}\{z^2+[(h-1-\sigma+\lambda)+(\mu-1)\\&\times h(1-\sigma+\lambda)]z+(1-h)(\sigma-\lambda)\}\end{aligned}$$

可得 μ 的稳定界域为

$$0<\mu<\frac{2[1+(1-h)(\sigma-\lambda)]}{h(1-\sigma+\lambda)}$$

$l=1$ 时

$$\begin{aligned}f(z)=&z^2+(\lambda-\sigma+h-1)z+(1-h)(\sigma-\lambda)\\&+(\mu-1)h(1-\sigma+\lambda)\end{aligned}$$

可得

$$0<\mu<\frac{h+1-\sigma+\lambda}{h(1-\sigma+\lambda)}$$

$l=2$ 时

$$\begin{aligned}f(z)=&z^3+(\lambda-\sigma+h-1)z^2+(1-h)(\sigma-\lambda)z\\&+(\mu-1)h(1-\sigma+\lambda)\end{aligned}$$

可得

$$0<\mu<1+\frac{\lambda-\sigma+h-1+\sqrt{\Delta}}{2h(1-\sigma+\lambda)}$$

其中

$$\Delta=(\lambda-\sigma+h-1)^2+4[1-(1-h)(\sigma-\lambda)]$$

根据上述式子，还可看出设计参数对鲁棒性的影响。例如，由 $l=0$，1时 μ 的失配范围可明显看出，增大优化时域中的 P（即减小 λ）或减小校正系数 h，均可提高系统在增益失配时的鲁棒性。即使是对于 $P=1$ 这种最敏感的情况（这时优化时域为 $l+1$ 即跨越了时滞后的一步预测优化），所允许的增益失配也有很宽的范围：

$$0<\mu<\frac{2}{h},\quad l=0$$

$$0<\mu<1+\frac{1}{h},\quad l=1$$

$$0<\mu<1+\frac{2}{h[\sqrt{(1-h)^2+4}+(1-h)]},\quad l=2$$

由于 $0<h\leqslant 1$，μ 的上界至少可达2，当 h 减小时，这一上界可进一步提高直至无穷。

当对象纯滞后 l 较大时，由式（5-17）解析导出 μ 的稳定界域便遇到了困难。虽然在文献中已提出不少离散系统稳定性的充要判据，但当方程（5-17）的维数较高且有参数出现在系数中时，由这些判据导出的关系式相当复杂，难以给出这些参数应满足的解析条件。尽管如此，当系数均为数值时，用这些判据判断稳定性仍是十分快速有效的。因此，我们可对固定的 T、l、P、h，通过式（5-17）和稳定性判据数值地搜索 μ 的界域。为了使搜索过程简化且有规律，下面首先分析一下稳定性与 μ 的关系。

引理5.1 设有离散特征方程

$$f(z)=p(z)+k=0 \tag{5-18}$$

其中，n 阶多项式

$$p(z)=z^n+p_1z^{n-1}+\cdots+p_n$$

的全部根都在单位圆内，k 为参数。当 $k=k_1>0$ 时，若 $f(z)$ 稳定，则 $f(z)$ 对一切 $k_1\geqslant k\geqslant 0$ 均稳定。当 $k=k_2<0$ 时，若 $f(z)$ 稳定，则 $f(z)$ 对一切 $k_2\leqslant k\leqslant 0$ 均稳定。

证明 将式（5-18）改写为

$$1+k\cdot\frac{1}{p(z)}=0$$

由根轨迹性质可知，$k\to+\infty$时，n条根轨迹将趋于Z平面无穷远处。由于$k=0$时$p(z)$的根均在单位圆内，故每一条根轨迹至少与单位圆相交1次。同理，对应于$k<0$的补根轨迹，当k从$-\infty$向0变化时，它们将从Z平面无穷远处趋于单位圆内$p(z)$的根，每一条补根轨迹也至少与单位圆相交1次。

现在我们要证明，对应于式（5-18）的每一条根轨迹或补根轨迹与单位圆必定且只能相交1次，为此只需证明全部根轨迹和补根轨迹穿越单位圆的总次数不大于$2n$。

首先，注意到Z平面上的实轴不是属于根轨迹便是属于补根轨迹，因此，根轨迹或补根轨迹必然穿越$z=-1$点，对应的$k=-p(-1)$。在$p(z)$的根全部位于单位圆内时，不难证明多项式$p(z)+k$在$z=-1$处只有单根。因此，根轨迹或补根轨迹在$z=-1$处穿越单位圆的次数为1。

进一步作变换

$$z=\frac{1+w}{1-w}$$

则将Z平面上除$z=-1$外的单位圆变换为W平面上的虚轴，并将Z平面上单位圆内的区域变换为W的左半平面。经此变换后，式（5-18）成为

$$A(w)+kB(w)=0$$

其中，$A(w)=(1+w)^n+p_1(1+w)^{n-1}(1-w)+\cdots+p_n(1-w)^n$的阶次$n_A\leqslant n$，$B(w)=(1-w)^n$的阶次$n_B=n$。由于根轨迹和补根轨迹在$Z$平面上对单位圆（$z=-1$除外）的穿越对应着$W$平面上对虚轴的穿越，故穿越点应是下述方程的解：

$$A(jy)+kB(jy)=0$$

其中y为实数，是W平面虚轴上穿越点的纵坐标。将实部与虚部分离可得

$$A(jy)=A_1(y)+jA_2(y)$$
$$B(jy)=B_1(y)+jB_2(y)$$

其中，$A_1(y)$、$A_2(y)$ 的阶次分别为n_A、n_A-1（n_A为偶数）或n_A-1、n_A（n_A为奇数），$B_1(y)$、$B_2(y)$ 的阶次分别为 n_B、n_B-1（n_B为偶数）或n_B-1、n_B（n_B为奇数）。

在穿越点处，y必须满足方程

$$A_1(y)+kB_1(y)=0$$
$$A_2(y)+kB_2(y)=0$$

消去k后有

$$A_1(y)B_2(y)-A_2(y)B_2(y)=0$$

容易证明，这一方程的阶次不超过$2n-1$，其实根（包括重根）至多只有$2n-1$个。所以，根轨迹和补根轨迹在Z平面上至多穿越单位圆（包括$z=-1$点）$2n$次。结合前面的分析可知，每一根轨迹和补根轨迹必定穿越且只能穿越单位圆1次。

若对$k=k_1>0$，$f(z)$的根都在单位圆内，说明由$k=0$出发的n条根轨迹都还未曾穿越单位圆，因此$f(z)$对一切$0\leqslant k\leqslant k_1$稳定。若对$k=k_2<0$，$f(z)$的根均在单位圆内，则说明它的$n$条补根轨迹由对应于$k=-\infty$的$Z$平面无穷远处出发后，已全部进入了单位圆，当$k$继续向0变化时，它们不可能再返回单位圆外，故$f(z)$对于一切$k_2\leqslant k\leqslant 0$均稳定。证毕。

引理5.2 设有离散特征方程

$$f(z)=(z-\alpha)(z-\beta)z^n+k=0 \qquad (5\text{-}19)$$

其中，$0\leqslant\alpha<1$，$0\leqslant\beta<1$均为实常数，n为非负整数，k为参数。如果

$$|k|<(1-\alpha)(1-\beta)$$

则$f(z)$的全部根在单位圆内。

证明 首先我们讨论方程（5-19）在单位圆上的根必须满足的条件。为此，设$z=x+jy(x^2+y^2=1)$为单位圆上的点，代入式（5-19）后，有

$$(x-\alpha+jy)(x-\beta+jy)(x+jy)^n+k=0$$

复数的模必须满足条件

$$|k|=\sqrt{(x-\alpha)^2+y^2}\cdot\sqrt{(x-\beta)^2+y^2}\cdot(\sqrt{x^2+y^2})^n$$
$$=\sqrt{1-2\alpha x+\alpha^2}\cdot\sqrt{1-2\beta x+\beta^2}\geqslant(1-\alpha)(1-\beta)$$

由此可见，在引理条件下，根轨迹不可能与单位圆相交。由于 $k=0$ 时方程的根均在单位圆内，故 $f(z)$ 的全部根都在单位圆内。证毕。

根据上述引理和特征方程 (5-17)，立即可得到下面的定理。

定理5.1 对于有增益失配的典型工业过程式 (5-5)，只要增益变化范围为

$$0<\mu<2$$

则其 DMC 系统必定稳定，且与时滞拍数 l、采样周期 T 及设计参数 P、h 无关。

证明 在增益失配时，DMC 系统的闭环特征方程为式 (5-17)，它可写作为

$$[z-(1-h)][z-(\sigma-\lambda)]z^{l+1}$$
$$+(\mu-1)h(1-\sigma+\lambda)=0$$

由于 $0\leqslant 1-h<1$，$0\leqslant\sigma-\lambda<1$，故当 $l\geqslant 1$ 时，由引理 5.2 可知，闭环系统对一切

$$|(\mu-1)h(1-\sigma+\lambda)|<[1-(1-h)]$$
$$\times[1-(\sigma-\lambda)]$$

均稳定，上式即意味着

$$0<\mu<2$$

而当 $l=0$ 时，由前面推出的 μ 的稳定界域可知，在定理条件下闭环系统亦稳定。证毕。

定理5.1 的结论与前面 $l=0$，1，2 时推出的结论是相容的，它适用于更一般的情况，表明了 DMC 算法对于典型工业过程的增益失配有很强的鲁棒性。这一定理还表明，当对象实际增益小于模型增益即 $0<\mu<1$ 时，DMC 闭环系统总是稳定的，因此 μ 的失配下界为 $\mu_{min}=0$，不须再去寻找。

应该注意，定理5.1给出的条件是充分的。$\mu=2$并不是失配的临界值，实际的失配上界μ_{max}将取决于l、T、P和h。前面的讨论中已给出了对应于$l=0$，1，2时的μ_{max}，显然，μ_{max}不可能用T、P和h的解析式一般地给出，因此它的确定必须借助于数值方法。在l、T、P和h已经确定的情况下，由引理5.1可知，存在一个μ_{max}，使系统在$\mu\leqslant\mu_{max}$时必稳定，而在$\mu>\mu_{max}$时必不稳定。因此，可用一般的单变量搜索法，如黄金分割法，对μ的确定值判断方程（5-17）的稳定性，并逐步缩小搜索区间，直至找到具有一定精度的μ_{max}。

2）滞后失配

设实际对象的Z传递函数为

$$G_p(z)=z^{-(k+1)}\frac{1-\sigma}{1-\sigma z^{-1}}$$

上式亦可写作

$$G_p(z)=z^{l-k}G_M(z)$$

其中，k为实际对象的纯滞后拍数，为简化讨论计，设k为非负整数。这种情况表明了模型与对象间的滞后失配。

由闭环传递关系（4-10），可得到这时的闭环特征方程

$$\left.\begin{aligned}f(z)=&z^{k+1}+(h-1-\sigma+\lambda)z^k+(1-h)\\&\times(\sigma-\lambda)z^{k-1}-h(1-\sigma+\lambda)z^{k-l}\\&+h(1-\sigma+\lambda)=0,\quad k\geqslant l\\f(z)=&z^{l+1}+(h-1-\sigma+\lambda)z^l+(1-h)\\&\times(\sigma-\lambda)z^{l-1}+h(1-\sigma+\lambda)z^{l-k}\\&-h(1-\sigma+\lambda)=0,\quad k<l\end{aligned}\right\}\tag{5-20}$$

由于k是离散变化的，并且它不是出现在特征方程的系数中而是出现在z的次数中，所以在这种情况下，不能像增益失配那样用根轨迹分析导出失配的界域。但由于$k=l$即无失配时闭环系统是稳定的，所以在l、T、P、h已经确定的情况下，可从$k=l$的标称情况出发，令k递增与递减，对每一个k，可由特征方程（5-20）用稳定性判据判断闭环系统是否稳定，直至相应

的 k 引起不稳定为止。这样，就可得到 k 的稳定界域

$$k_{\min} \leqslant k \leqslant k_{\max}$$

为了快速有效地进行稳定性判别，我们建议采用文献[14]中的如下朱里（Jury）判据：

朱里稳定性判据

离散系统的特征方程

$$f(z)=a_0z^n+a_1z^{n-1}+\cdots+a_n=0\ ,\quad a_0>0$$

的全部根都在 Z 平面单位圆内的充要条件是：

$$|a_n|<a_0$$

且 $n-1$ 阶多项式

$$\begin{aligned} f_1(z) &= \frac{1}{z}[a_0f(z)-a_nz^nf(z^{-1})] \\ &= b_0z^{n-1}+b_1z^{n-2}+\cdots+b_{n-1} \end{aligned}$$

的全部根也都在单位圆内。这里

$$b_k=a_0a_k-a_na_{n-k},\quad k=0\ ,\ \cdots,\ n-1$$

这一判据通过递推方法把稳定性判别归结为对不同阶次多项式相应于 $|a_n|<a_0$ 的条件，在这里的计算量主要是由 $f(z)$ 导出低一阶多项式 $f_1(z)$ 时的系数计算，但相应的公式十分简单。所以，可以快速有效地进行稳定性判别。

3．典型环节DMC控制的自动参数设计[15]

对于式（5-4）描述的典型工业对象，我们可以在上述分析的基础上，建立计算机辅助设计程序，来自动寻找满足设计要求的参数 P 和 h，而不必通过凑试结合仿真来整定参数。这一程序建立在模型无失配时的闭环 Z 传递函数（5-14）和模型有增益或时滞失配时的闭环特征方程式（5-17）、式（5-20）的基础上，充分利用了所得到的解析结果，并结合了有效的稳定性判据和搜索策略，可以快速地给出所需的 P、h 值，其计算量远远小于凑试结合仿真的整定方法。

在根据对象特性选定采样周期 T 后，假定我们对典型对象的DMC控制提出如下设计要求：

（1）在无模型失配时，闭环系统的时间常数$T_s^* \leqslant T_{max}^*$；

（2）不论模型是否失配，闭环系统无静差；

（3）作用于对象的有界干扰能快速且完全得到抑制；

（4）对象增益在$(0, \mu_{max}]$间变化时，闭环系统能保持稳定；

（5）对象滞后在$[k_{min}, k_{max}]$间变化时，闭环系统能保持稳定。

由于采样周期T已确定，l也随之确定，满足上述设计要求的可调参数便只有P和h。因为无模型失配时的动态快速性只取决于P，故可先确定P的范围以满足闭环响应快速性的要求，然后再寻找h使之满足鲁棒性要求。注意，虽然$h \to 0$可使允许失配范围变得很大，但为了快速地抑制干扰，应尽可能将h取大一些，以兼顾抗干扰性要求。至于闭环系统的无差性质则是由算法的结构特点保证的（见4.2节），故在设计中可不予考虑。

这一计算机辅助参数自动设计的具体步骤如下：

第1步　根据式（5-16）寻找P的范围，使相应的$T_s^* \leqslant T_{max}^*$。这里虽因涉及超越方程无解析解，但因$T_s^*$随$P$的增大而单调增大，故可将$P$由小设置到大，通过计算相应的$T_s^*$检验其是否满足快速性要求，直至找到满足要求的最大$P$为止。计算出对应于该$P$值的$\lambda$。

第2步　对于$h=1$，用黄金分割法结合稳定性判据寻找闭环系统稳定的增益失配上限，检验其是否大于给定的μ_{max}。若不满足，则按一定步长减小h，重复上述步骤，直至找到闭环系统在增益失配范围$(0, \mu_{max}]$内均稳定的h的上限。

第3步　对于上述h，用稳定性判据检验k从k_{min}变化到k_{max}时闭环系统是否均稳定。若不满足，则以适当步长减小h，直至找到满足上述滞后失配条件的h值。

下面，我们用一个例子来说明上述参数设计方法的有效性。设被控对象的传递函数为

$$G(s)=\frac{e^{-0.4s}}{1+s}$$

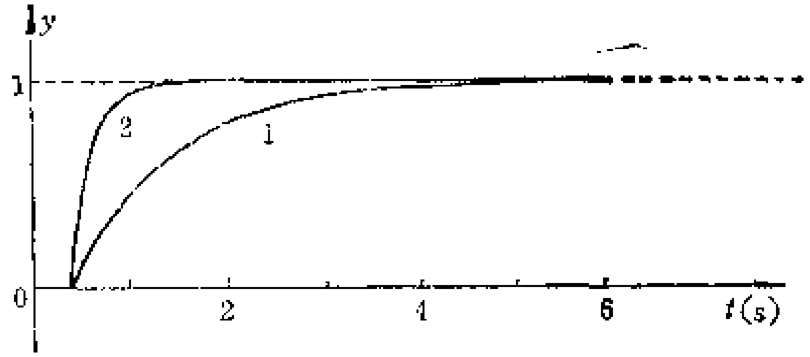


图5-8 对象（曲线1）和被控系统（曲线2）的动态响应

其单位阶跃响应如图 5-8 中曲线 1 所示。

取采样周期 $T=0.1$ s，在用DMC算法控制该对象时，闭环系统应满足以上所述的要求，其中给定

$$T^*_{max}=0.2\ \text{s}\ ,\ \mu_{max}=2.5$$

$$k_{min}=0\ ,\ k_{max}=10$$

经计算机辅助设计，可找出设计参数

$$P=8\ ,\ \ h=0.8$$

由图 5-8 中曲线 2 可见，无模型失配时的闭环响应仍表现为一阶惯性加纯滞后，但惯性环节的时间常数经控制后已减小为约 0.2 s，动态快速性满足了设计要求。

图 5-9和图 5-10 分别画出了在有增益和时滞失配时闭环系统的动态响应，所选参数恰好满足了设计要求。尽管存在模型失配，但在所有稳定情况下闭环系统均无静差。

在图 5-11 中，画出了一个幅值为 $d=0.2$的恒值扰动作用在对象输入端时的闭环响应。由图可见，控制系统能完全抑制干扰。至于干扰影响消除不够快，则是因为我们采用了侧重于鲁棒性的校正策略 (5-1)。对于抗干扰性只能在满足鲁棒性要求的基础上尽量加大 h 予以兼顾。

本节给出的分析和设计方法虽然都是针对典型环节 (5-4) 的，但其结果也近似适用于具有S型响应的其他过程。由于这类过程在实际工业生产中大量存在，对它们DMC控制的解析设计，对于DMC的推广应用是很有意义的。

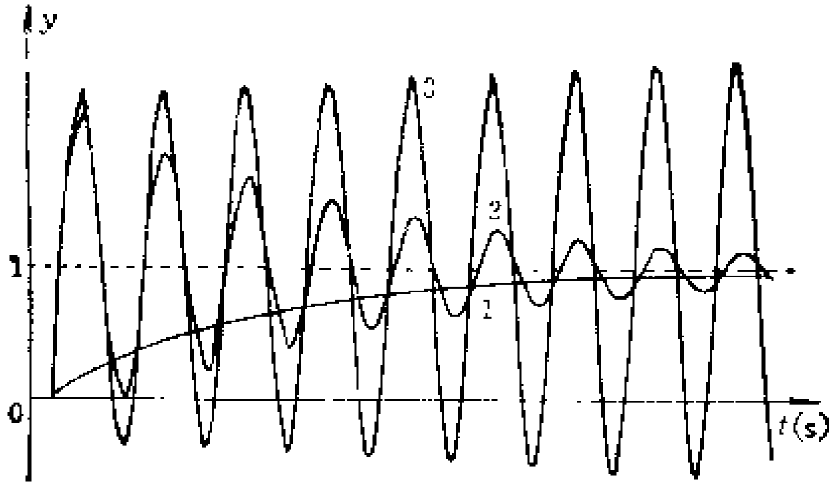


图5-9　增益失配时的闭环响应

1 — $\mu=0.2$；2 — $\mu=2.4$；3 — $\mu=2.6$。

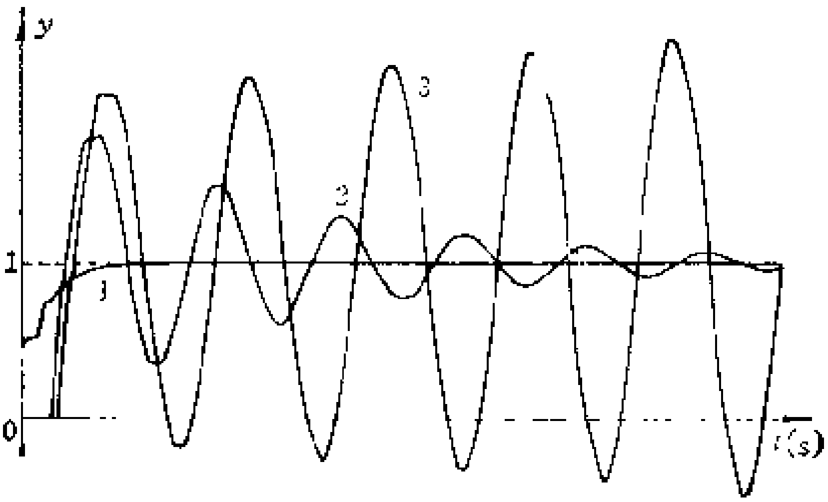


图5-10　滞后失配时的闭环响应

1 — $k=0$；2 — $k=10$；3 — $k=11$。

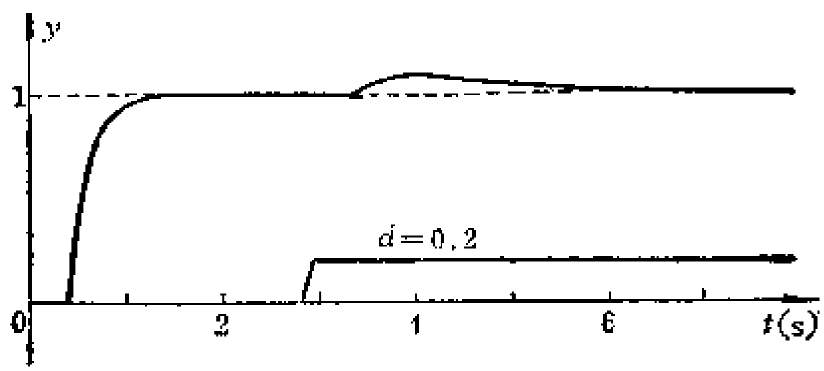


图5-11　闭环系统的干扰响应

第六章　多变量预测控制算法

以上各章，我们主要从原理角度介绍了预测控制算法。特别是以单变量DMC算法为典型，分析了其中的预测控制机理和闭环系统的性质，并给出了单变量DMC控制的设计要点。需要指出的是，预测控制算法固然可以取代PID算法用于单回路控制，但作为一种优化控制算法，其效用的发挥则更多地体现在对多变量系统和具有复杂结构系统的控制上。因此，从本章起我们将介绍DMC在复杂系统中的实现，这对其实际应用于复杂的工业过程具有更重要的意义。

§6.1　多变量系统的动态矩阵控制

3.1节所介绍的单变量DMC算法，是建立在下述基本原理基础上的：

（1）基于预测模型和线性系统比例、叠加性质的输出预测；

（2）基于最优跟踪和控制软约束性能指标的在线滚动优化；

（3）基于实时检测信息的误差预测与校正。

显然，这些原理也很容易推广到多变量系统。

设被控对象有m个控制输入、p个输出，假定已测得每一输出y_i对每一输入u_j的阶跃响应$a_{ij}(t)$，则可由它们在采样点上的值组成模型向量

$$a_{ij}=[a_{ij}(1)\cdots a_{ij}(N)]^T$$
$$i=1,\ \cdots,\ p,\qquad j=1,\ \cdots,\ m$$

这些就是多变量DMC控制的出发点。其中，N为模型长度，其意义同3.1节所述。对于这$p\times m$个阶跃响应，为统一记号及编程方便起见，可取同一个N值。

以下，类似于3.1节，我们根据预测控制的基本原理来推导

多变量 DMC 控制的算法。

1. 预测模型

对于线性多变量系统，其输出预测可通过单变量预测后叠加得到。为此，首先考虑在输入 u_j 作用下对于输出 y_i 的预测。类似于式（3-9），可写出 u_j 有一个增量 $\Delta u_j(k)$ 时 y_i 在未来 N 个时刻的输出预测值

$$\tilde{\boldsymbol{y}}_{i,N1}(k)=\tilde{\boldsymbol{y}}_{i,N0}(k)+\boldsymbol{a}_{ij}\Delta u_j(k) \tag{6-1}$$

其中

$$\tilde{\boldsymbol{y}}_{i,N1}(k)=\begin{bmatrix}\tilde{y}_{i,1}(k+1\mid k)\\ \vdots\\ \tilde{y}_{i,1}(k+N\mid k)\end{bmatrix},\quad \tilde{\boldsymbol{y}}_{i,N0}(k)=\begin{bmatrix}\tilde{y}_{i,0}(k+1\mid k)\\ \vdots\\ \tilde{y}_{i,0}(k+N\mid k)\end{bmatrix}$$

这里除了加上了 y，u 的编号 i，j 外，其余符号的含义与 3.1 节相同。$\tilde{\boldsymbol{y}}_{i,N0}(k)$ 的各分量表示在 k 时刻全部控制量 u_1，…，u_m 保持不变时对 y_i 在未来 N 个时刻的初始预测值。

同样，在 u_j 依次有 M 个增量变化

$$\Delta u_j(k),\ \cdots,\ \Delta u_j(k+M-1)$$

时，类似于式（3-4），可得 y_i 在未来 P 个时刻的预测值为

$$\tilde{\boldsymbol{y}}_{i,PM}(k)=\tilde{\boldsymbol{y}}_{i,P0}(k)+\boldsymbol{A}_{ij}\Delta \boldsymbol{u}_{j,M}(k) \tag{6-2}$$

其中

$$\tilde{\boldsymbol{y}}_{i,PM}(k)=\begin{bmatrix}\tilde{y}_{i,M}(k+1\mid k)\\ \vdots\\ \tilde{y}_{i,M}(k+P\mid k)\end{bmatrix},\quad \tilde{\boldsymbol{y}}_{i,P0}(k)=\begin{bmatrix}\tilde{y}_{i,0}(k+1\mid k)\\ \vdots\\ \tilde{y}_{i,0}(k+P\mid k)\end{bmatrix}$$

$$\boldsymbol{A}_{ij}=\begin{bmatrix}a_{ij}(1) & & 0\\ \vdots & \ddots & \\ a_{ij}(M) & \cdots & a_{ij}(1)\\ \vdots & & \vdots\\ a_{ij}(P) & \cdots & a_{ij}(P-M+1)\end{bmatrix}$$

$$\Delta \boldsymbol{u}_{j,M}(k)=\begin{bmatrix}\Delta u_j(k)\\ \vdots\\ \Delta u_j(k+M-1)\end{bmatrix}$$

式（6-1）和式（6-2）就是 y_i 在u_j 单独作用下的预测模型。若 y_i 受到 u_1，…，u_m 的共同作用，则可按线性系统的性质进行叠加。

若各 u_j 只有即时变化 $\Delta u_j(k)$，则对应于式（6-1），有

$$\tilde{\mathbf{y}}_{i,N_1}(k)=\tilde{\mathbf{y}}_{i,N_0}(k)+\sum_{j=1}^{m}\mathbf{a}_{ij}\Delta u_j(k)$$

若各 u_j 从 k 时刻起均变化M次，即有控制增量，$\Delta u_j(k)$，…，$\Delta u_j(k+M-1)$（$j=1$，…，m），则对应于式（6-2），有

$$\tilde{\mathbf{y}}_{i,PM}(k)=\tilde{\mathbf{y}}_{i,P0}(k)+\sum_{j=1}^{m}\mathbf{A}_{ij}\Delta\mathbf{u}_{j,M}(k)$$

我们把所有的 y_i 合并在一个向量中，为使符号简洁，记

$$\tilde{\mathbf{y}}_{N_1}(k)=\begin{bmatrix}\tilde{\mathbf{y}}_{1,N_1}(k)\\ \vdots\\ \tilde{\mathbf{y}}_{p,N_1}(k)\end{bmatrix},\qquad \tilde{\mathbf{y}}_{N_0}(k)=\begin{bmatrix}\tilde{\mathbf{y}}_{1,N_0}(k)\\ \vdots\\ \tilde{\mathbf{y}}_{p,N_0}(k)\end{bmatrix}$$

$$\tilde{\mathbf{y}}_{PM}(k)=\begin{bmatrix}\tilde{\mathbf{y}}_{1,PM}(k)\\ \vdots\\ \tilde{\mathbf{y}}_{p,PM}(k)\end{bmatrix},\qquad \tilde{\mathbf{y}}_{P0}(k)=\begin{bmatrix}\tilde{\mathbf{y}}_{1,P0}(k)\\ \vdots\\ \tilde{\mathbf{y}}_{p,P0}(k)\end{bmatrix}$$

$$\overline{\mathbf{A}}=\begin{bmatrix}\mathbf{a}_{11}&\cdots&\mathbf{a}_{1m}\\ \vdots&&\vdots\\ \mathbf{a}_{p1}&\cdots&\mathbf{a}_{pm}\end{bmatrix},\qquad \mathbf{A}=\begin{bmatrix}\mathbf{A}_{11}&\cdots&\mathbf{A}_{1m}\\ \vdots&&\vdots\\ \mathbf{A}_{p1}&\cdots&\mathbf{A}_{pm}\end{bmatrix}$$

$$\Delta\mathbf{u}(k)=\begin{bmatrix}\Delta u_1(k)\\ \vdots\\ \Delta u_m(k)\end{bmatrix},\qquad \Delta\mathbf{u}_M(k)=\begin{bmatrix}\Delta\mathbf{u}_{1,M}(k)\\ \vdots\\ \Delta\mathbf{u}_{m,M}(k)\end{bmatrix}$$

则可得到一般的多变量系统预测模型

$$\tilde{\mathbf{y}}_{N_1}(k)=\tilde{\mathbf{y}}_{N_0}(k)+\overline{\mathbf{A}}\Delta\mathbf{u}(k) \tag{6-3}$$

$$\tilde{\mathbf{y}}_{PM}(k)=\tilde{\mathbf{y}}_{P0}(k)+\mathbf{A}\Delta\mathbf{u}_M(k) \tag{6-4}$$

2. 滚动优化

类似于单变量的情况，在多变量 DMC 的滚动优化中，要求每一输出 y_i 在未来 P 个时刻紧密跟踪相应的期望值 w_i，并对 M 个控制增量的大小均加以软约束，这一性能指标可写作

$$\min J(k)=\|\boldsymbol{w}(k)-\tilde{\boldsymbol{y}}_{PM}(k)\|_Q^2+\|\Delta\boldsymbol{u}_M(k)\|_R^2 \qquad (6\text{-}5)$$

其中

$$\boldsymbol{w}(k)=\begin{bmatrix}\boldsymbol{w}_1(k)\\ \vdots\\ \boldsymbol{w}_p(k)\end{bmatrix},\quad \boldsymbol{w}_i(k)=\begin{bmatrix}w_i(k+1)\\ \vdots\\ w_i(k+P)\end{bmatrix},\ i=1,\cdots,p$$

$$\boldsymbol{Q}=\text{block-diag}(\boldsymbol{Q}_1,\ \cdots\boldsymbol{Q}_p),\quad \boldsymbol{Q}_i=\text{diag}[q_i(1)\cdots q_i(P)],\quad i=1,\ \cdots,\ p$$

$$\boldsymbol{R}=\text{block-diag}(\boldsymbol{R}_1,\ \cdots\boldsymbol{R}_m),\quad \boldsymbol{R}_j=\text{diag}[r_j(1)\cdots r_j(M)],\quad j=1,\ \cdots,\ m$$

很明显，误差权矩阵 $\boldsymbol{Q}$ 的分块 $\boldsymbol{Q}_1$，…，$\boldsymbol{Q}_p$ 对应着不同的输出，而 $\boldsymbol{Q}_i$ 中的元素则对应于 y_i 在不同时刻的跟踪误差。同样，控制权矩阵 $\boldsymbol{R}$ 的分块 $\boldsymbol{R}_1$，…，$\boldsymbol{R}_m$ 对应着不同的控制输入，而 $\boldsymbol{R}_j$ 中的元素则对应于对 u_j 在不同时刻增量的抑制。所以，在性能指标（6-5）中，$\boldsymbol{Q}$、$\boldsymbol{R}$ 的每一元素都有直观的物理意义，这对控制系统的整定是十分有利的。

在不考虑约束的情况下，由预测模型（6-4），可求出使性能指标（6-5）最优的全部控制增量

$$\Delta\boldsymbol{u}_M(k)=(\boldsymbol{A}^T\boldsymbol{Q}\boldsymbol{A}+\boldsymbol{R})^{-1}\boldsymbol{A}^T\boldsymbol{Q}[\boldsymbol{w}(k)-\tilde{\boldsymbol{y}}_{P0}(k)] \qquad (6\text{-}6)$$

而即时控制增量可由下式给出

$$\Delta\boldsymbol{u}(k)=\boldsymbol{D}[\boldsymbol{w}(k)-\tilde{\boldsymbol{y}}_{P0}(k)] \qquad (6\text{-}7)$$

其中

$$\boldsymbol{D}=\boldsymbol{L}(\boldsymbol{A}^T\boldsymbol{Q}\boldsymbol{A}+\boldsymbol{R})^{-1}\boldsymbol{A}^T\boldsymbol{Q}\triangleq\begin{bmatrix}\boldsymbol{d}_{11}^T & \cdots & \boldsymbol{d}_{1p}^T\\ \vdots & & \vdots\\ \boldsymbol{d}_{m1}^T & \cdots & \boldsymbol{d}_{mp}^T\end{bmatrix} \qquad (6\text{-}8)$$

而

$$L=\begin{bmatrix}1 & 0 & \cdots & 0 & & & & 0 \\ & & & & \ddots & & & \\ & & 0 & & & 1 & 0 \cdots & 0\end{bmatrix}$$

为 $m\times mM$ 维矩阵。

注意，这里 d_{ji}^T（$i=1$，…，p，$j=1$，…，m）均为 P 维向量。在阶跃响应已知且控制策略已定的情况下，A、Q、R 均属已知，D的元素可由式（6-8）一次离线算出。在线只须计算

$$\Delta u_j(k)=\sum_{i=1}^{p} d_{ji}^T[w_i(k)-\tilde{y}_{i,P0}(k)],\quad j=1,\cdots,m$$

及

$$u_j(k)=u_j(k-1)+\Delta u_j(k),\qquad j=1,\cdots,m \tag{6-9}$$

即可得到 m 个要实施的即时控制量

3．反馈校正

在 k 时刻实施控制后，即可根据预测模型（6-3）算出对象在未来时刻的各输出值，其中也包括了各输出量在 $k+1$ 时刻的预测值 $\tilde{y}_{i1}(k+1|k)$，$i=1$，…，p。到 $k+1$ 时刻测得各实际输出 $y_i(k+1)$ 后，即可与相应的预测值比较并构成误差向量

$$e(k+1)=\begin{bmatrix}e_1(k+1)\\ \vdots \\ e_p(k+1)\end{bmatrix}=\begin{bmatrix}y_1(k+1)-\tilde{y}_{11}(k+1|k)\\ \vdots \\ y_p(k+1)-\tilde{y}_{p1}(k+1|k)\end{bmatrix} \tag{6-10}$$

利用这一误差信息用加权方法预测未来的误差，并以此补偿基于模型的预测，可得到经校正的预测向量

$$\tilde{y}_{cor}(k+1)=\tilde{y}_{N1}(k)+He(k+1) \tag{6-11}$$

其中

$$H=\begin{bmatrix}h_{11} & \cdots & h_{1p}\\ \vdots & & \vdots \\ h_{p1} & \cdots & h_{pp}\end{bmatrix},\qquad h_{st}=\begin{bmatrix}h_{st}(1)\\ \vdots \\ h_{st}(N)\end{bmatrix},\qquad s,t=1,\cdots,p$$

为误差校正矩阵，它是由一系列误差校正向量 $\boldsymbol{h}_{st}$ 构成的。由于对误差形成的原因缺乏了解，不同输出的预测误差对输出预测值的交叉影响程度更属不可知，所以，交叉校正向量 $\boldsymbol{h}_{st}(s \neq t)$ 的选取是无规则可循的。为简化计，通常只保留 $\boldsymbol{H}$ 中的主对角块，即只用 y_i 自身的误差通过加权修正其预测输出值。

由于时间基点已从 k 时刻移到 $k+1$ 时刻，故这一校正后的预测向量 $\tilde{\boldsymbol{y}}_{cor}(k+1)$ 可通过移位构成 $k+1$ 时刻的初始预测值

$$\tilde{\boldsymbol{y}}_{N0}(k+1)=\boldsymbol{S}_0\tilde{\boldsymbol{y}}_{cor}(k+1) \tag{6-12}$$

其中

$$\boldsymbol{S}_0=\begin{bmatrix} \boldsymbol{S} & & 0 \\ & \ddots & \\ 0 & & \boldsymbol{S} \end{bmatrix}, \qquad \boldsymbol{S}=\begin{bmatrix} 0 & 1 & & 0 \\ & \ddots & \ddots & \\ & & 0 & 1 \\ 0 & & & 1 \end{bmatrix}$$

整个控制就是按照这一过程反复进行的。图 6-1 画出了 $m=p=2$ 时的多变量 DMC 算法结构。

由上述推导及图 6-1 可以看出，DMC 算法在推广到多变量系统时，除了对不同时刻控制作用的叠加外，还增加了对不同控制量作用的叠加。除了由于变量数的增加使相应的计算量增大外，在原理上没有任何困难，只是在预测、优化的各式中把单纯时间方向的叠加扩展为时间（不同时刻作用）与空间（不同输入量作用）的双重叠加而已。

与单变量情况相比，在这里只须指出以下几点：

（1）对于有 m 个控制输入、p 个控制输出的多变量对象，必须测出所有输出对全部输入的阶跃响应。为此，可在其他输入量保持不变的条件下，逐个地给 u_j 以一单位阶跃，并测定相应的各 y_i 在 N 个时刻的输出，得到 $\boldsymbol{a}_{ij}$。这样的试验一共要进行 m 次，所得到的阶跃响应一共有 $p \times m$ 个。

（2）对于设定值控制，在到达稳态时，对象的输入输出间有关系：

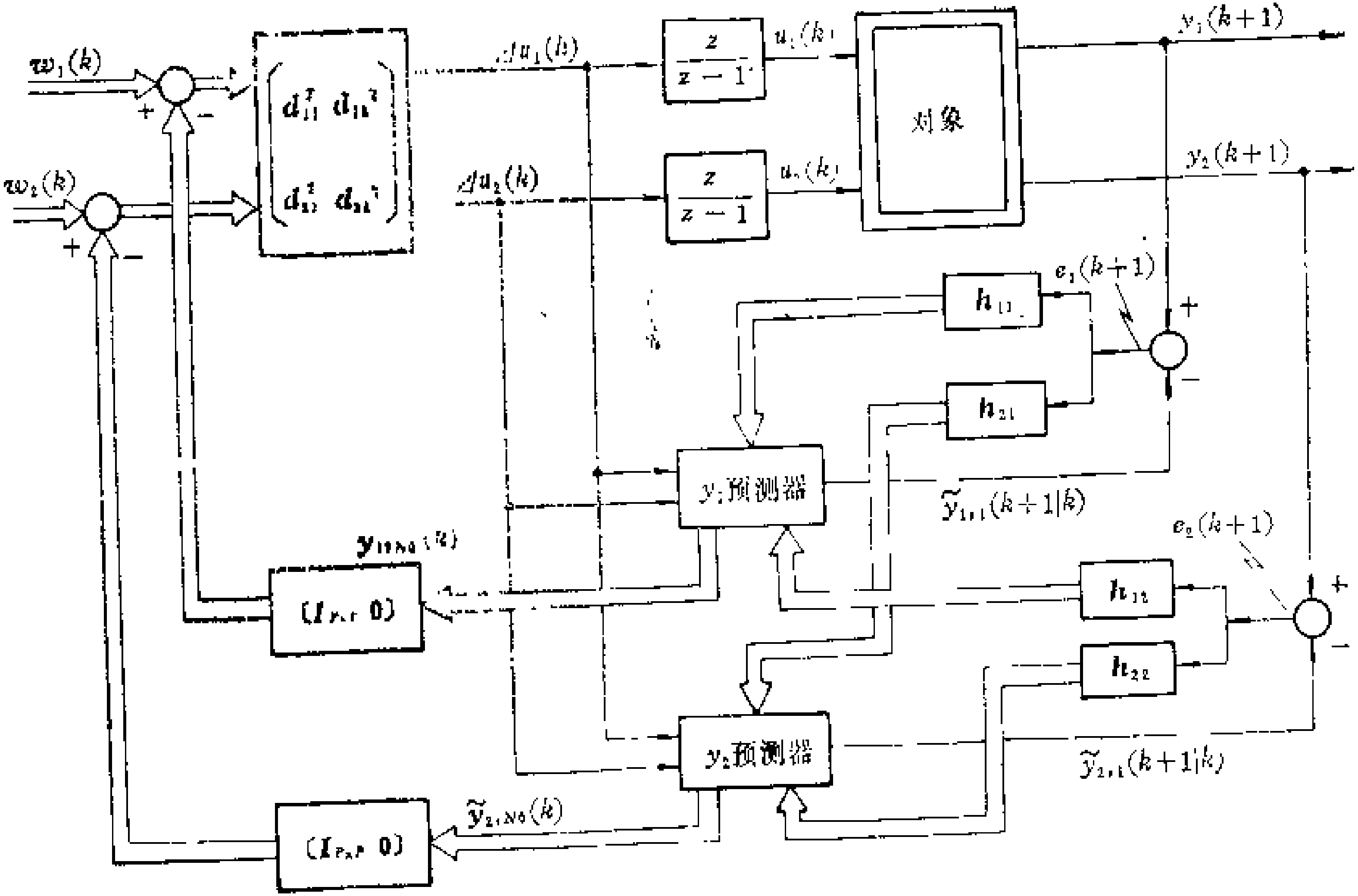


图6-1　多变量动态矩阵控制（$m=p=2$）

$$\begin{bmatrix} y_{1s} \\ \vdots \\ y_{ps} \end{bmatrix} = \begin{bmatrix} a_{11s} & \cdots & a_{1ms} \\ \vdots & & \vdots \\ a_{p1s} & \cdots & a_{pms} \end{bmatrix} \begin{bmatrix} u_{1s} \\ \vdots \\ u_{ms} \end{bmatrix}$$

