〔美〕A. 拉尔斯登 H. S. 維尔夫 著

数字计算机上用的数学方法

徐 献 瑜 等

数字计算机上用的数学方法

数字計算机上用的数学方法

〔美〕A. 拉尔斯登，H. S. 維尔夫 著
徐 献 瑜 等 譯

上 海 科 学 技 术 出 版 社

內 容 提 要

本书全面地介紹在电子数字計算机上行之有效的近代数学方法．全书共分六篇，計初等函数的生成，矩陣与綫性方程，常微分方程，偏微分[illegible]括 26 章．每章的叙述，除第 1 章外，均采用大体上統一的格式，分成职能，数学討論，計算过程摘要，框图，框图説明，子程序，例題，存儲需要量，机器工作时間的估計，參考文献等十大項目．对于各种方法的适用范圍作了詳細的分析和説明．可作为实际計算机工作者的一本完备的手册，也可供研究数値分析的讀者作參考．

MATHEMATICAL METHODS FOR DIGITAL COMPUTERS

Anthony Ralston, Herbert S. Wilf

John Wiley & Sons, Inc. 1960

数字計算机上用的数学方法

徐献瑜等 譯

上海科学技术出版社出版（上海瑞金二路 450 号）

上海市书刊出版业营业許可証出 093 号

商务印书館上海厂印刷　新华书店上海发行所发行

开本 850×1168 1/27　印張 17 3/27　排版字数 425,000

1963 年 9 月第 1 版　1963 年 9 月第 1 次印刷　印数 1—2,800

統一书号 13119·529　定价（十四）2.85 元

序

目前在应用数学、物理学、工程科学方面，有越来越多的科学工作者正在利用大型数字計算机来解决問題．大多数工作者已經掌握了机器上解題的数学知識，并把这些知識看作他們业务上所需求的知識的一部分；而还有一些工作者觉得这里所牵涉到的数学仍是属于新的，或相对地未曾探索过的領域．对于第一批人，本书将作为一本参考书来满足他們迫切的需要，从中能看到許多在机器上极常用的数学方法．第二批人則既可将本书作为参考資料，也可作为对数値分析这一領域的一个导引，虽然本书并不是一本数値分析的教科书．

第1章討論机器上生成初等函数的一些方法．其他各章都用一定的标准格式来叙述．这里包括所考虑的方法的数学討論，接着就是一系列的、对計算机使用者特别有兴趣的課題，例如，計算过程、框图、例題等等．另外，各章分别由某个人执笔，紧密联系着在他所掌握的領域中最新近的进展．总加起来，这二十六章包含的不仅有許多数字計算机上較常用的数学方法，也有若干关于嶄新发展的、有前途的某些方法，还有一些章說明新的技巧（例如，Monte Carlo 技巧），有时把它們用来在机器上解决問題．

为了要能从每一章中获得全面的了解，在讀者方面虽然要求备有基础数学的成熟知識，但著者們深信这里所需的数学背景并不多于从事数値分析領域工作的一般知識．甚至对于缺乏这方面数学素养，而不能全面鉴賞各个章节的工作者，仍可有益地利用本书作为一本有关計算步驟、框图、参考文献等的手册．应該指出，为了了解每章中面向机器方面的那些部分，并不要求讀者具有个人的計算經驗，而只須对一般机器的基本概念有所理解．

目　　录

引　言

(一) 目　的

数字自动电子計算机的最近发展，已經在数值分析中起了有力的冲击作用. 首先，为了解决一些只有在近代計算工具的飞快速度下才属可解的問題，已經建立了嶄新的方法論. 其次，数值分析的一些經典方法經受了严格的檢驗，由此，发現了多数經典方法中的若干缺陷，而这些缺陷一般在运用这些方法来解决計算量較少的問題时并不十分明显. 本书的主要目的在于陈述許多——而并不是一切——近代数值分析学家所較常用的算法工具，也讲到某些最近发展起来的較有前途的方法. 这样陈述的动机，不仅是为了对近代的各种数值方法集中地作一个局部性的鳥瞰，也在于使讀者在每一情形中熟悉計算机功能和分析过程間的相互影响. 每一章的目的在于描述从一个問題的形成直到在計算机上进行实际編碼工作之前的整个过程. 著者希望用这样的叙述方法可使提供原始問題的人們能更好地了解，在近代企业或大学的計算实驗室內，一群数学工作者怎样准备問題編碼的奇妙的过程. 应着重指明，在本书的任何部分，并不假定任何一架特殊的計算机器.

