数　值　方　法

〔美〕Robert W. Hornbeck

刘元久　郭耀煌　荣廷玉　译

钱冬生　周其刚　审校

中　国　铁　道　出　版　社

1 9 8 2 年 · 北京

内 容 简 介

本书是把数值法和电算结合起来，为科学技术和工程中较为复杂问题求算其数值解的。在数学理论上，本书并不深奥，凡具有微积分初步和算法语言基本知识的读者，掌握这些方法并不困难。

全书共十一章，内容为绪论，TAYLOR级数，有限差分，内插法和外推法，方程的根，线性代数方程组的求解及矩阵求逆，最小二乘曲线拟合和函数逼近，数值积分，常微分方程的数值解，矩阵固有值问题，偏微分方程介绍等，并在每一章后附有例题和习题。

本书可供作大专院校工科参考教材和工程技术人员自修用书。

NUMERICAL METHODS
ByRobert W. Hornbeck
Quantum publishers Inc. 1975

数 值 方 法

刘元久　郭耀煌　荣廷玉　译
钱冬生　周其刚　审校
中国铁道出版社出版
新华书店北京发行所发行
各 地 新 华 书 店 经 售
中国铁道出版社印刷厂印
开本：787×1092 1/16 印张：16.5 字数：406 千
1982年8月 第1版　1982年8月 第1次印刷
印数：0001—10,000 册　定价：1.75 元

校 者 序

在当前，要为工程或科学技术较为复杂的问题求算其数值解，势必要使用电算。要想有效地利用电算，势必要精通数值方法。在过去的二、三十年之中，电子计算机经历了飞速的更新和换代，为了使不断地新出现的高效电子计算机能够充分发挥效用，世界上已经有不少专家毕生致力于促使数值方法和计算机密切结合，并积累了丰富的经验；其中还有一些作者擅长于结合教学需要，已经将这些经验溶合于数值方法之中，深入浅出，写成了教材。本书的原版，就是这样的一本教材。在数学理论上，它并不深奥，但它的广度几乎已将当代所常用的解题方法概括无遗。在计算技术上，经验之谈堪称十分丰富。这本书已在美国理海大学理工学院使用了好几年，它是作为“工程引论”这一门课程的教材，供工学院所有各系一年级学生研读的。去年，我校邀请这所大学的黄棣教授来校讲学一个月。在几次座谈之后，我深深感到汲取国外使用电子计算机经验的重要性，认为它有助于培养高质量的工程建设人材。经将这书介绍于有关的老师，咸蒙好评。当即组织翻译和校核，拟作为一种引进教材来推荐。承中国铁道出版社的支持，列入出版计划。且喜付梓有期，特作此序。

西南交通大学教授
桥梁教研室主任　钱冬生

1980年5月

译者的话

目前，我国关于数值方法的书已有不少，但就更适合于工科学生学习以及便于工程技术人员自修这方面来说，这本书确有不少优点。本书在讲述各种数值方法时，都考虑到在电子计算机上实现的问题，层次分明，说理清楚，并附有大量精心安排的例题和习题，凡具有微积分方面和算法语言基本知识的读者，掌握这些方法均不会感到很大困难。

本书的翻译工作是在钱冬生教授的主持和指导下进行的。本书的第一章至第五章由刘元久同志翻译，第六章至第八章由郭耀煌同志翻译，第九章至第十一章由荣廷玉同志翻译。

由于译者水平所限，错误和不当之处在所难免，欢迎读者批评指正。

译者　于西南交通大学

1980年5月

前　言

用尽可能清晰的方式，将数值方法中的基本手段按逻辑分类编写，以飨读者，这就是本书的目的。其所列各法，都是很适合于用计算机为科学和工程的不同领域的许多问题求解的。为了在不忽略那些曾经较早地建立起来，且现今仍在广泛使用的好的技术的条件下，将正在飞速发展的数值方法中最新和最有效的技术包括到本书之内曾经作了多方面的努力。

本书的重点是放在对所列方法的了解和使用；关于方法的证明，只是在它们能够加深了解，或激发人们对于所说方法进行研究的兴趣时，才酌予列入。在一种方法的立即付诸使用可以使它的阐述更加有力时，本书就在那些地方将许多例题和正文结合到一起。在每一章的后面，还都精选了不同类型的例题，详列求解过程，既用以阐明该章所包括的各种课题，也用以解释方法的细节和其潜在的难点。读者所需具备的数学知识，只是微积分的初步。而由于采用了这一套措施，本书就不仅适用于充当结构专业的教材，还适合自学，并可充当这一学科所有其它教材的补充读物。

