

熊光楞 沈被娜 宋安澜 编著

控制系统仿真 与模型处理

科学出版社

控制系统仿真与模型处理

熊光楞 沈被娜 宋安澜 编著

科学出版社

1993

(京)新登字 092 号

内 容 简 介

仿真与模型处理是控制系统计算机辅助设计中的两个重要内容。本书是在总结作者 10 年来研制控制系统计算机辅助设计软件包工作经验的基础上写作而成。全书共分七章,内容包括:绪论,模型转换,模型分析,模型简化,线性系统仿真,非线性系统仿真,仿真与模型处理软件包等。

本书的主要特点是:从体系上力求将原理与方法介绍和具体算法程序相结合,并落实到通用的软件包上,使读者读后能真正解决将先进的控制理论应用于工程实际中去的难题。本书各章末均有若干个经多次测试通过的实用程序,可供学生上机实验及工程技术人员实际应用。本书所列全部程序都已制成软件,软件介质为 5 英寸软盘,可运行在各种类型的微机上,此软件与书同时发行。

本书可供从事自动控制和计算机仿真等领域工作的科研和设计人员学习参考,也可作为大专院校自动控制及有关专业的参考教材。

控制系统仿真与模型处理

周德庆 姚波娜 宋安澜 编著

责任编辑 鞠丽娟

科学出版社 出版

北京东黄城根北街 16 号

邮政编码 100724

中国科学院印刷厂印刷

新华书店北京发行所发行 各地新华书店经售

*

1993 年 7 月第 一 版 开本: 850×1168 1/32

1994 年 7 月第一次印刷 印张: 11 3/4

印数: 1—2 600 字数: 309 000

ISBN 7-03-003407-4/TP·250

定价: 13.50 元

前 言

控制系统计算机辅助设计是控制理论和计算机结合用于控制系统分析与设计的一门综合性技术,主要包括三大方面:

(1) 设计原理与方法,主要包括:模型建立,模型处理,设计与仿真等原理及方法。

(2) 有关的数值计算方法及实用程序研究,如:矩阵变换运算,数值积分算法和模型的结构分解算法等。

(3) 软件技术,包括:软件结构,人-机交互技术和工程数据库管理技术等。

本书作者参加了由国家自然科学基金资助的控制系统计算机辅助设计的研究工作,并负责其中的模型处理与仿真软件包的研制工作。这项研究工作前后历时 10 年,共研制了两个版本,其中,第一版本是采用 FORTRAN 语言编写的,第二版本是采用 C 语言编写的。上述两个版本的软件包分别于 1986 年和 1991 年进行了鉴定,专家们一致认为:该软件包功能齐全,算法先进,达到了国际上同类软件包的先进水平。

作者在总结 10 年来研制工作的经验以及用户们在使用中所提出问题的基础上写成了本书,本书写作的主要出发点是:

(1) 从体系上将设计原理、方法介绍与实用程序相结合,并落实到模型处理与仿真软件包上,使上述三部分内容有机地结合在一起。

(2) 本书除可作为大专院校有关控制专业或自动化专业进行控制系统模型处理与仿真的教学参考书外,还可作为广大工程技术人员的自学参考书。书中每章均提供了若干个经过多方面测试的实用程序以供读者使用。

本书共分七章。第一章是绪论,主要介绍控制系统计算机辅

助设计的发展历史、技术内容及本书的写作意图。第二章讲述线性多变量系统模型转换,包括线性多变量系统的描述形式,连续系统模型转换,离散系统模型转换,连续与离散模型之间的转换。第三章讲述线性多变量系统的模型分析,包括稳定性分析,标准形与结构化分解,并着重分析数值计算中的精度、误差及稳定性。第四章讲述模型简化技术,包括在状态空间模型上进行简化及在传递函数模型上进行简化。第五章讲述线性系统仿真,包括离散相似法及时域矩阵法,同时还将介绍线性系统仿真模型建立中的几个重要问题:仿真模型与最小实现,初值问题与能观标准形,结构图变换等。本章还要讲述几个特殊问题,如:线性采样系统仿真,具有延迟环节的线性系统仿真,多帧速仿真等。第六章讲述非线性系统仿真,包括数值积分法及扩展的离散相似法。与第五章一样,也要讲述几个与非线性系统仿真模型建立的有关问题及非线性系统仿真的几个特殊问题。第七章介绍模型处理及仿真软件包。