式中下标 s 均表示相应量的稳态值。上式中的 $p \times m$ 维矩阵的秩如果为 p，我们就有足够的自由度把对象的各输出控制到任意期望值。由于在 $m < p$ 时这一矩阵的秩总要小于 p，故这一对象是输出不完全可控的。这时，不可能用较少的输入使较多的输出均达到任意的期望值。但通过协调优化性能指标（6-5）中的权矩阵 $\boldsymbol{Q}_i$，可人为地指定不同输出值对于各自期望值的贴近程度。

（3）如果对象有相同的输入输出数 $m = p$，则与单变量控制的情况相仿，在不考虑约束且采用优化策略 $P = M$，$\boldsymbol{Q} = \boldsymbol{I}$，$\boldsymbol{R} = \boldsymbol{0}$ 时，导出的即时控制律与一步预测优化（即 $P = M = 1$）时的控制律完全相同。为了说明这一点，考虑式（6-6）并假定 $\boldsymbol{A}$ 是可逆的，在上述条件下，可得最优解满足

$$\boldsymbol{A} \Delta \boldsymbol{u}_M(k) = \boldsymbol{w}(k) - \tilde{\boldsymbol{y}}_{P0}(k)$$

即

$$\begin{aligned}
&a_{11}(1)\Delta u_1(k) + \cdots + a_{1m}(1)\Delta u_m(k) \\
&= w_1(k+1) - \tilde{y}_{1,0}(k+1 \mid k) \\
&\qquad \vdots \\
&a_{21}(1)\Delta u_1(k) + \cdots + a_{2m}(1)\Delta u_m(k) \\
&= w_2(k+1) - \tilde{y}_{2,0}(k+1 \mid k) \\
&\qquad \vdots \\
&a_{m1}(1)\Delta u_1(k) + \cdots + a_{mm}(1)\Delta u_m(k) \\
&= w_m(k+1) - \tilde{y}_{m,0}(k+1 \mid k) \\
&\qquad \vdots
\end{aligned}$$

不管 $\Delta u_i(k+j)$ 是多少，即时控制 $\Delta u_i(k)$ 可完全由这 m 个等式解出，这正对应着 $P = M = 1$ 时的最优解。

（4）这种多变量控制算法与一般的解耦、分散控制算法不同。它采用了全局整体优化的概念，考虑了所有输入输出间的交互影响，因而各控制输入与输出量在算法中具有同等的地位，只

是编号不同而已，因此，对它们的控制不存在配对问题。

§6.2 多变量动态矩阵控制的算法实现及参数整定

由上节介绍可知，多变量DMC算法的在线运算式(6-10)～式（6-12)、式（6-7)、式（6-3）只涉及到三类参量：模型参量$\boldsymbol{a}_{ij}$，控制参量$\boldsymbol{d}_{ji}^T$以及校正参量$\boldsymbol{h}_{ri}$。其中，$\boldsymbol{a}_{ij}$是由阶跃响应特性及采样周期确定的，$\boldsymbol{h}_{ri}$可自由选取，而$\boldsymbol{d}_{ji}^T$即$\boldsymbol{D}$则要根据优化性能指标（6-5）由式（6-8）事先离线算出。由于$\boldsymbol{D}$的计算涉及到许多矩阵，当输入输出数较多且P、M较大时，直接按式（6-8）的形式进行计算势必会占用很多内存并花费很长的计算时间。为此，本节将首先对其计算进行分析，在保持计算有效性的同时尽可能提高其经济性。

为了简化计算，首先把式（6-8）中的矩阵记为

$$\boldsymbol{\Phi}=\boldsymbol{A}^T\boldsymbol{Q}\boldsymbol{A}$$

$$\boldsymbol{\Psi}=\boldsymbol{L}(\boldsymbol{A}^T\boldsymbol{Q}\boldsymbol{A}+\boldsymbol{R})^{-1}=\boldsymbol{L}(\boldsymbol{\Phi}+\boldsymbol{R})^{-1}$$

$$\boldsymbol{D}=\boldsymbol{\Psi}\boldsymbol{A}^T\boldsymbol{Q} \tag{6-13}$$

根据$\boldsymbol{A}$及$\boldsymbol{Q}$的分块结构，可将$\boldsymbol{\Phi}$记为

$$\boldsymbol{\Phi}=\begin{bmatrix}\boldsymbol{\Phi}_{11} & \cdots & \boldsymbol{\Phi}_{1m}\\ \vdots & & \vdots\\ \boldsymbol{\Phi}_{m1} & \cdots & \boldsymbol{\Phi}_{mm}\end{bmatrix}$$

其中，
$$\boldsymbol{\Phi}_{ij}=\sum_{l=1}^{p}\boldsymbol{A}_{li}^T\boldsymbol{Q}_l\boldsymbol{A}_{lj}$$

且可知$\boldsymbol{\Phi}_{ij}=\boldsymbol{\Phi}_{ji}^T$，$\boldsymbol{\Phi}_{ii}$为对称阵。

组成$\boldsymbol{\Phi}_{ij}$的每一个$\boldsymbol{\Phi}_{ij,l}=\boldsymbol{A}_{li}^T\boldsymbol{Q}_l\boldsymbol{A}_{lj}$是一个$M\times M$阵，根据$\boldsymbol{A}_{li}$，$\boldsymbol{Q}_l$，$\boldsymbol{A}_{lj}$的元素，可直接写出该乘积矩阵的第（$s$，$t$）个元素为

$$\varphi_{ij,l}(s,\ t)=\sum_{h=\max(s,\ t)}^{P}a_{li}(h-s+1)q_l(h)\times a_{lj}(h-t+1)$$

由此可知，$M\times M$阵 $\boldsymbol{\Phi}_{ij}$ 的第（s，t）个元素为

$$\varphi_{ij}(s,\ t)=\sum_{l=1}^{p}\sum_{h=\max(s,\ t)}^{P}a_{li}(h-s+1)q_l(h)\times a_{lj}(h-t+1) \tag{6-14}$$

这样，为了得到 $\boldsymbol{\Phi}$，我们不需首先构成 $pP\times mM$ 维矩阵 $\boldsymbol{A}$ 并进行 $\boldsymbol{A}^T\boldsymbol{Q}\boldsymbol{A}$ 运算，而只须由式（6-14）设置 $\boldsymbol{\Phi}_{ij}$ 中的元素。注意到 $\boldsymbol{\Phi}$ 的对称性，故实际上只须设置 $\boldsymbol{\Phi}$ 的下三角部分。由于在式（6-14）中不出现任何矩阵，这里只需用一个 $mM\times mM$ 维矩阵直接储存 $\boldsymbol{\Phi}$ 的元素即可。

算出 $\boldsymbol{\Phi}$ 后，可构成 $\boldsymbol{\Phi}+\boldsymbol{R}$ 并对其求逆，得到 $(\boldsymbol{\Phi}+\boldsymbol{R})^{-1}$，然后取该矩阵的第1，$M+1$，…，$(m-1)M+1$ 行组成矩阵 $\boldsymbol{\Psi}$，并将其分块记作

$$\boldsymbol{\Psi}=\begin{bmatrix}\boldsymbol{\psi}_{11}^T & \cdots & \boldsymbol{\psi}_{1m}^T\\ \vdots & & \vdots\\ \boldsymbol{\psi}_{m1}^T & \cdots & \boldsymbol{\psi}_{mm}^T\end{bmatrix}$$

这里，$\boldsymbol{\psi}_{ij}=[\psi_{ij}(1)\ \cdots\ \psi_{ij}(M)]^T$，$i$，$j=1$，…，$m$

根据式（6-13）及式（6-14），可以得到

$$\boldsymbol{d}_{ji}^T=\sum_{l=1}^{m}\boldsymbol{\psi}_{jl}^T\boldsymbol{A}_{il}^T\boldsymbol{Q}_i,\quad i=1,\ \cdots,\ p,\quad j=1,\ \cdots,\ m$$

其元素亦不必用矩阵乘法来算而可直接计算。P维向量 $\boldsymbol{d}_{ji}^T$ 的第 s 个元素为

$$d_{ji}(s)=\sum_{l=1}^{m}\sum_{h=1}^{\max(s,\ M)}\psi_{jl}(h)a_{il}(s+1-h)q_i(s) \tag{6-15}$$

根据以上分析，可以得到下述离线计算 D 的紧凑算法。

算法6.1　控制矩阵D的离线计算

第1步　按式（6-14）设置所有 $\boldsymbol{\Phi}_{ij}(i\geqslant j)$ 的元素，其中 $\boldsymbol{\Phi}_{ii}$ 只须设置 $s\geqslant t$ 的情况。由此得到一下三角阵，根据对称性设置上三角部分，可得到矩阵 $\boldsymbol{\Phi}$。然后在对角元上逐个加上权矩阵 $\boldsymbol{R}$ 的元素 $r_1(1)$，…，$r_1(M)$，…，$r_m(1)$，…，$r_m(M)$，

构成 $mM\times mM$ 矩阵 $\boldsymbol{\Phi}+\boldsymbol{R}$；

第 2 步　对 $\boldsymbol{\Phi}+\boldsymbol{R}$ 阵求逆，并取其第 1，$M+1$，…，$(m-1)M+1$ 行组成 $M\times mM$ 维矩阵 $\boldsymbol{\Psi}$；

第 3 步　按式（6-15）计算 $\boldsymbol{d}_{ji}^T$ 的元素，组成矩阵 $\boldsymbol{D}$。

在上述计算步骤中，所需内存数为：

$p\times m$ 组 N 维阶跃响应系数 $a_{ij}(l)$；

p 组 P 维权系数 $q_i(l)$；

m 组 M 维权系数 $r_j(l)$；

$mM\times mM$ 维矩阵 $\boldsymbol{\Phi}+\boldsymbol{R}$ 及其求逆所需内存数，$m\times mM$ 维矩阵 $\boldsymbol{\Psi}$ 可利用同一内存；

$m\times p$ 组 P 维控制系数 $d_{ji}(h)$。

这比起用式（6-8）形式计算 $\boldsymbol{D}$ 来，既节省了内存，又使运算以最紧凑的方式进行。

在得到了 $\boldsymbol{a}_{ij}$、$\boldsymbol{d}_{ji}^T$、$\boldsymbol{h}_{ii}$ 这三组参量后，多变量DMC 的在线运算便可按式（6-10）～式（6-12）、式（6-7）及式（6-3）进行。由于所有量都是按输入输出分块的，在运算时均可采用叠加形式，因此在线计算变得十分简单。

算法6.2　控制量 $u(k)$ 的在线计算

初始化：在控制启动时，测量实际输出 y_{i0}，$i=1$，…，p，并设置预测初值为

$$\tilde{y}_i(l)=y_{i0},\quad l=1,\ \cdots,\ N;\quad i=1,\ \cdots,\ p$$

到下一采样时刻即转入下述实时运算。

第 1 步　检测实际输出 y_i，对应于式（6-10），构成误差

$$e_i=y_i-\tilde{y}_i(1),\quad i=1,\ \cdots,\ p$$

并根据式（6-11）进行校正

$$\tilde{y}_i(l)+h_{ii}(l)e_i\rightarrow\tilde{y}_i(l)$$

$$l=1,\ \cdots,\ N;\quad i=1,\ \cdots,\ p$$

第 2 步　对应于式（6-12），移位构成本时刻的预测初值

$$\tilde{y}_i(l)=\tilde{y}_i(l+1)$$

$$l=1,\ \cdots,\ N-1;\quad i=1,\ \cdots,\ p$$

第 3 步　根据式（6-7）计算控制增量

$$\Delta u_j=\sum_{i=1}^{p}\sum_{l=1}^{P} d_{ji}(l)[w_i(l)-\tilde{y}_i(l)],\quad j=1,\cdots,m$$

第 4 步　计算控制量

$$u_j+\Delta u_j\to u_j,\qquad j=1,\cdots,m$$

并实施控制。

第 5 步　根据式（6-3）计算在这些 u_j 作用下的预测值

$$\tilde{y}_i(l)+\sum_{j=1}^{m} a_{ij}(l)\Delta u_j\to\tilde{y}_i(l)$$

$$l=1,\cdots,N,\quad i=1,\cdots,p$$

为下一时刻作准备。

在算法 6.2 中，在线计算所需的内存为：

$p\times m$ 组 N 维阶跃响应系数 $a_{ij}(l)$；

$m\times p$ 组 P 维控制系数 $d_{ji}(l)$；

p 组 N 维校正系数 $h_{ii}(l)$；

p 组 N 维预测输出值 $\tilde{y}_i(l)$；

p 个实测输出值 y_i 及误差值 e_i；

m 个控制增量 Δu_j 及控制量 u_j；

p 组 P 维期望输出值 $w_i(l)$。

多变量 DMC 控制的参数整定要比单变量情况复杂得多，这是因为，在对象的输入输出之间存在着复杂的关联，而且，即使是对最简单的控制策略，也缺乏反映设计参数与控制性能关系的解析结论。所以，参数的整定只能直接立足于性能指标（6-5）的物理意义，并结合仿真结果反复调整。下面，我们简单指出多变量 DMC 参数整定的一些参考规则[16]：

（1）对应于输出 y_i 的优化时域 P_i，必须覆盖它对所有输入 u_j（$j=1,\cdots,m$）的阶跃响应的主要动态部分，即 P_i 必须跨越所有 $a_{ij}(t)$ 的时滞和非最小相位反向部分，并使

$$\sum_{l=1}^{P_i} a_{ij}(l)q_i(l)$$

与 $a_{ij}(t)$ 的稳态值同号，这在物理上意味着 y_i 的优化是针对其主要动态部分进行的，这是得到稳定而有意义的控制的必要条件。

（2）对应于输入 u_j 的控制时域 M_j，必须根据所有 y_i（$i=1,\cdots,p$）对它的阶跃响应的动态复杂性加以选择。如果所有的 $a_{ij}(t)$（$i=1,\cdots,p$）都有比较简单的动态，则 u_j 无须变化多次就可得到平稳的响应，M_j 的值可取小些；反之，则需加大 M_j 以增加控制的自由度。

（3）在上节的算法介绍中，为计算公式的统一和编程的规范，对不同输出的优化时域和不同输入的控制时域采用了统一的 P 和 M。但通过选择权矩阵 $\mathbf{Q}_i$ 和 $\mathbf{R}_j$，可以得到实际相异的 P 和 M_j。例如，若令 y_i 对应的权矩阵 $\mathbf{Q}_i$ 中元素 $q_i(P_i+1)=\cdots=q_i(P)=0$，则可将 y_i 对应的优化时域缩短到 P_i。同样，若令 u_j 对应的权矩阵 $\mathbf{R}_j$ 中元素 $r_j(M_j+1)=\cdots=r_j(M)=0$，则可将 u_j 对应的控制时域减小为 M_j。

（4）在选定了统一的 P 和 M 后，控制系统的整定主要是通过改变权矩阵 $\mathbf{Q}$ 和 $\mathbf{R}$ 中的元素来实现的。除了上述设置方法可对不同的输入或输出获得不同的 M_j 或 P_i 外，还可在仿真过程中根据输出响应及控制量的变化情况调整 $\mathbf{Q}$、$\mathbf{R}$ 的元素，改善控制性能。根据这些元素在性能指标（6-5）中的物理意义，可以知道：

① 如果某一 y_i 的响应过程特别慢，则可相应加大 $\mathbf{Q}_i$ 中的元素，以增强 y_i 跟踪误差的权重，加快 y_i 的动态过程；

② 如果某一 u_j 的变化过于剧烈，则可加大相应 $\mathbf{R}_j$ 中的元素，以加强对 u_j 变化的抑制，使 u_j 的变化趋于平缓。

（5）通过理论分析可知，在模型准确时，校正向量 $\mathbf{h}_{ij}$ 的选取不影响系统的动态特性。所以，在参数整定中，仍可遵循分

离原理，即调整优化性能指标（6-5）中的参数，以在模型无失配时获得稳定而良好的动态响应，而 h_{ii} 的选择则用于获得良好的抗干扰性及模型失配时的鲁棒性。由于我们在诸 h_{ij} 中只保留了主对角块部分 h_{ii}，理论上能证明各 h_{ii} 可以相互独立地选取，并且它们对稳定性的影响类似于单变量的情况，因此，可以根据控制的要求，类似于式（5-1）或（5-2）选择不同的校正策略，用以增强系统的鲁棒性或抗干扰性。

§6.3　多变量动态矩阵控制的解耦设计

上节已经指出，多变量系统由于其输入输出间存在着复杂的关联，DMC 设计参数的整定要比单变量复杂得多。为了简化多变量 DMC 控制的参数设计，在本节中，我们以降低控制的最优性为代价，通过分散化和关联预测，导出一种建立在解耦基础上的多变量 DMC 设计方法[17]。由于解耦后的子系统可以充分利用单变量 DMC 设计的经验和规则，从而可达到简化设计的目的。

设对象的输入数和输出数相等，即 $m=p$。为了克服集中优化时综合设计的困难，首先我们以分散设计取代集中设计，即把整体对象分解为 m 个单输入单输出的子系统，并取代整体的性能指标（6-5），而对每个子系统分别提出各自的优化性能指标

$$\min J_i=\|w_i(k)-\tilde{y}_{i,PM}(k)\|^2_{Q_i}+\|\Delta u_{i,M}(k)\|^2_{R_i}$$

$$i=1,\ \cdots,\ m \qquad (6\text{-}16)$$

同时也把相应的预测模型（6-4）分解为

$$\tilde{y}_{i,PM}(k)=\tilde{y}_{i,P0}(k)+A_{ii}\Delta u_{i,M}(k)$$

$$+\sum_{\substack{j=1\\ j\neq i}}^{m}A_{ij}\Delta u_{j,M}(k),\quad i=1,\cdots,m \qquad (6\text{-}17)$$

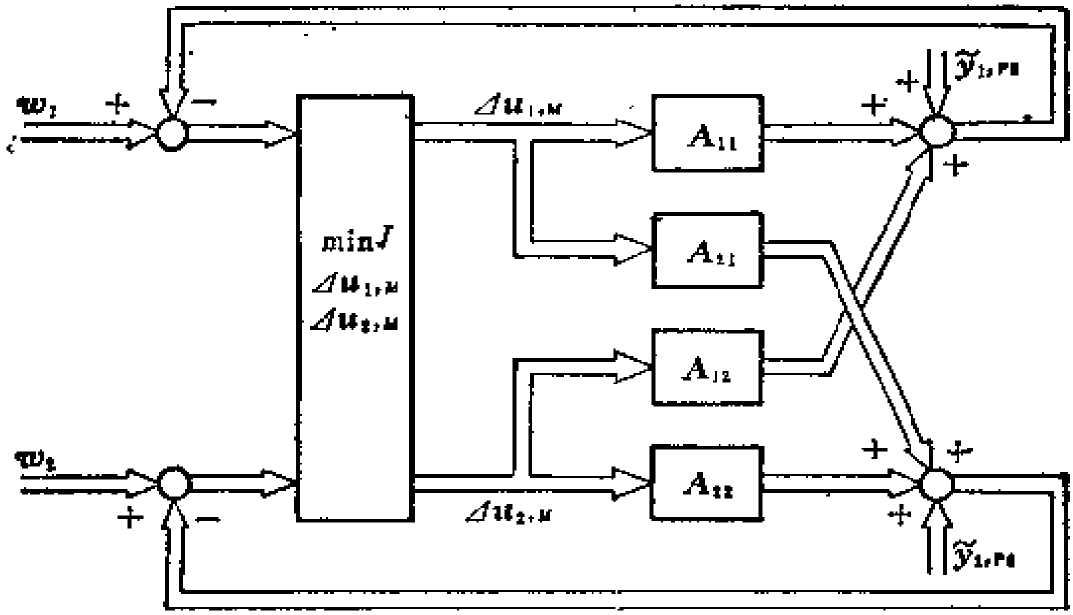
所谓分散设计，就是对每一个子系统，在模型约束（6-17）下，寻找使性能指标（6-16）最优的 $\Delta u_{i,M}(k)$。这可由单变量 DMC 优化问题解出为

$$\Delta \boldsymbol{u}_{i,M}(k)=(\boldsymbol{A}_{ii}^T\boldsymbol{Q}_i\boldsymbol{A}_{ii}+\boldsymbol{R}_i)^{-1}\boldsymbol{A}_{ii}^T\boldsymbol{Q}_i[\boldsymbol{w}_i^*(k)-\tilde{\boldsymbol{y}}_{i,P0}(k)]$$
$$=\boldsymbol{D}_{ii}[\boldsymbol{w}_i^*(k)-\tilde{\boldsymbol{y}}_{i,P0}(k)] \qquad (6\text{-}18)$$

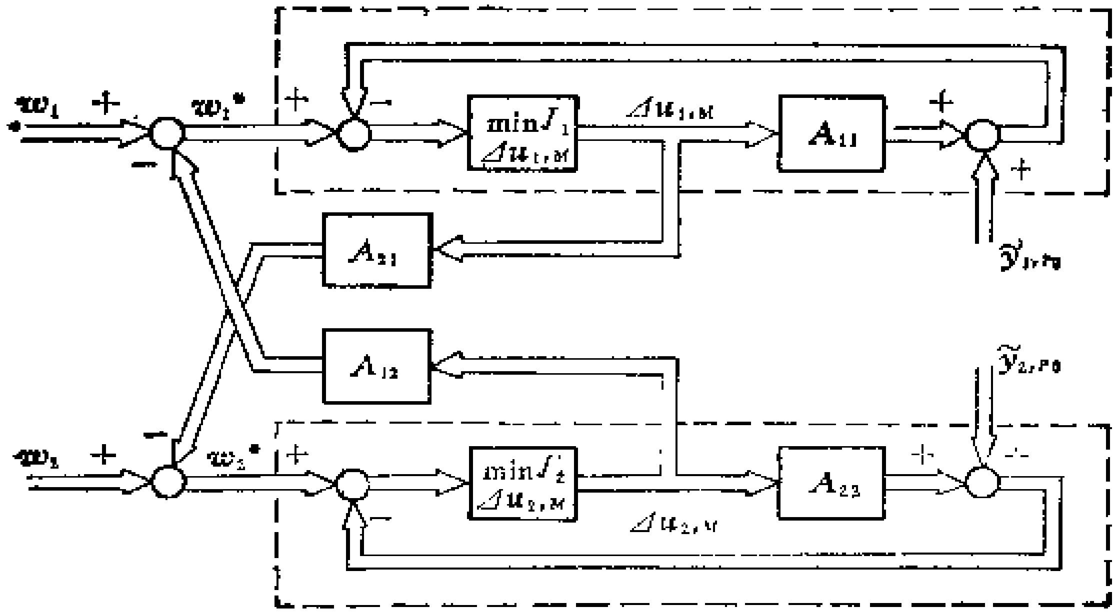
其中

$$\boldsymbol{w}_i^*(k)=\boldsymbol{w}_i(k)-\sum_{\substack{j=1\\ j\neq i}}^{m}\boldsymbol{A}_{ij}\Delta \boldsymbol{u}_{j,M}(k)$$

$\boldsymbol{D}_{ii}$ 为单变量 DMC 所算得的控制矩阵。



(a)



(b)

图6-2　多变量 DMC 的优化设计

(a) 集中设计；(b) 分散设计。

这表明，一旦把其他输入 $u_j(j\neq i)$ 对 y_i 的影响从期望输出 w_i 中扣除，则子系统的设计就具有解耦的形式，因而可按单变量 DMC 的规则对性能指标（6-16）中的设计参数进行整定，使之具有良好的动态响应。只要 $u_j(j\neq i)$ 对于 y_i 的影响是精确扣除的，则参数的整定就只影响 y_i 的动态，而不会对其他 $y_j(j\neq i)$ 产生影响。这种分散设计与前面所讨论的集中设计在原理上的差别可见图 6-2。

显然，要实现优化设计的完全解耦，所存在的明显困难是：为了求出最优的 $\Delta \boldsymbol{u}^*_{i,M}(k)$，必须事先知道最优的 $\Delta \boldsymbol{u}^*_{j,M}(k)$，而为了求出最优的 $\Delta \boldsymbol{u}^*_{j,M}(k)$，也必须首先得到最优的 $\Delta \boldsymbol{u}^*_{i,M}(k)$。这种以已知对方最优解来设计自身最优解的优化称为纳什（Nash）优化，是分散优化中的一种典型情况。为了克服这种互为设计前提的困难，可对所有的 $\Delta \boldsymbol{u}_{i,M}(k)$ 进行预估，在此基础上进行分散设计求出最优的 $\Delta \boldsymbol{u}_{i,M}(k)$，再作为新的较准确的预估值代入重新设计，如此反复进行。这一过程可根据式(6-18)写成迭代形式：

$$\Delta \boldsymbol{u}_M^{l+1}(k)=\boldsymbol{D}_0\Delta \boldsymbol{u}_M^l(k)+\boldsymbol{D}_1[\boldsymbol{w}(k)-\tilde{\boldsymbol{y}}_{P0}(k)]$$

$$l=0,\ 1,\ \cdots$$

其中

$$\boldsymbol{D}_0=\begin{bmatrix} \boldsymbol{0} & -\boldsymbol{D}_{11}\boldsymbol{A}_{12} & \cdots & -\boldsymbol{D}_{11}\boldsymbol{A}_{1m} \\ -\boldsymbol{D}_{22}\boldsymbol{A}_{21} & \boldsymbol{0} & \cdots & -\boldsymbol{D}_{22}\boldsymbol{A}_{2m} \\ \vdots & \vdots & & \vdots \\ -\boldsymbol{D}_{mm}\boldsymbol{A}_{m1} & -\boldsymbol{D}_{mm}\boldsymbol{A}_{m2} & \cdots & \boldsymbol{0} \end{bmatrix} \qquad (6\text{-}19)$$

$$\boldsymbol{D}_1=\begin{bmatrix} \boldsymbol{D}_{11} & & & \boldsymbol{0} \\ & \boldsymbol{D}_{22} & & \\ & & \ddots & \\ \boldsymbol{0} & & & \boldsymbol{D}_{mm} \end{bmatrix}$$

l 为迭代次数。由于在 k 时刻，$\boldsymbol{w}(k)$ 及 $\tilde{\boldsymbol{y}}_{P0}(k)$ 均为已知常

量，故只要

$$S(\boldsymbol{D}_0)<1 \tag{6-20}$$

其中 $S(\boldsymbol{D}_0)$ 表示矩阵 $\boldsymbol{D}_0$ 的谱半径，则这一迭代过程就是收敛的，并且最终可导出

$$\Delta\boldsymbol{u}_M(k)=(\boldsymbol{I}-\boldsymbol{D}_0)^{-1}\boldsymbol{D}_1[\boldsymbol{w}(k)-\tilde{\boldsymbol{y}}_{P0}(k)] \tag{6-21}$$

这一控制律满足完全解耦的条件

$$\Delta\boldsymbol{u}_M(k)=\boldsymbol{D}_0\Delta\boldsymbol{u}_M(k)+\boldsymbol{D}_1[\boldsymbol{w}(k)-\tilde{\boldsymbol{y}}_{P0}(k)]$$

不难验证它与式（6-18）是一致的，并且其中 $\Delta\boldsymbol{u}_M(k)$ 的各分量 $\Delta\boldsymbol{u}_{i,M}(k)$ 都是各子系统在性能指标（6-16）下的最优解。

由式（6-21）进一步可得到即时控制律

$$\Delta\boldsymbol{u}(k)=\tilde{\boldsymbol{D}}[\boldsymbol{w}(k)-\tilde{\boldsymbol{y}}_{P0}(k)] \tag{6-22}$$

其中

$$\tilde{\boldsymbol{D}}=\boldsymbol{L}(\boldsymbol{I}-\boldsymbol{D}_0)^{-1}\boldsymbol{D}_1$$

可离线算出，式中 L 的表达式可参见式（6-8）。

对比式（6-22）与式（6-7）可知，在解耦设计下的在线控制律仍具有集中控制的形式。两者的不同仅在于优化设计分别采用了分散优化和集中优化而已。但由于这里的分散优化是在解除其他输入量耦合的单变量情况下进行的，所以其参数设计要容易得多。

〔例〕 设对象的传递函数矩阵为

$$G(S)=\begin{bmatrix}\dfrac{e^{-2s}}{1+100S} & \dfrac{e^{-6s}}{1+100S}\\[2ex] \dfrac{-1.25e^{-2s}}{1+50S} & \dfrac{3.75e^{-6s}}{1+50S}\end{bmatrix}$$

首先我们采用集中优化的设计方法，选择 $T=20\,s$，$P=5$，$M=3$，$\boldsymbol{Q}=\boldsymbol{I}$，$\boldsymbol{R}=0.8\boldsymbol{I}$，可以得到图 6-3（$a$）所示的动态响应，其中 y_1 的输出响应较慢。为了使其得到加速，调整设计参数 $\mathbf{R}=\mathbf{0}$，这时可得到图 6-3（b）所示的响应曲线。虽然能使 y_1 与 y_2 同步上升，但 y_2 超调十分严重。可见，在集中设计下对不同输出响应的协调并不是很容易的。

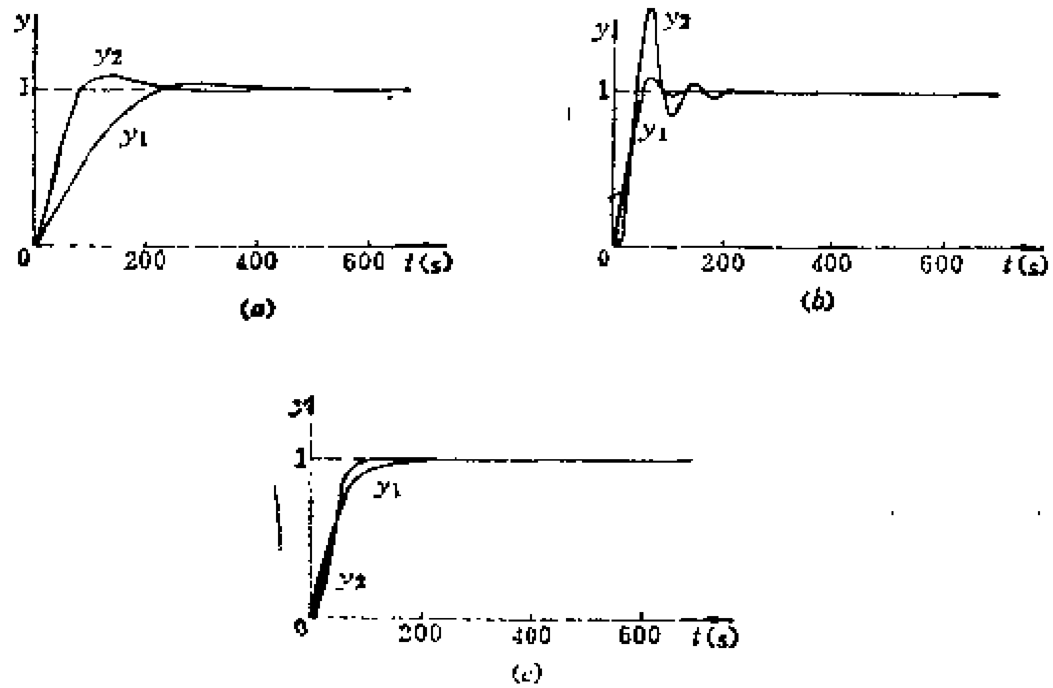


图6-3　不同优化方式下的动态响应

(a) 集中优化$R=0.3I$；(b) 集中优化$R=0$；(c) 分散优化。

如果采用本节提出的分散优化方法设计多变量DMC控制器，取$T=20$ s，$P=2$，$M=1$，$\boldsymbol{Q}=\boldsymbol{I}$，$\boldsymbol{R}=\boldsymbol{0}$，则可得到图6-3(c)所示的输出响应。在这种情况下，由于y_1和y_2的控制是在解耦的情况下用单变量优化方法设计的，故y_1和y_2可以同步地迅速上升，且无超调，从而较好地达到了协调不同输出响应的目的。

最后，我们还要指出本节解耦设计方法的一些注意点：

(1) 虽然以上在分析优化原理及导出最优纳什解的过程中采用了预估值的迭代计算，但这并不意味着一定要通过迭代才能得到最优解。事实上，最优解只须满足精确预估解耦的条件，它可由式(6-21)解析给出，因此，可直接依此离线计算$\widetilde{\boldsymbol{D}}$而无须迭代。尽管如此，我们还是建议计算$\widetilde{\boldsymbol{D}}$所用的$\boldsymbol{D}_0$应满足条件(6-20)。这是因为，式(6-20)可以保证矩阵$\boldsymbol{I}-\boldsymbol{D}_0$可逆，而且当由于模型失配或干扰使实际$\Delta\boldsymbol{u}_M(k)$偏离其最优值$\Delta\boldsymbol{u}_M^*(k)$时，式(6-20)可以保证存在$\Delta\boldsymbol{u}_M^*(k)$的一邻域，只要$\Delta\boldsymbol{u}_M(k)$落在这一邻域内，则扣除了实际交互影响

后由式（6-18）算出的各子问题的最优解，相当于迭代式中把实际$\Delta \boldsymbol{u}_M(k)$作为$\Delta \boldsymbol{u}_M^l(k)$算出的$\Delta \boldsymbol{u}_M^{l+1}(k)$，仍可落在$\Delta \boldsymbol{u}_M^*(k)$的这一邻域内，因而可充分接近作为关联假设的$\Delta \boldsymbol{u}_M^l(k)$。这表明，精确的解耦仍能近似实现，而各子问题的实际解仍可近似视为其最优解。反之，若条件（6-20）不满足，则实际解对于理想最优解的微小偏离，也可能造成各子问题的最优解远远偏离实际解，这时分散设计时的预估是很不准确的，因而控制性能将大为下降。鉴于以上分析，在对各子问题进行分散设计得到$\boldsymbol{D}_{ii}$后，在构成$\boldsymbol{D}_0$时应首先检验条件（6-20）是否满足。若不满足，则应调整相应的分散设计参数，改变$\boldsymbol{D}_{ii}$的值，直至条件（6-20）满足后再计算$\widetilde{\boldsymbol{D}}$。

（2）解耦设计涉及到变量的配对问题。虽然在精确预估的理想情况下，不论输入输出怎样配对，都可把设计问题转化为单变量系统在完全解耦条件下的独立优化设计问题，但实际上，由于不可避免地存在模型失配与干扰，理想的解耦是难以实现的。在控制偏离精确预估最优解的情况下，由于不同的配对方式，将对实际解耦程度及最优性的下降程度产生不同的影响。由图6-2（b）可见，如果A_{12}或A_{21}较大，则Δu_1或Δu_2对于理想解的微小偏离将对两个单回路设计的最优性产生很大的影响。因此，应尽可能选择相互影响最显著的输入输出进行配对，而这种配对正是以条件（6-20）来刻划的。以两输入两输出对象为例，假定在各自的单变量优化中取$M=1$，$\boldsymbol{Q}=\boldsymbol{I}$，$\boldsymbol{R}=0$，优化时域为$P$，则可写出

$$\boldsymbol{A}_{ij}=[a_{ij}(1)\cdots a_{ij}(P)]^T,\quad i,\ j=1,\ 2$$

$$\boldsymbol{D}_{ii}=\frac{1}{\sum\limits_{l=1}^{P}a_{ii}^2(l)}\cdot[a_{ii}(1)\cdots a_{ii}(P)],\quad i=1,\ 2$$

由此可得

$$\boldsymbol{D}_0=\begin{bmatrix} 0 & -\boldsymbol{D}_{11}\boldsymbol{A}_{12} \\ -\boldsymbol{D}_{22}\boldsymbol{A}_{21} & 0 \end{bmatrix}$$

$$=\begin{bmatrix} 0 & -\dfrac{\sum\limits_{l=1}^{P} a_{11}(l)a_{12}(l)}{\sum\limits_{l=1}^{P} a_{11}^2(l)} \\ -\dfrac{\sum\limits_{l=1}^{P} a_{22}(l)a_{21}(l)}{\sum\limits_{l=1}^{P} a_{22}^2(l)} & 0 \end{bmatrix}$$

条件（C-20）此时可写作

$$\left|\frac{\sum\limits_{l=1}^{P} a_{11}(l)a_{12}(l)}{\sum\limits_{l=1}^{P} a_{11}^2(l)}\cdot\frac{\sum\limits_{l=1}^{P} a_{22}(l)a_{21}(l)}{\sum\limits_{l=1}^{P} a_{22}^2(l)}\right|<1$$

这表示变量配对应使系统响应具有主对角动态优势，即在优化时域中，$a_{11}(l)$一般大于$a_{12}(l)$，$a_{22}(l)$一般大于$a_{21}(l)$。这样的配对、可以减小各子优化问题间的交互影响，在控制偏离最优值时仍能使优化近似具有最优的意义。

（3）对于多变量系统，采用不同的优化思想和控制策略，可以得到不同的控制律。这里有三种典型情况：

1）集中优化，集中控制

这就是6.1节中的多变量DMC算法。它采用整体的预测模型（6-4）对性能指标（6-5）进行整体优化，所导出的最优控制律（6-6）

$$\Delta\boldsymbol{u}_M(k)=(\boldsymbol{A}^T\boldsymbol{Q}\boldsymbol{A}+\boldsymbol{R})^{-1}\boldsymbol{A}^T\boldsymbol{Q}[\boldsymbol{w}(k)-\tilde{\boldsymbol{y}}_{P0}(k)]$$

无论在计算时或实现时都是整体的。这是采用集中优化的方法设计控制律，并以集中的方式实现控制。

2）分散优化，集中控制

这是本节中采用的解耦设计算法，它把整体性能指标（6-6）分解为局部性能指标（6-16），并以局部模型（6-17）对其进行优化。虽然在局部模型 i 中也考虑了其他控制作用 $\Delta u_j(j\neq i)$ 的影响，但用来对性能指标 J_i 优化的只是 Δu_i，而不是全部控制作用 $\Delta \boldsymbol{u}$，这是它与 6.1 节中集中优化的最大区别。由此导出的最优控制律（6-21）为

$$\Delta \boldsymbol{u}_M(k)=(\boldsymbol{I}-\boldsymbol{D}_0)^{-1}\boldsymbol{D}_1[\boldsymbol{w}(k)-\tilde{\boldsymbol{y}}_{P0}(k)]$$

其中，$\boldsymbol{D}_1$ 是分散优化的结果，它具有块对角的形式，而 $(\boldsymbol{I}-\boldsymbol{D}_0)^{-1}$ 项则反映了模型的关联。这使上述控制律的实现仍是集中形式的。

3）分散优化，分散控制

这是指对分解的性能指标（6-16）完全不考虑子系统间的相互关联，而用分散的预测模型

$$\tilde{\boldsymbol{y}}_{i,PM}(k)=\tilde{\boldsymbol{y}}_{i,P0}(k)+\boldsymbol{A}_{ii}\Delta \boldsymbol{u}_{i,M}(k)$$

对其实现优化。这就是通常所说的分散控制。由此导出的控制律为

$$\Delta \boldsymbol{u}_M(k)=\boldsymbol{D}_1[\boldsymbol{w}(k)-\tilde{\boldsymbol{y}}_{P0}(k)]$$

由于 $\boldsymbol{D}_1$ 是分块对角的，故这一控制律是以全分散的形式实现的。这种控制策略完全忽略了子系统间的相互关联，其控制性能与集中控制相比有较大程度的下降。为此，需采取附加的预测手段进行补偿，我们在 6.5 节中将进一步详细讨论这种分散预测控制算法。

对比上述三种情况可知，本节的解耦设计算法是全集中与全分散两种情况的折衷。它放弃了优化时的整体性考虑，但通过保持模型关联仍具有控制的整体性。它的控制律中系数矩阵既不是全关联的，又不是全分散的，而是由全关联与全分散两部分组合而成的，它们分别对应着控制的整体性及优化的分散性。所以，这种解耦设计的多变量 DMC 算法在设计上具有分散控制的简易性，而在性能指标上，虽不及全集中设计的多变量 DMC 算法，但仍远优于分散控制的算法。

§6.4 有约束的多变量动态矩阵控制

以上讨论的DMC算法，无论是单变量还是多变量的情况，都没有考虑系统中存在的约束。然而在实际工业过程中，系统中的物理量是不可能无限取值的，例如，当执行元件为阀门时，阀门开度只可能在一定范围内变化；在对锅炉进行液位控制时，液位高度超过一定的上限或下限都会引起事故。因此，在实现控制时，必须根据实际要求，把控制量和输出量约束在一定范围内，即

$$u_{\min} \leqslant u \leqslant u_{\max} \tag{6-23}$$

$$y_{\min} \leqslant y \leqslant y_{\max} \tag{6-24}$$

在这种情况下，如果还是按无约束优化求出最优控制 $\boldsymbol{u}(k)$，并根据式（6-23）把其超界的分量用临界值代替，则就失去了优化的意义。更为严重的是，这样做至多只能解决控制量 u 的约束问题，而在其作用下的输出 y 是否满足约束条件（6-24）仍是无法保证的。为了得到满足约束条件的真正可行的优化解，必须把约束条件（6-23）和（6-24）作为滚动优化的组成部分加以考虑。本节中，我们将讨论这类有约束多变量DMC控制的策略。