以下的二十六章并不全部都属于“較常用的”或“較有前途的”近代数值方法的范圍. 有几章是因为适于說明在計算机上可以运用以及有时正在运用的一些方法的新穎性. 属于这方面的，有这里陈述的 Monte Carlo 方法的某些应用. 这些随机方法在别的領域 (例如中子迁移，排队論等) 中发揮了很大的效用，但因过于专門，不能在本书內討論. 此外，第 26 章并不在于太多地涉及数值方法，而在于討論怎样把一大类实际問題，卽网絡問題，用数值分析簡化为便于編碼的形式.

又如在第 11 章，叙述了如何用一个很特殊的方法来解决有限的一类問題，其目的在于指明一些特别方法的用处，这些方法在特殊情况下可替

換一般方法的应用.

然而大多数的章确是討論今日数值分析学家认为标准的方法，或逐漸成为标准的方法. 为了使本书保持合理的篇幅，主編者不得不删弃某些可列为章的課題，因为这些課題在別的章中应該提到. 此外，由于本书的目的肯定地不是为了介紹数值分析学，所以把有关計算技巧的許多方面也作了删节. 例如,本书中的許多方法是作为独立的概念提出的,对于它們之間的相互联系和共同形式，这里仅用了比数值分析教科书上少得多的篇幅来論述. 但是如果讀者已在数值分析方面备受全面的訓练，我們期望他能在这里发現一些新的观点.

必須指出，本书并不一定能給讀者提供求解任一給定問題的最好方法. 一般說来,本书各篇所討論的問題是多种多样的,要对規定的某一类問題(例如矩陣求逆)推荐一个"最好"的方法是不可能的. 虽然如此,每章的执笔者一般总是仔細地指出所討論的方法在哪一类問題中具有主要的应用,而在哪些情况(如果有的話)下最好能避免使用. 如果讀者对討論某一类問題的各章作仔細的鉆研,可以預料,为了寻求"最好"的方法,他将会得到某些启发.

我們相信,如果讀者对本书細心閱讀,他将在学习到所討論的个別方法外,还能另外有所获益. 例如請讀者研讀各章中的框图,追查某些似乎奇离的步驟的动机,以及寻找一些框图中的过失,这些过失想来在多数的情形中一定还有,尽管大部分的框图已經通过广泛的檢查. 照这样的做法,讀者将学到許多有关数值分析以及准备問題上机器的知識.

(二) 章节的格式

由于这里所包函的題材是各式各样的，因此使各章都保持絕对严謹的統一格式,旣是不可能,也是不需要的. 特別,由于題材的性质,第1章具有独特的格式. 但是,在此以后的各章,除少数例外,都包含下述一些陈述內容:

职能

所要描述的方法的职能是些什么? 各章的执笔者力图在他所写的一

章中对所考虑的特殊問題，給出一个簡洁而且正确的提法.

数学討論

当这章的范圍扩展时，著者力图給出所要求解問題的一个完全的数学描述，并讲明他所提議的解題的一种或几种方法. 这里讀者可找到适当的数学定理及其証明，或証明的一些参考資料；如果有适用的誤差分析，讀者总会找到其中包含对所提議的技巧和其他有用技巧相比之下是否适用的各种情况的討論，以及有关的文献的援引.

計算过程摘要

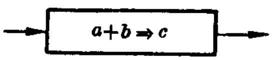
在数学討論与方法推导交叉陈述时，讀者到后来常常弄不清需用哪一些精确的步驟来实現問題的解案. 因此在这一节中，讀者能找到以前导得的方法，用开“药方”的形式所作的叙述，即首先做这个，其次做这个，等等.