总之,全书除绪论外大致可分成三大部分,第一部分介绍模型处理(第二、三、四章),第二部分介绍仿真(第五、六章),第三部分介绍模型处理与仿真软件(第七章)。这三部分相对独立,所以读者在自学时,可根据需要加以选择,而不必完全从头读起。这也是本书在安排上的一大特点。

考虑到目前大多数大专院校学生及工程技术人员对 FORTRAN 语言比较熟悉,本书中所提供的各种实用程序均用 FORTRAN 语言编写。若读者对用 C 语言编写的实用程序感兴趣,则可与作者联系。

本书第一、五、六章由熊光楞编写,第二、三章由沈被娜编写,第四、七章由宋安澜编写,全书由熊光楞主编。在本书写作过程中,得到了控制系统计算机辅助设计研制组许多同志的帮助,特别是得到了研制组组长、中国科学院系统科学研究所韩京清研究员,南开大学王治宝教授和王秀峰教授的热情指导,特在此向他们表示衷心的感谢。

作者

1992年6月

目 录

第一章 绪论	1
1.1 控制系统计算机辅助设计的发展历史及其特点	1
1.2 CADCS 的技术内容	5
1.2.1 开发模型	6
1.2.2 设计要求的形式化	6
1.2.3 进行设计	8
1.3 模型处理及仿真在 CADCS 中的地位	8
第二章 线性多变量系统模型转换	11
2.1 引言	11
2.2 线性多变量系统的描述形式	12
2.2.1 线性连续系统模型描述	12
2.2.2 线性离散系统模型描述	19
2.3 线性多变量系统模型转换	22
2.3.1 连续系统模型转换	22
2.3.2 离散系统模型转换	37
2.3.3 连续与离散系统之间的转换	44
2.4 连续系统模型转换算法介绍及评注	46
2.5 多变量系统模型变换部分子程序说明及程序 清单	51
2.5.1 子程序说明	51
2.5.2 程序清单	55
第三章 线性多变量系统的模型分析	63
3.1 引言	63
3.1.1 矩阵的初等运算	63
3.1.2 矩阵的相似变换	64

3.1.3	矩阵的秩	64
3.1.4	特征多项式和特征值	65
3.1.5	正定矩阵	65
3.1.6	正交性	66
3.2	线性多变量系统的稳定性分析	66
3.2.1	连续系统的稳定性判别	66
3.2.2	离散系统的稳定性判别	72
3.3	线性系统的标准形与结构分解	75
3.3.1	概述	75
3.3.2	线性系统的标准形	76
3.3.3	线性系统的结构分解	79
3.3.4	小结	83
3.4	数值计算中的精度、误差及稳定性	85
3.5	线性多变量系统模型分析部分子程序说明	88
3.5.1	子程序说明及程序清单	88
3.5.2	程序清单	91
第四章	模型简化技术	98
4.1	引言	98
4.2	在状态空间模型上进行简化	99
4.2.1	集结法	100
4.2.2	摄动法	103
4.3	在传递函数模型上进行简化	106
4.3.1	Pade 逼近法	107
4.3.2	连分式法	111
4.3.3	结合稳定判据的混合方法	128
4.3.4	频率拟合 Pade 法	130
4.3.5	评注	136
4.4	模型简化部分子程序说明及程序清单	136
4.4.1	程序简要说明	136
4.4.2	程序清单	145
第五章	线性系统仿真	162
5.1	引言	162