在多变量预测控制中，每一时刻的优化涉及到各输入量在未来M个时刻的增量以及各输出量在未来P个时刻的预测值。这些输入量均应满足约束条件（6-23），即

$$u_{i,\min} \leqslant u_i(k) = u_i(k-1) + \Delta u_i(k) \leqslant u_{i,\max}$$

$$\vdots$$

$$\begin{aligned} u_{i,\min} \leqslant u_i(k+M-1) & \\ = u_i(k-1) &+ \Delta u_i(k) + \cdots \\ &+ \Delta u_i(k+M-1) \\ \leqslant u_{i,\max} & \end{aligned}$$

$$i = 1, \cdots, m$$

它可用向量形式记为

$$\Delta \boldsymbol{u}_{\min} \leqslant \boldsymbol{B} \Delta \boldsymbol{u}_M(k) \leqslant \Delta \boldsymbol{u}_{\max} \tag{6-25}$$

其中 $\boldsymbol{B} = \text{block-diag}\ (\boldsymbol{B}_0 \cdots \boldsymbol{B}_0)$ （m 块）

$$B_0=\begin{bmatrix}1 & & & \\ 1 & 1 & 0 & \\ \vdots & & \ddots & \\ 1 & \cdots & & 1\end{bmatrix}_{(M\times M)}$$

$$\Delta\boldsymbol{u}_{\min}=\begin{bmatrix}u_{1,\min}-u_1(k-1)\\ \vdots\\ u_{1,\min}-u_1(k-1)\\ \vdots\\ u_{m,\min}-u_m(k-1)\\ \vdots\\ u_{m,\min}-u_m(k-1)\end{bmatrix}_{(mM\times 1)}$$

$$\Delta\boldsymbol{u}_{\max}=\begin{bmatrix}u_{1,\max}-u_1(k-1)\\ \vdots\\ u_{1,\max}-u_1(k-1)\\ \vdots\\ u_{m,\max}-u_m(k-1)\\ \vdots\\ u_{m,\max}-u_m(k-1)\end{bmatrix}_{(mM\times 1)}$$

同样，各输出预测值 $\tilde{y}$ 亦均应满足约束条件（6-24），这可直接引用式（6-4）写成向量形式

$$\boldsymbol{y}_{\min}-\tilde{\boldsymbol{y}}_{P0}(k)\leqslant \boldsymbol{A}\Delta\boldsymbol{u}_M(k)\leqslant\boldsymbol{y}_{\max}-\tilde{\boldsymbol{y}}_{P0}(k)\qquad(6\text{-}26)$$

其中

$$\boldsymbol{y}_{\min}=[y_{1,\min}\cdots y_{1,\min},\cdots,\ y_{p,\min}\cdots y_{p,\min}]^T_{(pP\times 1)}$$

$$\boldsymbol{y}_{\max}=[y_{1,\max}\cdots y_{1,\max},\ \cdots,\ y_{p,\max}\cdots y_{p,\max}]^T_{(pP\times 1)}$$

由式（6-25）和式（6-26）可知，对于输入或输出的约束均可归结为如下形式的对于输入量的不等式约束

$$\boldsymbol{C}\Delta\boldsymbol{u}_M(k)\leqslant\boldsymbol{l}\qquad(6\text{-}27)$$

其中，$\boldsymbol{C}$，$\boldsymbol{l}$ 均为 k 时刻已知的量。这样，在 k 时刻考虑约束的滚动优化问题，就是利用预测模型（6-4），在不等式约束(6-27)

下，求出使性能指标（6-5）最优的 $\Delta\boldsymbol{u}_M(k)$，即

$$\min_{\Delta u_M(k)} J(k)=\|\boldsymbol{w}(k)-\tilde{\boldsymbol{y}}_{PM}(k)\|_Q^2+\|\Delta\boldsymbol{u}_M(k)\|_R^2 \tag{6-28}$$

$$\text{s.t.}\ \tilde{\boldsymbol{y}}_{PM}(k)=\tilde{\boldsymbol{y}}_{P0}(k)+\boldsymbol{A}\Delta\boldsymbol{u}_M(k)$$

$$\boldsymbol{C}\Delta\boldsymbol{u}_M(k)\leqslant\boldsymbol{l}$$

这类具有二次型性能指标且带有线性等式和不等式约束的优化问题通常称为二次规划问题。

很明显，由于不等式约束的存在，上述优化问题不可能像无约束时那样导出解析解，因此，在线滚动优化就不再是简单地利用公式（6-6）计算 $\Delta\boldsymbol{u}_M(k)$，而要运用有效的非线性规划方法数值地解出 $\Delta\boldsymbol{u}_M(k)$。对于这类二次规划问题，现有的优化理论已提供了有效的求解算法，典型的有罚函数法（SUMT 法）。这是通过把不等式约束乘以一惩罚因子加入到性能指标中，这一惩罚因子仅在不等式约束被破坏时才起作用。整个算法采用了迭代逼近方法，即从解和惩罚因子的某一初值出发，求解无约束优化问题，然后从新的解出发，改变惩罚因子，继续优化以逐步逼近其最优解。这种算法需要在线反复计算，初值的选择和函数的形式对算法的收敛性有很大影响。在多变量系统的输入输出数较多时，由于附加的不等式约束数目很多，会导致很大的在线计算量。因此，只有当过程优化采用长周期时，例如把 DMC 作为高层优化控制算法，隔很长时间设置一次下级的最优设定点时，才有可能实现这样的大规模优化算法。

事实上，在正常控制中，上述不等式约束（6-27）中有很大一部分是不起作用的，真正出现超界而必须考虑的不等式只是其中极少数。把式（6-27）的所有条件一并在优化中加以考虑虽则全面，实际上是过于冗余了，因而造成了极大的计算量。而且，对于预测控制来说，由于其优化不是一次性而是反复在线进行的，故在 k 时刻的优化问题中，实际上只须考虑较短时段内的输入输出是否满足约束条件即可，完全没有必要把未来很远时间的输入

输出约束考虑在内。基于这样的想法，我们在下面介绍两种更为实用的多变量 DMC 控制处理约束的策略。与上述正规算法相比，它们显得更为简易。

1. 基于矩阵分解法的有约束优化算法[18]

如果不考虑不等式约束（6-27），则按无约束优化方法可求出问题（6-28）的最优解 $\Delta \boldsymbol{u}_M(k)$。在用式（6-27）加以检验时，可能会发现其中有些约束不能满足。根据优化性能指标（6-28）中权矩阵 $\boldsymbol{Q}$、$\boldsymbol{R}$ 的物理意义，可以改变相应的权系数，以改善输入或输出项对于约束条件的满足程度。但这种改变软约束的方法不可能一次成功而需在线反复凑试。由于每一次都涉及到矩阵 $\boldsymbol{A}^T\boldsymbol{Q}\boldsymbol{A}+\boldsymbol{R}$ 的求逆，故对于多输入多输出系统来讲，在线的计算量往往是不能接受的。因此，这种通过在线调整优化性能指标中的权系数来满足约束条件的做法是不可取的。

普雷特（Prett）在文献〔18〕中提出了一种多变量有约束优化算法，克服了上述困难。他指出，在有约束的优化问题中，优化变量 $\Delta \boldsymbol{u}_M$ 在其空间构成一可行域，其边界对应于约束条件的临界值。在不考虑约束时，若优化结果落在此可行域内，则全部约束条件都能满足，这组解是可实现的。但若优化结果落在此可行域外，则这组解是不可实现的，因为它至少破坏了某些约束条件。在这种情况下，最优解必须向可行域靠拢，直至落在可行域的一个顶点上，这时它对本来被破坏的约束条件达到了临界满足，因而是可实现的，但其最优性显然要比无约束时差，这是约束加入而造成的必然结果。

现在我们讨论如何根据上述分析导出较为简易的有约束优化算法。首先，考虑无约束条件（6-27）时的最优解 $\Delta \boldsymbol{u}_M(k)$，它已由式（6-6）给出为

$$\Delta \boldsymbol{u}_M(k)=(\boldsymbol{A}^T\boldsymbol{Q}\boldsymbol{A}+\boldsymbol{R})^{-1}\boldsymbol{A}^T\boldsymbol{Q}[\boldsymbol{w}(k)-\tilde{\boldsymbol{y}}_{P0}(k)]$$

然后，我们用约束条件（6-27）来检验其是否破坏对于输入与输出的约束。注意到优化是反复在线进行的，故只须检验有限的输入和输出即可，而不必历遍式（6-27）中的所有条件。通过检验

可把不满足约束的式子抽取出来，并强令其满足临界约束，这样就可构成下面的等式约束

$$C_1 \Delta u_M(k) = l_1 \tag{6-29}$$

其中，C_1，l_1 分别由 C，l 中对应于不满足约束的那些行的元素组成。很显然，C_1 或 l_1 的行数等于无约束初始解所破坏的约束条件的数目，它要远远小于式（6-27）中矩阵 C 的行数。

为了使解进入可行域，需将等式约束（6-29）作为一个强约束考虑在优化问题中。为此，修改优化性能指标（6-28）为

$$\min J'(k) = J(k) + \|C_1 \Delta u_M(k) - l_1\|_S^2$$

其中，$S = sI$，s 为充分大的正数，以强调对临界约束的满足。

结合预测模型（6-4）对上述性能指标优化，可以得到最优解

$$\begin{aligned}\Delta u_M'(k) = &(A^T QA + R + C_1^T SC_1)^{-1}[A^T Q(w(k)\\ &-\tilde{y}_{P0}(k)) + C_1^T Sl_1)]\end{aligned}$$

记 $P = (A^T QA + R)^{-1}$，利用矩阵分解技术，上式中的逆阵可化为

$$(P^{-1} + C_1^T SC_1)^{-1} = P - PC_1^T (C_1 PC_1^T + S^{-1})^{-1} C_1 P \tag{6-30}$$

从而可把上述解写作

$$\begin{aligned}\Delta u_M'(k) = &PA^T Q[w(k) - \tilde{y}_{P0}(k)] - PC_1^T (C_1 PC_1^T\\ &+ S^{-1})^{-1} C_1 PA^T Q[w(k) - \tilde{y}_{P0}(k)] + PC_1^T Sl_1\\ &- PC_1^T (C_1 PC_1^T + S^{-1})^{-1} C_1 PC_1^T Sl_1\\ = &\Delta u_M(k) - PC_1^T (C_1 PC_1^T + S^{-1})^{-1} C_1 \Delta u_M(k)\\ &+ PC_1^T (C_1 PC_1^T + S^{-1})^{-1} l_1\\ = &\Delta u_M(k) - PC_1^T (C_1 PC_1^T\\ &+ S^{-1})^{-1} [C_1 \Delta u_M(k) - l_1]\end{aligned} \tag{6-31}$$

由此可见，考虑了临界约束（6-29）后，最优解可通过无约束最优解 $\Delta u_M(k)$ 减去一附加项加以改进。这一附加项反映了无约束最优解对于约束条件的破坏程度 $C_1 \Delta u_M(k) - l_1$。值得注意的是，在附加项中出现的 C_1，l_1 是根据对约束条件的检验在线确定的，因而矩阵 $C_1 PC_1^T + S^{-1}$ 须在线求逆。但该求逆矩

阵的维数即 C_l 的行数仅是约束条件不满足的个数，一般说来它远远低于原求逆矩阵 A^TQA+R 的维数。此外，由于 P 可事先离线算出，$\Delta u_M(k)$可按无约束时的解析解算出，而 $C_l\Delta u_M(k)-l_l$ 在检验约束条件是否满足时已算出，所以，由式（6-31)计算改进的控制作用 $\Delta u'_M(k)$ 是很简易的。

在上式中，如果令 $s\to\infty$，则可得到解

$$\Delta u'_M(k)=\Delta u_M(k)-PC_l^T(C_lPC_l^T)^{-1}[C_l\Delta u_M(k)-l_l]$$

这组解一定满足约束边界条件（6-29)，这是因为，

$$C_l\Delta u'_M(k)=C_l\Delta u_M(k)-C_lPC_l^T(C_lPC_l^T)^{-1}\cdot[C_l\Delta u_M(k)-l_l]=l_l$$

应该注意，无论 s 取有限值或无穷大，所得到的 $\Delta u'_M(k)$ 有可能对某些约束条件仍不满足。在 s 为有限值时，对应于式（6-27）中的某些约束仍可能超限，而当 $s\to\infty$时虽保证了式（6-27）的临界约束，但另一些原已满足的约束条件有可能重新被破坏。因此，这一过程往往要迭代进行。在迭代时，仍可用式（6-31)，其中 $\Delta u_M(k)$ 可用上一次的值 $\Delta u'_M(k)$ 代替，P 用式（6-30）算得的结果代替，C_l 及 l_l 则根据仍未满足约束的那些项重新设置。

这种建立在矩阵分解技术基础上的有约束优化算法，其显著优点是利用了在优化时大部分约束条件都能满足这一事实，把在线求逆的矩阵维数降到了最低，从而避免了反复大量计算 A^TQA+R 逆阵，使在线考虑实时约束成为可能。它还可推广应用到限幅约束以外的情况。例如，在多变量控制中，当某一计算机输出通道发生故障而使相应控制转向手动保持恒值时，计算机不能按原来情况计算所有的即时控制增量。为了保证控制整体的最优性，必须把相应的输入增量等于 0 作为附加的等式约束，来求其他输入控制量的最优值。这时，可利用式（6-31）得到在此约束下的最优解，它保证了在故障存在时控制的最优性，而当故障恢复后，也能把控制光滑地继续下去。

2. 基于启发式约束预测的优化算法[18]

上面介绍的有约束优化算法，强调了控制的最优性，它适用于控制要求较高的情况。特别当 DMC 作为分级控制中优化层的算法，允许采用较长的控制周期，并由较高性能的计算机实现时，它是十分有效的。但当 DMC 作为直接控制级算法并且采用一般微型计算机实现时，由于上述算法的在线迭代涉及到矩阵求逆，计算量仍然较大。这时，为了实时控制的需要，就不得不降低对最优性的要求，而更强调计算的简易性。下面介绍的启发式约束预测优化算法就是基于这一考虑提出的[18]。

由于预测控制的优化是在线反复进行的，在 k 时刻实际实现的控制增量只是 $\Delta\boldsymbol{u}(k)$，而不是全部 $\Delta\boldsymbol{u}_M(k)$，所以，在 k 时刻只须对 $\Delta\boldsymbol{u}(k)$ 是否会引起约束条件被破坏进行检验，而对 $\Delta\boldsymbol{u}(k+1)$，$\Delta\boldsymbol{u}(k+2)$，…的影响可留待到 $k+1$、$k+2$ 等时刻再加以考虑。利用预测模型(6-3)，可导出在这组 $\Delta\boldsymbol{u}(k)$ 作用下对象在下一时刻的输出预测值

$$
\begin{aligned}
&\tilde{y}_{i,0}(k+1|k)+\sum_{j=1}^{m}a_{ij}(1)\Delta u_j(k)\\
&=y_{i,1}(k+1|k),\quad i=1,\cdots,p
\end{aligned}
\tag{6-32}
$$

如果某些 $y_{i,1}(k+1|k)$ 破坏了约束条件，则可强制它们保持在约束边界，而对其他满足约束条件的预测输出，应使其保持为原值。这意味着我们要修正 $\Delta u_j(k)$ 为 $\Delta u_j'(k)=\Delta u_j(k)+\varepsilon_j$，使得

$$
\left.
\begin{aligned}
&\tilde{y}_{i,0}(k+1|k)+\sum_{j=1}^{m}a_{ij}(1)\Delta u_j'(k)\\
&=y_{i,\max}\text{或}y_{i,\min}，\quad\text{对于约束破坏的 } i\\
&\tilde{y}_{i,0}(k+1|k)+\sum_{j=1}^{m}a_{ij}(1)\Delta u_j'(k)\\
&=\tilde{y}_{i,1}(k+1|k)，\quad\text{对于约束满足的 } i
\end{aligned}
\right\}
\tag{6-33}
$$

由式（6-32）和式（6-33）可得

$$\sum_{j=1}^{m} a_{ij}(1)\varepsilon_j = y_{i,\max} - y_{i,1}(k+1|k)$$

$$\text{或} y_{i,\min} - y_{i,1}(k+1|k)\text{，对于约束破坏的 } i$$

$$\sum_{j=1}^{m} a_{ij}(1)\varepsilon_j = 0\text{，对于约束满足的 } i$$

如果对象输入数 m 大于等于输出数 p，则不难由上式求出一组 ε_j，经它修正后的 $\Delta u_j'(k)$ 必能保证下一时刻的预测输出满足约束条件。在此基础上，对于输入量的约束可进一步采用直接限幅方法加以考虑。

可以看出，上述启发式修正算法是通过修正优化变量 $\Delta \boldsymbol{u}_M(k)$ 中的局部变量 $\Delta \boldsymbol{u}(k)$ 实现的，因而所得到的解不再有全局最优的意义。它不过是有约束优化问题的一个可行解而不是最优解。但是，由于求解过程尽可能保持了输出与无约束最优解的一致性，仅对破坏约束条件的那些输出作了修正，因而所带来的影响只是局部的。由于预测控制不是采用一成不变的全局优化性能指标，而是在每一时刻提出一个有限的局部优化性能指标，所以这里对控制量的修正恰如在 k 时刻出现了干扰或模型失配要作修正一样，只影响本时刻性能指标的最优性，并不对控制全过程的最优性产生决定性影响，因而它与滚动优化的思想是相容的。这种算法以牺牲局部最优性为代价，但通过简单的计算即可实现可行的控制，因而在实用中仍有一定的参考价值。

以上对约束的预测称为一步约束预测，即在 k 时刻仅对下一步输出是否破坏约束条件进行检验。由于 $\Delta \boldsymbol{u}(k)$ 不仅对 $k+1$ 时刻的输出产生作用，而且也影响到以后各时刻的输出，故采用这种一步预测方法有时仍不能防止约束条件被破坏。系统输出在这组 $\Delta \boldsymbol{u}(k)$ 的作用下虽然在下一时刻满足约束条件，但在 $k+1$ 时刻后仍可能超越约束边界，即使在 $k+1$ 时刻重复进行约束预测并修正控制作用，也来不及防止输出的超界。为了克服这一

困难，可将一步约束预测改为多步约束预测，更为保守地修正控制作用。即在 k 时刻利用无约束基准解 $\Delta u(k)$ 预测 $k+1$，$k+2$，…时刻的输出，并检验约束条件是否被破坏。对于约束条件未破坏的输出，希望修正后的控制作用仍将其在 $k+1$ 时刻的值维持在原值，而对于约束条件被破坏的输出，则找出其中约束条件破坏最严重的时刻，并强制其在该时刻的值保持在约束边界。由此仍可解出一组修正量 c_j，使在其作用下未来若干时刻的输出值均满足约束条件。这种基于多步约束预测的算法可以更有效地防止输出约束条件被破坏。这里所取的预测步数是启发式的，步数越多，对约束越易保证，但控制偏于保守。这是因为，它只考虑了 $\Delta u(k)$ 对以后各时刻输出的影响，而没有顾及到以后各时刻还要有新的控制作用 $\Delta u(k+1)$，$\Delta u(k+2)$，…，它们通过预测修正后，会对以后各时刻的输出保持在约束范围内产生积极的影响。例如，由 $\Delta u(k)$ 的作用预测出 $\tilde{y}_i(k+3|k)$ 将超越约束边界，并不意味着在 $\Delta u(k)$、$\Delta u(k+1)$、$\Delta u(k+2)$ 共同作用后 $\tilde{y}_i(k+3|k)$ 也会超越约束边界。因此，在实行多步约束预测时，过多的步数是没有意义且过于保守的。经验表明，对于平稳变化的过程，只须预测 2～3 步即可满足实际要求。

对于有时滞的对象，由于控制作用要经过一段时滞后才反映到对象输出上，所以对输出约束的一步或多步预测都是指跨越了最大时滞后的一步或多步预测。

上述启发式方法充分利用了预测控制对于未来的可预见性，以预测模型的输出预报约束条件是否满足，并由此获得启发逆推控制作用，从而抑制了输出超越约束界域。这是预测控制与 PID 算法相比明显的优点。在过程中间状态存在某些约束时，只要建立了对象输入与这些中间状态的预测模型，即可依据同样的原理实现对中间状态的有效约束。在这种情况下，这些中间状态可增广为对象的输出，但对它们并不要求最优的跟踪，而只须保持在约束范围内。下面，我们通过一个例子说明如何实现对中间状态

的约束预测。

〔例〕 图6-4所示的液位控制系统，是要通过施加在气动阀门上的压力 u 把第4管的液位高度H_4控制到期望值。这是一个典型的四阶系统，u 的作用是调节阀门开度，改变进水量，通过液位H_1、H_2、H_3，最终影响被控液位H_4的高度。

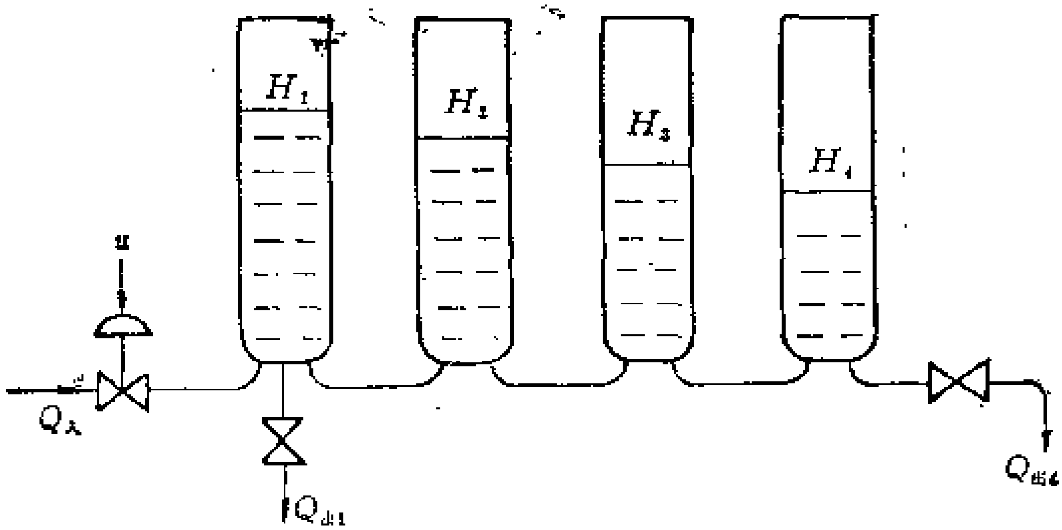


图6-4 液位控制系统

在这里，根据物理条件的限制，可给出对控制量 u，中间状态H_1、H_2、H_3及被控量H_4的约束条件：

$$0 \leqslant u \leqslant 1\text{kg/m}^2$$

$$0 \leqslant H_i \leqslant 1\text{m}, \quad i = 1, \ 2, \ 3, \ 4$$

我们把这个问题看作是单输入多输出的问题，其中H_4是被控量。由于在一般情况下，H_1的液位总要高于其他三管的液位，所以可只考虑对H_1的约束问题。图6-5中，曲线H_{11}和H_{41}对应于不考虑约束时液位H_1和H_4的控制结果，H_{41}的响应看起来很好，但相应的H_{11}超出约束上界达0.59m，实际上早已溢出，所以这是在物理限制下不可能实现的控制。采用这里的启发式方法对输出H_1进行一步约束预测后，可得到曲线H_{12}和H_{42}，如果再考虑对输入的约束，则可得到曲线H_{13}和H_{43}，从图中可以看到，H_{12}和H_{13}被有效地抑制在约束范围内，而曲线H_{43}则是在输入输出均有约束的情况下所能得到的实际响应。

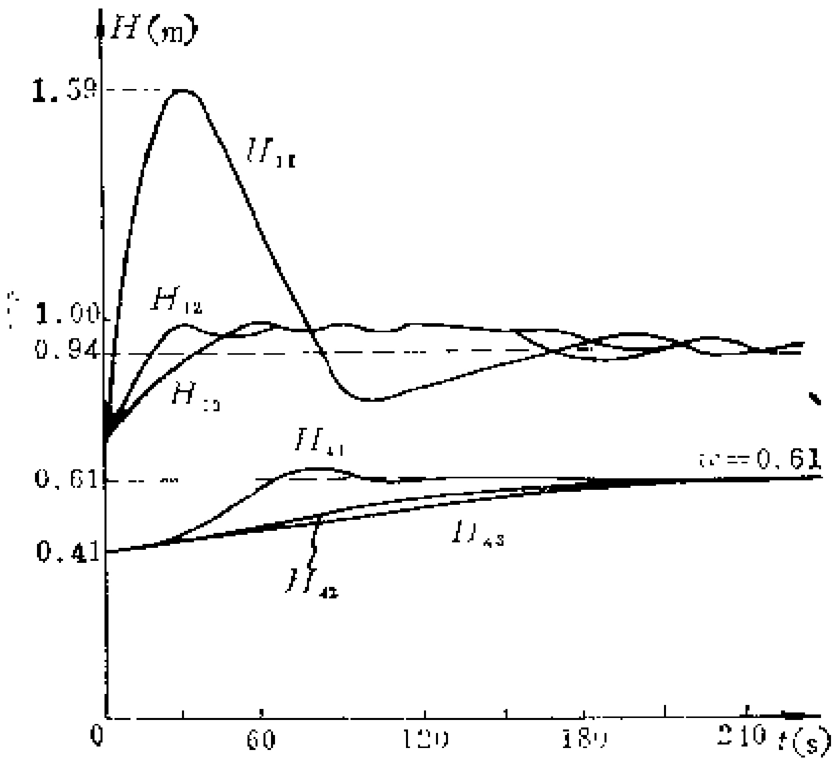


图6-5 液位 H_1和H_4的响应

§6.5 多变量系统的分散预测控制[20]

6.3节所讨论的多变量系统预测控制的解耦设计，着眼点在于以分散优化取代集中优化，以克服多变量系统整体优化的困难。它涉及的是优化控制律的离线设计问题，而不是在线控制问题。由控制律（6-22）可知，每一$\Delta u_i(k)$的计算需要所有y_j的预测值$\tilde{y}_{j,P_0}(k)$。因此，这一控制律仍是集中型的，与通常的多变量预测控制律（6-7）在线实现时并无差别。当系统的输入输出数较多时，这种集中型的在线预测和控制律计算一般需要采用内存大、速度快的高性能计算机。

由于实际工业系统的控制必须考虑经济性，在很多场合下，人们往往宁可采用多台廉价的微机取代高性能计算机对复杂工业对象进行控制。即使在集散型控制装置中，由于上位机承担了过程监控的许多任务，如故障诊断、图形显示等，控制的任务也往往由基础级控制器来承担。在这种情况下，要由分散的多台基础

控制器来实现集中的控制律，必须在它们之间进行大量的数据通信，这不但影响了控制的实时性，而且由于每一控制器的计算都涉及到其他子系统的输出，一旦某一控制器失效，就会对控制的全局性能产生影响，系统的可靠性也将下降。因此，针对这种由多台低性能基础控制器实现的硬件环境，考虑与之相适应的多变量系统预测控制算法，将是很有实际意义的。本节中，我们把分散控制的结构与预测控制的原理结合起来，介绍一种既易于分散实现、又有较好控制性能的多变量系统分散预测控制算法。

所谓分散控制，是在处理高维系统的大系统理论中发展起来的一种有效控制结构，其基本特点是利用分散的信息实现分散的控制，这里不但要求控制是分散实现的，而且还强调控制所依据的信息是分散的局部信息而不是系统的全局信息。为了说明这一概念，我们考虑一有较多输入输出数的大系统，它可按物理组成的特点划分为N个子系统，所有的输入量和输出量都可分别划归到相应的子系统中。设第 i 个子系统的输入量和输出量分别为$\boldsymbol{u}_i$和$\boldsymbol{y}_i$，它们均可能是多维的。现采用N个分散的控制器分别产生各子系统的控制作用$\boldsymbol{u}_i$，记控制律为

$$\boldsymbol{u}_i=\boldsymbol{u}_i(\boldsymbol{I}_i),\quad i=1,\cdots,N \tag{6-34}$$

式中，$\boldsymbol{I}_i$ 表示构成 $\boldsymbol{u}_i$ 时所依据的信息集合。如果对$\boldsymbol{I}_i$ 没有任何限制，即 $\boldsymbol{I}_i$ 可以包括系统的全部输出信息$\boldsymbol{y}_1$，…，$\boldsymbol{y}_N$，则这时的控制律具有形式

$$\boldsymbol{u}_i=\boldsymbol{u}_i(\boldsymbol{y}_1,\cdots,\boldsymbol{y}_N)\quad i=1,\cdots,N$$

可以看出，这实际上就是通常的集中控制律

$$\boldsymbol{u}=\boldsymbol{u}(\boldsymbol{y})$$

只是把$\boldsymbol{u}$分解为$\boldsymbol{u}=[\boldsymbol{u}_1^T\cdots\boldsymbol{u}_N^T]^T$由不同的分散控制器实现而已。所以，在这种情况下，虽然控制是分散实现的，但每一控制量的计算仍建立在系统全局信息的基础上，这样的控制只是集中控制律的分散实现，而不是真正的分散控制。

但是，如果对式（6-34）中的信息集合$\boldsymbol{I}_i$加以限制，使之不能包含系统的全局信息 $\boldsymbol{y}_1$，…，$\boldsymbol{y}_N$，则这样得到的就是分散控

制。对信息集合 I_i 的限制意味着在构成 u_i 时不必从所有其他子系统获取信息，这就减少了在线的数据通信。特别当我们限制信息集合$I_i=\{y_i\}$，即第 i 个控制器只能获得本子系统的输出信息 y_i 时，相应的控制律具有形式

$$u_i=u_i(y_i),\quad i=1,\ \cdots,\ N \tag{6-35}$$

这种对信息严格限制所构成的分散控制称为全分散控制。在这种情况下，每一控制器 i 只需检测本子系统的输出信息y_i，便可构成本子系统的控制输入u_i。由于不需要其他子系统的输出信息 $y_j(j\neq i)$，故子系统间的在线通信可以完全避免。显然，这种全分散的控制律在实现时最为简单，而且由式（6-35）可以看到，一旦某一测量信息 y_i 失效，它只会引起 u_i 的变化，而不会影响 $u_j(j\neq i)$。因此，系统的局部故障只会产生局部的影响，从而提高了控制系统的全局可靠性。

由式（6-35）所描述的分散控制律体现了信息分散、控制分散的思想。在经典控制中，对多变量系统采用多个单回路PID控制，就是这种分散控制的体现。在以现代控制理论为基础的大系统理论中，通常是用最优控制设计来导出分散控制律（6-35)的。

这种分散控制律的缺点也是显而易见的。由于多变量系统是一个输入输出高度耦合的系统，每一输出量 y_i 不但受到 u_i 的影响，而且也受到$u_j(j\neq i)$ 的影响。采用上述分散控制律，无论其形式是经典的 PID 控制或现代最优控制，都是建立在忽略子系统间耦合影响的基础上的，这就不可避免地引起了控制性能的下降，这是分散控制律因信息不完全而造成的本质弱点。正是这个原因，这种忽略子系统间相互影响的分散控制，尽管有着计算简易、可靠性高的特点，但在实际上却不能导致高性能的控制。

由上述分析可知，要提高分散控制系统的性能，必须对分散控制器所缺乏的信息进行适当的弥补。由于在全分散控制结构下，不可能通过在线通信交换全局的实际信息，信息的不足只能用预测的手段加以弥补，因此，把预测控制的原理引入分散控制结构，形成分散预测控制，将是提高分散控制系统性能的一个有

效途径。

根据预测控制的一般原理，控制系统的性能、主要取决于滚动优化能否建立在准确的输出预测基础上。在这里，除了以模型预测作为输出预测的主体外，其他未包含在模型中的因素对输出的影响都可通过反馈校正环节中的误差预测加以补偿。而误差预测通常不是基于模型，而是根据输出误差的历史信息用投影预测来实现的。由于投影预测不需要其他子系统的信息，故恰好适合于这里的分散控制结构。因此，在分散预测控制中，每一局部控制器除了依据本子系统的模型进行输出预测外，还可利用预测控制的反馈校正原理，根据本子系统的实际误差信息，用投影预测方法预测未包含在局部模型中的子系统间的耦合影响。这种预测虽然不如用全局模型和整体信息预测的结果精确，但至少是弥补了一部分分散控制所丧失的信息。与原来不考虑信息补偿的分散控制算法相比，提高了预测精度并改善了控制性能。另一方面，由于投影预测只需要本子系统的输出误差信息，不需与其他子系统进行信息交换，分散控制的优点则仍能保持。因此，这种分散预测控制以预测控制弥补了分散控制的信息不足，改善了控制性能，同时又以分散控制简化了预测控制律，避免了在线通信，减少了计算量，易于用多台低性能微机在线实现。因此，这是一种把分散控制的结构优点与预测控制的性能优点结合起来的控制策略。

下面，我们就来推导多变量动态矩阵控制的分散控制算法。为说明原理并与其集中控制算法相比较，我们将采用 6.1 节中的记号，并假设多变量系统的输入输出数均为 n，且整体系统被划分为 n 个子系统。

1. 预测模型

对于第 i 个分散控制器来说，其对应的子系统输入为 u_i，输出为 y_i，其他子系统的输入输出 u_j，$y_j(j \neq i)$ 均属未知，故其模型预测只能建立在子系统局部模型的基础上

$$\tilde{y}_{i,N1}(k)=\tilde{y}_{i,N0}(k)+a_{ii}\Delta u_i(k), \quad i=1, \cdots, n \tag{6-36}$$

式中各记号的意义同式（6-1）。

若u_i从k时刻起变化M次，则对应于式（6-2），有预测公式

$$\tilde{\boldsymbol{y}}_{i,PM}(k)=\tilde{\boldsymbol{y}}_{i,P0}(k)+\boldsymbol{A}_{ii}\Delta\boldsymbol{u}_{i,M}(k),\ i=1,\cdots,n \tag{6-37}$$

2. 滚动优化

由于采用了分散控制，每一分散控制器都有其局部的优化性能指标

$$\min_{\Delta\boldsymbol{u}_{i,M}(k)} J_i(k)=\|\boldsymbol{w}_i(k)-\tilde{\boldsymbol{y}}_{i,PM}(k)\|^2_{Q_i}+\|\Delta\boldsymbol{u}_{i,M}(k)\|^2_{R_i} \tag{6-38}$$

它与6.3节中分散优化的性能指标（6-16）相同。

在不考虑约束的情况下，结合预测模型（6-37），可导出分散的最优控制解

$$\Delta\boldsymbol{u}_{i,M}(k)=(\boldsymbol{A}_{ii}^T\boldsymbol{Q}_i\boldsymbol{A}_{ii}-\boldsymbol{R}_i)^{-1}\boldsymbol{A}_{ii}^T\boldsymbol{Q}_i[\boldsymbol{w}_i(k)-\tilde{\boldsymbol{y}}_{i,P0}(k)]$$

分散的最优控制律为

$$\Delta u_i(k)=\boldsymbol{d}_i^T[\boldsymbol{w}_i(k)-\tilde{\boldsymbol{y}}_{i,P0}(k)] \tag{6-39}$$

其中

$$\boldsymbol{d}_i^T=[1\ 0\ \cdots\ 0](\boldsymbol{A}_{ii}^T\boldsymbol{Q}_i\boldsymbol{A}_{ii}+\boldsymbol{R}_i)^{-1}\boldsymbol{A}_{ii}^T\boldsymbol{Q}_i=[d_{i1}\cdots d_{iP}]$$

3. 反馈校正

预测控制在从k时刻推移到$k+1$时刻后，为了得到$k+1$时刻的预测初值$\tilde{\boldsymbol{y}}_{N0}(k+1)$，应将$k$时刻所得的$\tilde{\boldsymbol{y}}_{N1}(k)$进行移位。但为了使$\tilde{\boldsymbol{y}}_{N0}(k+1)$比较准确地反映系统的实际情况，在将$\tilde{\boldsymbol{y}}_{N1}(k)$移位前必须根据实测信息对其进行校正（见式（6-10)、式（6-11））。在集中控制的情况下，这一校正的目的是为了把模型失配、干扰等因素考虑在预测中。但在分散控制的情况下，即使模型无失配且无干扰，这一反馈校正仍是不可少的。这是因为，分散控制器所用的预测模型（6-36）是一个局部模型，它忽略了其他子系统的输入对本子系统输出的影响，因此，按式（6-36）预测所得的子系统输出$\tilde{y}_{i,1}(k+1|k)$必然与实际输出$y_i(k+1)$不相吻合，它们可构成预测误差

$$e_i(k+1)=y_i(k+1)-\tilde{y}_{i,1}(k+1/k) \qquad (6\text{-}40)$$

预测误差不但包含了模型失配与干扰的信息，更重要的是包含了由于实现分散控制而丢失的反映子系统间耦合影响的信息 $\sum_{\substack{j=1\\j\neq i}}^{n} a_{ij}\Delta u_j(k)$。显然，这部分信息由于控制结构的限制不可能采用模型预测的方法得到，因此，我们只能采用投影预测的方法来预测它对 y_i 未来值的影响并加以校正。

在以前介绍的单变量或多变量DMC算法中，在得到预测误差 $e_i(k+1)$ 后，都采用简单的加权办法来预测未来的误差。在分散预测控制中，由于预测误差中包含了子系统间的耦合因素，它的变化不一定是缓慢光滑的，因此，有必要寻找更有效的时间序列预报方法。为了提高误差预测的精度，首先我们要求：

（1）在优化性能指标（6-38）中的预测时域 P 应取得短些；

（2）在每一采样时刻应不断更新预测初值使之与实际相符；

（3）应迅速抑制旧数据对预测量的影响。

而时间序列预报中的霍特-温特斯（Hott-Winters）方法是很适合上述要求的[21]。其做法是，首先根据式（6-40）得到的 $e_i(k+1)$ 和上一时刻得到的 $e_i(k)$ 构成误差的差分项

$$\Delta e_i(k+1)=e_i(k+1)-e_i(k)$$

然后以 $\Delta e_i(k+1)$ 作为预测变量应用上述方法。如果 $\Delta e_i(k+1)$ 的变化没有周期性的因素，则可得到预测其未来值的公式

$$\left.\begin{aligned}
\Delta\tilde{e}_i(k+1+j\,|\,k+1)&=m_i(k+1)\\
&\quad+j\gamma_i(k+1),\ j=1,\ \cdots,\ N\\
m_i(k+1)&=\alpha\cdot\Delta e_i(k+1)+(1-\alpha)\\
&\quad\times[m_i(k)+\gamma_i(k)]\\
\gamma_i(k+1)&=\beta[\Delta e_i(k+1)-m_i(k)]\\
&\quad+(1-\beta)\gamma_i(k)\\
&\quad 0<\alpha<1,\ \ 0<\beta<1
\end{aligned}\right\} \qquad (6\text{-}41)$$

式中，$m_i(k+1)$ 是 $\Delta\tilde{e}_i$ 的基准值，$\gamma_i(k+1)$ 是步长增量，

而初值$m_i(0)$，$\gamma_i(0)$ 及滤波参数α、β的选择都是比较简单的，可参见文献〔21〕。

在$k+1$时刻，由于$m_i(k)$、$\gamma_i(k)$ 已知，$\Delta e_i(k+1)$由实测误差信息的差分构成，故即可按上式算出所有的$\Delta\tilde{e}_i(k+1+j|k+1)$，$j=1,\cdots,N$，再按$\Delta e_i$的定义，可推算出未来的输出误差$\tilde{e}_i(k+1+j|k+1)$，$j=1,\cdots,N$。以此对预测输出进行校正并移位后，可得

$$\tilde{y}_{i,0}(k+1+j|k+1)=\tilde{y}_{i,1}(k+1+j|k)+\tilde{e}_i(k+1+j|k+1),\quad j=1,\cdots,N \qquad (6\text{-}42)$$

这样，就可得到经校正后的预测初值$\tilde{\boldsymbol{y}}_{i,N0}(k+1)$。

由上述分散预测控制算法的原理可知，为适应分散控制结构的要求，在模型预测和滚动优化方面都把多变量系统分解为独立的子系统来进行。由于其输出预测采用了局部模型（6-36），忽略了子系统之间的耦合影响，若不加以校正，则因优化控制律（6-39）中的$\tilde{\boldsymbol{y}}_{i,P0}(k)$与实际不符，将导致控制性能的显著下降。因此，必须充分利用预测控制的反馈校正机理，根据实时信息采用合适的预测手段修正模型预测，使输出的预测值适当地包含子系统间的耦合影响。但这种预测不能依靠子系统外部的信息，而只能利用本子系统的输出误差信息，因此，选择合适的时间序列预报方法是提高预测精度并进一步改善控制性能的关键。这种方法既保持了分散控制不需在线通信、计算量小、整体可靠性高的优点，又在一定程度上用预测手段弥补了因分散而引起的信息不足，提高了分散控制系统的控制质量。

为了说明分散预测控制算法的优点，我们考虑图6-6所示的锅炉温度控制的例子。

该锅炉的炉膛上缠绕着4组加热线圈供测量用，膛内装有4组燃烧器进行加热，构成了一个4输入4输出的多变量系统。很明显，由于它们工作在共同的炉膛内，每一加热器对所有线圈测得的温度都会产生影响，从图6-7所示的系统的开环阶跃响应曲线就可明显地看到这种耦合关系。图中因对称的原因，相应的阶