框图

框图是借通用的語言，而不是某一架机器的語言，来写成的計算机上的編碼. 一个框图可以是粗描的，或是細描的，或是不太粗不太細的. 但是編碼工作者只能从一个細描的框图去工作，所以这里所包含的框图是尽量地細描，做到图内資料的普遍性所容許的地步.为了使討論尽可能普遍化，本书中的框图是針对一架臆設的机器而描繪的，这架机器假定有无穷个快速取数的內存单元，沒有輸入机件，有一輸出設备，当指令要打印意想中的什么解答时，即刻印出. 照这样的做法，把邏輯結构从特定机器的需要条件尽可能分别开来. 但应指出目前对机器設計的趋向正是朝着上述那架机器来实現的.

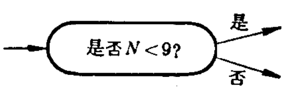
框图中的每一框的类型，只限于最少的几个. 所有的框图总是以标志着“启动”的一框开始，而以标志着“停机”的一框結束. 其他类型的框，用到的有

(a) 叙述框：



这框叙述框中含有的操作，并在通过这框的时候执行.

(b) 檢查框：



这框具有一根入口綫同二根出口綫，出口綫的选擇取决于框中含有的問句的答案是否肯定。

(c) 远接框：　　$\longrightarrow$ ⑤

这圓圈指明邏輯控制轉移到另一个含有同一数字的圓圈。在含有变量的远接框 $\longrightarrow$ ⓐ 情形下，出口框中的数字，象框图中所指明，由程序本身来安排，因此在框图的另一个地方可找到一个分离的进口圓框，对应于連接变量的每个可能的安排。

框图說明

这里給出框图中逐框的說明来帮助讀者有所依从。

子程序

列出程序中所需用的任何标准子程序。

例題

把一个有代表性的例題逐步做出，使讀者能領悟实际解題过程中的情况。

存儲需要量

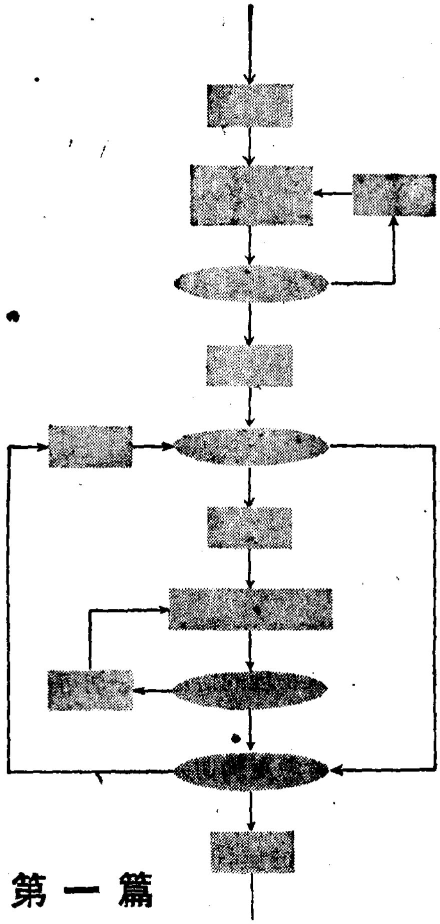
借用問題的一些参数，給出对問題本身所需用的內存单元，工作单元，数据单元的个数。这里假定在机器上的一个字具有一列的数字，既可作为指令，也可认为計算用的数量。

机器工作时間的估計

借用問題的一些参量，以及用任何机器的加乘时間，給出一个問題大約所需用的机器工作时間。

参考文献

列出文中所援引的参考文献(并加上有特别意义的其他参考資料)。文中方括弧內的数字相应于参考书目的序数。



第 一 篇

初等函数的生成

1

初等函数的生成

E. G. Kogbetliantz
徐献瑜 譯

引 言

物理学家、化学家或工程师經常使用电子計算机，以各种数学方法来探索在新的化学的或物理的情形下一切可能的結果，这些結果依賴于很多的可变的参数 $\alpha_k(1 \leqslant k \leqslant n)$.