5.2 线性系统的数学模型	163
5.2.1 连续系统的数学模型	164
5.2.2 离散时间系统的数学模型	166
5.2.3 连续-离散混合系统的数学模型	168
5.3 线性系统仿真方法	170
5.3.1 离散相似法	171
5.3.2 时域矩阵法	181
5.4 线性系统仿真模型的建立	186
5.4.1 仿真模型与最小实现	186
5.4.2 初值问题与能观标准形	191
5.4.3 面向结构图的仿真	194
5.5 特殊问题讨论	207
5.5.1 矩阵指数 e^{AT} 的数值计算	207
5.5.2 采样数据系统的仿真	211
5.5.3 多帧速算法	212
5.5.4 延迟环节的仿真	212
5.6 线性系统仿真中的有关程序	214
5.6.1 利用泰勒级数展开法计算 $\phi(T)$ 及 $\phi_m(T)$ 的程 序 (EAT)	214
5.6.2 离散相似法中计算状态转移矩阵的子程序(EA) (缩方与乘方算法)	217
第六章 非线性系统仿真	220
6.1 概述	220
6.2 非线性系统仿真方法	221
6.2.1 数值积分法	221
6.2.2 采用数值积分法时对步长的控制	232
6.2.3 扩展的离散相似法	234
6.3 非线性系统仿真模型的建立	235
6.3.1 面向方程的非线性系统仿真模型	235
6.3.2 面向结构图的非线性系统仿真模型	235
6.3.3 非线性采样系统的仿真	236
6.3.4 延迟环节的仿真	237

6.4 特殊问题的讨论	238
6.4.1 病态系统仿真	238
6.4.2 病态性探测	248
6.4.3 多帧速算法	252
6.5 非线性系统仿真程序	256
6.5.1 程序简要说明	256
6.5.2 程序清单	264
第七章 模型处理与仿真软件包	287
7.1 问题的提出	287
7.2 模型处理与仿真软件包的构成	288
7.2.1 功能选取	288
7.2.2 算法选取	290
7.2.3 人机交互方式选取	293
7.3 软件技术介绍	295
7.3.1 数据结构	295
7.3.2 软件结构	299
7.3.3 编译技术	301
7.3.4 调试方法	302
7.4 模型处理与仿真软件包 SMPACK 的使用	303
7.4.1 SMPACK 的命令系统	303
7.4.2 SMPACK 的运行	309
7.4.3 线性系统仿真命令 (SL)	311
7.4.4 面向微分方程的仿真命令 (SE)	316
7.4.5 面向结构图的仿真命令 (SB)	317
7.4.6 模型处理命令	323
7.4.7 其它命令	325
7.5 应用实例	327
7.6 SMPACK 软件包控制、译码模块的 FORTRAN 程序清单	351
7.6.1 程序简要说明	351
7.6.2 程序清单	355
参考文献	366

第一章 绪 论

1.1 控制系统计算机辅助设计的发展历史及其特点

计算机辅助设计技术(简称 CAD)是在计算机技术得到迅速发展后出现的一门新兴技术,它可以利用计算机的快速计算能力及计算机图形技术帮助设计人员进行复杂的设计。因此,CAD已在机械制造、建筑设计、服装剪裁、大规模集成电路以及控制系统设计等技术领域内得到广泛的应用。

英国控制论方面的著名学者 H. H. Rosenbrock 认为^[1]:“人与机器之间的信息交互技术对人类未来的影响将远远大于它们对人类现在的影响。计算机的能力与人脑的能力具有十分不同的特点,但又具有互相补充的性质。计算机擅长于分析与数值计算,而人脑则擅长于模式识别和对复杂情况的判断。因此,如果把‘人’和‘机器’的不同能力和特点有效地结合起来,其作用力将会更加强大。”因此,CAD 对科学研究,对未来人类文明发展将具有难以估量的巨大意义。