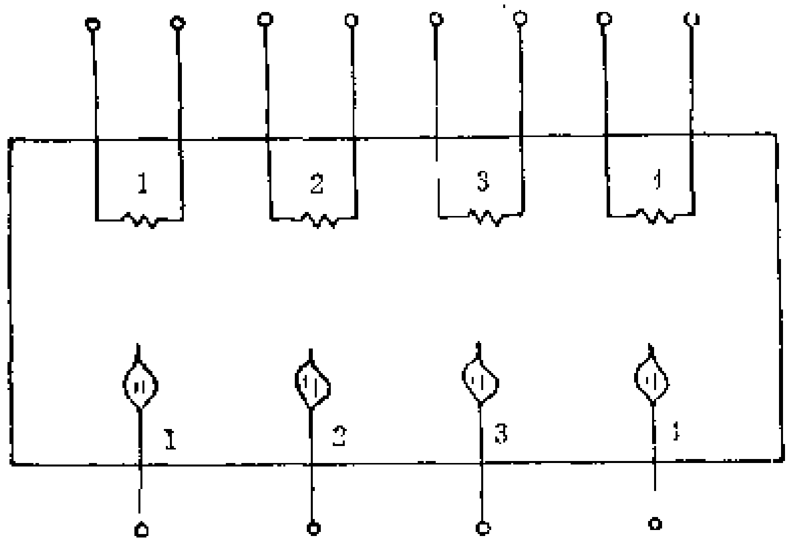


图6-6　锅炉燃烧系统

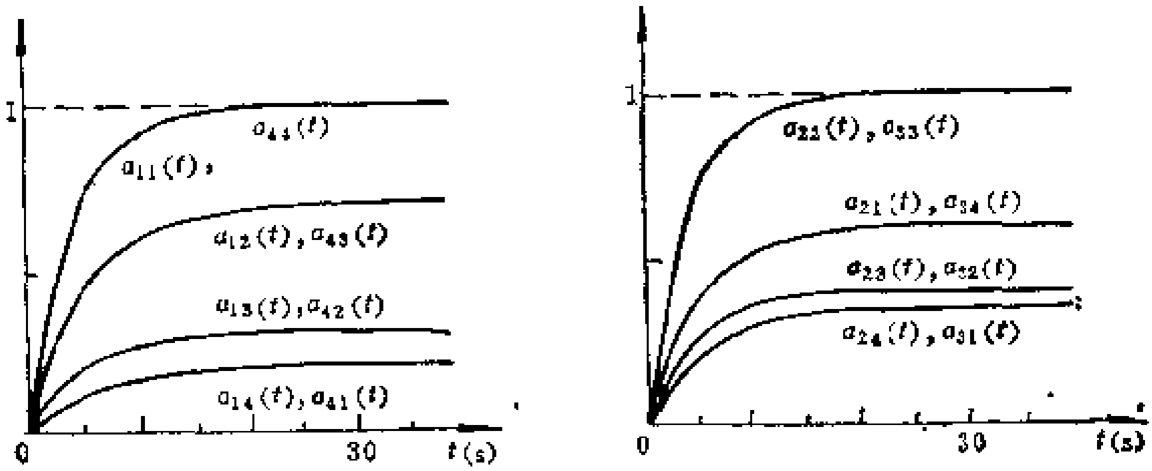


图6-7　燃烧系统的开环阶跃响应

跃响应曲线是重合的。

对于这一多变量系统，我们采用三种方案对其进行控制：

（1）集中控制，即把系统作为一个整体用 6.1 节的多变量 DMC 算法控制；

（2）分散控制，系统划分为 $n=2$ 个子系统，分别包含 1，2 组、3、4 组线圈和燃烧器，按本节原理结合 $n=2$ 的多变量 DMC 算法控制；

（3）分散控制，系统划分为 $n=4$ 个子系统，分别包含一

组线圈和对应的燃烧器，按本节原理及单变量 DMC 算法控制。

在仿真研究中，取采样周期 $T=1\text{s}$. 各时域参数 $N=26$, $P=3$, $M=1$, 权矩阵 $\boldsymbol{Q}_i=\boldsymbol{I}$, $i=1,\cdots,n$, $\boldsymbol{R}_i=\boldsymbol{I}$, $i=1,\cdots,n$, 时间序列预报公式中的 $\alpha=0.8$, $\beta=0.2$, 并对期望值进行了适当的“柔化”后，可得到图 6-8 所示的仿真结果。其中，曲线 1 、2 、3 分别对应于上述三种控制方案。可以看出，由于采用了合适的预测手段进行补偿，因分散化而引起的控制性能的下降是微不足道的。

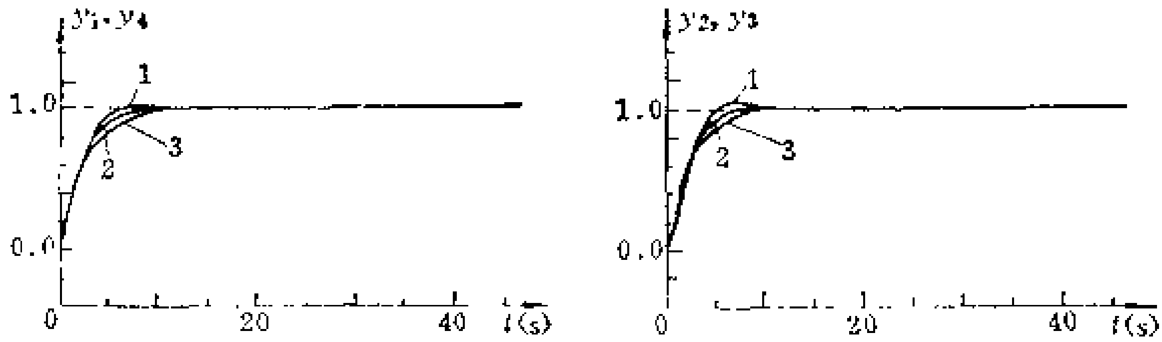


图6-8 不同控制方案的结果比较

1—集中控制，$n=1$；2—分散控制，$n=2$；3—分散控制，$n=4$。

在控制性能十分接近的情况下，我们感兴趣的是采用分散控制后究竟有什么好处。下面给出了这三种控制方案所需内存容量及计算时间的相对比较值：

		所需内存	计算时间
集中控制	$n=1$	1.000	1.000
分散控制	$n=2$	0.654	0.245
分散控制	$n=4$	0.339	0.144

很明显，由于信息的分散化，分散控制所需的内存容量和计算时间与集中控制相比都有了明显的下降。这表明，分散预测控制确是一种把预测控制与分散控制的优点结合起来的优良控制策略。特别当多变量系统有较多的输入输出数而硬件环境又不允许进行大量复杂计算时，采用这种策略、可使预测控制算法得以实

现，并同样能获得良好的控制性能。

以上讨论的分散预测控制算法是针对全分散控制结构的。如果子系统之间允许进行有限的通信，只要各分散控制器不能获得系统的全部实时信息，则仍属于分散控制的范畴。对于这种情况下的分散预测控制，可按照同样的原理，尽可能利用能得到的信息作出较完善的模型预测，而对因信息不足而不能用模型进行预测的部分，则可通过时间序列方法预测补偿，以改善系统的控制性能，这方面的例子可参见文献〔22〕。

第七章　预测控制在实际应用中的发展

本书第三章介绍的预测控制基础算法，从原理上为工业过程的优化控制提供了新的途径。但这并不意味着只依靠这些基础控制算法就可满足复杂工业控制的种种要求。近年来，随着预测控制应用的日益广泛，人们根据过程的复杂性和控制的要求，把这些基础控制算法与先进的控制结构和控制思想相结合，提出了一系列复杂系统预测控制的策略，这是预测控制面向实际应用的进一步发展。在本章中，我们将对这些发展作若干示例性的介绍。这些内容不仅有助于加深对预测控制基本原理的理解，而且展示了预测控制在实际应用中的灵活性，为在复杂的工业过程中如何应用预测控制提供了有益的参考。

§7.1　具有前馈-反馈结构的预测控制

对象的输入通常可分为两类，一类是可控输入，即控制量；另一类是不可控输入，它包含了可测量或可预知但无法加以改变的外部作用以及由对象、环境不确定性引起的未知干扰。控制的目的，就是要不断调整可控输入，用于克服不可控输入量的影响，并使对象输出具有期望的动态特性。对于无规律可循的未知干扰，这一调整只有在它反映到可测量后才能进行，因而必须采用反馈的方式。但对于变化规律已知的不可控输入，由于其对输出的影响有一定的可预见性，则可通过前馈预先加以补偿。这种前馈结合反馈的控制结构，已被普遍地应用于工业过程中。

在前面介绍的预测控制算法（如DMC）中，我们已经详细讨论了其中反馈校正这一重要组成部分。它利用实测信息与预测信息间的误差构成对未来输出误差的预测

$$\tilde{e}(k+i\mid k+1)=h_i e(k+1),\quad i=1,\cdots,N \tag{7-1}$$

并用它校正对未来输出的预测值

$$\tilde{y}_{cor}(k+i\,|\,k+1)=\tilde{y}_{N1}(k+i\,|\,k)+\tilde{e}(k+i\,|\,k+1)$$

这一环节的主要作用是克服环境干扰、参数时变、模型失配等不可知因素的影响。这种反馈校正还可以用更一般的形式实现，例如代替式（7-1）可采用时间序列的预报公式

$$\begin{aligned}&\tilde{e}(k+i\,|\,k+1)\\&=E_i[e(k+1),\ e(k),\ \cdots,\ e(k-l+1)]\end{aligned}\qquad(7\text{-}2)$$

其中，$E_i(\cdot)$ 为线性或非线性函数，l 为预报所用的历史数据长度。注意，这里对误差的预测只能建立在误差信息自身的基础上，而无法采用任何因果性的预报方式，这是由于对误差产生的原因一无所知所致。

然而，对于有规律的不可控输入，如果也把它当作不可知干扰用上述反馈方式加以校正，显然是不合理的。这是因为，反馈校正只有当误差产生后才起作用，带有一定的被动性。而且采用非因果的误差预报和校正，不能充分利用这部分误差产生的已知因果信息。在这种情况下，更合理的处理方式应是充分利用这部分输入的规律性及算法的预测功能，构成前馈控制，及时有效地补偿由这部分输入引起的误差。

我们以单变量 DMC 算法为例，来讨论其前馈补偿问题。这里的核心问题是要把已知但不可控输入对系统输出的影响作为因果信息包含在预测模型中。为此，除了测定对象输出对可控输入量的阶跃响应序列$\{a_i\}$外，还应测定输出量对于规律已知的不可控输入量 v 的阶跃响应序列$\{b_i\}$。在每一采样时刻，对象输出值的变化是由控制量 u 和不可控输入量 v 共同引起的，因而可在预测模型（3-4）中补充不可控输入量的影响而改写为

$$\tilde{\boldsymbol{y}}_{PM}(k)=\tilde{\boldsymbol{y}}_{P0}(k)+\boldsymbol{A}\Delta\boldsymbol{u}_M(k)+\boldsymbol{B}\Delta\boldsymbol{v}_P(k)\qquad(7\text{-}3)$$

其中

$$\boldsymbol{B}=\begin{bmatrix}b_1 & & 0\\ \vdots & \ddots & \\ b_P & \cdots & b_1\end{bmatrix}\qquad \Delta\boldsymbol{v}_P(k)=\begin{bmatrix}\Delta v(k)\\ \vdots\\ \Delta v(k+P-1)\end{bmatrix}$$

而 $\Delta v(k)=v(k)-v(k-1)$。

此外，类似于式（3-9），一步后的输出预测值可用向量形式记为

$$\tilde{\boldsymbol{y}}_{N1}(k)=\tilde{\boldsymbol{y}}_{N0}(k)+\boldsymbol{a}\Delta u(k)+\boldsymbol{b}\Delta v(k) \tag{7-4}$$

其中，$\boldsymbol{b}=[b_1\cdots b_N]^T$

采用同样的优化性能指标（3-5），在不考虑约束的情况下，可得到最优控制向量

$$\Delta\boldsymbol{u}_M(k)=(\boldsymbol{A}^T\boldsymbol{Q}\boldsymbol{A}+\boldsymbol{R})^{-1}\boldsymbol{A}^T\boldsymbol{Q}[\boldsymbol{w}_P(k)-\boldsymbol{y}_{P0}(k)-\boldsymbol{B}\Delta\boldsymbol{v}_P(k)]$$

即时控制增量为

$$\Delta u(k)=\boldsymbol{d}^T[\boldsymbol{w}_P(k)-\tilde{\boldsymbol{y}}_{P0}(k)-\boldsymbol{B}\Delta\boldsymbol{v}_P(k)] \tag{7-5}$$

与式（3-7）给出的控制律相比较可知，控制律（7-5）中多了一项 $\boldsymbol{B}\Delta\boldsymbol{v}_P(k)$。其物理含义是，把规律已知但不可控的输入量在优化时域中对输出的影响从期望值中扣除，构成新的期望值

$$\boldsymbol{w}'_P(k)=\boldsymbol{w}_P(k)-\boldsymbol{B}\Delta\boldsymbol{v}_P(k)$$

然后再考虑只有可控输入时的滚动优化问题。由图 7-1 可见，它具有前馈补偿的性质。

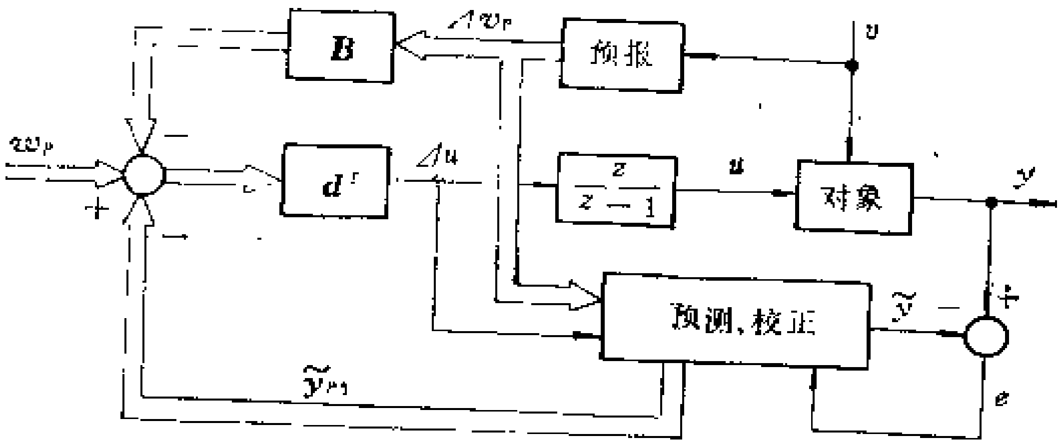


图7-1　带有前馈补偿的动态矩阵控制

上述控制结构和控制律是建立在 y 对 v 的动态响应 $\{b_i\}$ 完全已知，并且对 v 的变化完全可预知的基础上的。在这种情况下，控制律（7-5）将完全补偿不可控输入 v 对于输出 y 的影响。虽然这一条件在实际上往往过于苛刻，但用前馈处理有规律的不可控输入、用反馈处理未知干扰的思想仍是具有普遍意义的。

假定输出 y 对于不可控输入 v 的阶跃响应 $\{b_i\}$ 是已知的，但我们只能检测即时的 $v(k)$ 构成 $\Delta v(k)$，而不知道未来的 $v(k+1)$，$v(k+2)$，…，则可将预测模型（7-3）修改为

$$\tilde{\boldsymbol{y}}_{PM}(k)=\tilde{\boldsymbol{y}}_{P0}(k)+\boldsymbol{A}\Delta\boldsymbol{u}_M(k)+\boldsymbol{b}\Delta v(k) \tag{7-6}$$

相应得到的控制律为

$$\Delta u(k)=\boldsymbol{d}^T[\boldsymbol{w}_P(k)-\tilde{\boldsymbol{y}}_{P0}(k)-\boldsymbol{b}\Delta v(k)]$$

这时，在 k 时刻用式（7-4）预测 $k+1$ 时刻的输出仍准确地计入了 v 的影响，因而误差 e 中只包含了其他未知干扰对输出的影响，不再含有 v 所引起的误差。但在考虑优化时由于对 v 的未来值无先验信息而采用了预测模型（7-6），其实质是假设 $v(k)$ 保持不变，这与实际情况有可能不符，因而建立在这一模型基础上的控制律的最优性将受到影响。即便如此，由于最大限度地利用了未来的信息，这一控制律仍然要胜过不考虑 $\boldsymbol{b}\Delta v(k)$ 项的优化。

另一种情况是对不可控输入量 v 的变化规律有预见，但缺乏其对输出动态影响的阶跃响应模型。例如在锅炉燃烧控制系统中，用户负荷作为不可控输入，可能具有已知的日统计规律，但很难测定负荷对被控参量影响的准确模型。这时，我们固然可以通过输入输出数据进行辨识，获得 $\{b_i\}$ 后再用上述方法进行前馈补偿。但也可通过数据分析，导出 $v-y$ 间关系的粗略模型或模糊模型，以避免繁复的辨识运算。这种模型虽然较之阶跃响应显得粗糙，但只要其反映了影响的主要趋势，则在预测模型中加入其影响形成前馈部分，仍可在一定程度上补偿不可控输入 v 的影响。

上述前馈加反馈的 DMC 算法同样适用于多变量系统，其原理也适用于其他预测控制算法。可以看出，预测控制中的预测模型可以把不可控的已知输入包括在内，通过模型预测形成前馈去除其影响，反馈校正部分则针对未知干扰的影响通过非因果预测形成反馈去除其影响。因此，预测控制从原理上就是一种集前馈和反馈于一体的控制算法，它对不同的不可控输入采用了不同的预测策略，充分利用过程的已知信息最大限度地补偿不可控输入

对过程动态的影响，因而能取得较好的优化控制效果。

应该指出，对于不可控输入的模型预测，其目的在于尽可能提取它对过程动态影响的因果性信息，因此并不局限于基于精确数学模型的预测。结合预报技术和人工智能方法，可以在更广泛的意义上采用因果预测削弱不可控输入的影响。例如在分散控制中，如果可以得到各子系统间关联的简化模型或规则模型，则同样可通过模型预测构成前馈。虽然这种预测未必精确，但它至少可在一定程度上补偿子系统间的关联，因而可以改善分散控制系统的性能。

§7.2 串级预测控制

上节所讨论的前馈控制结构，只能用于处理有规则的不可控输入。同时我们也指出，对于部位不确定、变化不可知的无规则干扰，预测控制只能采用反馈的方式进行被动补偿，即在干扰引起误差后，用非因果的误差预测予以校正。但在无自校正的预测控制算法（如 DMC、MAC）中，要用第三章介绍的基础控制算法获得理想的抗干扰性，仍然存在着本质的困难。首先，抗干扰性要求对干扰作出快速的反应，通常希望有较小的采样周期，但这些算法采用了非参数模型，在线计算也比一般 PID 算法复杂，为了使模型维数不致过高以影响控制的实时性，往往不能把采样周期取得过小。其次是在采用加权校正方式（3-11）对预测输出进行校正时，不能分辨误差究竟是由模型失配还是由干扰引起的。在 5.1 节关于设计参数的讨论中，已指出对这两种情况的校正策略是不同的。因此，为了增强对模型失配的鲁棒性，校正参数 h_i 的选择往往难以兼顾对干扰反应的快速性。

上述对于采样周期和校正策略相互矛盾的要求，在预测控制基础算法的单层次结构中是很难协调的。解决这一矛盾的途径之一是引入分层控制结构，即在不同的层次上采用不同的采样周期分别满足优化计算和快速抗干扰的需要，并把抗干扰性和鲁棒性这两个难以用同一设计参数协调的性能要求划分到不同的层次中

分别处理。这样，才有可能在保持预测控制系统优良跟踪性能和鲁棒性的同时，增强对干扰的抑制能力。为了达到这一目的，可以借鉴串级控制的思想，把预测控制结合到串级结构中，形成串级预测控制。以下，我们以 DMC-PID 串级控制[23] 为例，来说明串级预测控制的设计思想。

首先，我们注意到密集采样的数字 PID 控制对于干扰有良好的抑制作用，故可在对象最易发生干扰的部位后面取出信号先构成 PID 闭环控制。这一层控制采用了比 DMC 控制高得多的频率，其目的在于快速有效地抑制突发性的干扰。这一被控回路与对象的剩余部分可视为一广义对象。在外层，再对这一广义对象实施通常的 DMC 控制。由于干扰的主要部分已及时得到抑制，体现在广义对象中可视为模型略有失配，所以外层可以良好的动态性能和鲁棒性为设计目标。这种分层的 DMC-PID 控制具有串级控制的结构形式（见图 7-2），只是以 DMC 算法取代了通常外层的 PID 控制。这样，它既能保持串级控制的结构优点，又能保持预测控制的性能优点。

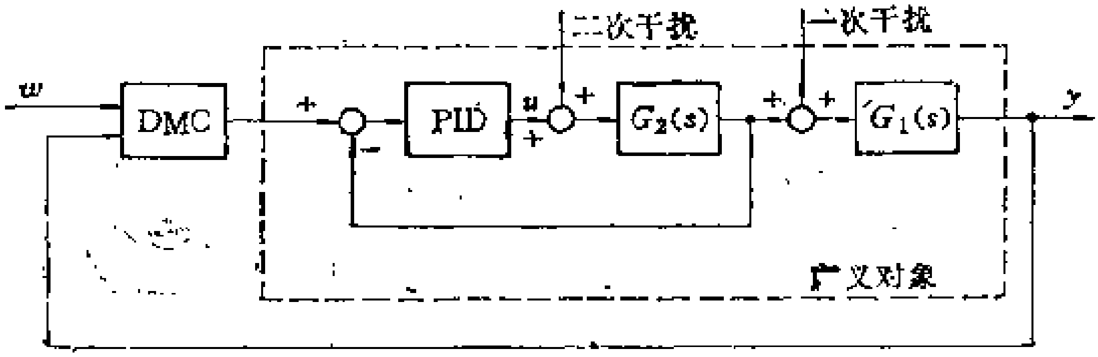


图7-2　DMC-PID串级控制结构

根据上述分层控制的目的，在图 7-2 所示的串级控制结构中，内层（副回路）和外层（主回路）的设计要点分别如下：

1. 副回路设计

副回路的选择应使副对象 $G_2(s)$ 包含有系统的主要干扰，并有较小的纯滞后或时间常数。副回路采用频率较高的数字 PID 控制，主要目的在于及时克服进入对象的二次干扰。

由于副回路采用了准连续的数字 PID 控制，故可采用通常的工程整定方法，如临界比例法、响应曲线法等结合经验公式确定 PID 控制参数。由于大量工业过程呈现为纯滞后加惯性的动态环节，而副对象$G_2(s)$只包含了其中纯滞后和时间常数较小的部分，所以一般采用 P 调节器，即可得到快速的响应。这不但有利于减小干扰引起的偏差幅值，而且也有利于主回路控制的设计。

2．主回路设计

主回路的被控对象是包含副回路和对象的剩余部分 $G_1(s)$ 在内的广义对象（见图 7-2 中虚框所示部分）。主回路控制采用 DMC 算法，主要目的在于获得良好的跟踪性能并对模型失配有较强的鲁棒性。

应该注意，在主回路的 DMC 控制中，阶跃响应模型应取自上述广义对象而不是原始对象。因此在测试时，输入阶跃应加于副回路的设定值，而不是作用于对象的直接控制量。在整个广义对象中，由于副对象 $G_2(s)$ 与主对象 $G_1(s)$ 相比，本身已有较小的时间常数，经控制后，副回路的动态响应将更快，所以主要影响广义对象动态的将是主对象 $G_1(s)$。它通常可包含有较大的纯滞后，并且有较大的时间常数，因此采用具有预测功能的 DMC控制显然要比PID控制更为有利，并可获得较好的跟踪性能。由于主要干扰的影响经副回路控制后已大为减弱，在主回路中可认为是广义对象的模型略有失配，因此主回路 DMC 中校正策略的选择可主要考虑增强控制的鲁棒性。这样，我们就可参照 5.1 节提供的设计要点对 DMC 参数进行整定。特别当被控过程近似具有 5.2 节中所讨论的典型形式时，还可借助于定量分析的结果辅助 DMC 参数的设计。

下面，我们举一个仿真比较的例子来说明串级预测控制的优点。设被控过程的传递函数为

$$G(s)=\frac{10}{(s+1)(s+2)(6s+1)}$$

考虑下面三种控制方案：

（1）DMC 控制，采样周期为 $T=1\ \mathrm{s}$。

（2）PID 串级控制，内、外环对象分别为

$$G_2(s)=\frac{1}{(s+1)(s+2)},\quad G_1(s)=\frac{1}{6s+1}$$

内、外环采样周期分别为 $T_2=0.1\ \mathrm{s}$，$T_1=1\ \mathrm{s}$。

（3）DMC-PID 串级控制，内、外环同（2）。

图 7-3 中曲线 1、2 分别为采用方案（1）和（3）的控制结果，它们有相同的动态响应，但当输入端有一幅值为 d 的阶跃干扰时，用 DMC 控制对小幅值 $d=0.2$ 的干扰响应也比不上用 DMC-PID 串级控制对大幅值 $d=1$ 的干扰响应。这表明，串级结构的引入将大大改善 DMC 控制系统的抗干扰性。

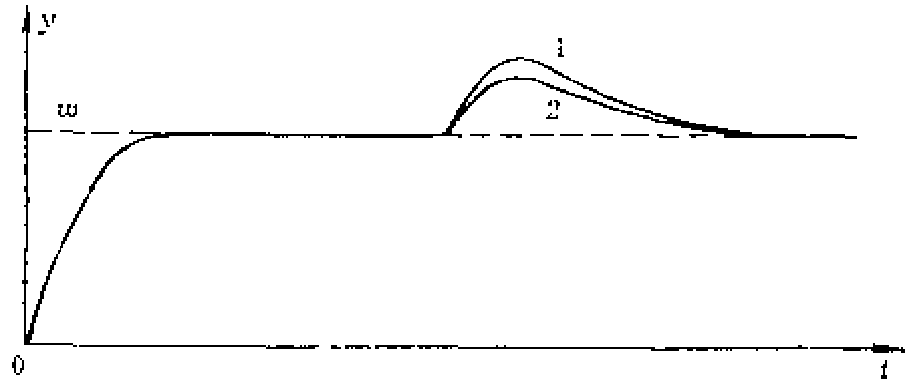


图7-3 DMC（曲线 1）和DMC-PID（曲线 2）抗干扰性比较

图 7-4 则是采用方案（2）和（3）的控制结果比较，其中，曲线 1 为传统 PID 串级控制的结果，而曲线 2 则为 DMC-PID 串级控制的结果。可以看出，除了抗干扰性还略差一些外，DMC-PID 串级控制的跟踪性能及鲁棒性都要胜过 PID 串级控制。就综合质量而言，可认为前者优于后者。这是引入预测控制所带来的性能优点。对于无纯滞后的对象尚且如此，对于有纯滞后对象的仿真表明，串级预测控制的性能优点将更为突出。

DMC-PID 串级控制具有良好的控制性能，并且易于设计整定，因而在工业过程中有一定的应用价值。图 7-5 是某合成氨厂变换工段的工艺流程简图。下面我们简要介绍对其应用串级预测控制的方案。

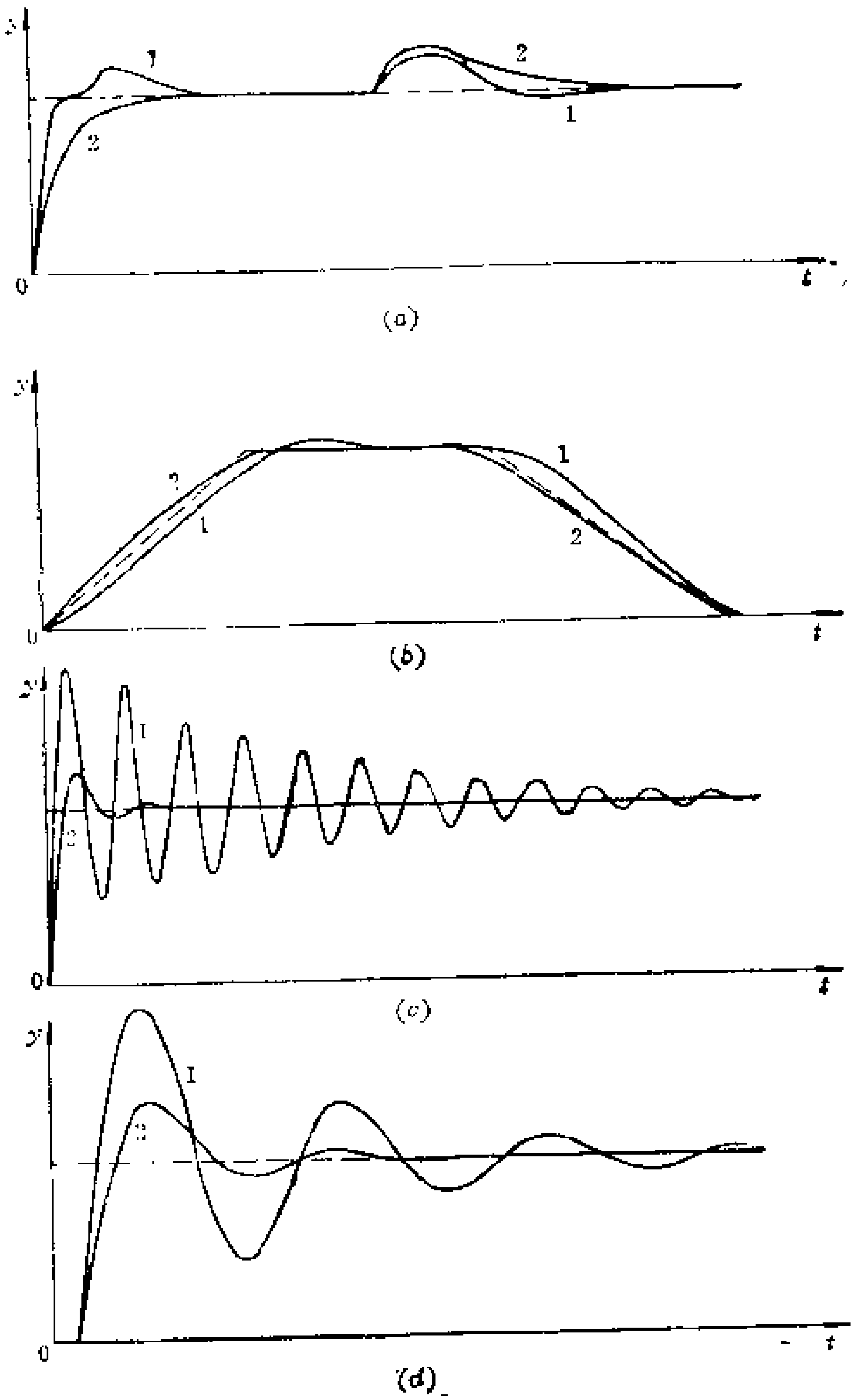


图7-4 PID串级控制（曲线1）与DMC-PID串级控制（曲线2）的比较

（*a*）动态响应与抗干扰性；（*b*）跟踪性能；（*c*）增益失配时的响应（对象增益为原来的2倍）；（*d*）滞后失配时的响应（对象有 2 s 时滞）。

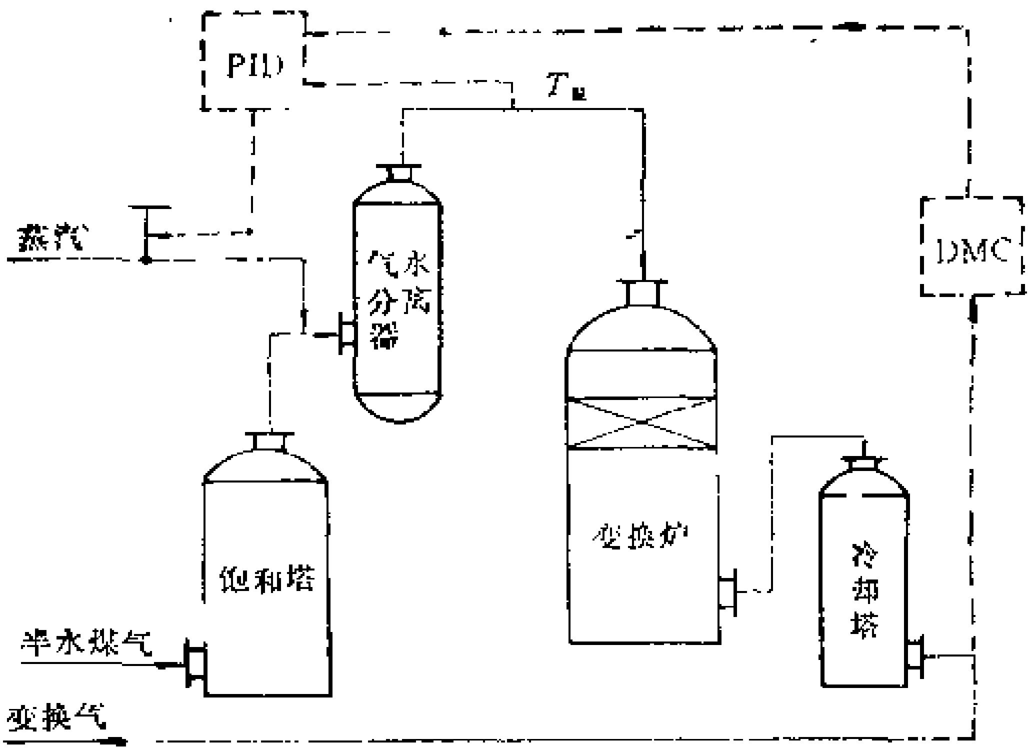


图7-5　变换工段的工艺流程及控制方案

变换工段是合成氨生产的重要环节。从压缩工段来的半水煤气经过饱和塔预热后，又在饱和塔出口管道上添加适量的水蒸汽，以达到合适的汽水比。半水煤气经气水分离器后，再经过一系列换热装置（图中未画）后升温到380℃左右进入变换炉。在变换炉中发生如下的变换反应：

$$CO+H_2O(汽)\xrightleftharpoons{触媒}CO_2+H_2+Q(放热)$$

这样，既除掉了半水煤气中的无用气体CO，又获得了合成反应所需要的H_2。变换后的气体称为变换气，经冷却塔冷却后送下一工段碳化。在工艺上，为保证CO的彻底清除，要求变换气中残存的CO含量在3％以下。

对于这样一个变换过程，由于化学反应的速度和平衡均受到压力、温度、催化剂等多种因素的影响，故给控制带来了一定的难度。在这里，蒸汽流量是控制量，而半水煤气的压力和流量以及催化剂各层的温度都会使平衡发生变化，但其中主要的前期干

扰是来自蒸汽压力的变化。因此,我们选择 DMC-PID 控制结构,把气水分离器出口的气体温度 $T_{\text{汽}}$ 作为内环被调量,用以快速反映蒸汽压力这一主要干扰的影响,并通过内环的 P 调节器迅速加以调节。而外环则对变换工段的主要性能要求——变换气中的 CO 含量进行控制,用 DMC 算法克服因工况引起的、内环又无法克服的模型失配。这一控制结构如图 7-5 中虚线所示。该控制方案在某合成氨厂变换工段试验后,取得了良好的效果。变换气中 CO 含量围绕设定值 2 %的波动比起 PID 串级控制有所减小,每天所用的蒸汽量明显节约,达到了提高控制性能、降低能耗的目的。

上述 DMC-PID 串级控制算法充分利用了串级控制的结构优点,可以通过内回路较为频繁的 PID 控制快速抑制进入对象的二次干扰。同时,它又发挥了预测控制的算法优点,在外回路可以有效地处理纯滞后,并有良好的跟踪性能和鲁棒性,因而它与单纯的 DMC 控制或 PID 串级控制相比都显现出更为优越的综合控制性能。由于采用了分层结构,不同的性能要求被划分到不同的层次分别予以处理,不但解决了单层次 DMC 控制难以解决的矛盾,而且设计的冗余性提高了,参数也易于整定。因而它在复杂的工业过程中有一定的吸引力和实用性。

这种串级预测控制的策略还可推广到多变量系统,其中内环可由单回路 PID 控制构成,其目的在于以频繁的数字 PID 控制克服各回路中主要干扰的影响,而外环则可对局部闭环后得到的广义对象实现多变量 DMC 控制,它考虑了各回路之间的耦合,从整体上对系统实现以优化为目的的控制。

在过程中间量不可测时,由于缺乏中间信息,故不能构成上述形式的串级控制。但按分层控制的原理,同样可把对象输出直接反馈构成内环回路,以密集的 PID 控制尽快抑制干扰。外环则在此基础上再用预测控制实现优化。这种分层控制方式称为透明控制 (transparent control)[4],它与传统的串级控制在形式上略有区别,但处理信息的策略思想是相同的。由于采用了分层

概念，控制的要求可有侧重地分解到不同的层次，分别采用不同的策略予以满足，因而可取得较好的控制效果，并简化了设计过程。

§7.3 带有自校正的预测控制算法

在第二章关于预测控制基本原理的介绍中，我们已经指出，反馈校正可以取各种不同的形式。它可以像 3.1 节的 DMC 算法和 3.2 节的 MAC 算法那样，通过对误差的预测和补偿实现，也可以像 3.3 节的 GPC 算法那样，通过模型的在线辨识和控制律的在线修正，以自校正的方式实现。后者实际上也是一种分层的控制结构。

上一节已经分析了无自校正的预测控制算法在选择校正参数时所面临的抗干扰性和鲁棒性的矛盾，这从它们的 IMC 结构图 4-3 中也可看出。其中滤波器 $G_F(z)$的设计既要考虑模型 $G_M(z)$与对象 $G_P(z)$ 的失配，又要考虑对作用于对象的干扰 v 的抑制，两者很难兼顾。因此，上节中提出了串级控制的结构和分层控制的思想克服这一困难。但从图 4-3 中也可看出，如果能使模型尽可能跟踪对象的变化，以致在控制过程中能保持 $G_M(z)\approx G_P(z)$，那么鲁棒性就能得到基本的保证，滤波器 $G_F(z)$ 的设计就只须考虑抗干扰性。不仅如此，控制器 $G_C(z)$ 也可根据模型的变化及时调整，以使它对于对象的变化仍能保持良好的控制性能。这表明，在对控制综合性能有较高要求时，可把自校正机制与基础 DMC、MAC 算法结合起来，形成类似于 GPC 的另一类分层控制结构。这种带有自校正的预测控制算法，可以把抗干扰性和鲁棒性划分到基础预测控制层和自校正层分别处理，因面可获得较好的控制综合性能。下面，我们以 DMC 算法为例，讨论它如何与自校正机制相结合。

显然，在自校正 DMC 算法中，自校正层的核心问题是在线辨识对象的阶跃响应系数，并以此在线修正模型与控制律。在这里，阶跃响应系数出现在由增量卷积公式描述的预测模型中

$$\Delta y(k)=\sum_{i=1}^{N} a_i \Delta u(k-i) \tag{7-7}$$

其中，$\Delta y(k)=y(k)-y(k-1)$，$\Delta u(k)=u(k)-u(k-1)$

引入记号

$$\boldsymbol{a}=\begin{bmatrix} a_1 \\ \vdots \\ a_N \end{bmatrix} \qquad \Delta \boldsymbol{u}_N(k-1)=\begin{bmatrix} \Delta u(k-1) \\ \vdots \\ \Delta u(k-N) \end{bmatrix}$$

则可将上式记为

$$\Delta y(k)=\Delta \boldsymbol{u}_N^T(k-1)\boldsymbol{a} \tag{7-8}$$

对此，可采用带有遗忘因子的递推最小二乘法在线估计参数 $\boldsymbol{a}$，其计算公式为

$$\left.\begin{aligned} \hat{\boldsymbol{a}}(k)&=\boldsymbol{a}(k-1)+\boldsymbol{K}(k)[\Delta y(k)-\Delta \boldsymbol{u}_N^T(k-1)\hat{\boldsymbol{a}}(k-1)] \\ \boldsymbol{K}(k)&=\boldsymbol{P}(k-1)\Delta \boldsymbol{u}_N(k-1) \\ &\quad\times[\lambda+\Delta \boldsymbol{u}_N^T(k-1)\boldsymbol{P}(k-1)\Delta \boldsymbol{u}_N(k-1)]^{-1} \\ \boldsymbol{P}(k)&=\frac{1}{\lambda}[\boldsymbol{I}-\boldsymbol{K}(k)\Delta \boldsymbol{u}_N^T(k-1)]\boldsymbol{P}(k-1) \end{aligned}\right\} \tag{7-9}$$

其中，$\hat{\boldsymbol{a}}(k)$ 为 k 时刻对参数 $\boldsymbol{a}$ 的估计值，初值 $\hat{\boldsymbol{a}}(0)$ 可由离线辨识得到，$\boldsymbol{P}(0)=\alpha^2\boldsymbol{I}$，$\alpha^2=10^5\sim10^{10}$，遗忘因子 λ 通常选为 $0.95\leqslant\lambda\leqslant0.995$。

上述辨识算法是否需要在整个过程中进行，取决于过程的特点及所用计算机的性能。一种可取的方案是：以实际输出与模型预测输出间的误差 e 为参考，如果在一段过程中 e 的存在有一定规律性，例如它始终保持正值且无减小趋势，则可启动辨识算法重新计算阶跃响应系数 $\{a_i\}$。在每一次辨识出新的 a_i 后，应重新计算控制参数 d_i。

当过程有弱非线性或缓慢时变时，采用辨识的方法修正模型，并通过模型参数修改控制参数，可以使控制系统有较好的适应性。但是应该注意到，在这个过程中，我们并未修改优化性能指标的