为所介绍的各法分别提供一完整的计算机程序，那是一种流行的办法，但本书没有那样做。从作者的广阔经验看来，那一种办法会促使学生简单地重演那些程序并照着它们去操作，而不利于鼓励学生去深入地了解那些方法。同时，那些程序还会将一本书的视野限制于一种计算机语言（时常是FORTRAN），而我们却拥有不同的理由认为：让读者也会利用其它的语言（例如PL/I，APL或 BASIC），那将是方便的，或者是符合愿望的。

虽然本书所举的许多例子和例题实际上代表了物理现象的数学模型的成果，但它们时常还是用数学形式来给出。这样，本书就不是按照工程或科学中的某一门课来铸制成型的，教师也就可以将所说的方法和问题联系到任意一个领域中去。但是，作者要劝告学生：在还没有对基本数值技术掌握之前，不要去取用包括许多物理内容的复杂问题，以免在物理学中迷路，从而不能在数值方法方面获得所希冀的经验。

第一章介绍了数值方法的功效（和限制），对工程师或科学工作者在学习这些方法方面给出一些启发，并且从用户的观点对数字计算作出讨论。第二和第三章阐述了数值方法的基本构件——Taylor级数和差分法。

第四章的主题是内插法。尽管这一主题在实用中仍然有很大的重要性，而且是许多数值分析的理论基础，但由于某种原因，许多流行的教材或是将它完全忽略，或是没有给予足够的注意。第五章是致力于为方程式求根。好几种求根的法子可以直接从 Taylor 级数导出，但是，反内插法的概念也是有用的。

在第六章内，将线性代数方程组求解所用的直接法和迭代法，均皆给出。病态问题，很大的方程组的求解问题，经特予注意。在第七章内，经将插值的概念引伸到函数的逼近；利用在第六章所取得的手段，可以有效地考察数据的最小二乘拟合。

第八章讨论了数值积分。它包括一些很精确很有效的技术，并且较细致地考察了处理奇异点的方法。

第九章对于常微分方程的求解提供了很多方法。对于初值问题和边值问题的方法都有讨

论。对于每一方法的精度和效率，也都有精心的论述。

第十章的主题是代数的固有值问题。对于为所遇问题选择最有效的技术，经给予重视。

本书是以第十一章介绍偏微分方程（着重抛物型和椭圆型）的求解结束的。对于有限元法的功效和潜力也有简略的讨论。

对于任何一种工程或科学学科讲，在大学的低年级或高年级，本书的内容都比一门“一学期的课”所需要的为丰富。若教学时间实在不够用，可以将下列几部分删去：7.2 节，第十章的较深的部分，以及第十一章。这样的删简对本书结构仍是合乎逻辑的。

（以下为致谢内容，从略）。

Robert W. Hornbeck

目　　录

第一章 绪 论

1.0 引 言

我们将首先简要地讨论数值法的目的、能力以及它们所受的限制，再讲述仔细地学习这一类方法的必要性。

1.1 什么是数值法？

数值法是解决许多数学问题的一类方法。当然，这些问题往往是来源于各种物理现象的数学模型。这类方法的特点，在于它们只使用算术运算和逻辑，因此，它们能够直接地使用数学计算机。

如果把话说得绝一点，从手指头到算盘，其中任何一件都可以说作是数学计算机，但在这里，这名词所指的则是从本世纪五十年代中期以来开始广泛使用的贮存程序的电子计算机。数值法实际上要比电子计算机早许多年。事实上，现在所用的许多方法在某种形式上在现代数学的萌芽期就已出现。然而，在台式计算机出现前，这些方法的使用很有限；只是在引用电子计算机的年代到来后，这些方法的使用才迅猛地发展。

数值法和数学计算机的结合，在数学分析方面形成一个强有力的工具。例如，对于由许多物理现象所需的精确的模拟提出的非线性问题、复杂几何问题、大量的耦合方程，数值法就能够处理它们。若用古典的数学，即使有最精明的应用数学家，也很难按照当前的技术所需的水平去解决。因此，在工业和研究领域，数值法已经在极大的程度上代替了古典的数学分析；由于数值法容易执行、需费不多、和时常有现成的程序可用，即使是对于那一些能用古典数学求解的题，古典方法（不管好坏）往往也很少被采用了。