为要使我們的考慮更确切一些，試考虑由一个定积分来表示的最后的数值結果．設这个积分的被积函数包含积分变量的某些初等函数，而变量又依賴于参数 α_k 的值；但在积分过程中把 α_k 认为固定的常数．

为了刻划一切可能的物理情形，对各种 α_k 的值的許多組合要計算出相应的許多次积分．如果 α_k 的区間很大，則需用 α_k 的平均个数 m 亦会很大，总共这积分要算 m^n 次．在每一个这样的数值积分过程中，积分区間 (a, b) 被分为 $p=(b-a)/h$ 个子区間，其中 h 为子区間的长度．因此，在被积函数中包含的每一初等函数，在每一次积分要計算 p 次．总共，每一初等函数要計算的次数可用乘积 $m^n p$ 来表示，而这个数目的大小經常达到 10^5 的数量級．还須提到有些問題需要計算很多这样的积分．

换言之，科学家經常使用电子計算机来做多变数函数的列表工作，并为了完成这个結果对某些初等函数的計算就要重复到許多千次．大家知道，电子計算机具有如此的速度使对一个初等函数的一次計算可在几毫

秒的时間內来完成(自然，計算的准确时間长度取决于所需的精确度，所用的計算方法，所考虑的哪一个初等函数，以及所用的机器的类型)。但是象这样的一次計算在一年的时間內可能重复到几千万次或几亿次，所以很明显，对一个初等函数的生成，如果能用更經济更快速的子程序来完成，只要有1毫秒的节約，那末在一年之中，就可省掉20～30个机器小时，这就等价于10,000到20,000美元. 应指出，在許多情形中2～3毫秒的节約是有可能的.

这一章的目的在于对初等函数的生成建立最經济的数学方案. 为了节約机器时間，必須尽量縮减乘除次数，而还要不太多地增加指令的条数，亦即不使子程序的邏輯部分太复杂化. 加法，减法和移位运算已是如此快速，可不必計較它們的次数.

在建立一个更經济数学方案的工作中，重要的一点是使方案灵活：即应具备充分的普遍性，使它在任何給定的精确度下，可得到近似值. 換句話說，对同样一个初等函数，一个方案要建立在同一的普遍計算方法的基础上，定义出精确度递增的一族子程序，不过它們具有不同的乘除次数 M 和預算好的存儲常数个数 PC. 这两个重要的参数 M 和 PC 依賴于近似的种类(多項式或有理式)；依賴于所应用的近似中自变量 N 区間的大小；也依賴于給定的精确度，亦即在自变量的整个范圍內結果中所期待的正确数字的个数 Dg.

这里我們必須分清定点和浮点計算. 对于一个浮点的子程序，我們用位于前面的正确有效数字的个数 Dg 来定义精确度，所以涉及到相对誤差(即絕对誤差与計算所得的值的比). 但在一个定点計算中，大半情形下以絕对誤差作为精确性的度量.

因此对定点編制的子程序，我們必須考虑而且企图使絕对誤差的上界极小化. 于是，如果这一界限沒有超过 $5\times10^{-(n+1)}$，我們就称这子程序在小数点后产生 $Dg=n$ 个正确数字，即使一个初等函数的結果数值在 $n-m$ 个有效数字以前具有一串的 m 个零，也这样看待.

自然，对浮点編制的子程序，相对誤差的上界是刻划子程序的精确度的. 因此，如果这个界限最多等于 $5\times10^{-(n+1)}$，我們就称这子程序产生在前面的 $Dg=n$ 个正确有效数字.

应着重指出，有些机器乘法运算非常之快，而除法操作如此的緩慢，使在这些机器上的子程序，只能根据多项式近似来演算，而把除法的利用全部取消了．

只在几年以前，差不多对初等函数的一切子程序是以多项式近似为根基，但是目前用有理式逼近的优越性似乎成为一般承认的事实．在这一章中我們将同时討論这两种逼近，因为对具有緩慢除法的机器，多项式逼近有它突出的重要性．

对于同样的精确度（Dg 的个数），有可能編制出若干个子程序，具有不同个預算好的存储常数的个数 PC．

一般說来，增加了 PC 可以少用些运算操作，亦卽减少些 M．但是这句話是对自变量 N 的区間而說的，在这区間內以給定精确度 $Dg=n$ 确定的誤差的界限不超过 $5\times10^{-(n+1)}$．所以，这里 N 的区間 $0\leqslant N\leqslant N_0$ 是另一重要参数，因为一个近似式的誤差依賴于区間的长度，而且通常是随 N_0 的增加而很快增大的．这是为什么在实际应用中，任何近似都用在短縮的区間 $(0, N_0)$ 內，而后者只是自变量全程的一部分．