形式。由于对不同的模型参数 a_i，在优化性能指标不变即时域 P、M 及权矩阵 $\boldsymbol{Q}$、$\boldsymbol{R}$ 不变时，会得到不同的动态响应。所以，如果对象动态变化较大，虽经辨识后模型能与对象较好地匹配，控制系统的闭环响应却有可能变坏甚至引起不稳定。因此，所谓自校正不应只局限于由模型参数修改控制参数，而要进一步考虑如何改变优化性能指标使闭环响应适应过程动态的变化。对于一般过程，由于优化性能指标中的设计参数与控制性能间缺乏直接的解析关系，通过在线修改性能指标的整定方式来保持期望的动态响应是不现实的，故通常只能采用专家控制的思想，把阶跃响应分成几类典型情况，先在离线情况下分别整定优化性能指标中的设计参数，使闭环系统具有满意的动态响应，并将相应求得的控制参数与模型参数对应地一组组存入计算机内存。在线控制时，可根据辨识所得的阶跃响应系数对于典型模型参数的匹配情况，相应地改变控制参数。

对于可用一阶惯性加纯滞后环节近似描述其动态的典型工业过程，由于5.2节已导出了其DMC控制的某些解析关系，特别是式（5-16）给出了模型无失配时闭环时间常数与设计参数 P 的解析关系，故可用较为定量的方法实现其自校正DMC控制[24]。在辨识出对象的阶跃响应系数后，可通过特征抽取求出其时滞 l，增益 K 及时间常数 T_0，将其规范化后，即可按式（5-16）搜索优化时域 $P+l$ 中的参数 P，使闭环系统的时间常数达到给定值 T_c^* 而具有期望的动态响应。然后可由此算出新的控制参数 d_i，导出新的控制律。这一新的控制律不但产生于 a_i 的变化，也产生于优化时域 $P+l$ 的变化，在IMC结构中保持了前向传递特性的不变性。由于这里选择了 $M=1$，计算 d_i 时不涉及矩阵求逆问题，故很容易在线实现。此外，根据控制系统抗干扰性的好坏，还可在线修改校正系数 h_i。这种带有启发式的自校正DMC控制的结构可见图7-6。

为了说明在DMC算法中引入自校正的好处，考虑对象每隔125 s按下述顺序变化一次：

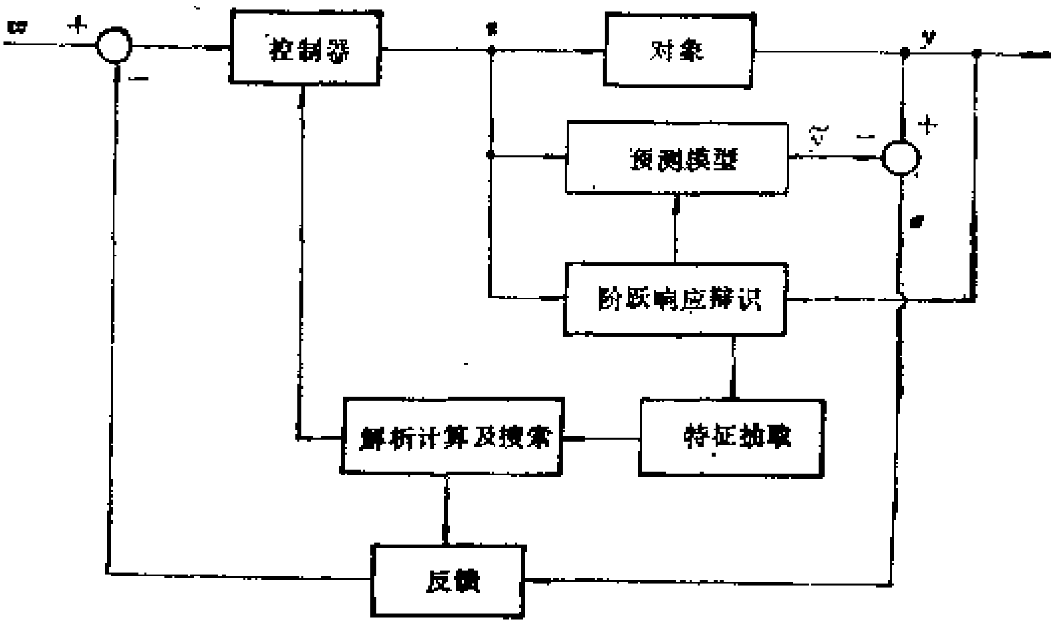


图7-6　自校正动态矩阵控制

(1) $\dfrac{e^{-s}}{1+2.5s}$　　(2) $\dfrac{5e^{-s}}{1+2.5s}$

(3) $\dfrac{e^{-4s}}{1+2.5s}$　　(4) $\dfrac{e^{-s}}{1+7.5s}$

取采样周期为$T=0.5$ s。在无自校正的DMC控制中，取(1)为不变的预测模型，则(2)、(3)、(4)分别表示对象发生了增益、时滞和时间常数的变化。设期望闭环时间常数为$T_s^*=1$ s，则由图7-7(a)的仿真结果可见，虽然在模型无失配时DMC的动态响应十分好，并在模型失配时仍有一定的鲁棒性，但因失配较大，闭环系统稳定性变差，动态响应与无失配时的期望响应相差甚大。这是因为，在这种算法中缺乏适应机制，鲁棒性仅是被动地维持稳定性，而不可能在对象改变后仍保持原有的期望响应。

现采用自校正DMC控制策略，每隔5 s(相当于$10T$)抽取一次特征，并相应调整模型、控制律和反馈系数，图7-7(b)给出了同样情况下的仿真结果。可以看到，由于引入了自校正，使模型能较好地保持与实际过程的一致。在此基础上调整控制律，

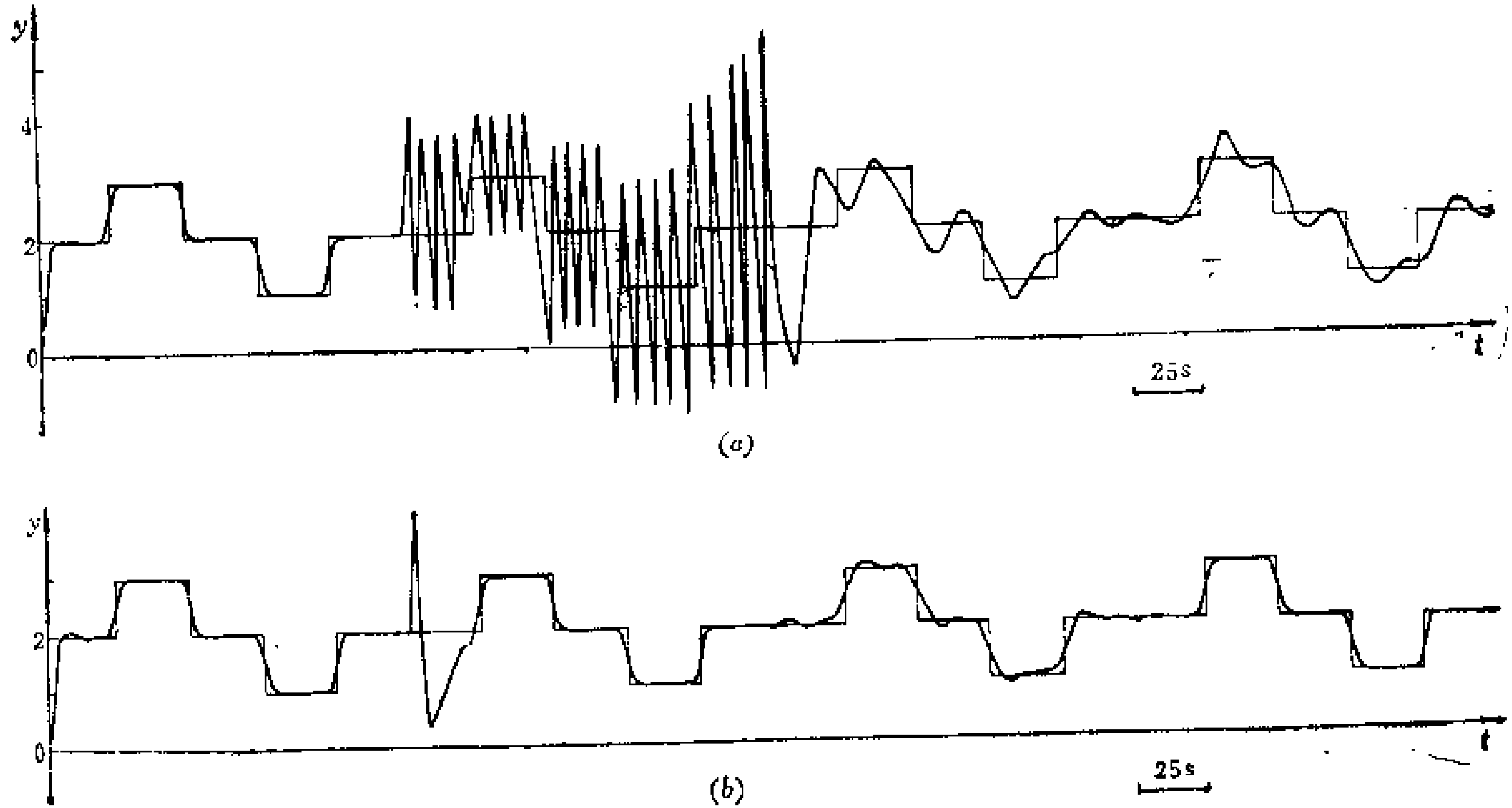


图7-7　无自校正与有自校正的动态矩阵控制比较
（a）基础DMC控制；（b）自校正DMC控制。

因其同时考虑了优化时域的改变，故使系统在整个控制过程中都表现出良好一致的动态响应。与无自校正时相比，明显地增强了对于复杂过程的适应性。

以上介绍的自校正DMC算法，采用了分层的控制结构。其要点在于通过在线辨识尽可能保持模型与对象的匹配，从而增强了控制系统的鲁棒性，使基础DMC控制的设计可集中于对付不可知干扰的影响。这一原理与从自校正控制发展起来的GPC算法是相同的。但在对控制的校正中，这里除了通过模型参数的改变修正控制律外，进一步提出了对优化性能指标的校正，其目的在于保持期望的动态响应，在对象改变时不但具有鲁棒稳定性，而且能保持良好的控制质量。这种主动的适应性需要以一定的解析关系为基础，并且通常是采用启发式方法予以实现的。

自校正控制结构与预测控制基础算法的结合，构成了自校正预测控制，其形式是多种多样的。根据工业现场的实际需要和可能，对模型的辨识与对控制的校正都可采用不同的策略。这里不但需要定量的传统控制理论，而且也需要人工智能中的各种先进方法。

§7.4　高维大系统的递阶预测控制

6.1节介绍的多变量预测控制算法，是一种集中型的控制算法。当对象的输入输出数较多时，其离线计算要对维数很高的矩阵求逆，这对所用计算机的性能提出了高要求。为了便于用低性能的微机系统实现高维系统的预测控制，我们在6.5节中曾提出了分散预测控制算法。其实质是以减少信息量为代价，换取计算的简易性，但同时也降低了控制的最优性。本节中，我们将在保持控制最优性的前提下，采用大系统分解-协调的思想，提出一种高维大系统预测控制的两级递阶算法[25]，并以DMC算法为例加以说明。注意，在这里"高维"是指有较多的输入输出数，而不是通常意义上状态的维数。

为简化讨论计，首先假设对象有相同的输入与输出数，即

$m=p$。根据6.1节介绍的多变量DMC控制，可将其算法归结为下述公式：

预测模型

$$\tilde{y}_{i,PM}(k)=\tilde{y}_{i,P0}(k)+\sum_{j=1}^{m}A_{ij}\Delta u_{j,M}(k),\quad i=1,\cdots,m \tag{7-10}$$

优化性能指标

$$\min J(k)=\sum_{i=1}^{m}\|w_i(k)-\tilde{y}_{i,PM}(k)\|_{Q_i}^2+\sum_{j=1}^{m}\|\Delta u_{j,M}(k)\|_{R_j}^2 \tag{7-11}$$

下一时刻输出预测

$$\tilde{y}_{i,N1}(k)=\tilde{y}_{i,N0}(k)+\sum_{j=1}^{m}a_{ij}\Delta u_j(k),\quad i=1,\cdots,m \tag{7-12}$$

反馈校正

$$\tilde{y}_{i,cor}(k+1)=\tilde{y}_{i,N1}(k)+h_{ii}e_i(k+1),\quad i=1,\cdots,m \tag{7-13}$$

移位设初值

$$\tilde{y}_{i,N0}(k+1)=S\,\tilde{y}_{i,cor}(k+1),\quad i=1,\cdots,m \tag{7-14}$$

在不考虑约束时，根据预测模型式（7-10）求解优化问题式（7-11），可以直接得到解析解

$$\Delta u_M(k)=(A^TQA+R)^{-1}A^TQ[w(k)-\tilde{y}_{P0}(k)] \tag{7-15}$$

即时控制增量可由下式给出

$$\Delta u(k)=D[w(k)-\tilde{y}_{P0}(k)]$$

其中，控制矩阵D可按式（6-8）离线算出。

为了克服计算D时对高维矩阵（A^TQA+R）求逆的困难，

同时又不降低控制的最优性，我们可以保持集中控制对信息的利用方式，但应用大系统分解-协调的思想，把集中的计算分散到多微机系统实现。具体地说，就是放弃离线集中计算$\boldsymbol{D}$，而代之以用分解-协调算法直接在线求解控制作用。由于经分解后每台微机只涉及到较低维数的计算，故其内存和速度都能满足要求。而且，由于采用了多微机并行计算，虽然分解-协调的在线计算量比原来的直接计算有所增加，但仍能满足实时的需要。

为了用分解-协调方法求解滚动优化问题，首先考虑性能指标（7-11）在模型约束（7-10）下的整体优化。为此，组成整体问题的拉格朗日函数

$$L(k)=J(k)+2\sum_{i=1}^{m}\boldsymbol{\lambda}_i^T(k)[\tilde{\boldsymbol{y}}_{i,PM}(k)-\tilde{\boldsymbol{y}}_{i,P0}(k)$$
$$-\sum_{j=1}^{m}\boldsymbol{A}_{ij}\Delta\boldsymbol{u}_{j,M}(k)] \tag{7-16}$$

其中，$\boldsymbol{\lambda}_i^T(k)=[\lambda_i(k+1)\cdots\lambda_i(k+P)](i=1,\cdots,m)$为拉格朗日乘子。这样，原始的整体优化问题就转化为$L(k)$的无约束优化问题。为了使整体计算得到分解，可取$\boldsymbol{\lambda}_i^T(k)(i=1,\cdots,m)$为协调因子，即首先固定

$$\boldsymbol{\lambda}(k)=\hat{\boldsymbol{\lambda}}(k)=[\hat{\boldsymbol{\lambda}}_1^T(k)\cdots\hat{\boldsymbol{\lambda}}_m^T(k)]^T$$

在此基础上求出最优解，再通过协调算法改进$\boldsymbol{\lambda}$，反复迭代，直至满足协调条件得到真正的最优解。这种分解-协调算法可以用一个两级优化算法加以描述：

$$\min_{\Delta u}L(\Delta\boldsymbol{u})\Longleftrightarrow\max_{\hat{\lambda}}\{\min_{\Delta u}L(\Delta\boldsymbol{u},\hat{\boldsymbol{\lambda}})\} \tag{7-17}$$

以下，我们给出每一级优化的详细内容。

第1级　$\min\limits_{\Delta u}L(\Delta\boldsymbol{u},\hat{\boldsymbol{\lambda}})$

这是指在$\hat{\boldsymbol{\lambda}}$相对固定时，求解使$L$最优的$\Delta\boldsymbol{u}$。

通过将式（7-16）改写为

$$L(k)=J(k)+2\sum_{i=1}^{m}\hat{\lambda}_i^T(k)[\tilde{y}_{i,PM}(k)-\tilde{y}_{i,P0}(k)$$

$$-\sum_{j=1}^{m}A_{ij}\Delta u_{j,M}(k)]$$

$$=J(k)+2\sum_{i=1}^{m}\hat{\lambda}_i^T(k)[\tilde{y}_{i,PM}(k)-\tilde{y}_{i,P0}(k)]$$

$$-2\sum_{i=1}^{m}\left(\sum_{j=1}^{m}\hat{\lambda}_j^T(k)A_{ji}\right)\Delta u_{i,M}(k)$$

$$=\sum_{i=1}^{m}L_i(k)$$

其中

$$L_i(k)=\|w_i(k)-\tilde{y}_{i,PM}(k)\|_{Q_i}^2+\|\Delta u_{i,M}(k)\|_{R_i}^2$$

$$+2\hat{\lambda}_i^T(k)[\tilde{y}_{i,PM}(k)-\tilde{y}_{i,P0}(k)]$$

$$-2\left(\sum_{j=1}^{m}\hat{\lambda}_j^T(k)A_{ji}\right)\Delta u_{i,M}(k)$$

可以使 $L_i(k)$ 只与 Δu_i，$\tilde{y}_i$有关，在全部$\tilde{\lambda}_i^T$ 相对固定可视为常量的情况下，对 $L(k)$ 的优化问题便可分解为各 Δu_i 对$L_i(k)$ 的独立优化问题，即

$$\min_{\Delta u}L(k)\Longleftrightarrow\min_{\Delta u_i}L_i(k),\quad i=1,\cdots,m$$

这样，就可以用多台微机独立地求解规模较小的 $L_i(k)$ 的优化子问题。对于第 i 个问题，根据极值必要条件可写出

$$\frac{\partial L_i(k)}{\partial \tilde{y}_{i,PM}(k)}=2\{-Q_i[w_i(k)-\tilde{y}_{i,PM}(k)]+\hat{\lambda}_i(k)\}=0$$

$$\frac{\partial L_i(k)}{\partial \Delta u_{i,M}(k)}=2\left\{R_i\Delta u_{i,M}(k)-\left(\sum_{j=1}^{m}A_{ji}^T\hat{\lambda}_j(k)\right)\right\}=0$$

由此可求出

$$\left.\begin{aligned}\widetilde{\boldsymbol{y}}_{i,PM}(k)&=\boldsymbol{w}_i(k)-\boldsymbol{Q}_i^{-1}\hat{\boldsymbol{\lambda}}_i(k)\\ \Delta\boldsymbol{u}_{i,M}(k)&=\boldsymbol{R}_i^{-1}\sum_{j=1}^{m}\boldsymbol{A}_{ji}^{T}\hat{\boldsymbol{\lambda}}_j(k)\end{aligned}\right\}\tag{7-18}$$

因此，对于第 i 个优化子问题，只需根据已知的 $\boldsymbol{A}_{ji}$，$\boldsymbol{Q}_i$、$\boldsymbol{R}_i$ 及给定的$\hat{\boldsymbol{\lambda}}(k)$ 计算式（7-18）即可，这可由多微机系统中的第 i 台微机实现，并且所有 m 个子问题的优化解（7-18）可以并行计算。注意为了防止出现奇异解，在采用递阶算法时要求对角权矩阵 $\boldsymbol{Q}>0$，$\boldsymbol{R}>0$。

第 2 级 $\max\limits_{\hat{\lambda}}\varphi(\hat{\boldsymbol{\lambda}})$，其中 $\varphi(\hat{\boldsymbol{\lambda}})=\min\limits_{\Delta u}L(\Delta\boldsymbol{u},\ \hat{\boldsymbol{\lambda}})$

通过第 1 级优化所得到的 $\varphi(\hat{\boldsymbol{\lambda}})=\min\limits_{\Delta u}L(\Delta\boldsymbol{u},\ \hat{\boldsymbol{\lambda}})$，并不是 $L(k)$ 的最优值，因为它是在$\hat{\boldsymbol{\lambda}}$相对固定的情况下求出的。所以，第 2 级即协调级的任务在于修改$\hat{\boldsymbol{\lambda}}$ 的值，使之接近于满足全局最优的 $\boldsymbol{\lambda}$ 的值。这可以通过下述梯度算法实现

$$\hat{\boldsymbol{\lambda}}_i^{l+1}(k)=\hat{\boldsymbol{\lambda}}_i^{l}(k)+\alpha(k)\frac{\partial L(k)}{\partial\hat{\boldsymbol{\lambda}}_i^{l}(k)}\tag{7-19}$$

其中，$l=0$，1，…为迭代次数，$\alpha(k)$ 为迭代步长，梯度向量 $\partial L(k)/\partial\hat{\boldsymbol{\lambda}}_i^{l}(k)$ 可由式（7-16）给出为

$$\frac{\partial L(k)}{\partial\hat{\boldsymbol{\lambda}}_i^{l}(k)}=\widetilde{\boldsymbol{y}}_{i,PM}^{l}(k)-\boldsymbol{y}_{i,P0}(k)-\sum_{j=1}^{m}\boldsymbol{A}_{ij}\Delta\boldsymbol{u}_{j,M}^{l}(k)\tag{7-20}$$

由于$\widetilde{\boldsymbol{y}}_{i,P0}(k)$ 是已知的，$\widetilde{\boldsymbol{y}}_{i,PM}^{l}(k)$ 及 $\Delta\boldsymbol{u}_{j,M}^{l}(k)$ 均已在第 1 级算出，所以很容易算出 $\boldsymbol{\lambda}$ 的新值，这可由承担协调任务的协调级微机进行。这一过程可反复进行下去，直至

$$\|\hat{\boldsymbol{\lambda}}_i^{l+1}(k)-\hat{\boldsymbol{\lambda}}_i^{l}(k)\|<\varepsilon,\quad i=1,\ \cdots,\ m\tag{7-21}$$

这里，ε 是事先给定的充分小的正数。这时，可认为 $\boldsymbol{\lambda}$ 已达到了其最优值，相应 $\Delta\boldsymbol{u}_{i,M}^{l}(k)$ 中的第一个元素 $\Delta u_i^{l}(k)$ 便可用于构成实际控制作用。

上述采用分解-协调原理的两级递阶 DMC 算法具体步骤

如下：

第 1 步　m 台基础级微机分别检测对应的实际输出 $y_i(k)$，构成误差 $e_i(k)$，并按式（7-13）、式（7-14）进行误差校正和移位，构成预测初值 $\tilde{y}_{i,P0}(k)$，并经通信将其送至协调级微机。

第 2 步　协调级微机给出 λ 的初值 $\hat{\lambda}^0(k)$，经通信传送到各基础级微机。

第 3 步　各基础级微机按式（7-18）计算 $\Delta u^l_{i,M}(k)$ 及 $\tilde{y}^l_{i,PM}(k)$，并将计算结果传送到协调级微机。

第 4 步　协调级微机按式（7-19）、式（7-20）计算 λ 的新值 $\hat{\lambda}^{l+1}(k)$。

第 5 步　协调级微机检验条件（7-21）是否成立。如不成立，则将 $\hat{\lambda}^{l+1}(k)$ 传送至各基础级微机，并返回第 3 步重新计算。否则，将 $\Delta u^l(k)$ 传送至各基础级微机。

第 6 步　各基础级微机计算 $u_i(k)=u_i(k-1)+\Delta u^l_i(k)$，实施控制，并按式（7-12）计算一步预测值 $\tilde{y}_{i,N1}(k)$，为下一时刻的优化控制作准备。

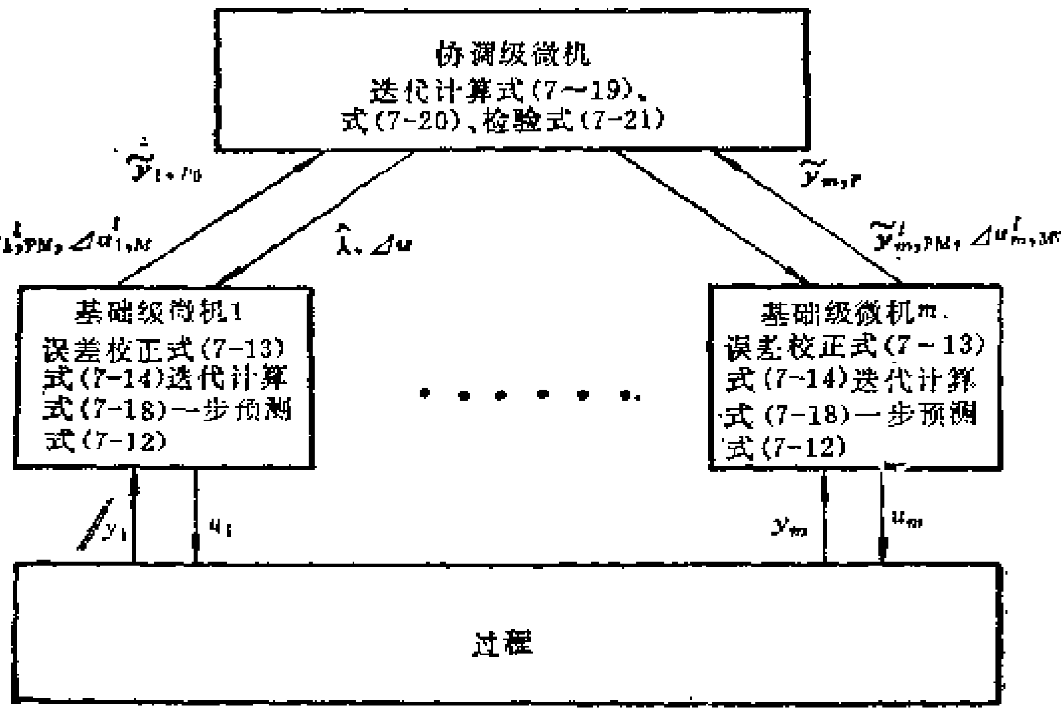


图7-8　递阶动态矩阵控制算法

可以看出，无论是基础级微机或协调级微机，其计算所涉及的向量和矩阵数都要比集中计算时少得多，而且它们的在线计算都比较简单，因而在内存和速度上对计算机的性能没有过高的要求。特别是这一算法以增加在线计算为代价，避免了任何求逆运算，在高维情况下有一定的优越性。图 7-8 画出了这种分解-协调算法的信息结构。

下面，我们讨论采用递阶算法的最优解与集中计算的最优解之间的关系。可以证明，只要迭代过程收敛，通过迭代算得的最优解 $\Delta \boldsymbol{u}_M(k)$ 就是多变量 DMC 算法所得到的最优解 (7-15)。因为在迭代结束时，条件 (7-21) 意味着

$$\frac{\partial L(k)}{\partial \hat{\boldsymbol{\lambda}}_i^l(k)}=0$$

由式 (7-20) 可知

$$\widetilde{\boldsymbol{y}}_{i,PM}^l(k)-\widetilde{\boldsymbol{y}}_{i,P0}(k)-\sum_{j=1}^{m} \boldsymbol{A}_{ij}\Delta \boldsymbol{u}_{j,M}^l(k)=0$$

或把它写成整体形式

$$\widetilde{\boldsymbol{y}}_{PM}^l(k)=\widetilde{\boldsymbol{y}}_{P0}(k)+\boldsymbol{A}\Delta \boldsymbol{u}_M^l(k)$$

此外，把式 (7-18) 也写成整体形式

$$\widetilde{\boldsymbol{y}}_{PM}^l(k)=\boldsymbol{w}(k)-\boldsymbol{Q}^{-1}\hat{\boldsymbol{\lambda}}^l(k)$$

$$\Delta \boldsymbol{u}_M^l(k)=\boldsymbol{R}^{-1}\boldsymbol{A}^T\hat{\boldsymbol{\lambda}}^l(k)$$

由此消去 $\hat{\boldsymbol{\lambda}}^l(k)$、$\widetilde{\boldsymbol{y}}_{PM}^l(k)$，可以得到

$$\Delta \boldsymbol{u}_M^l(k)=(\boldsymbol{A}^T\boldsymbol{Q}\boldsymbol{A}+\boldsymbol{R})^{-1}\boldsymbol{A}^T\boldsymbol{Q}[\boldsymbol{w}(k)-\widetilde{\boldsymbol{y}}_{P0}(k)]$$

对比式 (7-15)，可知 $\Delta \boldsymbol{u}_M^l(k)=\Delta \boldsymbol{u}_M(k)$，即由递阶算法所得的控制作用与集中算法所得的结果完全一致。这表明，采用递阶算法，保持了解的最优性，只是把集中算法的解析解通过另一种途径实现而已。

在上述递阶算法中，十分关键的问题是迭代步长 $\alpha(k)$ 的选取，它涉及到算法是否收敛以及以多快的速度收敛。由迭代式 (7-19)、式 (7-20) 以及式 (7-18) 可得

$$\hat{\lambda}^{l+1}(k)=\hat{\lambda}^{l}(k)+\alpha(k)[\tilde{y}_{PM}^{l}(k)-\tilde{y}_{P0}(k)-A\Delta u_{M}^{l}(k)]$$
$$=\hat{\lambda}^{l}(k)+\alpha(k)[w(k)-Q^{-1}\hat{\lambda}^{l}(k)-\tilde{y}_{P0}(k)$$
$$-AR^{-1}A^{T}\hat{\lambda}^{l}(k)]$$
$$=[I-\alpha(k)(Q^{-1}+AR^{-1}A^{T})]\hat{\lambda}^{l}(k)$$
$$+\alpha(k)[w(k)-\tilde{y}_{P0}(k)]$$

记$S=Q^{-1}+AR^{-1}A^{T}$，要使迭代收敛，必须有

$$\max_i|\sigma_i(I-\alpha(k)S)|<1$$

其中，$\sigma_i(\cdot)$ 表示矩阵的第 i 个特征值。由此式可导出满足收敛条件的迭代步长的范围

$$0<\alpha(k)<\frac{2}{\max\limits_i\sigma_i(S)} \tag{7-22}$$

特别当$Q=qI>0$，$R=rI>0$时，有

$$\sigma(S)=\frac{1}{q}+\frac{1}{r}\sigma(AA^{T})$$

由于 $AA^{T}\geqslant0$，$\sigma(AA^{T})\geqslant0$，故

$$\max_i\sigma_i(AA^{T})=\max_i\sigma_i(A^{T}A)$$

由此可得

$$0<\alpha(k)<\frac{2}{\frac{1}{q}+\frac{1}{r}\max\limits_i\sigma_i(A^{T}A)} \tag{7-23}$$

这从理论上给出了使算法收敛的迭代步长范围。在实际运算时，由于计算$\sigma(A^{T}A)$ 不现实，故通常可用试探法寻找$\alpha(k)$。

本节介绍的递阶预测控制算法，不是离线求出最优解的解析形式，而是在线通过迭代算法直接求解优化问题。这样，在线计算量与集中算法相比将有所增加。但它避免了高维矩阵求逆时对计算机内存和精度的要求。同时，因为采用了多机并行计算，在线迭代也能较快地完成。由于计算分散化，对每台计算机的性能要求远远低于集中计算，因而在缺乏高性能计算机时是一种可取的控制方案。特别在考虑约束的情况下，由于集中计算也没有解

析解，必须在线优化，其在线计算简易的优点不复存在，故采用这种分解-协调算法的优越性将更为突出。这时，只需把约束同样分解到子问题中，整个算法并无本质的改变。

§7.5 利用无穷范数优化的鲁棒预测控制

以上介绍的预测控制算法，都采用了二次型的优化性能指标，它对应于预测偏差的二范数，反映了对于输出值良好跟踪期望值的性能要求。但在许多工业过程中，控制的要求并不侧重于跟踪，而在于良好的镇定性能，即所有输出值都应保持在设定值附近，不希望出现过大的偏差。在这种情况下，用以刻划控制要求的更直接的数学度量是要极小化所可能出现的最大偏差，即

$$\min_{\Delta u_M(k)} \max_{l=1,\cdots,P} \max_{i=1,\cdots,p} |w_i(k+l)-\tilde{y}_i(k+l\mid k)| \tag{7-24}$$

它相当于把二次型性能指标（6-5）推广为无穷范数的形式

$$\min J(k)=\|\boldsymbol{w}(k)-\tilde{\boldsymbol{y}}_{PM}(k)\|_\infty \tag{7-25}$$

其中，向量 $\boldsymbol{x}=[x_1\cdots x_n]^T$ 的无穷范数定义为

$$\|\boldsymbol{x}\|_\infty=\max_i |x_i|$$

当式（7-25）中的无穷范数对空间（对全部输出量）和时间（对优化时域中的全部时刻）双重取值时，其具体表达就是式(7-24)。显然，这一优化要使优化时域中所有输出的最大可能偏差取极小，因而直接表达了良好镇定的物理要求。

由于性能指标（7-24）具有线性形式，如果对象模型与物理约束也是线性的，则这一优化问题可采用线性规划（LP）方法求解。然而，由于式（7-24）中涉及到极值及绝对值项，它与标准LP问题的形式尚存在一定距离，所以还要运用一些技巧将其转化为标准LP问题。以下，我们介绍文献[26]中对这类问题的求解方法及其在参数有摄动时设计鲁棒预测控制器中的应用。

设有 m 个输入、p 个输出的对象模型可由 $p\times m$ 维脉冲响应矩阵序列 $\boldsymbol{H}=\{\boldsymbol{H}_i,\ i=1,\cdots,N\}$ 给出，其中 N 为模型长度，

即在 $i>N$ 时可认为 $\boldsymbol{H}_i\approx\boldsymbol{0}$。类似于3.2节中MAC算法的多步闭环预测，推广到这里的多变量情况，可写出在 k 时刻起 M 个控制量 $\boldsymbol{u}(k)$，$\boldsymbol{u}(k+1)$，…，$\boldsymbol{u}(k+M-1)$ 作用下对未来 P 步输出的闭环预测

$$\tilde{\boldsymbol{y}}(k)=\boldsymbol{G}_1\boldsymbol{u}_1(k)+\boldsymbol{G}_2\boldsymbol{u}_2(k)+\tilde{\boldsymbol{e}}(k) \tag{7-26}$$

其中

$$\tilde{\boldsymbol{y}}(k)=\begin{bmatrix}\tilde{\boldsymbol{y}}(k+1|k)\\ \vdots\\ \tilde{\boldsymbol{y}}(k+P|k)\end{bmatrix},\quad \boldsymbol{u}_1(k)=\begin{bmatrix}\boldsymbol{u}(k)\\ \vdots\\ \boldsymbol{u}(k+M-1)\end{bmatrix}$$

$$\boldsymbol{u}_2(k)=\begin{bmatrix}\boldsymbol{u}(k-1)\\ \vdots\\ \boldsymbol{u}(k-N+1)\end{bmatrix}$$

$$\boldsymbol{G}_1=\begin{bmatrix}\boldsymbol{H}_1 & & \boldsymbol{0}\\ \vdots & \ddots & \boldsymbol{H}_1\\ & & \vdots\\ \boldsymbol{H}_P & \cdots & \boldsymbol{H}_1+\cdots+\boldsymbol{H}_{P-M+1}\end{bmatrix},\quad \boldsymbol{G}_2=\begin{bmatrix}\boldsymbol{H}_2 & \cdots & \boldsymbol{H}_N\\ \vdots & ⋰ & \\ \vdots & & \boldsymbol{0}\\ \boldsymbol{H}_{P+1}\cdots\boldsymbol{H}_N & & \end{bmatrix}$$

$$\tilde{\boldsymbol{e}}(k)=\begin{bmatrix}\tilde{\boldsymbol{e}}(k+1|k)\\ \vdots\\ \tilde{\boldsymbol{e}}(k+P|k)\end{bmatrix}$$

$$\tilde{\boldsymbol{e}}(k+l|k)=\boldsymbol{y}(k)-\sum_{i=1}^{N}\boldsymbol{H}_i\boldsymbol{u}(k-i),\quad l=1,\cdots,P$$

设在 k 时刻的滚动优化问题是在对输入输出有约束情况下的无穷范数优化问题

$$\left.\begin{aligned}&\min_{\boldsymbol{u}_1}\|\boldsymbol{w}(k)-\tilde{\boldsymbol{y}}(k)\|_\infty\\ &=\min_{\boldsymbol{u}_1}\max_{l}\|\boldsymbol{w}(k+l)-\tilde{\boldsymbol{y}}(k+l|k)\|_\infty\\ &\text{s.t.}\qquad \boldsymbol{\alpha}\leqslant\boldsymbol{u}_1\leqslant\boldsymbol{\beta}\\ &\boldsymbol{c}(k+l)\leqslant\tilde{\boldsymbol{y}}(k+l|k)\leqslant\boldsymbol{d}(k+l)\\ &\qquad l=1,\cdots,P\end{aligned}\right\} \tag{7-27}$$

引入记号

$$\boldsymbol{f}(\boldsymbol{u}_1)=\boldsymbol{G}_1\boldsymbol{u}_1(k)+\boldsymbol{G}_2\boldsymbol{u}_2(k)-\boldsymbol{s}(k)$$

其中，$\boldsymbol{s}(k)=\boldsymbol{w}(k)-\tilde{\boldsymbol{e}}(k)$，则结合预测模型(7-26)，可将

优化问题（7-27）改写为

$$\left.\begin{aligned}&\min_{u_1}\ \max_i |f_i(\boldsymbol{u}_1)|\\&\text{s.t.}\quad \boldsymbol{\alpha}\leqslant\boldsymbol{u}_1\leqslant\boldsymbol{\beta}\\&\qquad\quad \boldsymbol{\gamma}\leqslant\boldsymbol{f}(\boldsymbol{u}_1)\leqslant\boldsymbol{\delta}\end{aligned}\right\}\tag{7-28}$$

其中，$f_i(\boldsymbol{u}_1)$ 是 $\boldsymbol{f}(\boldsymbol{u}_1)$ 的第 i 个分量

$$\boldsymbol{\gamma}=\begin{bmatrix}c(k+1)-w(k+1)\\ \vdots\\ c(k+P)-w(k+P)\end{bmatrix},\ \boldsymbol{\delta}=\begin{bmatrix}\boldsymbol{d}(k+1)-w(k+1)\\ \vdots\\ \boldsymbol{d}(k+P)-w(k+P)\end{bmatrix}$$

为了把式（7-28）转化为标准 LP 问题，定义

$$\mu(\boldsymbol{u}_1)=\max_i |f_i(\boldsymbol{u}_1)|$$

显然，这一最优 μ^* 值是满足下式的最小 μ，

$$\boldsymbol{l}\mu\geqslant\boldsymbol{f}(\boldsymbol{u}_1)$$
$$-\boldsymbol{l}\mu\leqslant\boldsymbol{f}(\boldsymbol{u}_1)$$

其中 $\boldsymbol{l}=[1\ \cdots\ 1]^T$。这样，优化问题（7-28）可进一步转化为

$$\left.\begin{aligned}&\min_{u_1,\mu}\ \mu\\&\text{s.t.}\quad \boldsymbol{l}\mu\geqslant\boldsymbol{f}(\boldsymbol{u}_1),\ -\boldsymbol{l}\mu\leqslant\boldsymbol{f}(\boldsymbol{u}_1)\\&\qquad\quad \boldsymbol{\gamma}\leqslant\boldsymbol{f}(\boldsymbol{u}_1),\ \boldsymbol{\delta}\geqslant\boldsymbol{f}(\boldsymbol{u}_1)\\&\qquad\quad \boldsymbol{\alpha}\leqslant\boldsymbol{u}_1,\ \boldsymbol{\beta}\geqslant\boldsymbol{u}_1\end{aligned}\right\}\tag{7-29}$$

由于 $\boldsymbol{f}(\boldsymbol{u}_1)$ 具有 $\boldsymbol{u}_1$ 的线性形式，上述问题便成为一个以 μ，$\boldsymbol{u}(k)$，…，$\boldsymbol{u}(k+M-1)$ 为优化变量、具有线性目标函数和线性不等式约束的典型 LP 问题。它可进一步化作规范化形式

$$\left.\begin{aligned}&\min_{x}\ \boldsymbol{c}^T\boldsymbol{x}\\&\text{s.t.}\quad \boldsymbol{A}\boldsymbol{x}\geqslant\boldsymbol{b}\\&\qquad\quad \boldsymbol{x}\geqslant\boldsymbol{0}\end{aligned}\right\}\tag{7-30}$$

并采用单纯形法来求解。其中，$\boldsymbol{A}$ 为 $(4Pp+Mm)\times(Mm+1)$ 维矩阵，$\boldsymbol{x}$，$\boldsymbol{c}$ 均为 $Mm+1$ 维向量，$\boldsymbol{b}$ 为 $4Pp+Mm$ 维向量。

在用单纯形法求解 LP 问题时，迭代次数通常介于 $m+n$ 与 $2(m+n)$ 之间，其中 m 为约束数，n 为优化变量个数，而每次单纯形迭代需要 m^2 数量级的运算。注意到上述 LP 问题的约

束数 $4Pp+Mm$ 远大于优化变量数 $Mm+1$，为了简化计算，可首先求解与式（7-30）对偶的LP问题

$$\left.\begin{array}{ll} & \min\limits_{y} \ -\boldsymbol{b}'\boldsymbol{y} \\ \text{s.t.} & \boldsymbol{A}'\boldsymbol{y}\leqslant \boldsymbol{c} \\ & \boldsymbol{y}\geqslant \boldsymbol{0} \end{array}\right\} \tag{7-31}$$

它的约束数远小于原问题，因而可大大减少寻优计算。此外，注意到在上述表达式中只有向量 $\boldsymbol{b}$ 涉及到 $w(k)$、$u_2(k)$、$\widetilde{e}(k)$，而其他向量和矩阵都是时不变的，所以采用对偶形式(7-31）后，约束便成为时不变的。这样，在 k 时刻优化所得到的解可立即作为 $k+1$ 时刻优化的一个初始可行解。在每一时刻优化时，就不必再去寻找一个基础可行解。