1.2 数值法的能力有无限制？

对这个问题的回答，就是一个肯定的“是”。许多的外行，还有不少理应懂得多一些的科学家和工程师，经常有这样的观点：若是某一问题不能用其他任何方法解决，那就必须将它“送进计算机去”。无疑地，这一现象是由我们在上一节所讲到的数值法的巨大能力所产生的。然而，遗憾的是，现在确实有许多问题还不可能（对于有些情况，应该说“还难于实现”）用数值法去解。对于某些问题，由于它们还没有精确和完整的数学模型，其数值法求解也就显然是不可能的。其它一些问题，只是由于它们太庞大，在当前的计算机技术条件下，它们的解也就得不到了。例如，曾经有人估计，为求得一个紊流问题随时间变化的详细解，且将最小旋涡的影响也包括在内，那大概需要30年。这一估计是建立在1968年的技术水平上的，而按今天的水平讲，也许要除之以5这样一个大概的系数了。当然，能否“实现”，这问题是强烈地依赖于人们愿意花多少钱去求一个解。有些问题确乎重要，所以，工业或政府情愿花费很多的钱去获得所需的计算能力，这就使先前认为实现不了的求解很快地变成能实现。在任何一个方面，尽管能够求解的界限不断地向外推进，但就当前的技术水平

讲，或是由于数学模型还未建立，或是由于实际的计算能力有限，总是还有不少问题不能解出。

1.3 为什么要学习数值法？

在它的应用已经遍及于科学、技术和行政的每一方面的情况下，提出这一问题似乎有些奇怪。可是本书作者认为有义务来阐明这一问题。当然，对于现在和未来的数值分析家和计算机科学家讲，这一阐明是不需要的。

可是，对于工程师和科学家讲，有些人对于这一学习的必要性也许就不那么清楚。近几年来，已经为模拟复杂物理问题编制了不少大型程序（它们的每一个都需要化费几个人成年的时间来编制）。这些程序时常是为那些对计算机和内部工作懂得不多的人使用的。此外，利用精致的数值法来解决许多数学问题所用的子程序，现今的库存也越来越多。而对这些事实，确是有人怀疑：使工程师和科学家们取得数值法的工作知识，到底有无必要？事实却是：对于希望能够为解决自己的问题而找到现成的程序或子程序的工程师或科学家讲，情势还是要使他失望的。对于任何具体情况。数值法的选择和使用仍然较多的属于艺术，而不是属于科学；计算机用户如果没有为一具体问题选择和使用一个数值法的能力和知识，如果不会为它编制程序，他所能处理的问题在范围方面是极其有限的。

很明显，当有久经考验的，适合于求解所要解的问题的程序或子程序可资利用时，最有效的途径就是利用它们。即使这样，由于用户在使用这些程序或子程序之中难免遇到问题，对于数值法具有工作知识仍然是很有价值的。从各方面所难免出现的问题，包括以下这一些：

（a）没有那么一个复杂的物理现象是能够准确地用一个数学模型来模拟的（这是一件至关重要的事，但不属于现在所讨论的范围）。

（b）没有那么一个数值法是在任何条件下都不出麻烦的。

（c）没有那么一个数值法是完全不会发生误差的。

（d）没有那么一个数值法在任何条件下都是最优的。

（在（b），（c）和（d）之中可能有相当程度的重复。但我们在这里将不考虑定义精确，而只是讲个概念。）数值方法的这些困难能够使现成的程序或子程序产生错误，甚至会全然没有结果。此外，在为某一问题搜寻现成子程序之中，用户时常会见到数目很多、变化多端的子程序似属都可以使用，而从它们的说明材料之中，很难知道它们的效率，也很难判断它们是否适合于求解自己手上的问题。

当一个不懂数值方法的用户遇到这些问题时，如果有熟悉所需资料情况的人（也许是一位数值分析家）可以请教，他是应该去请教的。可是，由于双方的基础知识不一样，要用户能够正确提问，以及要被请教者给出有益的回答，都将是困难的。

这样，我们就能领会到工程师或科学家取得数值法工作知识的极端必要性。凭借这项知识，计算机用户就能为具体问题选择、修改所用方法乃至编写程序，就能在现成程序或子程序中进行选择和使用，就能在处理特殊难题时向计算机专家有效地精辟地进行商讨。最后，还应该知道：正在被称为“方法的开发”（实际上就是模拟复杂物理现象而编写大的程序）的大量工作，并不是由数值分析家，而是由工程师或科学家做出来的。很明显，在这样的工作中，必须使用最有效的精确的数值计算技巧；对于投入这类工作的工程师或科学家讲，通晓数值解法自是很重要的。