应用计算机进行控制系统的设计(简称 CADCS)是在 70 年代左右,在数字仿真技术的基础上逐步发展起来的,目的是解决如何把控制理论转换成工程上可以实现的方法。早在 50 年代,为了设计复杂的火炮控制系统,人们就已经采用了模拟计算机仿真技术,即事先构造被控对象与设计好的控制器的数学模型,然后将它们放在模拟计算机上,通过对整个系统的仿真实验,验证控制器是否满足设计要求,并根据某个控制指标调整控制器的参数。60 年代后,数字计算机逐渐普及,出现了大量通用的数字仿真软件及语言。控制系统的设计者开始在数字仿真软件及语言的基础上,增加有关

控制器设计的模块,从而产生了第一代 CADCS 软件。与此同时,为了发展空间技术,人们在古典控制理论中吸收了应用数学方面的新成就,从而发展成为现代控制理论。现代控制理论在实践中遇到的最大难题就是如何把由数学语言表达的理论转化为工程上实用的方法,因此,人们迫切希望开发出一种软件系统,使控制工程师可以方便地应用现代控制理论来解决工程实际中的问题。

较早期的 CADCS 系统是利用古典频域法、根轨迹法来设计单变量系统的校正环节及用线性二次型最优控制理论来设计单变量或多变量最优控制系统,即解矩阵黎卡提代数方程,并用仿真软件观察系统的时域响应。因此,这类 CADCS 系统大部分由数字仿真软件及控制器设计软件两部分组合而成。例如,Thomposon 和 Young 将 MIMIC 仿真语言及 NASA 分析程序改变为 EXATLAN 的 CADCS 软件包^[2], Chubb 将 IBM 公司开发的 CSMP/360 仿真程序改变后用于控制系统的设计^[3],加拿大 Alberta 大学数据采集控制和仿真中心研制了一套 CADCS 软件包——GEMSCOPE (GEneral Multipurpose Simulation and COntrol Package)^[4]等。其中 GEMSCOPE 的结构如图 1.1 所示,由图可以知道,其主线是系统仿真程序。

70 年代初,英国学者 Rosenbrock, Mayne, Macfarlane 和 Owens 等人开始研究多变量控制系统频域法及其与状态空间法之间的联系,他们用多项式矩阵理论及复变量代数理论来处理线性多变量系统,取得了显著成果。其中, Rosenbrock 提出了对系统的另一种描述形式——系统矩阵,并引入了“分状态”概念作为对系统内部状态的描述,使状态空间描述成为系统矩阵描述的一种特殊形式,这就使频率法与状态空间法之间存在的鸿沟开始被填平,并在此基础上形成了多变量控制系统的频域理论——现代频域理论。在这方面有代表性的工作为 Rosenbrock 提出的逆奈奎斯特法、Mayne 提出的回差序列法、Macfarlane 的特征轨迹法及 Owens 提出的并矢展开法。他们把古典理论中的零、极点、根轨迹、奈奎斯特图等概念扩展到多变量系统中,并借助于图形进行