举例来說，对指数函数 e^N 的子程序应該让自变量 N 在全程 $-\infty<N<\infty$ 中取任何值来計算 e^N．但要在全程 $(-\infty, \infty)$ 中以給定的精确度来近似 e^N，找不到任何計算公式．所以为了設計 e^N 的子程序，第一步要做的在于短縮无穷区間 $0<|N|<\infty$ 到有限区間 $0<|N|<N_0$；并对任何其他初等函数也应作同样的考慮．

在一般情形下，可以选擇充分小的函数自变量的短縮区間，把任何近似式的誤差做到这样的小使能达到給定的精确度．因此，如果短縮区間短小了，M 亦能减小．但在另一方面，使区間进一步短縮，需用更多的存儲常量，以及更多的指令，这就意味着消耗机器的时間及存儲单元．

由于存在着两种不同类型的电子計算机——二进制的和十进制的——就有需要对不同类型的机器做出不同的适用的子程序．因此，对于每一个初等函数，我們要有两种子程序——二进制的和十进制的子程序．特別是，使自变量区間短縮这一件事，对于二进制子程序和十进制子程序是很不相同的．

最后，还値得提到另外一点．在編制計算初等函数的子程序中，把这

函数的值預算好和列表存放起来是沒有什么用处的。列表的函数值总是不够用的，而且要計算任何一个不在表中的函数值，必須用到插值。但是插值計算，比直接用近似式来生成这函数值，更为費时費存儲单元。然而对某些函数，例如对数函数，則用一个很短的数值表，同时采用在一很小的短縮区間中的一个近似式，就可在相对少数的 M 运算的帮助下，达到給定的精确度。

在这章中，我們首先描述純粹型的二进制和十进制子程序，在这种型中沒有用到函数的数値表。混合型的子程序仅偶然提到。

这里的討論表明，要編制一个經济的和短的子程序不是一件容易的工作。我們必須对許多因素的相互矛盾的影响和它們間的平衡，作出一个冗长的全面的研究。在数字計算机上，初等函数的生成是一个新的，进展得很快的技巧。这里并不申說对最后結果的要求，而在本章中所叙述的子程序并不能看作最經济或最好的。我們希望讀者在应用这里所討論和所描述的方法时，能把它們加以改进。

这是为什么在討論初等函数的子程序时，我們的注意力总指向在方法上和概念上，而不在实际的結果和要存儲常数的实际計算上。按作者的意見，原則是更为重要，并且当已掌握了方法和近似的一般公式，則一个子程序的实际編制是相对地簡单和容易的一件工作。

(一)

数学工具

初等函数的子程序，在它們的結构中反映出每一特定函数的特性。它們是不相似的，所以对每一初等函数要分別研究。然而，在树立近似表达式所用到的基本数学工具方面，則对一切函数都是一样的。因此，值得首先来描述用作近似表达式的各种数学公式，然后把它們应用在第（二）部分中来計算各別的初等函数。

（1）Maclaurin 冪級数

多項式近似是首先被用来作計算的，并在使用电子計算机的初期，差

不多所有的子程序都以 Maclaurin 幂級数

$$f(x)=\sum_{n=0}^{\infty} f^{(n)}(0)\cdot x^n/n! \tag{1·1}$$

的部分和 $s_m(x)$ 作为根据，这里 m 的大小要选擇得这样大，使保証达到所需的精确度.

举例来說，考虑在区間 $(0, \pi/4)$ 上的 $\sin x$. 由展开式(1·1)的部分和 $s_m(x)$ 近似 $\sin(x)$ 而給出的絕对誤差是小于被删掉的第一项的絕对值①，因此对 $|x|\leqslant\pi/4$，它小于 $\left(\frac{\pi}{4}\right)^{2m+1}\Big/(2m+1)!$. 为了保証在小数点后有 Dg 个正确数字，$s_m(x)$ 中 m 的大小必須适合以下不等式:

$$(2m+1)\log(4/\pi)+\log(2m+1)!>Dg+0.30103,$$

而这不等式对 $Dg=10$ 就給出 $m>5$.