现在，我们简单地讲几个和计算机有关的问题，这些问题自身不是数值方法。可是，对于要将数值法配备到计算机去的人讲，这问题是相当重要的。

1.4 计算机语言

本书的多数读者，对于诸如FORTRAN，ALGOL或BASIC这些高级计算机语言的程序，大概即将具有一些经验。凭借这些语言，用户可以用代数公式、英语形式的逻辑、以及输入-输出语句来编写程序。这些高级语言实际上是和执行程序的计算机无关。利用编译程序，高级程序能够转变为执行程序的那一具体计算机的基本代码。

为了科学的目的，最广泛使用的代数语言是FORTRAN IV或它的同类（仅有微小修改者）。除了少数例外，ALGOL一般不用于科学计算，但却广泛地作为国际通用语言来描述算法。在分时系统之中，BASIC是很通用的一种语言；对于比较简单的程序编译工作，它是常被使用的。科学研究者所可能遇到的其它高级语言，是APL（在分时系统中广泛使用，对于很简单到很精细的问题，它都能适用），MAD（和ALGOL相近，但已废弃）和 PL-1（新近为计算机科学家感兴趣的一种强有力的语言）。

对于一种新的语言，一般用户往往是带着几分不安的心情来迎接它的；因为，他们感到又有一套新规则要学，而且新语言会和旧语言混淆起来。不过，任何一个较为灵活的人都会感到：若有必要，采用新语言也不会有什么困难。一个极其重要的结论则是在经济方面的，这是因为：大型计算机程序的开发非常昂贵，而将它们从一种语言变换到另一种，往往是一宗要化好几个月的大件工作。这就是 FORTRANIV 为什么是现行“标准的”科学语言，而且在近期内是不会被代替的一个主要原因。

1.5 检验问题

在为某一问题取得数值解之中，一个极端重要、必须进行而又很难进行的工作，就是检验最后得到的解答的正确性。首先必须确定的是：程序是否按编写者的意图进行工作，也就是说，编码是否正确。这事往往可以从输出大量的中间结果来确定；若有必要，可以用手算或台式计算机对某些点进行复核。检验过程的第二部分，就是确定所使用的算法能否给出正确的解答。由于问题的正确解不可能事先就知道（否则我们也不用为了获得数值解而伤脑筋了），这一部分的检验通常是间接的。例如查看所解的问题的各种极限情况，而这些极限情况的解是已知的。这些极限情况可以用所检验的程序这样来模拟：令某些项是零，令某些常数或条件是很大或很小，或将程序中某段暂时绕过，以及(或者)暂时地插入其它某一小段。

在许多情况下，这种检验过程实际上会比用程序去算出所需要的最终结果更昂贵和更费时。然而，人们对最终结果的信赖是直接和投在检验过程中的时间及精力联系着的。在估计为获得一个数值解所需的时间和费用时，将检验过程包括在内是重要的。

到这儿为止，我们已讲述了涉及一般用户为解决某一具体问题所编写的程序的检验过程。为众多用户解决各类问题的通用程序及库存子程序的检验，其过程自属相似，但必需更加广泛，更加操心，包括要试用所检验的程序对一系列“最坏的情况”进行计算，借以核实程序在应付各种已知难题方面的能力。

1.6 计算机会出错吗？

当然，在这种或那种意义上，计算机能够、而且确实会出现过错。然而，我们应该注

意：在计算过程中所出现的绝大多数错误是来自用户本身的错误。这有时很难使人接受，说一个难于寻找的错误是来自自己。可是，在产生错误的可能性被消除以前，总是照这种假定按最有效的搜寻错误的操作来检查的。

如果计算机出现错误，则或者是硬件的错误，或者是软件的错误。真正的硬件的错误是比较少见的，这里我们还不能够讨论它们。软件错误（它们实际上是另一些程序编写者的错误）是较为常见的，其有代表性的错误包括：计算机执行系统的错误，编译代码（机器语言）的错误，以及库存子程序的错误。