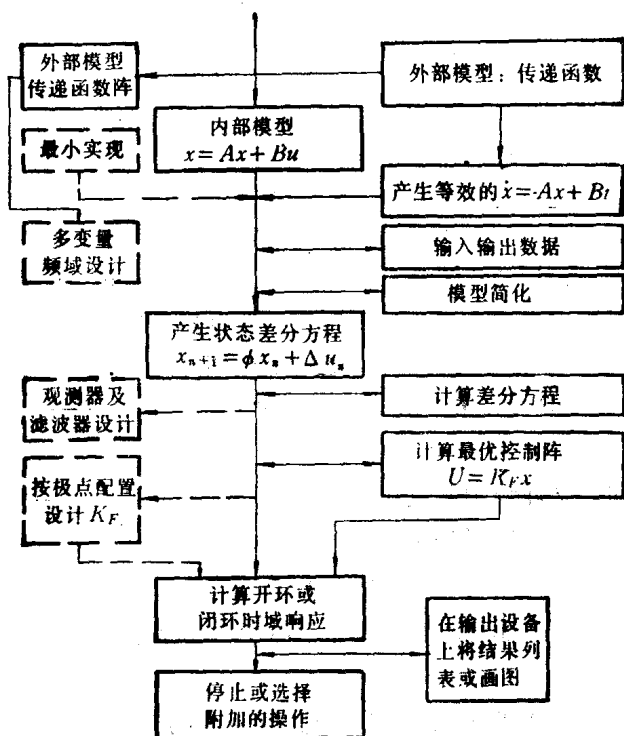


图 1.1 GEMSCOPE 软件包

分析设计。在他们的领导下，在英国曼彻斯特理工学院等高等院校中研究出了一套比较完整的，应用于多变量控制系统设计的计算机软件包，并已成功地应用于化工、造纸及飞机发动机等领域。

与此同时，在状态空间设计领域中也出现了许多计算机辅助设计的软件包，其中以日本东京大学古田（K. Furuta）为首的实验室开发研制的 DPACS-F 系统^[5]（Design PACKAGE for Control System—Developed by Furuta Lab.）有一定的代表性。

1974 年美国控制理论学者 Wolovich 在“Linear Multivariable System”一书中详细阐述了关于采用多项式矩阵理论进行复杂控制系统设计的原理。1979 年捷克学者 Kučera 用矩阵多项式理论开发了一套设计多变量离散系统的软件包。

总之,从 70 年代初到 80 年代初,世界各国的控制理论界、工程界及计算机软件界互相结合,已开发出一大批用于控制系统设计的软件包。虽然它们所采用的理论不尽相同,软件包的结构及人-机交互方式也各具特色,但是从功能上看,这些软件都是以控制系统设计为主的;从整体上看,它们已经独立于仿真语言或仿真软件包。通常,称它们为第二代 CADCS 软件包。

由于第二代 CADCS 软件包在完整性、一体化、使用环境、对用户的指导等方面还存在许多问题,因此在一定程度上阻碍了第二代 CADCS 软件包在工程界的推广应用。从 80 年代中期开始, Taylor, Frederick, Birdwell 及 Åström 等人先后论述了研究第三代 CADCS 软件包的必要性及可行性^[6,7,8],并给出了新一代软件包的概念性结构及它们的原型。与此同时,在仿真界,Ören 及 Zeigler 也提出了先进的建模方法学的新概念,并开发了功能完整的建模语言^[9,10]。第三代 CADCS 软件包利用数据库技术、图形技术及人工智能技术使第二代 CADCS 软件包得到了很大的改进。

我国在 CADCS 方面的研究比国外大约晚了 10 年时间。从 70 年代末期开始,我国的控制理论研究者及系统仿真、应用软件的研究人员逐渐认识到 CADCS 的重要性,并根据各自工作的领域研制了一些 CADCS 程序包。1984 年在国家自然科学基金的资助下,成立了由全国 15 个科研单位及高等院校组成的 CADCS 设计组,开始研制我国第一个多功能、有较完善的软件结构的、适用于科学计算及教学的 CADCS 软件包。经过三年的努力,该软件包已于 1986 年 6 月正式通过国家级鉴定。1987 年国家自然科学基金委员会再次资助这一有学术价值及实际应用意义的研究课题,目标是进一步完善与扩充该软件包,如:进一步实现一体化,嵌入人工智能技术等,使它接近第三代 CADCS 软件系统的水平,并要求在设计过程中严格遵循软件工程的设计方法,使它具有更强的应用价值。