在求出对偶LP问题的最优解后，原问题的最优解可由此根据互补松弛条件还原算出。单纯形法计算及用互补松弛条件还原计算原问题的最优解，都是LP问题中的典型内容，对此可参考有关书籍，在此不再介绍。在实用中，由于求解LP问题已有大量程序包，在线实现这一优化是毫不困难的。

上述具有无穷范数性能指标的预测控制算法，需要在线求解优化问题，其计算量当然要大于二次型性能指标优化时解析解的计算。但是，如果考虑了对输入输出的约束，后者的解析解就不再有效，而需在线求解二次规划问题，这时，本节中的线性规划方法便明显地具有计算上的优点。根据无穷范数的意义，这种优化控制将使输出偏差控制在一个充分小的范围内，这对于维持平稳的工况是十分重要的。

下面，我们利用无穷范数优化预测控制的算法和思路，进一步将其扩展到在有参数摄动时鲁棒预测控制器的设计问题。为此，假定对象的脉冲响应模型 $\boldsymbol{H}$ 依赖于参数 $\boldsymbol{\theta}$，记为 $\boldsymbol{H}=\boldsymbol{H}(\boldsymbol{\theta})$。为简化计，设参数标称值 $\boldsymbol{\theta}_0=\boldsymbol{0}$，并将参数的变化空间记作

$$\pi=\{\boldsymbol{\theta}\,|\,|\theta_j|<\varepsilon_j,\quad j=1,\ \cdots,\ q\}$$

其中，q 为参数的数目。显然，π 是 $\boldsymbol{\theta}$ 空间中以 $\boldsymbol{\theta}_0=\boldsymbol{0}$ 为中心的一个多面体。对于参数空间 π 中的每一点 $\boldsymbol{\theta}$，都可得到相应的

脉冲响应模型$H(\theta)$。这样，当允许参数在π空间摄动时，对象的模型也构成了一个模型族

$$\Pi=\{H(\theta)|\theta\in\pi\}$$

显然，由于参数空间π有无穷多个点，这一模型集合Π是无限维的，而标称模型$H(0)$只是集合中的一个元素。所谓鲁棒预测控制，就是要通过前面所述的无穷范数优化的预测控制，不但使系统对于标称模型$H(0)$的最大输出偏差取极小，而且当θ在参数空间π内变化时，要对所有可能的最大输出偏差取极小。这相当于无穷范数不但要对不同变量及不同时刻，而且要对不同参数θ导致的不同模型取值。在这种情况下，无穷范数优化问题（7-27）可进一步增广为下述形式：

$$\left.\begin{array}{l}\min\limits_{u_1}\max\limits_{\theta\in\pi}\max\limits_{l}\|w(k+l)-\tilde{y}(k+l|k)\|_\infty\\ \text{s.t.}\qquad \alpha\leqslant u_1\leqslant\beta\\ \quad c(k+l)\leqslant\tilde{y}(k+l|k)\leqslant d(k+l)\\ \qquad\qquad l=1,\cdots,P\end{array}\right\}\quad(7\text{-}32)$$

注意到参数存在不确定性时，由于模型$H(\theta)$依赖于参数θ，故所有涉及到模型预测的项也将依赖于θ。类似于前面的推导，可把这时的滚动优化问题描述为

$$\begin{array}{l}\min\limits_{u_1}\max\limits_{\theta\in\pi}\max\limits_{i}|f(\theta,u)|\\ \quad\text{t.}\qquad \alpha\leqslant u_1\leqslant\beta\\ \qquad\qquad \gamma\leqslant f(\theta,u_1)\leqslant\delta,\ \forall\theta\in\pi\end{array}$$

其中

$$f(\theta,u_1)=G_1(\theta)u_1(k)+G_2(\theta)u_2(k)-s(k)$$

$G_1(\theta)$、$G_2(\theta)$对应于式（7-26）中的G_1、G_2，但其中所有H_i均为$H_i(\theta)$代替，以表示其依赖于参数θ，而$s(k)$的组成形式同前，只是在构成$\tilde{e}(k)$时，用标称模型的响应矩阵$H_i(0)$作为式中的H_i。

这一优化问题可进一步转化为类似于式（7-29）的形式

$$
\begin{aligned}
&\min_{\mu, \boldsymbol{u}_1} \quad \mu \\
&\text{s.t.} \quad \left.\begin{array}{l} l\mu \geqslant \boldsymbol{f}(\boldsymbol{\theta}, \boldsymbol{u}_1), \ -l\mu \leqslant \boldsymbol{f}(\boldsymbol{\theta}, \boldsymbol{u}_1) \\ \gamma \leqslant \boldsymbol{f}(\boldsymbol{\theta}, \boldsymbol{u}_1), \ \delta \geqslant \boldsymbol{f}(\boldsymbol{\theta}, \boldsymbol{u}_1) \end{array}\right\} \forall \boldsymbol{\theta} \in \pi \\
&\qquad\qquad \alpha \leqslant \boldsymbol{u}_1, \qquad \beta \geqslant \boldsymbol{u}_1
\end{aligned}
$$

它与式（7-29）不同之处在于，由于 $\boldsymbol{H}_i$ 与参数 $\boldsymbol{\theta}$ 间的关系是以一般的形式给出的，并且 $\boldsymbol{\theta}$ 可在空间 π 中任意取值，上述约束条件可以是非线性的且有无限多个，因而很难给出一般的求解算法。但是，如果矩阵 $\boldsymbol{H}_i(\boldsymbol{\theta})$ 是 $\boldsymbol{\theta}$ 的仿射函数，则可推出 $f_i(\boldsymbol{\theta}, \boldsymbol{u}_1)$ 也是 $\boldsymbol{\theta}$ 的仿射函数，这时可证明 $f_i(\boldsymbol{\theta}, \boldsymbol{u}_1)$ 将在 π 的边界端点上达到其极值。我们把这些边界端点的集合记作 $\boldsymbol{S}$，它包含了多面体 π 的 2^q 个顶点。这时，上述有无限多个约束的优化问题就可简化为有限个约束的优化问题

$$
\begin{aligned}
&\min_{\mu, \boldsymbol{u}_1} \quad \mu \\
&\text{s.t.} \quad \left.\begin{array}{l} l\mu \geqslant \boldsymbol{f}(\boldsymbol{\theta}, \boldsymbol{u}_1), \ -l\mu \leqslant \boldsymbol{f}(\boldsymbol{\theta}, \boldsymbol{u}_1) \\ \gamma \leqslant \boldsymbol{f}(\boldsymbol{\theta}, \boldsymbol{u}_1), \ \delta \geqslant \boldsymbol{f}(\boldsymbol{\theta}, \boldsymbol{u}_1) \end{array}\right\} \forall \boldsymbol{\theta} \in \boldsymbol{S} \qquad (7\text{-}33) \\
&\qquad\qquad \alpha \leqslant \boldsymbol{u}_1, \qquad \beta \geqslant \boldsymbol{u}_1
\end{aligned}
$$

对 $\boldsymbol{S}$ 的 2^q 个元素中的每一个写出约束（7-33）的具体形式，最后可把上述优化问题写成式（7-30）的规范 LP 问题，其中与约束数对应的向量 $\boldsymbol{b}$ 的维数为 $2^{q+2}pP\times mM$。这一有庞大约束数的 LP 问题同样可通过求解其对偶问题得到简化。

$\boldsymbol{H}_i(\boldsymbol{\theta})$ 是 $\boldsymbol{\theta}$ 的仿射函数意味着 $\boldsymbol{H}_i$ 是已知线性定常对象通过 $\boldsymbol{\theta}$ 加权的任意线性组合，例如，它可取如下形式：

$$
\boldsymbol{H}_i(\boldsymbol{\theta}) = \boldsymbol{H}_i(\boldsymbol{0}) + \sum_{j=1}^{q} \theta_j \boldsymbol{H}_i^j, \quad i = 1, \cdots, N
$$

其中，$\boldsymbol{H}_i(\boldsymbol{0})$、$\boldsymbol{H}_i^j(j=1, \cdots, q)$ 均为已知的 $p\times m$ 维常值阵。由于 θ_j 为摄动参数，上式表示在标称响应 $\boldsymbol{H}(\boldsymbol{0})$ 的基础上，考虑了其他未建模动态 $\boldsymbol{H}^j$ 的扰动。按照上述方法设计的预测控制器，可以在有扰动的情况下，保持良好的鲁棒性。

〔例〕 设对象的标称传递函数为

$$P_0(s) = \frac{1}{10s+1}$$

现在要设计预测控制器，使系统在有未建模动态干扰时有良好的鲁棒性。这一未建模动态的传递函数为

$$P'(s) = \frac{1}{10s+1}\left(\frac{1-2s}{1+2s}\right)$$

所考虑的模型集由下式给出

$$\Pi = \{P(s) | P(s) = P_0(s) + \theta(P'(s) - P_0(s)), \quad 0 \leqslant \theta \leqslant 1\}$$

它可用脉冲响应形式表示为

$$\Pi = \{H(\theta) | H(\theta) = H_0 + \theta(H' - H_0), \quad 0 \leqslant \theta \leqslant 1\}$$

其中，H_0、H' 分别为 $P_0(s)$ 和 $P'(s)$ 的脉冲响应序列。

选择参考轨线为低通滤波形式

$$F(s) = \frac{1}{2.5s+1}$$

以确保闭环稳定性，采用上述无穷范数性能指标的优化方法，由图 7-9(a) 可以看出，通常的MAC算法在模型精确时（$\theta=0$）可得到很好的闭环响应，但当实际对象有扰动时（$\theta=1$），将出现大幅度振荡和超调。而采用无穷范数优化的MAC算法（见图 7-9(b)），在模型精确时（$\theta=0$）闭环响应良好，当存在扰动时（$\theta=1$）亦能获得良好的镇定作用，表现出优良的鲁棒性能。

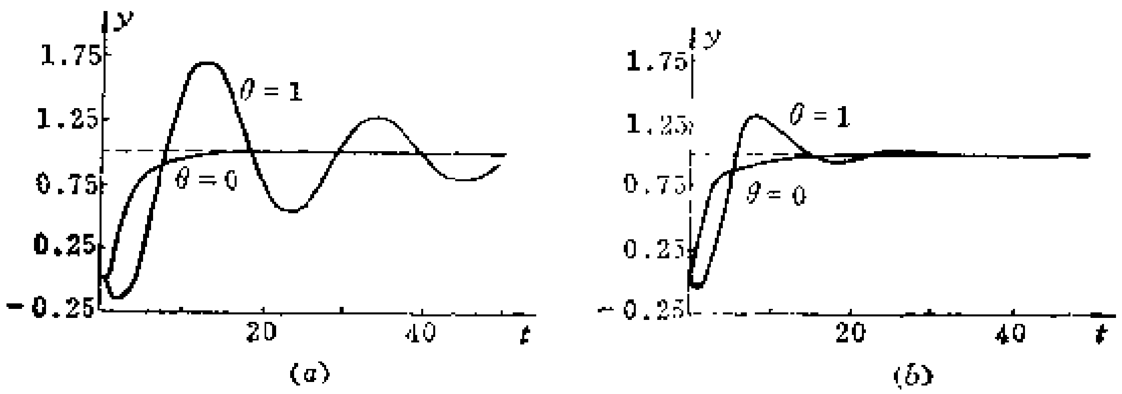


图7-9　不同性能指标预测控制的比较

(a) 二次型指标MAC；(b) 无穷范数指标MAC。

上述简单的例子仅仅给出了单变量情况下示例性的说明。实际上，采用无穷范数优化的意义在多变量情况下将更为突出，并且由于考虑了模型的不确定性，它对于约束条件的满足也比通常只基于标称模型的预测控制更能得到保证，因而它十分适用于存在输入输出约束和模型不确定性时把过程保持在工况条件的控制。

§7.6 非线性系统的预测控制

预测控制算法最初是针对线性系统提出的。在第三章所介绍的几种典型算法中，都用到了对象的线性模型，并通过线性系统的比例和叠加性质对未来输出进行预测。在对象只存在弱非线性时，采用这种线性预测控制算法是十分有效的，因为弱非线性可视为一种模型失配，其影响可通过系统的鲁棒性设计自然加以克服。必要时，还可通过在线辨识和自校正策略修改模型和控制律，以适应因弱非线性而引起的对象特性的变化。然而，当对象有强非线性时，由于采用线性模型的输出预测与实际偏离较大，达不到优化控制的目的，因而必须基于非线性模型进行预测和优化。从第二章预测控制的基本原理可知，它同样也适用于非线性对象。下面就从这些原理出发，来探讨非线性系统的预测控制问题。

设非线性系统的模型由下式给出：

$$\left.\begin{aligned} \boldsymbol{x}(k+1)&=\boldsymbol{f}(\boldsymbol{x}(k),\ \boldsymbol{u}(k)) \\ \boldsymbol{y}(k)&=\boldsymbol{g}(\boldsymbol{x}(k)) \end{aligned}\right\} \tag{7-34}$$

其中，$\boldsymbol{x}\in R^n$，$\boldsymbol{u}\in R^m$，$\boldsymbol{y}\in R^p$。根据这一模型，在k时刻只要知道了对象的初始状态$\boldsymbol{x}(k)$及其在未来的控制输入$\boldsymbol{u}(k)$，$\boldsymbol{u}(k+1)$，…，便可预测对象在未来各时刻的模型输出$\tilde{\boldsymbol{y}}_M$

$$\left.\begin{aligned} &\tilde{\boldsymbol{x}}(k+i\mid k)=\boldsymbol{f}(\tilde{\boldsymbol{x}}(k+i-1\mid k),\ \boldsymbol{u}(k+i-1)) \\ &\tilde{\boldsymbol{y}}_M(k+i\mid k)=\boldsymbol{g}(\tilde{\boldsymbol{x}}(k+i\mid k))\tilde{\boldsymbol{x}}(k\mid k)=\boldsymbol{x}(k) \\ &\qquad i=1,\ 2,\ \cdots \end{aligned}\right\} \tag{7-35}$$

通过递推关系，可以得到

$$\tilde{y}_M(k+i|k)=F_i[x(k),\ u(k),\ \cdots,\ u(k+i-1)]$$
$$i=1,\ 2,\ \cdots \tag{7-36}$$

其中，$F_i(\cdot)$ 是由$f(\cdot)$ 及$g(\cdot)$ 复合并组合而成的非线性函数。式（7-36）就是非线性系统的预测模型。

由于实际受控系统总包含有某些不确定因素，利用上述模型预测，不能完全精确地描述对象的动态行为，因此可在实测输出的基础上通过误差预测和补偿对模型预测进行反馈校正。记k时刻测得的实际输出为$y(k)$，则可由

$$e(k)=y(k)-\tilde{y}_M(k|k-1)$$

构成预测误差，并根据历史的误差信息$e(k)$，…，$e(k-q)$通过误差预测

$$\tilde{e}(k+i|k)=E_i[e(k),\ \cdots,\ e(k-q)] \tag{7-37}$$

校正基于模型的预测，并构成对输出的闭环预测

$$\tilde{y}_P(k+i|k)=\tilde{y}_M(k+i|k)+\tilde{e}(k+i|k) \tag{7-38}$$

其中，$E_i(\cdot)$ 为某一线性或非线性函数，其形式取决于所用的非因果预测方法，q为所用到的历史误差信息长度。式（7-38）就是在模型（7-36）基础上带有反馈校正（7-37）的闭环预测。

在k时刻，控制的目的是要求出该时刻起的M个控制量$u(k)$，…，$u(k+M-1)$(假设u在$k+M-1$时刻后保持不变)，使输出的某一性能指标最优

$$\min J(k)=F(\tilde{y}_P(k),\ w(k)) \tag{7-39}$$

其中

$$\tilde{y}_P(k)=\begin{bmatrix}\tilde{y}_P(k+1|k)\\ \vdots\\ \tilde{y}_P(k+P|k)\end{bmatrix},\qquad w(k)=\begin{bmatrix}w(k+1)\\ \vdots\\ w(k+P)\end{bmatrix}$$

式中，$w(k+i)$ 为$k+i$时刻的期望输出，M、P的含义与线性预测控制相同。这样，在线的滚动优化就是在闭环预测（7-38）约束下，寻找控制作用使性能指标（7-39）取极小的问题。如果可由此求出最优的$u^*(k)$，…，$u^*(k+M-1)$，则在k时刻实施控制$u^*(k)$。到下一采样时刻，检测系统的实际

输出进行误差校正后，又重复进行优化。这就是非线性系统预测控制问题的一般描述。

可以看出，非线性系统的预测控制在方法原理上与线性系统没有什么不同。但考虑其具体算法时，它的在线滚动优化却成为突出的困难。即使在性能指标（7-39）取二次型的情况下，由于模型的非线性，我们面临的仍是一个相当一般的非线性优化问题。由于其中的优化变量出现在复合函数中，无法单独分离出来，故应用参数优化的概念直接求解$u(k)$，…，$u(k+M-1)$存在着解析上的困难。而把性能指标（7-39）结合模型（7-35）当作离散的非线性函数优化问题来解，也缺乏有效的算法，即使应用离散极大值原理写出一系列极值必要条件，因计算量十分庞大，也难以满足实时控制的需要。因此，虽然非线性系统的预测控制问题可以用明确的数学形式描述，但其求解仍存在着由非线性带来的本质上的困难。如何实时有效地求解非线性滚动优化问题，至今仍是一个富有挑战性的课题。

尽管如此，近年来，人们对于非线性系统的预测控制仍然作了大量研究，并提出了不少有意义的方法。这些方法的核心在于如何克服非线性问题求解的困难。目前已提出的方法大致有如下几种：

（1）线性化方法，即把非线性模型线性化后，用线性预测控制的滚动优化设计控制器，同时保留了非线性模型用于预测。为了克服模型线性化带来的误差，可通过在线辨识不断修正线性化模型。

（2）数值计算和解析相结合的方法，即利用非线性模型进行在线仿真，并通过梯度法寻优，反复迭代求出非线性优化问题的解。

（3）分层方法，即通过递阶算法把非线性优化转化为线性优化与协调两级计算，或通过非线性反馈实现输入输出线性化后再用线性预测控制算法。

（4）逼近方法，即用广义卷积模型或广义正交函数逼近非线性模型，并在截断后求解线性的或简单非线性的预测控制问题。

（5）特殊非线性系统，如哈默斯坦（Hammerstein）模型、双线性模型的预测控制算法。

以下，我们简要地介绍其中几种方法的基本思想和算法概貌。

1. 分层递阶优化的非线性系统预测控制

鉴于非线性系统预测控制在线优化所遇到的计算上的困难，文献〔27〕借用了大系统理论中用递阶算法求解非线性最优控制的思想，通过对非线性部分的预估和协调，把非线性滚动优化问题转化为线性问题求解。

对于式（7-34）给出的非线性系统，考虑二次型的滚动优化性能指标

$$\min J(k)=\|w(k)-\tilde{y}_P(k)\|_Q^2+\|u(k)\|_R^2 \qquad (7\text{-}40)$$

其中，$\boldsymbol{u}(k)=[\boldsymbol{u}^T(k)\cdots\boldsymbol{u}^T(k+M-1)]^T$为优化变量。结合预测模型（7-35）和（7-38），可将这一优化问题归结为非线性规划问题。显然，它不能像线性模型那样立即导出解析解。

为了充分利用线性模型时可导出解析解的优点，我们首先通过固定模型中的非线性因素，将其转化为线性模型，然后再通过协调来改进解。假定$\boldsymbol{u}_0=\boldsymbol{0}$，$\boldsymbol{x}_0=\boldsymbol{0}$为系统（7-34)的一个平衡点，即

$$\boldsymbol{f}(\boldsymbol{0},\ \boldsymbol{0})=\boldsymbol{0},\quad \boldsymbol{g}(\boldsymbol{0})=\boldsymbol{0}$$

若$\boldsymbol{f}$，$\boldsymbol{g}$是连续可微的，则可将系统在平衡点（$\boldsymbol{0}$，$\boldsymbol{0}$）处线性展开，把其等价地写作

$$\left.\begin{aligned}&\boldsymbol{x}(k+1)=\boldsymbol{A}\boldsymbol{x}(k)+\boldsymbol{B}\boldsymbol{u}(k)+\boldsymbol{D}(\boldsymbol{x}(k),\ \boldsymbol{u}(k))\\&\boldsymbol{y}(k)=\boldsymbol{C}\boldsymbol{x}(k)+\boldsymbol{G}(\boldsymbol{x}(k))\end{aligned}\right\} \qquad (7\text{-}41)$$

其中

$$\boldsymbol{A}=\left.\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=0,\boldsymbol{u}=0},\quad \boldsymbol{B}=\left.\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{u}}\right|_{\boldsymbol{x}=0,\boldsymbol{u}=0},\quad \boldsymbol{C}=\left.\frac{d\boldsymbol{g}}{d\boldsymbol{x}}\right|_{\boldsymbol{x}=0}$$

$$\boldsymbol{D}(\boldsymbol{x}(k),\ \boldsymbol{u}(k))=\boldsymbol{f}(\boldsymbol{x}(k),\ \boldsymbol{u}(k))-\boldsymbol{A}\boldsymbol{x}(k)-\boldsymbol{B}\boldsymbol{u}(k)$$

$$\boldsymbol{G}(\boldsymbol{x}(k))=\boldsymbol{g}(\boldsymbol{x}(k))-\boldsymbol{C}\boldsymbol{x}(k)$$

从 k 时刻起逐时刻写出式（7-41），可得

$$\left.\begin{aligned}\boldsymbol{x}(k+i)&=\boldsymbol{A}\boldsymbol{x}(k+i-1)+\boldsymbol{B}\boldsymbol{u}(k+i-1)\\&\quad+\boldsymbol{D}(\boldsymbol{x}(k+i-1),\ \boldsymbol{u}(k+i-1))\\\boldsymbol{y}(k+i)&=\boldsymbol{C}\boldsymbol{x}(k+i)+\boldsymbol{G}(\boldsymbol{x}(k+i))\\&\quad i=1,\ \cdots,\ P\end{aligned}\right\}\quad(7\text{-}42)$$

这是一个形为式（7-35）的预测模型。为符号简洁起见，这里不再采用预测记号。

为了克服基于上述非线性模型求解优化问题的困难，注意到式（7-42）中$\boldsymbol{A}$、$\boldsymbol{B}$、$\boldsymbol{C}$均为常量，非线性只出现在$\boldsymbol{D}$、$\boldsymbol{G}$项中，故可通过适当手段固定$\boldsymbol{D}$、$\boldsymbol{G}$，使之变为线性模型。为此，引入附加的预估

$$\left.\begin{aligned}\boldsymbol{u}(k+i)&=\boldsymbol{u}^0(k+i),\quad i=0,\ \cdots M-1\\\boldsymbol{x}(k+i)&=\boldsymbol{x}^0(k+i),\quad i=1,\ \cdots P\end{aligned}\right\}\quad(7\text{-}43)$$

其中，$\boldsymbol{x}^0$可在$\boldsymbol{x}(k)$、$\boldsymbol{u}^0$已知的情况下由式（7-41）递推算出。把这些$\boldsymbol{u}^0$、$\boldsymbol{x}^0$代入式（7-42）中的非线性项$\boldsymbol{D}$、$\boldsymbol{G}$，可以得到

$$\left.\begin{aligned}\boldsymbol{x}(k+i)&=\boldsymbol{A}\boldsymbol{x}(k+i-1)+\boldsymbol{B}\boldsymbol{u}(k+i-1)+\boldsymbol{l}_i\\\boldsymbol{y}(k+i)&=\boldsymbol{C}\boldsymbol{x}(k+i)+\boldsymbol{h}_i,\quad i=1,\ \cdots,\ P\\\boldsymbol{u}(k+i-1)&=\boldsymbol{u}(k+M-1),\quad i\geqslant M\end{aligned}\right\}(7\text{-}44)$$

其中

$$\boldsymbol{l}_i=\begin{cases}\boldsymbol{D}(\boldsymbol{x}(k),\ \boldsymbol{u}^0(k)), & i=1\\\boldsymbol{D}(\boldsymbol{x}^0(k+i-1),\ \boldsymbol{u}^0(k+i-1)), & M>i\geqslant 2\\\boldsymbol{D}(\boldsymbol{x}^0(k+i-1),\ \boldsymbol{u}^0(k+M-1)), & P\geqslant i\geqslant M\end{cases}$$

$$\boldsymbol{h}_i=\boldsymbol{G}(\boldsymbol{x}^0(k+i)),\quad i=1,\ \cdots,\ P$$

它们在已知$\boldsymbol{x}(k)$且对$\boldsymbol{u}$、$\boldsymbol{x}$预估后均为已知常量。对于这样一个线性模型（7-44）下性能指标为式（7-40）的优化问题，采用通常的线性预测控制算法很容易得到其解析解，从而避免了直接求解非线性优化的困难。

然而应该注意，线性模型（7-44）是在预估了非线性模型

(7-42) 中的D、G项后得到的，它一般地不等于原来的非线性模型，因而上面算得的最优解也不是原问题的最优解。只有在解出的最优解恰好就是$\boldsymbol{u}$、$\boldsymbol{x}$的初始预估值时，这两个模型才是一致的。这意味着我们要把还未算出的$\boldsymbol{u}$、$\boldsymbol{x}$作为预估值，才能保持模型的一致性，这当然是不可能的。这一困难可通过迭代过程加以克服，即在求出最优解后，若其与初始预估值不一致，则把它们作为新的预估值

$$\left.\begin{aligned} \boldsymbol{u}^0(k+i)&=\boldsymbol{u}(k+i), \quad i=0, \cdots, M-1 \\ \boldsymbol{x}^0(k+i)&=\boldsymbol{x}(k+i), \quad i=1, \cdots, P \end{aligned}\right\} \quad (7\text{-}45)$$

重新求解线性模型的优化问题。在一定条件下，这一迭代过程是收敛的，最终达到了预估值与解的一致，这时得到的解就可作为实际解付诸实施。

这样，就可得到非线性模型优化的两级递阶算法。其中，第1级是在对$\boldsymbol{u}$、$\boldsymbol{x}$进行预估的基础上基于线性模型 (7-44) 的优化计算，第2级是在第1级所算出解的基础上按式 (7-45) 重设预估初值的协调运算。这一过程需反复迭代，直至精确预估条件满足为止。由于在第1级充分利用了线性预测控制解析解的简易性，第2级的协调计算又十分简单，所以，即使不是一次求解而需迭代，在线计算也能满足实时要求。

以上介绍的是这种控制方法的主要思路，在算法实现上，还有一些技巧问题和理论问题需要探讨，在此不再详述。

2. 利用广义卷积模型的非线性预测控制

如同线性系统的输入输出关系可以用卷积模型描述一样，非线性系统的输入输出关系可以用不同形式的广义卷积模型描述。文献[28]讨论了利用维纳 (Wiener) 模型作为广义卷积模型的MAC算法。这一模型的离散时间形式为

$$y(k+1)=h_0+\sum_{j=0}^{N} h_1(j)\,u(k-j)$$

$$+\sum_{i=0}^{N}\sum_{j=0}^{N}h_2(i,\ j)u(k-i)u(k-j)$$
$$+\cdots\cdots \tag{7-46}$$

其中，$h_l(l=0,\ 1,\ \cdots)$ 称为 l 阶维纳核，N 表示系统记忆的截断值，类似于脉冲响应模型中的模型时域。在实际应用时，这一无穷级数在项数上也应截断。此处考虑其到 2 阶时截断，则可得到粗略的维纳模型作为预测模型

$$y_M(k+1)=h_0+\sum_{j=0}^{N}h_1(j)u(k-j)$$
$$+\sum_{i=0}^{N}\sum_{j=0}^{N}h_2(i,\ j)u(k-i)u(k-j) \tag{7-47}$$

式（7-46）中的维纳核 h_l 可通过用高斯白噪声扰动系统后由互相关函数求出，但这涉及到大量数据的统计平均，在实际上难以实现。由于经截断后的维纳模型（7-47）只有较少的核参数，故可用辨识的方法求出它们。注意到式（7-47）可改写为

$$y_M(k+1)=\boldsymbol{\varphi}^T(k)\boldsymbol{\theta} \tag{7-48}$$

其中

$$\boldsymbol{\varphi}^T(k)=[1\quad u(k)\cdots u(k-N),\ u^2(k),$$
$$u(k)u(k-1),\ \cdots,\ u^2(k-N)]$$
$$\boldsymbol{\theta}=[h_0\ \ h_1(0)\cdots h_1(N),\ h_2(0,\ 0),\ h_2(0,\ 1),\ \cdots,$$
$$h_2(N,\ N)]^T$$

可见式（7-48）对于参数 $\boldsymbol{\theta}$ 是线性的，故可用标准的线性参数估计方法求出维纳核。

为了对该非线性系统进行预测控制，根据MAC算法的原理，设参考轨迹为

$$y_r(k+i)=\alpha^i y(k)+(1-\alpha^i)c,\quad i=1,\ 2,\ \cdots$$

其中，c 为设定值，$0<\alpha<1$，并取一步优化性能指标为

$$\min J_1(k)=[y_P(k+1)-y_r(k+1)]^2\omega_1$$

采用闭环预测可得到上述性能指标中的 $y_P(k+1)$：

$$y_P(k+1)=y_M(k+1)+(y(k)-y_M(k))$$

在无约束情况下优化，可以得到

$$(1-\alpha)(c-y(k))=y_M(k+1)-y_M(k)$$

以式（7-47）代入后，可把最优控制作用 $u(k)$ 的计算归结为求解下述二次方程

$$au^2(k)+bu(k)+d=0 \tag{7-49}$$

其中

$$a=h_2(0,\ 0)$$

$$b=h_1(0)+2\sum_{j=1}^{N}h_2(0,\ j)u(k-j)$$

$$\begin{aligned}d=&\left[h_0+\sum_{j=1}^{N}h_1(j)u(k-j)\right.\\&\left.+\sum_{i=1}^{N}\sum_{j=1}^{N}h_2(i,\ j)u(k-i)u(k-j)\right]\\&-y_M(k)-(1-\alpha)(c-y(k))\end{aligned}$$

由此即可求出 $u(k)$。式中的维纳核参数可以在线估计，这使形成的算法具有自适应性质，不但可以改进原始的粗略模型，而且在对象发生变化时可及时修改模型。

上述方法通过截断离散卷积模型，把对象输出近似表为不同时刻输入及其乘积项的线性组合。当维纳模型于第 n 阶被截断时，采用一步优化的 MAC 算法可归结为求解一元 n 次方程。n 越大，模型越精确，但所涉及的参数将急剧增加，在线解方程的次数也将增高。

采用同样的原理，也有人研究了采用沃尔泰拉（Volterra）模型作为广义卷积模型的MAC算法[29]。它把非线性系统的输入输出关系近似用多项式模型描述

$$\begin{aligned}y_M(k+1)=&\sum_{i=1}^{P}\sum_{i_1=1}^{N}\cdots\sum_{i_n=1}^{N}h_i(i_1,\ \cdots,\ i_n)\\&\cdot u(k+1-i_1)\cdots u(k+1-i_n)\end{aligned}$$

当采用一步优化MAC算法时，控制量$u(k)$的求取同样转化为一个一元n次方程的求根问题。这一求根也可在u的上下限约束区间内用直接搜索的方法进行，当n较大时，这比直接解方程更为实用有效。

3. 哈默斯坦模型的非线性预测控制

哈默斯坦模型是由一无记忆静态增益

$$x(k)=\sum_{i=1}^{l} r_i u^i(k) \tag{7-50}$$

和一动态线性环节

$$y_M(k)=\sum_{i=1}^{n_a} a_i y_M(k-i)+\sum_{j=1}^{n_b} b_j x(k-j) \tag{7-51}$$

所组成的非线性模型，其中l、n_a、n_b均为模型阶数，r_i、a_i、b_j为参数，u、y_M分别为模型输入及输出，而x则为中间变量。这种模型结构简单，能用于描述pH过程及具有幂函数、死区、开关等非线性特性的工业过程，是一类应用广泛的非线性模型。下面，我们简要介绍文献〔30〕中对这类系统预测控制的思路。

根据式（7-50）、式（7-51）所描述的模型形式，即可以发现输入$u(k)$是通过中间变量$x(k)$影响模型输出的。虽然u与y之间的关系是非线性的，但x与y之间的关系却是线性的。因此，首先可立足于线性关系（7-51）考虑把非线性预测控制问题转化为易于求解的线性预测控制问题。为此，考虑如下的优化性能指标

$$\min J(k)=\sum_{i=1}^{P}[y_r(k+i)-y_P(k+i)]^2+\lambda\sum_{j=1}^{M}\Delta x^2(k+j-1) \tag{7-52}$$

这里，把控制作用Δu的影响间接地通过Δx反映在性能指标中，从而在动态优化中排除了非线性关系（7-50）的影响。

基于预测模型（7-51）对性能指标（7-52）的优化问题，是

一个典型的线性模型二次性能指标优化问题。这里只要把 x 看作系统的输入并应用模型（7-51）中的系数a_i，b_j 递推导出 y 对 x 的阶跃响应，即可用DMC算法的最优控制律导出使性能指标（7-52）最优的$\Delta x(k)$，并进一步算出 $x(k)$。在此基础上，注意到式（7-50）表明 $u(k)$ 只取决于 $x(k)$，即可通过

$$x(k)=\sum_{i=1}^{l} r_i u^i(k)$$

求解一元 l 次方程，得到最优的控制输入 $u(k)$。

可以看出，针对式（7-50）、式（7-51）所描述的哈默斯坦模型，采用动态和静态分离的思想，提出适当的优化性能指标，即可将其预测控制问题分解为一线性模型的动态优化问题及一非线性模型的静态求根问题。在这种特殊模型形式下，实际实现的并不是基于非线性模型的滚动优化，而仍然是一个通常基于线性模型的滚动优化问题。由输入引起的非线性，在这里不过是作为一个附加的静态环节来处理的。

第八章　预测控制的工业应用实例

预测控制的产生，是复杂工业系统实现优化控制的需要。作为一种新型控制算法，预测控制有很多适合于工业环境的优点。例如，它可以用简易的方式建模，软件易于实时实现，参数整定意义直观从而便于对操作者进行培训，对环境和对象的不确定性有很强的鲁棒性，等等。然而，对于工业界来说，一种新算法如果只是现有算法的等价替代物，而不能带来更大的实际效益，则它仍然是缺乏吸引力的。因此，我们首先要分析一下复杂工业过程中应用预测控制的利益所在。

在现代的大工业过程中，控制结构通常具有多层递阶的形式，而且人们已经认识到，为了实现过程的优化，必须把严格的动态控制作为不可缺少的一层考虑在控制结构中。因此，这种大工业过程的递阶控制一般包含以下几个层次：

第 0 层　辅助系统（如随动阀）控制。在这里，PID控制器已相当有效。

第 1 层　受到状态和结构不可测扰动的多变量过程的动态控制。

第 2 层　设定值的优化。其目的是按某一性能指标最优确保产品的产量和质量。

第 3 层　生产过程在时间、空间上的计划、调度。

在这一多层递阶结构中，由第 0 层和第 1 层控制所能获得的直接经济效益实际上是微不足道的。与此相反，第 2 层的操作优化却能明显地提高经济效益。但是，这并不意味着前两层的控制是不重要的。事实上，要在第 2 层实现满意的优化，其必要条件是第 0 层和第 1 层必须已经优化。例如，人们在操作优化时常常将工作点设置得接近约束临界值，实现所谓"卡边控制"，以获

得较高的经济效益。如果第1层的动态控制质量很差，实际值对设定值的方差分布很广，为安全起见，工作点就不得不设置得偏离约束临界值较远，这样，第2层的优化性能指标就难以提高。相反，如果第1层能实现严格的动态控制，控制偏差很小，则在第2层优化时便可把工作点设置得临近其约束边界。在保证安全控制的同时，经济效益就可进一步提高。

预测控制通常采用输出方差最小化的优化性能指标，基于模型的预测可在优化中考虑各种实际约束，又有很好的鲁棒性和抗干扰性，因此很适宜用来作为上面所需要的动态控制算法。它的基于模型的预测和优化功能，使其在应用于多变量、大时滞、有约束的复杂工业过程时，与常规的PID控制相比质量有明显提高。而它的易于建模、鲁棒性强，便于用户理解和掌握的优点，使其比现有的复杂优化控制算法对工业用户有更强的适用性。在复杂的工业过程中，预测控制的实现方式一般有两种：

（1）直接控制，即对过程变量直接进行控制。

（2）透明控制，即在经PID控制后对闭环对象直接控制，它也可理解为对PID控制再进行动态设定值控制。

不论采取哪一种方式，预测控制的目的都是为了实现严格的动态控制，通过减少实际值偏离设定值的方差，使第2层的操作优化建立在更好的基础上，更接近理想的卡边控制。因此，它对生产过程必将带来显著的经济效益。预测控制在大工业过程中的产生和应用，一开始正是建立在上述理解的基础上的。论述预测控制原理及应用的第一篇重要文献〔1〕，已对此作了详细的阐述。

70年代中期，MPHC(MAC)算法在锅炉、分馏塔的控制中获得了成功的应用，标志着预测控制这一新型算法进入了工业控制领域。与此同时，DMC算法也在石油加工生产装置的优化控制中产生并得到成功应用，引起了过程控制界的普遍重视。80年代以来，一些专门从事过程控制系统的公司相继推出各种软件包，加速了预测控制技术在化工、石油等部门生产过程控制中的

推广应用。例如，Setpoint 公司研制开发的 IDCOM 软件包，于1981年应用于海湾石油公司在加拿大安大略省Clarkson炼油厂的润滑油加氢反应器的温度控制上，连同三个分馏塔的计算机控制系统一起，使产品粘度变化减少70%，燃料节省25%以上，并提高了操作的灵活性。Profimatic公司推出的催化裂化高级过程控制软件包，将预测控制应用于炼油厂催化裂化提升管反应器温度和回炼油流量的多变量系统及其他一些控制系统，均取得了很好的效果。伴随着这些成功应用所带来的经济效益，充分显示出预测控制在复杂工业系统中的作用及潜力。

80年代中期以来，随着预测控制的机理和作用日益为人们所认识，其应用也从化工、石油等主要部门的过程控制推向了更广阔的范围，并在与应用的结合中产生了各种新的控制算法和策略。作为一种动态优化控制算法，预测控制仍然大量应用于工业过程对设定点实现严格的动态控制，但其模型预测、滚动优化、反馈校正的原理，却有着更广泛的适用性，使其已推广到在复杂系统中实现高质量的跟踪控制，以及对具有复杂性能指标的系统实现动态优化等应用场合。在本章中，我们将介绍预测控制应用于大工业过程或复杂系统的若干实例，以反映预测控制的应用概貌。尽管由于篇幅所限只能取有限的例子，并且对每一例子不可能深入到细节，但读者可通过对应用环境、控制方案、实际效果的介绍，进一步了解预测控制在应用于实际系统时所面临的问题、原理的实现以及方法的多样性。

§8.1 汽轮发电机组蒸汽系统的控制

文献〔1〕报导了MAC算法在一个250MW汽轮发电机组控制中的应用，该汽轮发电机组蒸汽系统的工作原理如图 8-1 所示。

从汽包 1 经蒸发器 9 集成的蒸汽经过热器 2 加热后，送入高压缸 3 做功，并经再热器 4 后送入中压缸 5 再做功，工作后的蒸汽进入凝汽器 6 冷却成水，经水泵 7 打入省煤器 8 后送回汽包，完成蒸汽-水循环。燃料泵 10 和空气泵11分别打入燃烧加温所需

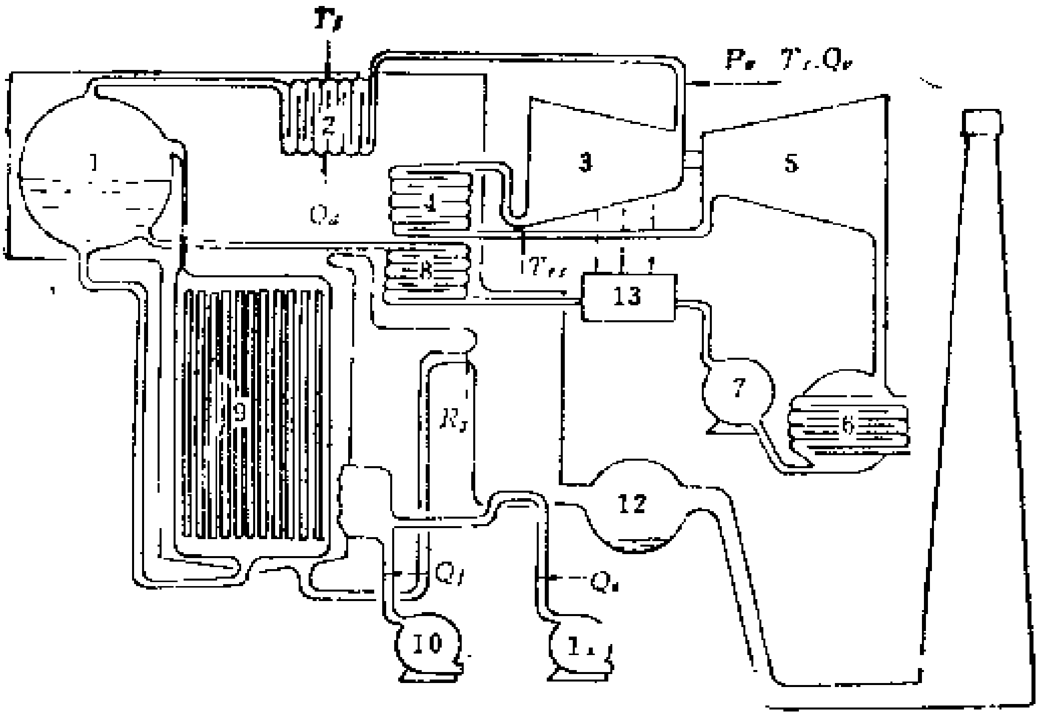


图8-1 250MW机组蒸汽系统

1—汽包；2—过热器；3—高压缸；4—再热器；5—中压缸；6—凝汽器；7—水泵；8—省煤器；9—蒸发器；10—燃料泵；11—空气泵；12—除尘器；13—主机。

的燃料和空气，除尘器12把燃烧杂质从烟囱中排除出去。在这里，被控量是传送到高压缸的蒸汽压力P_v、过热器及再热器中的蒸汽温度T_r和T_{rr}，而作为控制变量的则是冷却水流量Q_d、循环空气流量R_y以及燃料输入量Q_f。此外，蒸汽的出口流量Q_v是一个可测扰动量，可用来反映对象的负载变化。