在执行系统（也叫执行器，监察器，操作系统管理程序、乃至其它名称）中的错误，它们会使用户十分困惑。现代计算机系统时常为了最有效地利用硬件而有同时处理几个程序的能力（这叫多道处理），或者允许许多用户从远距离终端和计算机连续进行计算和“对话”（这叫分时系统）。有些系统甚至将这些能力结合起来。用户从计算机执行系统所遇到的困难大多来自未曾预料到的这一程序与另一程序的相互作用。其后果可以是整个系统的完全混乱，也可以使个别用户的结果发生错误。这些错误很少是重复的，只要重新执行程序，事情就被纠正过来了。

编译程序中的错误可以使一个完全正确的高级程序被译成不正确的机器编码，因此得出错误的结果，这对用户是一个很大的挫折。幸而由于广泛的检查，严重的编译错误一般是不会遇到的（常遇错误的一览表，通常可以从计算机管理人员得到）。然而，对那些体现优化的编译程序，则可能发生严重的和几乎无法预料的错误。在这里，优化可以理解为把一个用高级语言编写的程序，编译成可能的效率最高的机器编码的工作。这个优化工作就是：为了（有希望地）用较少的时间得到同样的结果，可以改变在高级语言中所规定的执行顺序。在许多情况下，优化能够显著地节约时间，但是，优化程度越高（就提高效率讲），越是可能产生错误的机器编码。对于许多情况，可以“去掉”编译程序的优化措施，或是采用没有优化的类似的编译程序，或是采用比较简单而无错误的优化措施*。建议这样来处理：在改正程序中的错误（通程序）及最初几次试算中不考虑优化，而只在正式计算时才使用高度优化的编译程序，在此时，效率是极度重要的。当然，高度优化的机器编码只有将其结果和未经优化所得的结果比较时才能看出来。

在库存子程序中的错误一般是由于检验过程的失误。除了将错误的情况报告给该系统的负责人外，用户一般地不能进行处理。

最后，我们应该注意：虽然就通常的意义说，误差并不是过错。按有限位数所进行的任何计算，将产生舍入误差；而任何一个数值法在其使用之中总是先天地具有某些误差。这些误差最好是在讲每一种方法时讨论，现在我们就不再作进一步的考虑了。

1.7 一点希望

为了理解数值法，就不能只停留在阅读它，而是必须去使用。因此，读者应用本书所讨论的数值法去解题，那是至关重要的。在结束这一章绪论时，作者根据本人经验乐于指出：检验一个人对某一方法是否真懂，最好的方法不是让他进行手算（虽然这在试图懂得逻辑的早期是能够有所助益的），而是让他写出计算机程序。在力图准确和不含糊的压力之下，如何使模糊不清的概念澄清，这明显地有其意义。

---

* 关于一个给定的编译程序是否是优化的编译程序，其优化措施是否可以“去掉”，其有关的资料时常是颇难得到的。然而，若在制造商手册中势力搜寻，时常可以看到这项资料以某种形式表现。

第二章 TAYLOR 级 数

2.0 引 言

Taylor级数是数值法的基础。许多数值法可以直接从 Taylor 级数得到；在使用这些技术时，所包含的误差估计也可直接得到。读者在其先前的学习中大概业已熟悉这个级数，但我们仍将作一简短介绍，因为我们的侧重点是和普通的微积分稍有不同的。

如果函数$f(x)$的值能在x接近$x=a$的邻域内用无穷幂级数表示

$$f(x)=f(a)+(x-a)f'(a)+\frac{(x-a)^2}{2!}f''(a)$$
$$+\frac{(x-a)^3}{3!}f'''(a)+\cdots+\frac{(x-a)^n}{n!}f^{(n)}(a)+\cdots \tag{2.1}$$

则我们说$f(x)$在$x=a$邻域内是解析的，级数（2.1）是唯一的，并称之为$f(x)$在$x=a$处的Taylor级数展开。指明级数（2.1）的存在和收敛的一般条件，那是困难的，但显而易见，若要这级数存在，则$f(x)$在点$x=a$处的各阶导数必须存在，而且应是有限的。而如果Taylor级数存在，只要知道$f(a)$和$f(x)$在点$x=a$处的各阶导数，我们就能够求出不同于a的若干个x的$f(x)$值，但应保持x“充分地接近”$x=a$。