1.2 CADCS 的技术内容

为了使大家了解 CADCS 的技术内容,首先让我们详细分析一下控制系统的设计过程。

设计过程本身是人的创造性与智能决策的结合过程。为了实现在设计中的决策,有两个条件是必需的:一是对各种问题进行形式化的创造能力,另一是说明它们相对优缺点的能力。

在设计过程中,建立系统的模型是至关重要的。模型本身可能有许多种形式:概念上的或形式上的(即数学形式的)。在控制系统设计过程中,设计者建立的模型通常被表示成一组微分方程、差分方程或代数方程,当然也可以表示成一组语言表达式,它们包含某种规定的逻辑。由于模型形式不同,常常导致不同的设计方法,比如:若用状态空间形式来描述系统,则可用状态空间的设计方法,而若用传递函数阵来描述系统,则可用多变量频域法或代数理论的算法。

在设计过程中,每个设计者都会提出以下一些问题:我的模型是否良好?我是否已经考虑了所有可能的设计方案?我是否已经正确地解释了一个设计决定的结果?一个品质优良的 CAD 系统应该帮助设计者来回答这些问题。

设计过程与其它工程活动之间有两个重要的接口,即设计者与顾客之间的交互及设计者与实现者之间的交互。在设计开始之前,设计者必须向顾客了解清楚设计的要求并制定一个设计协议,不仅如此,在设计过程中还将多次证实设计工作是否符合顾客的要求,即是否满足设计协议,因此 CADCS 系统常常需要一个能完全与精确表示顾客要求的语言,这种语言不仅设计者容易理解,而且顾客也很容易理解,另外还应能提出一种表示及证实设计协议的方法,它同样也要求很容易被设计者及顾客双方所理解。

在控制系统设计中,顾客的要求通常是以一种半形式化的方式表达出来的,它们描述了在不同条件下控制系统所期望的性能。

而设计协议的表示则通常是借助于仿真来实现的，也就是说是通过
对系统的形式化模型的实验来实现的。

设计者与实现者之间的交互要求将一个抽象的控制系统形式
化模型转变成一个具体的系统——指令、软件及电子硬件。

通过上述讨论可知，建立 CADCS 系统可以帮助设计者进行
复杂控制系统的设计，但是由此也带来一系列问题，它们集中起来
是：

(1) 需要一个表示模型及系统行为的语言，用这个语言，人们
能够形式化地改变设计要求。

(2) 需要一个软件工具来完成各种不同的计算任务。

图 1.2 中详细地表示了控制系统设计的全过程，同时也清 楚
地描述了一个 CADCS 系统的各个成分是如何支持这个设计的全
过程的。下面分三个部分来介绍它们。

1.2.1 开发模型

控制系统设计者通常采用两种模型来对问题进行形式化并解
决该问题，一种是形式化模型，它们通常是一组数学方程式。这种
模型一般是借助于已知的规律推导出来的，或者是通过对系统进行
某种试验，然后经过数据拟合及统计方法而获得的。另一种则是
系统的概念性模型，它包含了系统的有关信息，比如：模型的应用
限制，用它来精确表示实际系统时所存在的长处及弱点，扩展它
的各种可能性等。这些信息是与第一类模型密切相关的，但一般
并不表示在模型的形式化语言中，概念性模型还包含一些与形式
化模型并无直接关系的信息，比如有关操作模式的知识等。