对于这样一个有三个可控输入，一个可测扰动和三个被控量的多变量系统，其预测控制采用了图 8-2 所示的结构，其中，燃料量结合蒸汽出口流量直接用来控制蒸汽压力，构成了MAC-PI串级的透明控制结构（见7.2节），而两个过热器的出口温度T_r和T_{rr}则由冷却水流量Q_d及空气循环量R_y实行直接的多变量MAC控制。

用于这一控制的算法软件包称为IDCOM (Identification and Command，辨识与命令)，它的基础算法是MAC算法，但根据控制与辨识在算法上的对偶关系引入了参数辨识，并考虑了对

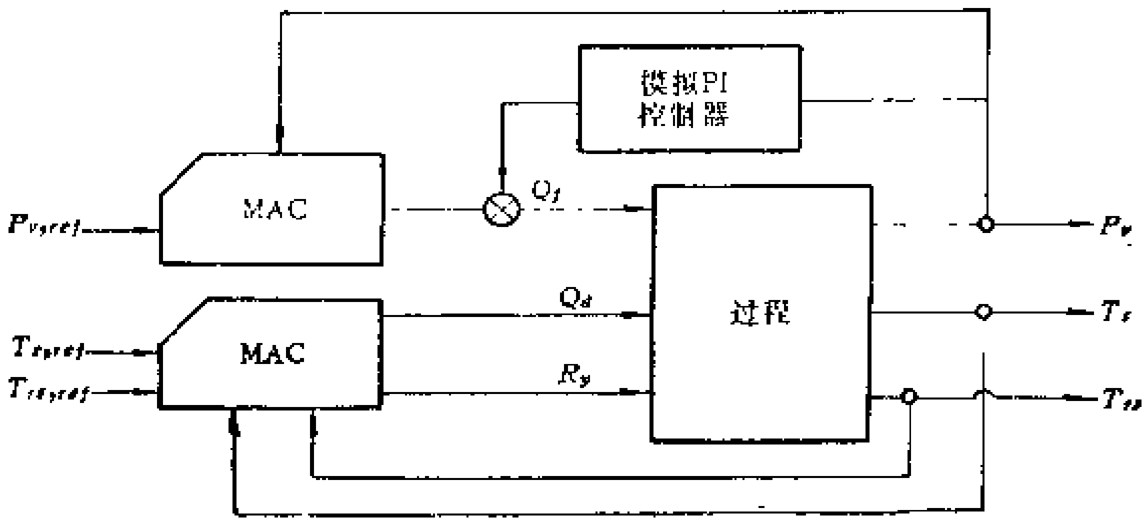


图8-2　蒸汽系统的预测控制方案

执行器和中间变量的约束（例如这里对冷却水注入流量Q_d及其变化速率的约束），约束的上下限可由过程操作人员在线设定。在控制启动时，脉冲响应模型和约束条件首先可设置得较为严格，然后在运行过程中再逐步优化。

应该指出，虽然这一软件包提供了在线辨识的可能性，但在工业环境中一般并不需要连续地进行辨识。这是因为，在设定值为常数时，算法自然具有的鲁棒性已能被动地适应对象结构的缓慢变化。而且在工业环境中，由于不允许加大幅度的测试信号，测试结果的信噪比很差，连续的辨识和自适应对用户来说反而显得危险。因此，通常可根据先验信息用启发式方法对预测模型进行整定。例如，通过离线辨识可以得到本例中的脉冲响应，但在不同的负载水平（100%，80%，50%）下，测试结果表明，与温度有关的脉冲响应是随着负荷Q_v的变化呈反方向变化的。通过对响应的进一步分析可以发现，如果采样周期也作相应的调整，则所测得的响应参数就只有增益的变化。这样，在线就不需辨识全部脉冲响应系数，而只需辨识每一脉冲响应的增益即可。

对于过程中执行器的各种非线性，应该独立地加以辨识，并在预测模型中加以考虑，这样可保证预测的正确性。在这里，循环

空气流量R_y的非线性，就是经辨识后作为内部模型的组成部分的。

这一汽轮发电机组采用预测控制确保了对蒸汽系统实现整体的优化控制，因而与用模拟仪表的PID控制相比，显现出明显的优越性，特别当系统负荷发生大幅度变化时，能很快把被控量调节到设定值。图8-3是系统受到10MW/min的三角形功率扰动时，用IDCOM和用模拟控制的效果比较。可以看出，采用预测控制后，控制量在扰动发生后较迅速地作出了反应，从而使被控量较快地得到调节，避免了大幅度的超调及振荡。

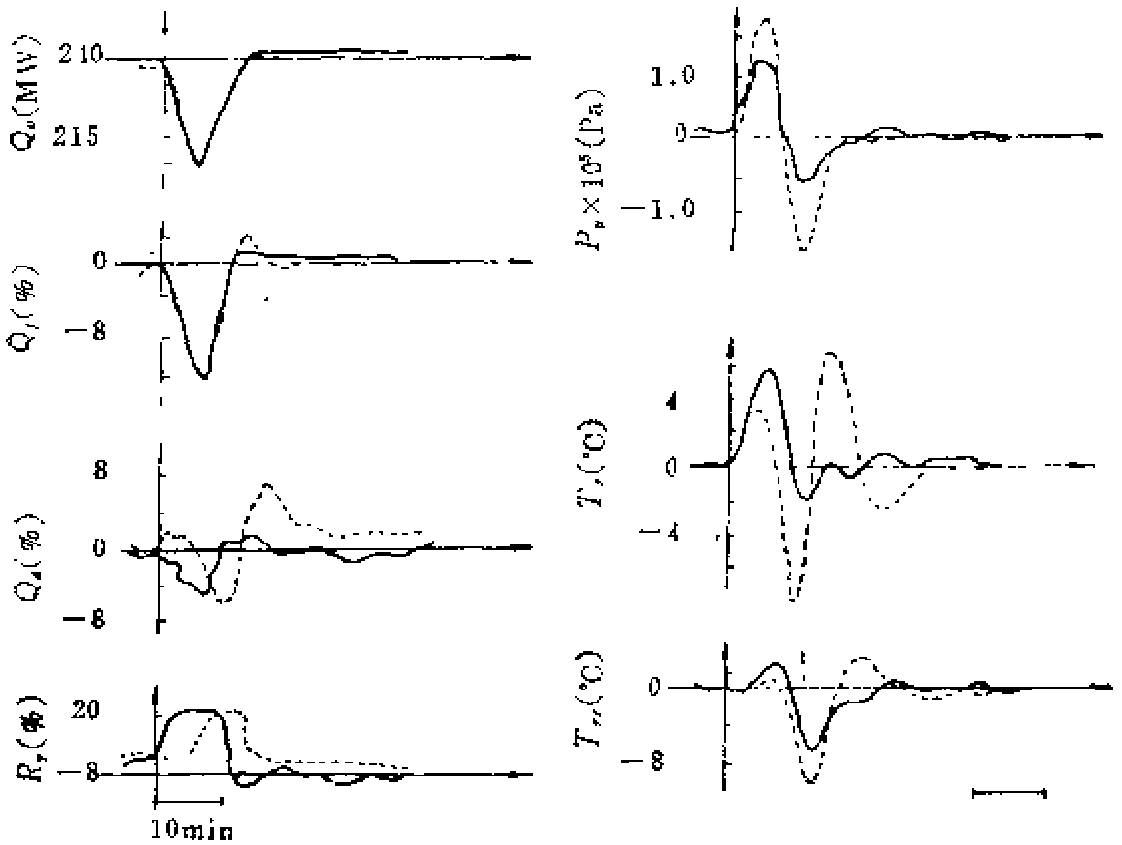


图8-3 负载扰动时的模拟控制（虚线）和IDCOM控制（实线）

§8.2 炼油厂加氢裂化单元的动态矩阵控制

在石油加工工业中，加氢裂化是对产品生成有重要影响的关键过程。70年代后期，美国、加拿大等国的许多石油公司，都先后采用预测控制对其实现多变量优化控制，取得了明显的效益。

在这里，我们将介绍壳牌石油公司加拿大Scotford炼油厂对加氢裂化单元的动态矩阵控制[31]。

该炼厂原油加工的核心部分是两组并行的加氢裂化单元，每一单元包含两个多床反应器即第1级的预处理塔及第2级的裂化塔。第1级主要用来去氮，裂化反应则主要在第2级发生。新鲜原油和循环料的馈入以及产品分馏则为两组单元所共有。整个过程的简化流图可见图8-4。

在图8-4所示的过程中，由附近工厂提供的高纯氢气在经过预热炉加热后，与高压馈入的烃料混合，然后经过多层催化床发生裂化反应，把原料中的硫和氮转化为硫化氢和氨气，并使未饱和的烃类饱和。由于这是放热反应，为了使反应保持在正常温度下进行，每一催化床顶部都引入了未加热的氢气直接作为冷却剂。经两级反应后的烃送入分馏塔分馏出不同的原油产品，其中最重的烃料将循环送回第二级反应器进一步裂化。

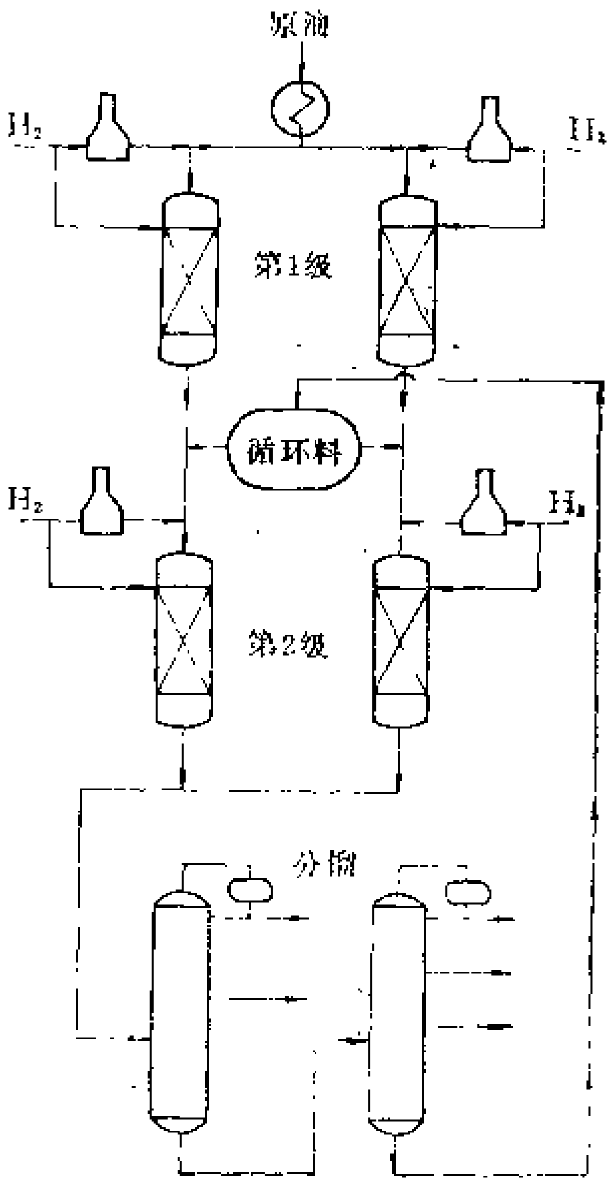


图8-4　加氢裂化过程简图

在该炼油厂中，采用了FOXBORO SPECTRUM分布式计算机系统（DCS）实现工艺设定点的基础控制。其中，反应器入口温度由送入预热炉的燃料气流量控制，各反应床顶部的温度则

通过调整冷却氢气输入阀门开度加以调节。配有动态矩阵控制和数据采集计算软件的上级监控计算机则进一步对各反应器实现动态优化控制。其基本目的是，通过对基础控制设定值的动态调整，保证各反应器的DCS系统得到严格和安全的控制。鉴于 4 个反应器的控制策略都是相同的，下面我们仅以图 8-5 所示的第 1 级反应器为例，来说明其 DMC 控制的实现。

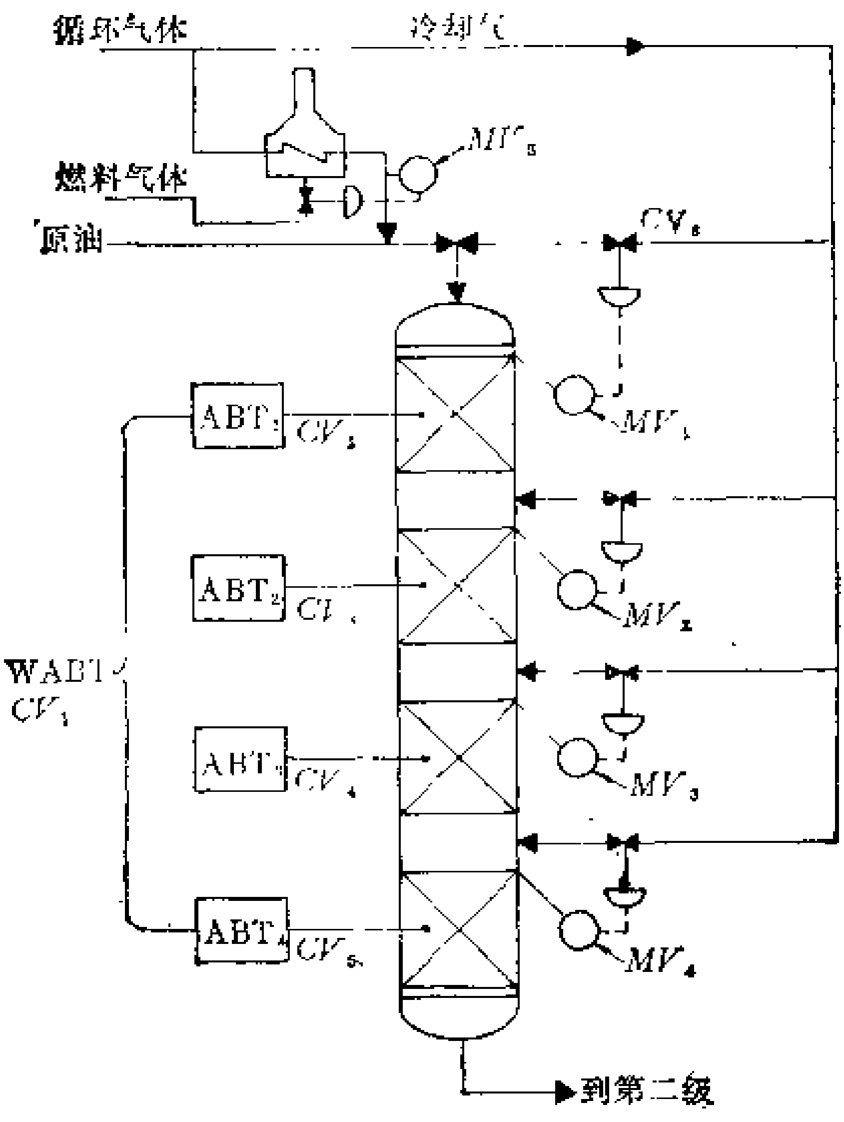


图8-5　第 1 级反应器的控制

图8-5所示的反应器由 4 层反应床组成。基础级的 DCS 系统已对反应器入口温度及各反应床顶部温度进行了控制，但这还不能保证反应在严格的工艺条件下进行。进一步实现DMC控制，就

是为了获得严格的动态控制性能，使反应过程按工艺要求正常进行。这是由加权平均床温（WABT）为度量标准的，它是各反应床平均温度（ABT）的加权平均。DMC控制的主要目标，就是要使反应器的WABT严格跟随其工艺设定值。控制的辅助目标，则可以是使各反应床的ABT按给定的温度轨线变化或是使控制能耗最小。

在DMC控制中，4个反应床入口温度和反应器入口温度的DCS设定值被选作为控制量，而反应器的WABT、4个反应床的ABT以及反应器入口处冷却剂的阀门开度则作为被控量。其中，WABT的设定值由工艺要求给出，而各ABT的设定值则可根据WABT的设定值和由用户输入的一组偏置系数算出。由于WABT是各ABT的线性组合，所以这实质上是一个五输入五输出的多变量优化控制问题。

除了有直接设定目标的被控量外，在控制方案中还把可测量的4个反应床的出口温度作为辅助变量加以考虑，其目的是在温度偏移时提供高度安全保证。在设计控制策略时，要通过调整相应的反应床入口温度设定值使其出口温度低于一定幅值，以保证安全的反应条件。同样，各反应床冷却氢气阀门开度因受到上下限约束，也应作为附加的约束条件在设计中加以考虑。

为了对这一复杂系统实现DMC控制，一开始就需要确定全部被控量对于控制量的阶跃响应，这是离线进行的。首先，由监控计算机对每一控制量产生M序列（伪随机双电平序列）测试信号，进行长达24小时的测试，然后再用监控计算机上的软件包对测试数据进行时间序列分析，这样可得到图8-6所示的被控量的阶跃响应。在这里，可以看到，每一反应床入口温度的设定值变化都会对其下一层反应床产生扰动。但由于各反应床经DCS的温度控制均工作在闭环状态，故这些扰动效应的稳态值将是0。图8-6所示的动态响应是很难用低阶模型加以描述的。事实上，不必进一步辨识其最小化模型，从这些阶跃响应中便可直接取出DMC控制所需要的数据。

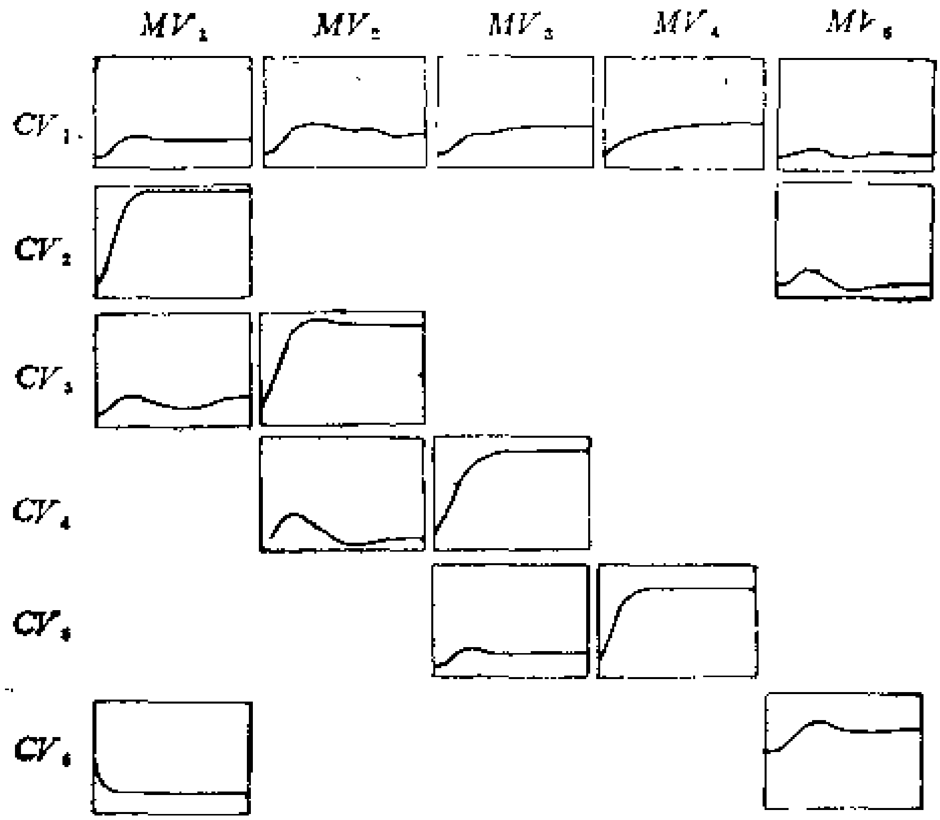


图8-6 过程的动态阶跃响应

利用同样的方法，也可得到各辅助变量对于控制量的阶跃响应，其响应曲线在此不再画出。

在上述阶跃响应模型的基础上，可在PC机上应用离线仿真软件包QDMC(二次型动态矩阵控制)对系统进行仿真设计。这一有约束的多变量DMC控制可转化为如下滚动优化问题：

$$\min_{\Delta u} J(k)=\frac{1}{2}\Delta \boldsymbol{u}^T \boldsymbol{H} \Delta \boldsymbol{u}-\boldsymbol{g}^T \Delta \boldsymbol{u}$$

其中，$\boldsymbol{H}=\boldsymbol{A}^T\boldsymbol{\Gamma}^T\boldsymbol{\Gamma}\boldsymbol{A}+\boldsymbol{\Lambda}^T\boldsymbol{\Lambda}$，$\boldsymbol{g}=\boldsymbol{A}^T\boldsymbol{\Gamma}^T\boldsymbol{\Gamma}\boldsymbol{e}$，$\boldsymbol{e}=\boldsymbol{w}-\tilde{\boldsymbol{y}}_0$，而约束条件为

$$\boldsymbol{C}\Delta \boldsymbol{u} \geqslant \boldsymbol{c}$$

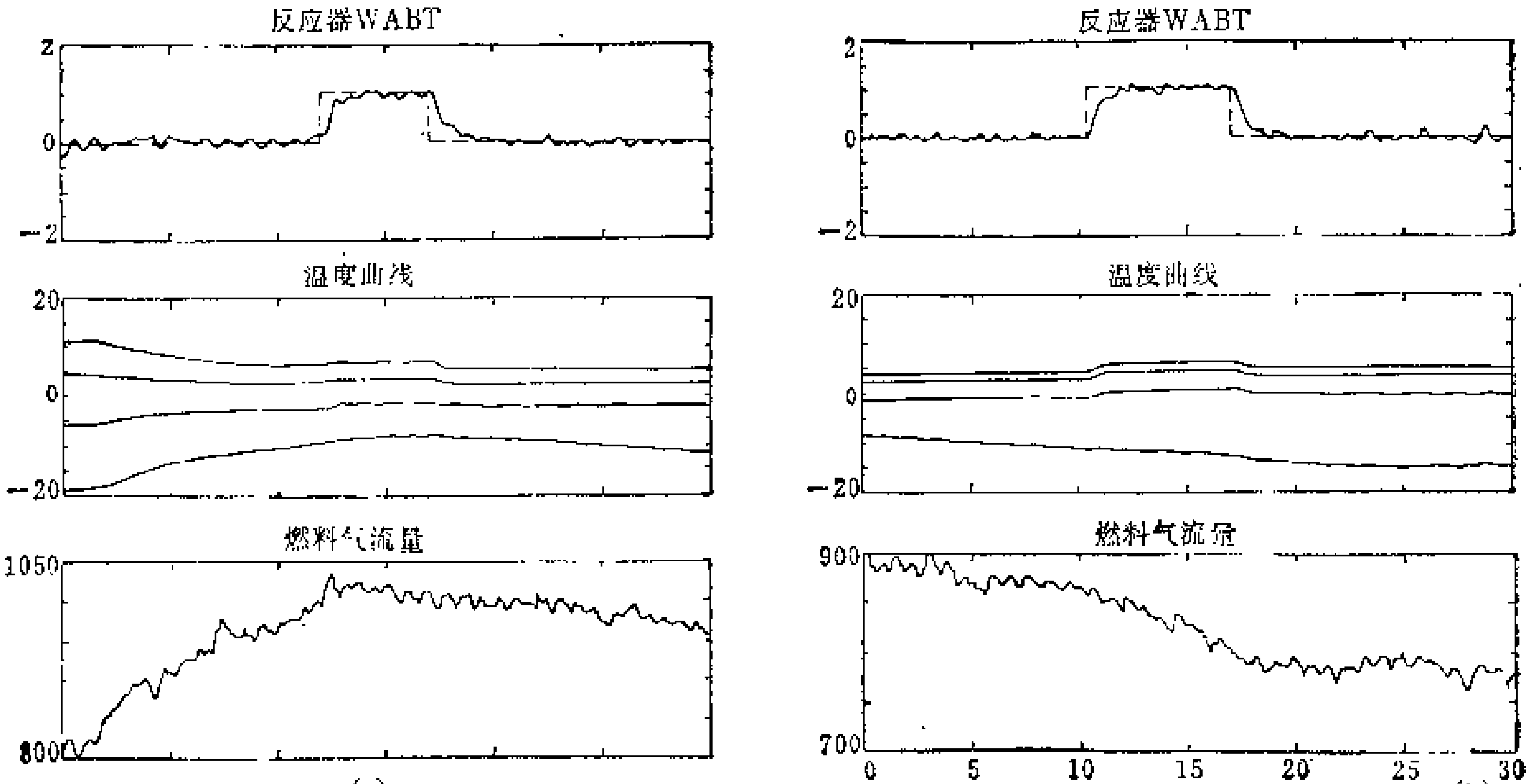
式中的$\boldsymbol{A}$为动态矩阵，$\boldsymbol{\Gamma}^T\boldsymbol{\Gamma}$和$\boldsymbol{\Lambda}^T\boldsymbol{\Lambda}$分别是3.1节DMC算法描述中的权矩阵$\boldsymbol{Q}$和$\boldsymbol{R}$，它们的元素有直观的物理意义，可作为整定参数离线或在线地加以改变。

如前所述，除了以WABT作为控制的主要目标外，根据辅

助目标的不同，可以采取两种不同的优化控制模式。第一种是跟踪模式，即对各反应床的ABT有期望的目标轨线。这种模式可用于延长催化剂的寿命或使反应器工作在特定的温度变化（递减、平滑或递增）模式下。这时，对应于各ABT的权矩阵 Γ 的元素通常要比WABT的小一个数量级，而比反应器入口冷却剂阀门开度的权系数大5～10倍。在反应床出口温度有持续偏高的趋势时，对辅助变量的加权系数应加大，可以为WABT控制的10～100倍。优化控制的另一种模式是节能模式，即使氢气预热炉的燃料气体消耗为最小。这种模式充分利用了过程放热反应的特点，降低了加热炉的出口温度并减少了各反应床的冷却气体流量，这时，将导致大幅度下降的温度变化曲线。适应于这种模式的每一ABT对应的权系数应设置为0，反应器入口温度的设定值将作为始终受到约束的控制量，并缓慢地下降到冷却剂阀门开度达到的约束界限。

通过离线仿真进行系统设计后，可有选择地产生面向应用的FORTRAN数据和维数语句，并把它们结合到在线通用标准程序中实施实时控制。监控级计算机可在检测实时信息的基础上在线计算各反应床的ABT和反应器的WABT，并根据WABT的工艺设定值和操作人员输入的偏置系数计算出各ABT的设定值。由于采用了批处理技术，操作人员可通过显示屏幕和键盘对DMC参数进行在线整定，修改约束的上下限，或选择优化控制的模式。

DMC控制策略在加氢裂化单元上的实现带来了明显的好处，跟踪期望值变化的响应在精确性和快速性方面都明显胜过手动控制。图8-7画出了第1级反应器在30 h内WABT设定值变化两次时用DMC控制的实际效果。图8-7（a）是反应器在第1种模式（跟踪模式）下平滑的响应。各催化床ABT缓慢变化的温度曲线呈现出期望的状态，同时对WABT的严格控制未产生干扰，燃料气体的消耗在这一模式下是逐渐增加的。图8-7（b）则表示由上述模式转换到第二种控制模式（节能模式）。在此期间，WABT在设定值发生两次变化时仍呈现出良好的跟踪性能，但燃料气体



(a)

(b)

图8-7　DMC控制的效果

（a）跟踪模式；（b）节能模式。

的消耗在这种模式下可节约25%。此外，在没有遇到其他约束条件时，反应器冷却流量的减少也可带来附加的好处，因为在氢气供应量有限的情况下，减少冷却用的氢气流量可提高生产能力。

类似的控制策略已广泛应用于石油加工工业中，并取得了显著的经济效益。例如，在 Suncor 公司对加氢裂化过程使用DMC控制，使反应器预热炉节能达到55%，取得了近36万美元的年经济效益[32]。目前，在国外的许多炼油厂，DMC 作为一种有效的多变量优化控制算法，正在得到日益广泛的应用。

§8.3 天然气传输网络的在线优化

在过去几十年里，天然气已逐渐成为世界的主要能源之一。例如，美国就有约 1/3 的能源是由天然气供应的。与石油相比，天然气因其价廉和污染小而有明显的优点，但因其体积大却难以储存，因而绝大多数天然气是通过管道直接传输的。在大型天然气传输网络中，为了满足各终端用户的需要并易于传输，需要使天然气保持很高的压力。由于传输距离长会引起气压下降，往往需要一些中间压缩站不断加压。因此，这种传输方式的成本很高。例如，加拿大天然气传输公司 1979 年的运行费就达到 6500 万美元，几乎是进入管道的天然气成本的 8～10%。

目前，天然气传输网的控制都未采用自动方式，而是由操作人员根据经验，通过中央控制单元来修改不同压缩站的设定点。在这里，控制的主要目标是使所传输的天然气保持合适的压力，以满足正常传送和用户的需要，而降低传输成本这一有重大经济意义的目标，则因缺乏有效的方法，往往难以兼顾。操作人员通常只能在保证前者的前提下，尽可能使能耗保持在适当的范围。在这种情况下，迫切需要发展新的控制策略，以实现能耗的最小，同时又使系统运行在可行、安全的区域内。这一问题的难点在于天然气传输网络是一个有大量物理约束的非线性高维大系统，计算求解十分困难，加之测量点少，缺乏关于内部状态和外部扰动的

知识，离线的优化在实际上很难奏效。因而，研究其在线优化的策略，便成为一个亟待解决的问题。

在过去很长时间里，人们已对天然气传输网络的建模、仿真和状态估计等进行了大量的研究，取得了很多成果。例如慕尼黑工业大学在70年代末研制的仿真程序包GANESI和80年代初研制的状态估计软件包，已成功地应用于工业部门对天然气传输网络进行离线仿真和在线估计。本节介绍的文献[33]的在线优化策略，就是建立在这些研究工作的基础上的。

典型的天然气传输网络是由源点、用户和压缩站通过传输管道联结而成的，其中源点汇集不同供气源和网络其他部分传送来的气体，可为网络提供一定已知压力的天然气。用户则是天然气的消耗者，可看作是网络的负荷。他们对天然气的压力和用量有不同的要求。通常可根据大量运行数据的统计平均，得到其对天然气需求量的一定规律，并依此对未来的负荷变化作出预报或估计，但这种预报往往是比较粗略的。压缩站的作用是提高气体的压力，使之能进行正常传输并满足用户的要求，它是一个耗能单元，把流量为 q 的气体从压力 PS 压缩到具有压力 P_d 时所需消耗的能量为

$$HP=C_1q(t)\left\{\left[\frac{P_d(t)}{PS(t)}\right]^{C_2}-1\right\}$$

其中，C_1、C_2取决于压缩机及所泵入的气体。

在整个网络中，刻划传输动态过程的最重要的单元是传输管道，它们的长度通常为数十到100多公里，直径为300～1200mm。对于一根长管道中的高压气体流，可根据连续性方程、运动方程及气体的状态方程写出描述其动态的偏微分方程

$$\left.\begin{aligned}&\frac{\partial p}{\partial t}+\frac{b^2}{A}\frac{\partial q}{\partial z}=0\\&\frac{\partial q}{\partial t}+A\frac{\partial p}{\partial z}+f\frac{b^2}{DA}\frac{|q|}{p}q=0\end{aligned}\right\}\tag{8-1}$$

其中，t、z 分别表示时间、空间坐标，D为管道直径，A为管

道的截面积，$p(t, z)$ 及$q(t, z)$则为反映天然气流状态的压力和流量，b、f为取决于气体类型和传输管道特性的参量。

对于这样的网络系统，其优化问题可归结如下：

（1）源点的气压和对各用户用气需求量的预报为已知；

（2）管道中高压气体流的压力和流量可由式（8-1）描述；

（3）系统中存在着物理的或需求的约束，这主要是指压缩机的工作范围受到激振、饱和以及最大最小流速的限制，它可近似地用每一压缩站流量和压缩比的上下限约束加以描述。此外，用户对所用气体压力有一定的要求，这一要求可归结为每一用户输出节点压力的下限约束；

（4）能耗最小的优化性能指标

$$\min J=\sum_{i=1}^{m}\int_{t_p}^{t_p+NP}C_{1i}q_i(t)\left\{\left[\frac{u_i(t)}{PS_i(t)}\right]^{C_{2i}}-1\right\}dt \quad (8\text{-}2)$$

其中，m为压缩站的数目，$[t_p, t_p+NP]$为优化时域，通常在几小时到几十小时数量级，并可分为相等或不等的NP个控制时间间隔，在每一时间间隔中，控制量u_i（即每一压缩站的期望输出压力P_{di}）都保持为常量。在优化过程中，优化时域长度、控制时域长度及所划分的步数都可以作为可调参数。

显然，对于上述复杂的优化问题，不可能有解析解，而只能通过数值方法求解。但在对控制量和状态量均存在不等式约束的情况下，采用基于变分原理和极值必要条件的优化求解方法也十分复杂，所以可首先把系统方程（8-1）按地点、时间离散化转化为代数方程、然后再通过参数优化方法求解控制量。鉴于对天然气传输网络已有十分有效的动态仿真程序，所以在参数优化时可采取“黑箱”技术，即以动态仿真程序GANESI作为预测模型，对性能指标及约束条件作出预报，然后配以梯度寻优方法迭代搜索最优的控制作用。其具体过程为：从给定初始状态及某一组假设的$u(1),\cdots,u(NP)$出发，根据已知的源点气压及对用户需求的预报，用动态仿真器计算目标函数值及约束条件值。然后通过对NP个控制量的每一分量进行摄动，用动态仿真器计算后可得到

目标函数及每一约束的梯度值。根据这些函数值和梯度值，可用梯度优化算法迭代改进控制量，直至求得其最优解。这是一个有 $m\times NP$ 个优化变量的梯度寻优算法。由于梯度的计算涉及到对各控制分量的摄动，一共要进行 $m\times\sum_{k=1}^{NP}k$ 次仿真，其数量是很大的，所以在“黑箱”程序中，只计算那些确实需要的信息，以避免不必要的梯度计算。这一“黑箱”优化方法的框图见图8-8。

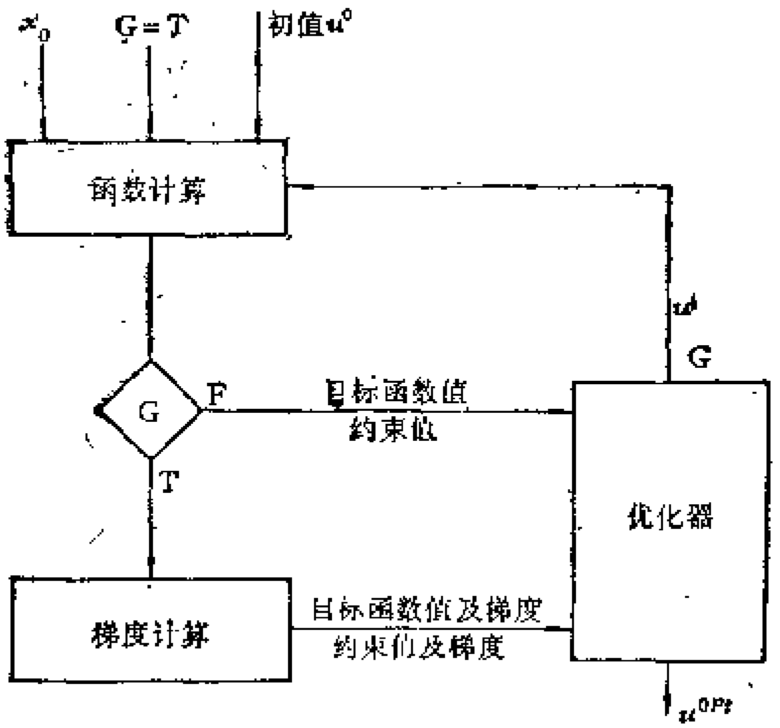


图8-8 天然气网络的“黑箱”优化
T—计算梯度，F—不计算梯度。

上述优化方法可用来离线计算出最优控制作用。但由于模型以及对用户需求预报的不精确性，实施离线优化的结果，不能保证性能指标的最优和约束条件始终满足。为此，可采用滚动形式的闭环优化，即在线进行上述优化，并将第一步的最优控制作用付诸实施。在这一步结束时，根据用户实际需求与预报值的误差，修正对未来需求的预报值。在此基础上，再重复进行上述优化过程。可以看出，这一优化方案是以预测控制的基本原理为基础的，只是在优化时不是利用模型的解析形式，而是用仿真器给出的结果结合梯度法来搜索最优的控制作用。由于仿真器能对性能

指标和约束条件进行直接有效的计算，故使我们在搜索过程中有可能使解进入可行域内并达到能耗最小。由于采用了闭环校正，即使在对用户需求预报不准确时，也能在线修正使优化建立在更加准确的基础上。这种在线修正相当于把动态仿真器进一步发展为动态状态估计器或观测器，在对初始条件缺乏先验知识以及存在大量不可测量状态（如管道内部的压力、流量）的情况下，利用观测器的误差校正方法可使重构状态反映不可测量的真实状态，并用于优化过程的计算。

上述在线优化策略对于图8-9所示的系统进行了研究。图中，源点A提供常压为 5×10^6Pa 的天然气，负荷B的流量需求在 320000～470000m^3/h内变化，平均流量为399000m^3/h，并以24h为周期重复出现（未画出）。压缩站K1和K2的最小流量为150000m^3/h，最大流量为550000m^3/h。要求系统为用户B提供的天然气压力不低于 4×10^6Pa。

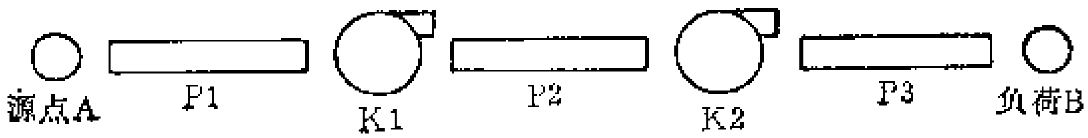


图8-9　天然气传输网络实例

对于上述系统采用了滚动的优化策略，每次滚动的优化时域分为3步，前两步均为1h，后一步为2h。为对比起见，采用了两种优化方式：

方案Ⅰ　只利用模型和需求预报的离线优化。

方案Ⅱ　利用误差校正和预报的在线优化。

当实际需求低于其预报值时（预报误差在－32000m^3/h～－47000m^3/h之间），可得到图8-10（*a*）所示的用户B处的输出气压，无论方案Ⅰ或Ⅱ都能使其保持在低限值以上，但方案Ⅱ的气压比方案Ⅰ更接近临界值，因而在保证用户需求的前提下，可使压缩站的能耗下降为方案Ⅰ的87.6%。

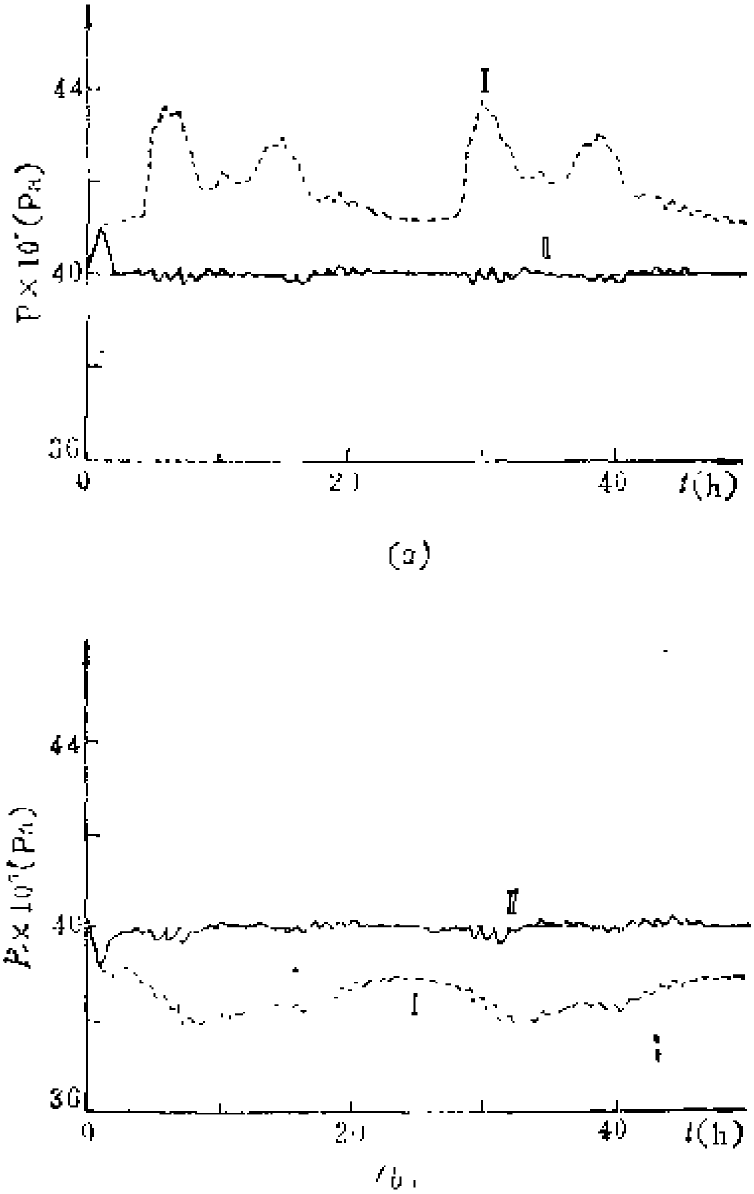


图8-10　用户B处的输出气压

（a）实际需求低于预报值；（b）实际需求高于预报值。

而在用户实际用气量高于原预报值时（预报误差在32000 m^3/h～47000m^3/h之间），由图8-10（b）可见，由于离线优化对用户需求估计不足，加压后的气体在输出点不能满足用户对压力的要求。采用方案Ⅱ后，由于在线优化是建立在闭环校正基础上的，故仍可向用户提供具有所要求压力的天然气。