如果随着$|x-a|$的增加，达到一个使幂级数（2.1）不再收敛的点，那时，我们就不再是“充分地接近”$x=a$，而是超出了幂级数的收敛半径。有些级数对于所有的$|x-a|$都收敛（其收敛半径是无穷大），而另一些级数仅当$|x-a|$在某些限值以下方才收敛。当级数收敛时，如果在级数中取无穷多项，则其$f(x)$的值将是精确的。然而，对我们更感兴趣和有用的却是：仅取用式（2.1）中的几项，它将在多大程度上接近于$f(x)$。

这可以用图2.1来说明。假如我们要求算$f(b)$。从式（2.1），

$$f(b)=f(a)+(b-a)f'(a)+\frac{(b-a)^2}{2!}f''(a)$$
$$+\frac{(b-a)^3}{3!}f'''(a)+\cdots \tag{2.2}$$

如果仅取用式（2.1）第一项，则函数被假定为常数，如图2.1a所示。如果取用式（2.1）前两项，那就是将函数f在$x=a$处的斜率考虑到了，也就是从$f(a)$引一条斜率为$f'(a)$的直线，如图2.1b所示。如果取用前三项，那就是将由于$f''(a)$所生的曲率考虑进来，如图2.1c所示；如此等等。每添加一项，就使$f(b)$的准确性得到一点改进。

对于截断的Taylor级数，需要采用一些标准的术语和代号。我们从考察截断 Taylor 级数所引起的误差开始。当级数在包含$(x-a)^n$项以后被截断时，表达$f(x)$的式（2.1）对$f(x)$的误差当不大于

$$\left|\frac{d^{n+1}f}{dx^{n+1}}\right|_{max}\frac{(|x-a|)^{n+1}}{(n+1)!} \tag{2.3}$$

这里的角标“max”表示从a到x区间上导数的最大值。

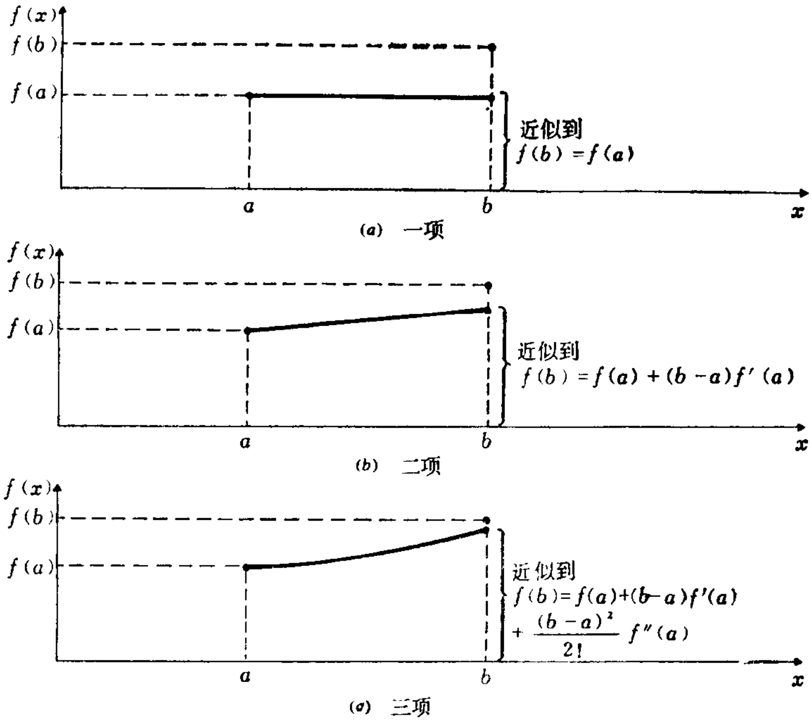

图2.1 由截断 Taylor级数引起的近似

从这个误差界限中似乎什么东西也没有得到，因为，如果为了估计误差，我们必须知道在整个区间上 f 的 $(n+1)$ 阶导数值，则我们也必须知道在这个区间上的 $f(x)$ 值，那就不需要首先将 $f(x)$ 展开了。这种挫折在数值分析中是并不罕见的；然而，有一个很有用处的信息可以从式（2.3）中得到。我们不能控制 f（或者它的导数）的表现；也不能控制常数 $(n+1)!$。但是，我们能够控制 x 和 a 的接近程度，也就是说，能够控制 $(x-a)^{n+1}$。于是，我们可以使用这样的术语：误差式（2.3）是 $(x-a)^{n+1}$ 阶，并用 $E(x-a)^{n+1}$ 来表示。如果 $f(x)$ 的级数表达式在前三项以后被截断，我们说 $f(x)$ 是精确到 $E(x-a)^3$，这是因为

$$f(x)=f(a)+(x-a)f'(a)+\frac{(x-a)^2}{2!}f''(a)+E(x-a)^3 \tag{2.4}$$

应该注意：使用 $E(x-a)^3$ 并不曾将和 $(x-a)^3$ 相乘的常数或导数给予什么限制，例如 $7(x-a)^3$ 就是一个 $E(x-a)^3$。所以，量 $E(x-a)^3$ 也可以当作随 $(x-a)^3$ 而变化的量来理解。