直到现在为止，大部分 CADCS 软件系统只支持系统形式化
模型的开发，而不支持对系统概念化模型的开发。

1.2.2 设计要求的形式化

建立了被控系统的模型后，设计者必须确定控制系统设计的
准则，也就是控制系统设计目标的描述。通常这种设计要求先由

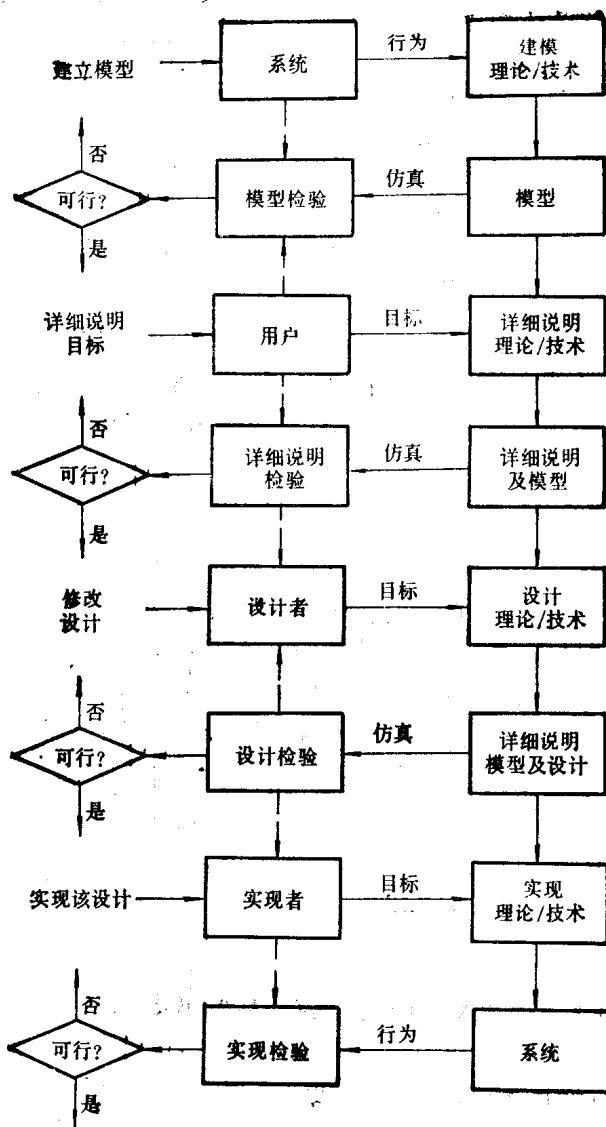


图 1.2 控制系统的设计过程

用户提出,然后要在设计过程中作多次修改,所以需要一种对话式的语言来描述设计要求。

1.2.3 进行设计

在设计阶段,设计者的主要活动是:

- 1) 给出各种不同的设计策略;
- 2) 说明与解释这些策略对模型的效果;
- 3) 决定最佳策略;
- 4) 基于设计中所获得的知识使模型更加完善。

在这个阶段中,CADCS 系统要求给出各种设计方法,并包含检查应用这些设计方法以后系统所达到的性能的模块,通常后者是利用仿真模块来实现的。另外,系统中常常还包含了最优化模块。在最近开发的一些 CADCS 系统中还引进了专家系统,它们能自动寻找最优的设计方案。

综上所述,一个完整的 CADCS 系统应该包括:模型建立、模型转换、模型分析、系统设计(时域、频域)及系统仿真等部分。要开发一套具有实用价值的 CADCS 系统,不仅要依赖于强有力的计算机硬件(如:高速的计算机,大容量的内存,高分辨率的图形显示器及 X-Y 绘图仪等);还要依赖于强有力的系统软件,如:更适合于 CADCS 的高级语言,数据库管理系统,图形软件等,更重要的是依赖于如何将控制理论中一切具有实用价值的方法与原理纳入到 CADCS 系统中去,而这又依赖于如何将控制系统设计工程师及专家们的经验加入到 CADCS 系统中去,因此要求发展各种 CADCS 的算法,并使 CADCS 与人工智能技术结合起来,

1.3 模型处理及仿真在 CADCS 中的地位

正如在上一节中所介绍的那样,控制系统设计是从建立系统的模型开始,然后确定系统的设计要求,并对它进行形式化的描述,继而再选择一种或几种设计方法对控制系统进行设计。为了分析设计结果,判别所设计的控制系统是否符合要求,需要对整个系统进行仿真,获取系统对某种典型信号的响应,如果不满足要