上述结果表明，滚动优化如果只建立在模型和先验信息的基础上，仍然是一种开环优化，并不能增强控制的鲁棒性。只有把它与闭环校正结合起来，才能通过在线校正实行在线的优化。它可

以在把系统保持在可行工作范围的条件下，达到降低能耗的目的。

对于一个规模庞大的天然气传输网络，按上述方法进行优化，涉及到很大的计算量。为了使在线优化成为可行，应通过递阶分解降低计算的复杂性。在上层，可利用已知的源点气压及所有用户输出点的需求预报，计算出未来一段时间内网络中各压缩站的最优输出压力。这一优化计算的周期与需求预报的频率相适应，一般是长周期（例如24 h一次）的，可由上级高性能的监控计算机实现。其规模虽大，但因周期长，故仍能满足实时需要。而在下层，则对所有与用气点直接邻接的压缩站，以较高的频率（例如1 h）和较短的优化时域（例如4 h）在线进行优化控制。在此过程中，其他压缩站的输出气压可保持为常数。由于时段短减少了优化变量，且规模较小，在线优化计算所需时间在几分钟数量级，与滚动周期1 h相比，仍可满足要求。在这种递阶结构中，上层的任务是对整个网络在控制量为常值的假设下进行简化了的全局优化，而在下层的在线优化，则因只涉及到控制量的一部分，其规模远小于全局优化，因而可提高控制的实时性。

本节提出的在线优化方法已应用在一个含有三个压缩站的天然气传输局部网中。随着优化技术的改进和计算机性能的提高，它将推广到更普遍的天然气传输网络。

§8.4 工业机器人的预测函数控制

预测控制算法一般要比PID控制算法复杂，在线计算量相对较大，对控制实时性的要求也较高，因此在一开始，它的应用主要局限于慢过程的控制，即过程量的控制。然而，随着预测控制的原理日益为人们深刻认识，方法日渐多样灵活，近年来，在快速随动系统中，预测控制也开始得到了应用。文献〔34～35〕提出了一种基于预测控制原理的预测函数控制（Predictive Functional Control，简称为PFC）方法，并成功地应用于工业机器人的快速高精度跟踪控制。以下，我们对此作一简要介绍。

PFC是在预测控制的原理基础上发展起来的。传统的预测控制算法在通过优化过程计算未来的控制作用时，并未注意到控制量的结构性质。它在用于快速随动系统时虽然也能导致满意的输出响应，但却可能伴随着规律不明的控制输入。PFC则把控制输入的结构视为关键问题，并且认为，在输入频谱有限的情况下，控制输入只能属于一组与设定轨线和对象性质有关的特定的函数族。因此，它在预测优化的具体做法上与一般的预测控制算法有所不同。每一时刻加入的控制输入被看作是由若干事先选定的基函数组合而成的，通过它们的已知响应合成系统的输出，在此基础上通过优化求出线性加权系数，再由此算出控制输入。这一算法的基本原理可描述如下：

（1）基函数

新加入的未来控制作用被表示为若干已知函数$f_n(n=1,\cdots,N)$的线性组合

$$u(k+i)=\sum_{n=1}^{N}\mu_n f_n(i),\quad i=0,\cdots,P-1 \tag{8-3}$$

其中，$f_n(n=1,\cdots,N)$称为基函数，$f_n(i)$表示基函数在$t=iT$时的值，P为预测优化时域长度。基函数的选取依赖于对象及期望轨线的性质，例如可取阶跃、斜坡、指数函数等。对于已选定的基函数f_n，可离线算出在其作用下对象的输出响应$g_n(i)$。

（2）误差预测及补偿

对象与模型输出之间的误差被送入一预测器，对未来优化时域中的误差进行预测，并作为前馈量引入到参考轨迹中加以补偿。这里，可选用各种预测方法，例如可取未来的预测误差为$e(k+i)=y(k)-y_M(k)$，其中$y_M(k)$是k时刻的模型输出。

（3）优化控制

在每一时刻进行局部线性化后，提出下面的滚动优化问题：

参考轨迹

$$y_r(k+i)=F_1(y(k),\ c(k+i))$$
$$i=1,\ \cdots,\ P \tag{8-4}$$

其中，$c(k+i)$ 为未来 $k+i$ 时刻的设定值。

模型自由输出

$$y_M(k+i)=F_2(x(k)),\quad i=1,\ \cdots,\ P \tag{8-5}$$

其中：$x(k)$ 为 k 时刻已知的信息，包括过去时刻的控制量和输出量，以及对未来控制输入的已知假设，F_2 是对象预测模型的数学表达式，可取线性差分方程、卷积公式等不同形式。所谓“自由”，是指 k 时刻的输出预测是在未考虑该时刻新加入的控制作用前作出的。

模型函数输出

$$w(i)=\sum_{n=1}^{N}\mu_n g_n(i),\quad i=1,\ \cdots,\ P \tag{8-6}$$

它表示在 k 时刻起加入形为式(8-3)的控制作用 $u(k+i)$ 后新增加的模型响应。在这里和式（8-3）中，PFC 表现出了它与一般预测控制算法的不同之处。新加入的控制输入不是在时间上各自独立的量，而是基函数的线性组合，因此，其引起的输出变化表现为不同基函数响应的叠加，而不是不同时间点控制效应的叠加。由于各基函数及其响应的采样值都已离线算出，所未知的只是线性组合系数 μ_n。

在上述基础上，可得到经误差补偿后的预测输出

$$y_P(k+i)=y_M(k+i)+w(i)+e(k+i)$$
$$i=1,\ \cdots,\ P$$

优化的目标，就是要寻找一组系数 μ_1，…，μ_N，使在整个优化时域内的预测输出尽可能接近参考轨迹。如果把 k 时刻的已知量和未知量分别记为

$$Y=\{y_r(k+i)-y_M(k+i)-e(k+i)$$
$$i=1,\ \cdots,\ P\}$$

$$W=\left\{w(i)=\sum_{n=1}^{N}\mu_n g_n(i),\quad i=1,\ \cdots,\ P\right\}$$

测优化性能指标可一般地表示为

$$\min D(Y, W) \tag{8-7}$$

例如，取二次型性能指标时，有

$$\min\left\{\sum_{i=1}^{P}(y_r(k+i)-y_P(k+i))^2\right\}$$

这是一个$\mu_1, \cdots, \mu_N$的参数优化问题。由此解出μ_n($n=1, \cdots, N$)后，即可由式（8-3）综合出k时刻应加入的函数输入$u(k+i)$($i=0, \cdots, P-1$)。其中，除$u(k)$可作为新加入的控制作用付诸实施外，其余量可叠加到对未来控制量的已知假设上，作为下一步计算模型自由输出的基础。这一优化过程将以滚动方式在线进行。每一时刻求取新加入控制作用的原理可见图8-11。

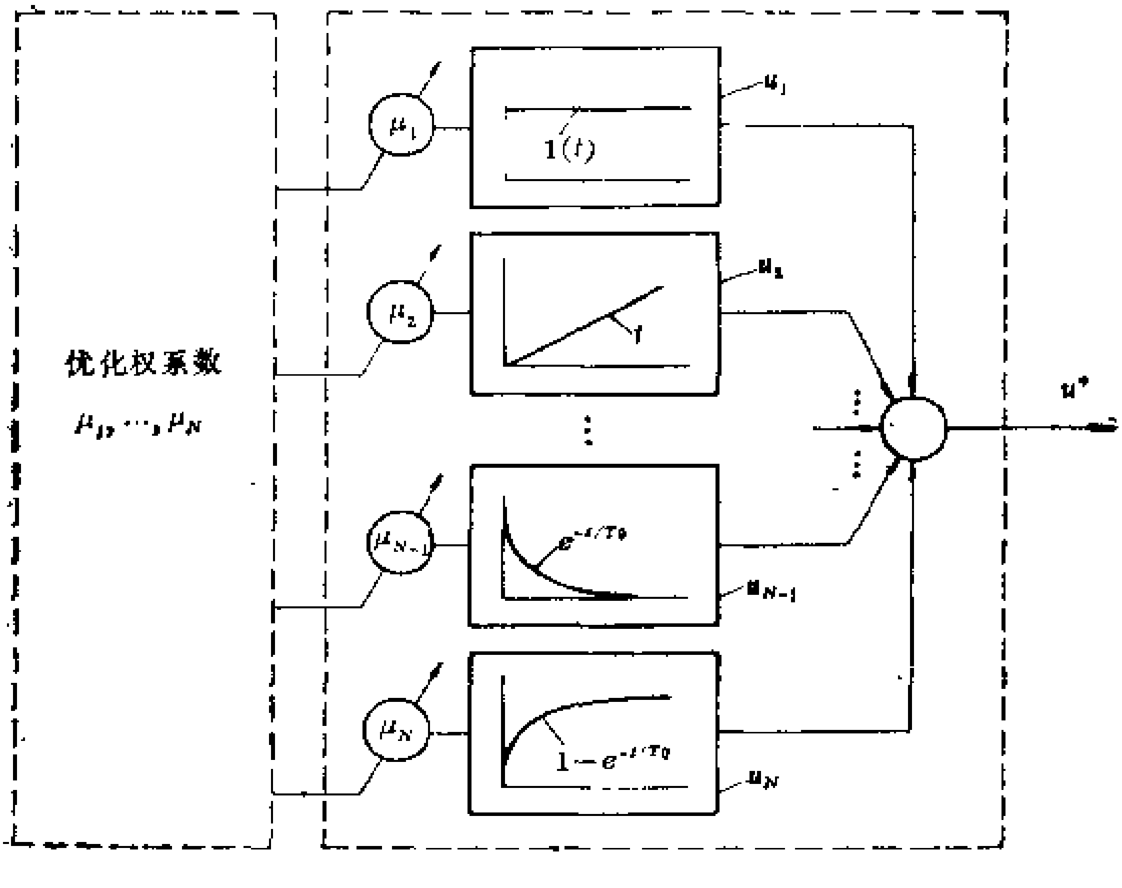


图8-11 利用基函数加权的优化

在PFC算法中，作为设计参数的是预测优化时域长度P、反映参考轨迹时间常数的参数α(见式(3-16))以及基函数f_1，…，f_N，它们将影响控制的精度、稳定性（包括鲁棒性)和动态响应。PFC的显著优点在于，这些设计参数对控制性能的影响恰好是各有侧重的。控制精度主要取决于基函数的选择，动态响应主要受参考轨迹的影响，而预测优化时域P则对控制的稳定性和鲁棒性起主要作用。这样，在控制系统设计时，便能根据性能要求很快地整定参数。

与其他基于模型的控制算法（如非线性解耦控制、自适应控制等）相比，PFC的在线计算量显著减少。这是因为，基函数及其响应的采样值都可事先离线算出，在线只须对少量线性加权系数进行参数优化即可。这一算法对采样周期没有特殊要求，甚至在采样周期相当大时（10ms≤Δt≤20ms）也能取得好的效果。

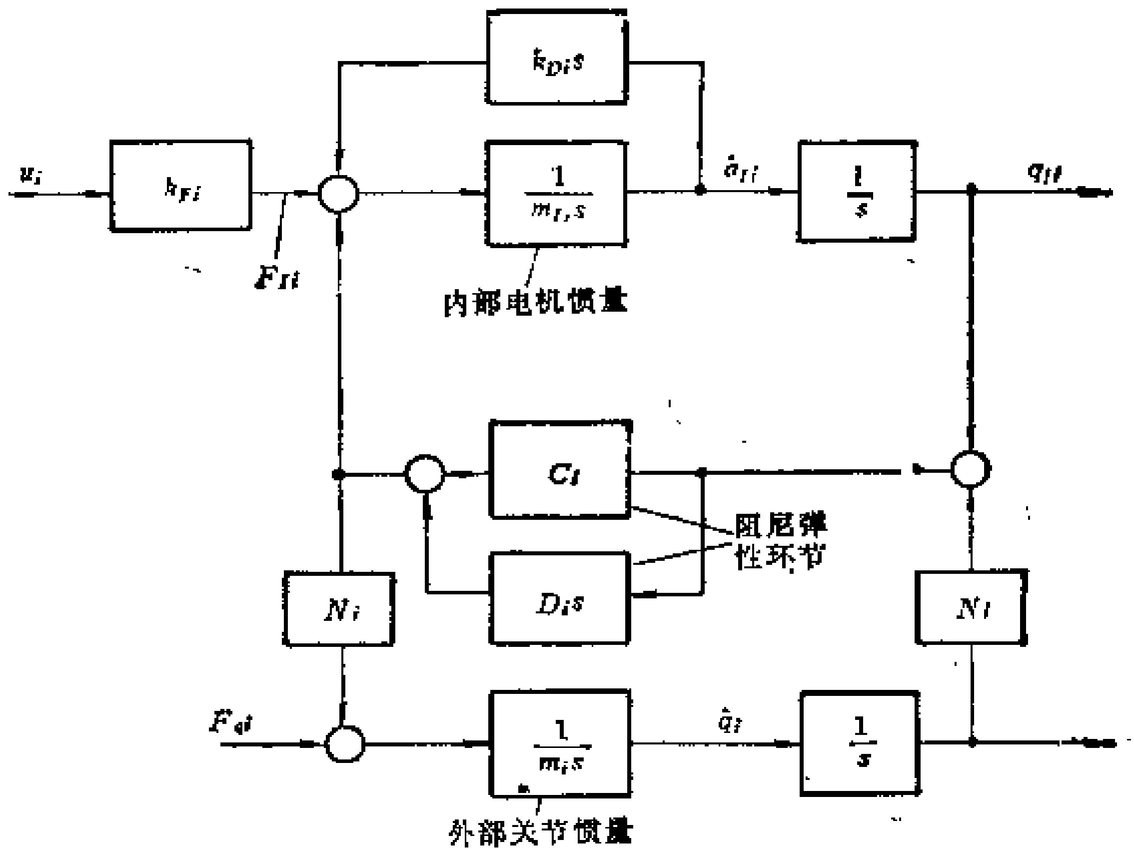


图8-12　各关节的内部模型

PFC算法用在具有6个自由度的工业机器人KUKA160上，取得了良好的效果。这一机器人的动力学可由具有弹性非线性结构的全局模型加以描述。为了简化设计，可从这一模型出发，把关节间的耦合和摩擦力视为扰动，并把其他关节惯量对关节 i 的影响 m_i 近似作为常数，这样，就可建立起图 8-12 所示的各关节独立的线性内部模型。

在图 8-12 中，m_{ri}为电机惯量，m_i为外部关节惯量，k_{Di}为电机阻尼，N_i为传动比。注意，其中用阻尼弹性环节 c_i、D_is 近似描述了滑动附着摩擦。利用测速码盘，可以求出关节坐标，它可用电机轴转角q_{ri}表示，也可用传动机构输出端的转角q_i表示，它们都可选作为系统的被控量。而作为控制量的则是电机驱动电流 I_i。

若选择关节坐标q_i为输出量，则可得到传递函数

$$\frac{Q_i(s)}{I_i(s)}=\frac{k_{Fi}(b_1s+b_0)}{s(a_3s^3+a_2s^2+a_1s+a_0)}$$

其中，参数 a_i、b_i取决于上述物理参数，可通过实验和物理分析求出。由于被控对象有一积分环节，故可取指数型的基函数

$$f_1(t)=e^{-t/T_0},\quad f_2(t)=1-e^{-t/T_0}$$

并对其进行PFC控制。

该算法由实时语言 PEARL编制，并在工业机器人专用多处理器控制器MPR5000上实现。图 8-13 为当取采样周期$\Delta t=10\mathrm{ms}$且第 2 个关节在设定值有1°阶跃变化时，分别用PFC和PI控制的控制量 I_2及输出偏差e_2的测量值。显然，前者有快速无超调的响应，控制量只有很小的峰值且无振荡。

此外，PFC控制对于模型参数的变化有很强的鲁棒性。当图 8-12中的阻尼弹性系数c_i有±50%变化、外部关节惯量m_i有±80%变化或阻尼常数有±50%变化时，仍能获得良好的控制性能。

市售的工业机器人很多带有测速电机，可构成速度反馈，并配备有模拟的电流—转速串级控制。针对这一硬件结构，应把模拟控制回路包含在预测模型中，再考虑相应的PFC控制方案，如图 8-14 所示。

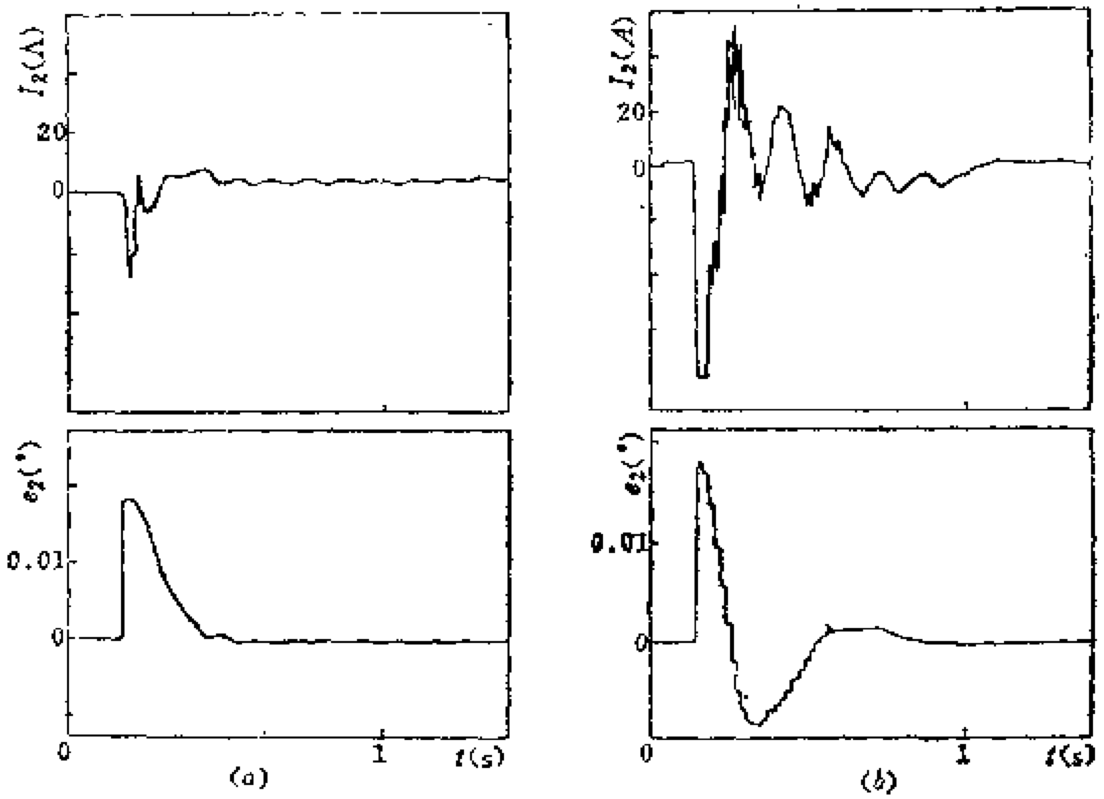


图8-13　设定值1°阶跃时的控制量和输出偏差
（a）PFC控制；（b）PI控制。

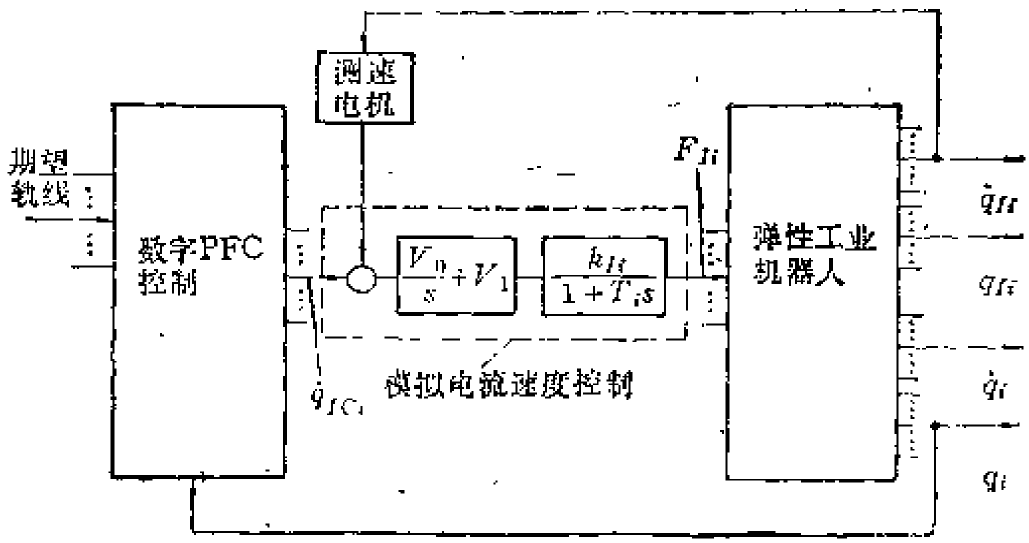


图8-14　带有速度反馈的PFC控制

这时，可把速度环的设定值 $\dot q_{Ici}$ 作为控制量，相应的模型传递函数为

$$\frac{Q_i(s)}{\dot Q_{Ici}(s)}=\frac{N_i}{s\,(s^2+2\,d_i\omega_i s+\omega_i^2)}$$

在对该机器人进行标准测试时，要求其以 $v=0.5\mathrm{m/s}$ 的速度在空间高精度地反复画一个直角三角形。在其直角转角处圆弧半径规定为 $R=25\mathrm{mm}$ 时，由图8-15（a）可见，PI 控制的最大偏差为8.5mm，而用PFC控制则只有2.0mm。当 $R=2\mathrm{mm}$ 即要求其相当严格地走直角时，PI 控制的偏差更大，达到了16.5mm，而用 PFC 控制的最大偏差只有3.5mm。由此可见，PFC 算法在工业机器人高速高精度跟踪控制中，可取得比PI控制好得多的效果，这主要是因为在PFC控制中包含了预测的机制。

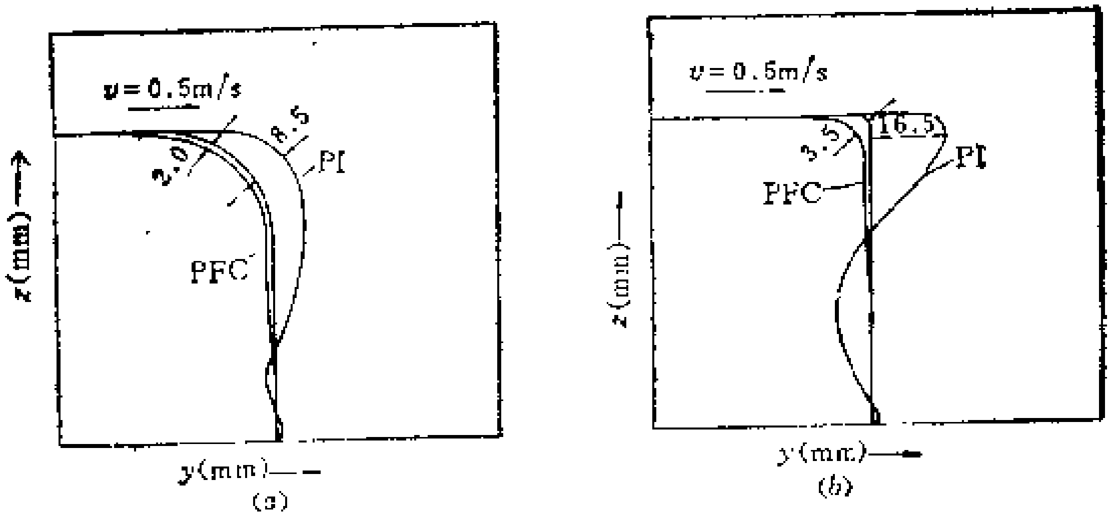


图8-15　对期望轨线的高速跟踪
（a）$R=25\mathrm{mm}$；（b）$R=2\mathrm{mm}$。

预测控制在快速随动系统中的应用还只是开始，其关键在于要有简易的在线算法以提高控制的实时性。因此，不仅需要采用简化的参数模型，而且需要发展新的控制策略和控制算法。这一领域所提出的挑战必将有力地推动预测控制的发展，并拓宽应用预测控制的思路。

第九章　预测控制的发展前景

以上各章，我们介绍了预测控制的基本原理和算法，讨论了其分析和设计问题，并展示了它在应用于复杂系统时实用化和多样化的发展。这些内容在一定程度上反映了当前预测控制领域的发展水平，也为正确认识预测控制的地位提供了基础。

在目前预测控制的文献中，理论分析大多集中于单变量的基础算法，成功应用的报导却大多是复杂的多变量系统。这种矛盾反映了预测控制的理论研究落后于实际应用，但并不意味着对其价值的认识应立足于单变量基础算法。作为一种适应复杂工业过程控制需要而产生的新型控制方法，预测控制应被看作是多变量优化控制的重要组成部分，并在这一认识的基点上，正确地评价和应用它。当然，预测控制可以作为基础级的控制算法，并因其适用于时滞或非最小相位对象、易于建模、跟踪性能好，而具有一定的优点。但要使它发挥更大的作用，更合适的是把它用作为动态优化级的多变量控制算法。这样，才有可能在多种性能要求和有约束的情况下，实现系统的综合优化控制，取得更为显著的效益。第八章中所介绍的预测控制在大工业过程中的应用范例，正是从实际上描绘了预测控制在工业控制中的适当地位。只有认识了这一点，才能开阔视野，提高预测控制的应用规模和应用水平。

预测控制自从问世以来，已在复杂工业过程的控制中展现出诱人的应用前景，从而自然引起了工业控制界和控制理论界的重视。近年来，关于预测控制研究的文献日渐增加，其应用领域也从最初的化工、石油、电力部门的过程控制延伸到网络、冶金、轻工、机械等部门或系统[36]。这些，正反映了人们对这一领域日益增长的兴趣。与70年代后期提出预测控制之时相比，当前人

们对于预测控制的研究和应用，无论在深度上还是广度上都有了很大发展，形成了算法、理论和应用的多方位研究。这些研究有力地推动了预测控制向更高层次的发展。

预测控制的算法研究，是对于现有预测控制算法的修正、推广或发展，也是当前这一领域研究的集中点。这里虽然包含了为简化设计和计算所进行的技术性修正，如导出递推算法、在控制间隔内采用多步预测等，但更重要的是包含了预测控制算法在概念上的发展，这主要表现在以下几个方面：

（1）预测控制的基础算法，除了最初从工业过程中直接产生的以非参数模型为基础的MAC、DMC等算法外，还出现了以辨识模型、状态方程模型为基础的算法。理查勒特当初提出的预测控制三要素之一的脉冲响应模型，已为更一般的预测模型概念所代替。

（2）在滚动优化中，尽管对原有的优化算法有不少策略上的修正，但线性目标、无穷范数目标、带状目标等的出现，已突破了原有算法二次型性能指标的局限。

（3）在求解优化问题时，面向实际工业过程提出的有约束优化问题，解析求解的可能性越来越小，注意力开始转移到如何确切地描绘在线参数优化问题以及寻找各种有效的求解算法上。这里，除了基于规划理论的数值寻优方法外，甚至出现了仿真器辅助寻优这类非经典的求解方法。

（4）变校正系数的误差预测、扰动观测器、自校正、自学习等引入预测控制算法，开辟了处理非模型信息的新途径，使原有的误差预测概念发展为更一般的闭环校正概念。

（5）随着预测控制的推广应用，适应于不同的复杂对象，各种新的预测控制算法陆续出现，如非线性预测控制、预测函数控制、分散预测控制等，它们的算法形式已与原来的基础控制算法有很大不同。预测控制的内涵已有了新的发展。

这些研究的多样性，拓宽了预测控制的适用范围，同时也标志着人们对于预测控制的理解正在发生深刻的变化。预测控制已突破了算法的局限，而以其共同蕴含的机理成为一种具有鲜明特

色的新型控制方法。第六到第八章所介绍的预测控制面向应用的发展，正体现了这一转变。当我们再回顾第二章所介绍的预测控制的基本原理时，就可看到，这形形色色的算法虽然各有特色，但都包含着共同的方法思想，即预测模型、滚动优化和反馈校正。

作为一种新型的控制方法，预测控制提供了在复杂环境下有效地利用过程信息实现优化控制的途径。它的上述三个基本特征，正是一般控制论中模型、控制、反馈概念的具体体现。由于强调模型的功能而不是结构，使它可以根据对象的特点和控制的要求，以最简易的方式集结信息建立预测模型。由于采用了对模型外信息的辅助预测和非经典的优化模式，使它可以把实际系统中的多种复杂因素考虑在优化过程中，形成动态的优化控制，并可处理约束和多种形式的优化目标。预测控制所蕴含的这些方法原理，不仅为现有算法的改进和应用提供了有益的启示，而且对于拓宽其应用的范围，指导新算法的开发以及预测控制向智能化的发展，也有着重要的意义。

工业生产优化控制的需要，是预测控制产生的直接动因，也始终是它发展的强大动力。随着计算机系统的普遍配置及工业生产规模水平的日益提高，人们对工业控制提出了越来越高的要求。这里不但要在设备的物理限制下追求良好的控制质量，而且要考虑经济性、安全性、对环境的适应性及用户友好性。面对用传统的模型和优化方法难以描述与处理的大量不确定性，更需要控制系统具有较高的智能水平。为此，工业界已提出要进一步解决不确定性的描述、不确定环境下的优化、多目标优化、专家系统、高速计算等面向实际的问题[37]。预测控制面临这一挑战，同样需要从人工智能、控制理论等多方面汲取有价值的思想，向智能化的方向发展。

预测控制与智能控制的结合，不是抛弃原有的预测控制算法，而是在有不确定性、多目标要求等复杂情况下对现有算法的补充和发展。预测控制的智能化，在结构上表现为控制的递阶分层，在算法上表现为非常规的模型预测和优化。而预测控制的一般方

法原理，正为这种智能化发展提供了可能性。

预测控制在选择有效的信息集合作为预测模型时，只强调其实现预测的功能，而对其结构类型没有限制，模型的形式甚至可以是非经典的。这样，就可根据实际对象的复杂程度，建立恰如其分的预测模型。在对象或环境的动态变化规律难以用通常的数学手段描述时，甚至还可利用模糊关系、逻辑关系或规则集建立预测模型。这里的原则只有两条：一是预测模型应能正确反映输入输出间的动态因果关系，二是建模过程和模型的表达应尽量简便。这种对模型结构约束的突破为复杂工业系统的建模提供了很大自由度，使我们可在常规模型的基础上，辅之以用人工智能技术和启发式方法建立起来的非常规模型，更全面准确地预测出系统未来的行为。

以误差预测补充模型预测的综合预测，是一类预测控制算法中常用的策略。其中，模型预测反映了预测模型所描述的确定的动态因果关系，而误差预测则描述了在该模型中未能包含的一切不确定性，它作为模型预测的重要补充，可以弥补在基础模型中已简化、不可知或无法加以考虑的一切其他因素的影响。在单层次的预测控制算法中，误差预测由于缺乏模型，只能建立在非因果的基础上。但如果对引起误差的原因还有某些知识，则可根据这些知识建立起误差的预测模型，进一步将原来的误差预测分解为因果和非因果的综合预测，形成多层预测的策略。这种多层预测与多层建模是相适应的。每一层的模型被用来进行因果预测，而在这一层不知原因的误差，到上一层又可通过不同形式的预测模型作进一步的因果预测。各层次模型预测的综合，可最大限度地利用已知信息中的因果部分来提高预测的准确度。而其分层分解实现，则可降低对每一层基础模型的要求，克服一揽子建模的困难，并有利于引入智能启发式的模型与预测概念，最大程度地克服不确定性的影响。

预测控制的优化模式具有鲜明的特点，它的离散时间的有限优化目标及滚动推进的实施过程，使其在控制的全过程中表现为

动态优化，而在控制的每一步所解决的只是参数优化即数学规划问题。这种规划问题的一般化，可使它处理有约束、多目标、非线性乃至非参数目标等更复杂的情况。汲取规划中的分层思想，我们可把目标按其重要性分层，在各层次建立相应的预测模型，并实施不同形式的优化。在这里，同样可以采用非定量的性能指标和非经典的寻优手段，以适应目标和模型的非经典性。由于多种目标被分散到不同层次及规划方式的多样性，无疑可以增强预测控制对于复杂系统综合性能要求的适应性。

由此可见，预测控制向智能化的发展，将形成多层智能预测控制的模式[7]。在这种模式中，控制的目标被划分到不同的层次。在每一层次，都有与目标相适应的预测模型和优化方式，它们可以是经典的，也可以是智能启发式的。对象的先验信息将按层次集结成不同类型的模型，而实时信息将反馈到各个层次中予以识别，并补充基于模型的预测。在较低层中的不确定性，可在较高层上进一步识别和描述，有针对性地加以优化处理，而智能启发方法将渗透到预测和优化策略中。与原来单一模型的预测控制算法相比，这种控制概念不但具有适应性质，而且可以有效地处理非参数的信息规律，因而可用来解决预测控制算法中的参数在线设计和调整问题，并可拓宽原有算法的应用范围，处理动态复杂（如非线性对象）、目标多样（包括约束和非经典的目标）、不确定性难以参数化的复杂系统的控制。事实上，目前预测控制的研究，已在不同程度上反映了这一趋势。模型、目标的多样性、控制的层次性、启发式寻优和校正，已在第六到第八章的许多部分可见。这种多层智能预测控制方法的研究，将有力地克服单一模型预测控制算法的不足，进一步增强预测控制处理复杂对象和综合目标的能力。

预测控制作为一种新型的控制方法，对于未来工业过程的控制必将产生重大的影响。虽然在这一领域内，还存在着许多迫切需要解决的理论和实际问题，它的工业应用也将不断提出各种新问题，但其基本原理对于复杂系统的适应性，无疑是富有吸引力的。预测控制的深入研究和推广应用，将对我国国民经济的发展

和工业自动化水平的提高产生积极的影响。而预测控制概念的进一步延拓，也将为大工业过程的集成优化提供有力的支持。

参考文献

1 Richalet J, et al. Model Predictive Heuristic Control, Applications to Industrial Processes. Automatica, 1978, 14(5): 413～428

2 Rouhani R, Mehra R K. Model Algorithmic Control (MAC); Basic Theoretical Properties. Automatica, 1982, 18(4): 401～414

3 Cutler C R, Ramaker B L. Dynamic Matrix Control——A Computer Control Algorithm. In: Proceedings of the 1980 Jiont Automatic Control Conference V.1. San Francisco: American Automatic Control Council, 1980. WP5-B

4 De Keyser R M C, Van Cauwenberghe A R. Extended Prediction Self-adaptive Control. In: Barker H A, Young P C eds. Identification and System Parameter Estimation 1985 V.2. Oxford: Pergamon Press, 1985. 1255～1260

5 Clarke D W, Mohtadi C, Tuffs P S. Generalized Preditive Control. Automatica, 1987, 23(2): 137～162

6 Garcia C E, Morari M. Internal Model Control, 1. A Unifying Review and Some New Results. IEC Process Des. Dev., 1982, 21(2): 308～323

7 席裕庚，许晓鸣，张钟俊. 预测控制的研究现状和多层智能预测控制. 控制理论与应用, 1989, 6(2): 1～7

8 Schmidt G, Xi Y. A New Design Method for Digital Controllers based on Nonparametric Plant Models. In: Tzafestas S G, ed. Applied Digital Control. Amsterdam: North-Holland, 1985. 93～109

9 席裕庚. 预测空间中数字控制器的设计. 控制理论与应用, 1987, 4(3): 66～73

10 Xi Y. Beiträge zur Analyse und Synthese eines prädiktiven Regelverfahrens: [Dr.-Ing. Dissertation]. München: Technische Universität München, 1984

11 Xi Y. Minimal Form of a Predictive Controller based on the Step-Response Model. Int. J. Control, 1989, 49(1): 57～64

12 袁璞. 预估控制及其在过程控制中的应用. 见: 孙优贤等编. 中国自动化学会首届过程控制科学报告会论文集. 杭州: 浙江大学出版社, 1988. 347～356

13 席裕庚，孟祥恺. 典型过程系统预测控制的鲁棒性分析. 见: 孙优贤等编. 中国自动化学会首届过程控制科学报告会论文集. 杭州: 浙江大学出版社, 1988 74～82

14 Jury E I . A Stability Test for Linear Discrete Systems in Table Form. Proc. IRE, 1961, 49(12): 1947~1948

15 Xi Y, Meng X. Analytical Design of Predictive Control for a Class of Industrial Processes. In: Preprints 11th IFAC World Congress V. 5. Tallinn: IFAC, 1990. 222~227

16 Xi Y. New Design Method for Discrete-Time Multivariable Predictive Controllers. Int. J. Control, 1989, 49(1): 45~56

17 席裕庚，季红彬．多变量系统预测控制的解耦设计．见：俞金寿等编．中国自动化学会第二届过程控制科学报告会论文集．上海：华东化工学院出版社，1989. 124~131

18 Prett D M, Gillete R D. Optimization and Constrained Multivariable Control of a Cataytic Cracking Unit. In: Proceedings of the 1980 Jiont Automatic Control Conference V. 1. San Francisco: American Automatic Control Couancil, 1980. WP5-C

19 张佩星，席裕庚．有约束的多变量系统预测控制．见：孙优贤等编．中国自动化学会首届过程控制科学报告会论文集．杭州：浙江大学出版社，1988. 83~90

20 Xu X M, Xi Y G, Zhang Z J. Decentralized Predictive Control (DPC) of Large Scale Systems. Information and Decision Technologies, 1988, 14(3): 307~322

21 Chartfield C. The Analysis of Time Series: An Introduction. 3rd ed. London: Chapman and Hall, 1984

22 席裕庚，许颂华．多关节机器人的分散预测控制．控制理论与应用，1992, 9(1): 37~42

23 席裕庚，李彤．广义动态矩阵—PID串级控制．见：俞金寿等编．中国自动化学会第二届过程控制科学报告会论文集．上海：华东化工学院出版社，1989. 145~151

24 席裕庚，许颂华，孟祥恺．典型工业对象的自校正广义动态矩阵控制．见：控制理论及其应用1988年年会论文集．山东曲阜：中国自动化学会控制理论专业委员会，1988. 599~602

25 Zhang Z, Xi Y, Xu X. Hierarchical Predictive Control for Large Scale Industrial Systems. In: Isermann R, ed. Automatic Control Tenth Triennial World Congress of IFAC V. 7. Oxford: Pergamon Press, 1988. 91~96

26 Campo P J, Morari M. Robust Model Predictive Control. In: Proceedings of the 1987 American Control Conference V. 2. Minneapolis: American Automatic Control Council, 1987. 1021~1026

27 孙浩．一类非线性系统预测控制的分层策略：[博士学位论文]．上海：上海交通大学，1990

28 Harris K, Bell R D. Model Algorithmic Adaptive Control for Non-

linear Systems. In: Proceedings of the 23rd IEEE Conference on Decision and Control V. 2. Las Vegas: IEEE Inc., 1984. 681~682

29 黄河清，黄道，俞金寿．一类非线性系统的模型算法控制的研究．见：孙优贤等编．中国自动化学会首届过程控制科学报告会论文集．杭州：浙江大学出版社．1988. 195~202

30 顾钟文，邹志云，罗文魁．一类非线性系统的预测控制算法．见：孙优贤等编．中国自动化学会首届过程控制科学报告会论文集．杭州：浙江大学出版社，1988. 186~194

31 Kelly S J, Rogars M D, Hoffman D W. Quadratic Dynamic Matrix Control of Hydrocracking Reactors. In: Proceedings of the 1988 American Control Conference V. 1. Atlanta: American Automatic Control Council, 1988. 295~300

32 Cutler C R, Hawkins R B. Constrained Multivariable Control of a Hydrocracker Reactor. In: Proceedings of the 1987 American Control Conference V. 2. Minneapoils: American Automatic Control Council, 1987. 1014~1020

33 Marques D, Morari M. On-line Optimization of Gas Pipeline Networks. Automatica, 1988, 24(4): 455~469

34 Richalet J, et al. Predictive Functional Control: Application to Fast and Accurate Robots. In: Isermann R, ed. Automatic Control Tenth Triennial World Congress of IFAC V. 4. Oxford: Pergamon Press, 1988. 251~258

35 Jacubasch A, et al. Anwendung eines neuen Verfahrens zur schnellen und robusten Positionsregelung von Industrierobotern. Robotersysteme, 1987, 3(3): 129~138

36 Garcia C E, Prett D M, Morari M. Model Predictive Control: Theory and Practice — a Survey. Automatica, 1989, 25(3): 335~348

37 Prett D M, Garcia C E. Design of Robust Process Controllers. In: Isermann R, ed. Automatic Control Tenth Triennial World Congress of IFAC V. 2. Oxford: Pergamon Press, 1988. 275~280