如果我们在将 $f(x)$ 的级数式（2.1）截断前是取用四项，就得到，

$$f(x)=f(a)+(x-a)f'(a)+\frac{(x-a)^2}{2!}f''(a)$$
$$+\frac{(x-a)^3}{3!}f'''(a)+E(x-a)^4 \tag{2.5}$$

一般地，对于一个已经给定的 x 值讲，四项表达式（2.5）要比三项表达式（2.4）给出 $f(x)$ 更为精确的近似。因此，我们可以期望这个函数的误差项 $E(x-a)^4$ 小于其在三项级数式（2.4）内的误差项 $E(x-a)^3$。我们可以把这个结果这样地概括一下：对于一个已知函数的 Taylor 级数讲，在其 n 项处截断后的误差与在其 $n+1$ 项处截断后的误差之间，有下述关系：

$$E(x-a)^{n+1}<E(x-a)^n \tag{2.6}$$

请注意：不论 $|x-a|<1$ 或否，只要是在级数收敛半径范围以内，这一关系都存在。（严格地说，对于某些级数讲，若只取前几项，式（2.6）可能并不对，特别是当所取的 $(x-a)$ 值很大时。然而，只要将 n 取得较大，这关系就是正确的。将式（2.6）看作是总的趋向，那就已经能符合我们的目的）。

还应注意：对于某些级数展开式讲，某一些项可能消失，例如，某 $f(x)$ 只取式（2.1）前四项者的误差，有可能和只取前五项者的误差相同。对于这一情况，请看例题2.2。

例　　题

2.1　用级数（2.1）求 $\sin x$ 在点 $x=0$ 附近的Taylor级数展开。

因为在 $x=0$ 附近展开，$a=0$。级数（2.1）成为

$$\sin(x)=\sin(0)+x\cos(0)-\frac{x^2}{2!}\sin(0)-\frac{x^3}{3!}\cos(0)+$$
$$+\frac{x^4}{4!}\sin(0)+\cdots$$

且因 $\sin(0)=0$，$\cos(0)=1$

所以　$\sin x=x-\dfrac{x^3}{3!}+\dfrac{x^5}{5!}+\cdots$

2.2　截断例题2.1所求出的正弦Taylor级数，给出 $E(x)^4$ 的表达式。说明这个表达式实际上和 $E(x)^5$ 一样。

从例题2.1，

$$\sin(x)=\sin(0)+x\cos(0)-\frac{x^2}{2!}\sin(0)-\frac{x^3}{3!}\cos(0)+E(x)^4$$
$$=x-\frac{x^3}{3!}+E(x)^4$$

如果我们在级数中再加一项，得：

$$\sin(x)=\sin(0)+x\cos(0)-\frac{x^2}{2!}\sin(0)-\frac{x^3}{3!}\cos(0)$$
$$+\frac{x^4}{4!}\sin(0)+E(x)^5$$

因为 $\sin(0)=0$，所添的这一项恰好是零。因此，

$$\sin x=x-\frac{x^3}{3!}+E(x)^5$$

所以，这一个两项式（指取右侧前两项所形成的式子）的误差实际上是 $E(x)^5$，而不是 $E(x)^4$。

2.3　用Taylor级数在 $x=0$ 处展开 e^x，求 $e^{0.5}$ 到 $E(0.5)^3$。使用误差表达式（2.3）来求误差界限。与实际的误差进行比较。

从方程（2.1），得

$$e^x=e^{(0)}+xe^{(0)}+\frac{x^2}{2!}e^{(0)}+\frac{x^3}{3!}e^{(0)}+\frac{x^4}{4!}e^{(0)}+\cdots$$

即　　$$e^x=1+x+\frac{x^2}{2!}+\frac{x^3}{3!}+\frac{x^4}{4!}+\cdots$$

如果 $x=0.5$，则