根据近似公式

$$\sin(x)\approx s_5(x)=x-\frac{x^3}{3!}+\frac{x^5}{5!}-\frac{x^7}{7!}+\frac{x^9}{9!}-\frac{x^{11}}{11!} \tag{1·1a}$$

編制在区間 $|x|\leqslant\pi/4$ 中定点計算的 $\sin x$ 的子程序是一很短小很簡单的子程序，但它是太慢了. 要得到 $Dg=10$，必須要用多到 7 次的乘法：1 次来計算 $z=x^2$，5 次来算 z 的 5 次多項式，还有 1 次把結果乘以 x. 在 $\pi/4\leqslant N\leqslant\pi/2$ 中 $\sin N$ 可用 $\cos(\pi/2-N)=\cos t$ 来計算，其中 $0\leqslant t\leqslant\pi/4$. 利用 $s_m(t)$ 作为 $\cos t$ 的近似，而有

$$2m\cdot\log(4/\pi)+\log(2m)!>10.30103,$$

給出了 $m>6$. 我們看出，要計算

$$s_6(t)=1-\frac{t^2}{2!}+\frac{t^4}{4!}-\frac{t^6}{6!}+\frac{t^8}{8!}-\frac{t^{10}}{10!}+\frac{t^{12}}{12!}, \tag{1·1b}$$

則也要算 7 次乘法. 因此，要产生 $Dg=10$，在 $|x|\leqslant\pi/4$ 中的 $\sin x$ 和 $\cos x$ 的子程序，让它們来計算这些函数在 $0\leqslant|x|\leqslant\pi/2$ 中的值，如果用 Maclaurin 幂級数，則要作 7 次乘法，并且对二进制机器，其預算好的存儲的常数要 10 个，对十进制机器要 11 个.

(2) Чебышев 展开

多項式近似的另一源泉是 Fourier 余弦級数. 这可写成在区間

---

① 以后当誤差上界的精确估計不能找到时，則仅考虑上界的近似估計. 在已找到函数的近似表达式以后，誤差上界必須再行檢驗.

$-a\leqslant z=ax\leqslant a$ 中函数 $f(z)$ 的 Чебышев 展开的形式

$$f(ax)=\frac{1}{2}c_0(a)+\sum_{n=1}^{\infty}c_n(a)\cdot T_n(x)\quad(-1\leqslant x\leqslant 1),\qquad(1\cdot2)$$

这里 $x=\cos\theta$，而 $T_n(x)\equiv\cos n\theta$ 是在区間 $(-1,1)$ 上，权函数为 $(1-x^2)^{-1/2}$ 的 x 的正交 n 次多項式．展开式 $(1\cdot2)$ 中的系数 $c_n(a)$ 为参数 a 的函数：

$$\pi c_n(a)=2\int_{-1}^{1}f(ax)\cdot T_n(x)\cdot(1-x^2)^{-1/2}dx.$$

对逼近的目的来說，式 $(1\cdot2)$ 的部分和 $\sigma_n(a,x)$ 比幂級数的部分和 $s_m(x)$ 确是好得多．原因是很簡单，因为 $(1\cdot1)$ 中系数，只依賴于被积函数在原点 $x=0$ 的紧邻范圍內的性质，只有在自变量 x 的很小值时，級数 $(1\cdot1)$ 很快收斂，且很好地代表这个函数．截断誤差 $f(x)-s_m(x)$，当 $|x|$ 增加时增加得很快．但是，Fourier 級数 $(1\cdot2)$ 中的系数 $c_n(a)$，依賴于函数 $f(z)$ 在 $(-a,a)$ 区間中所取得的一切函数值．一般說来，当 n 增加时，这些系数很快地減小，而截断誤差 $f(ax)-\sigma_n(a,x)$ 可以很好地用被删掉的第一項，即用 $c_{n+1}(a)\cdot T_{n+1}(x)$ 来作近似．显然，$|T_{n+1}(x)|\leqslant1$，而 $T_{n+1}(x)=\cos(n+1)\theta$，当 x 在从 -1 到 1 的区間中变化时，它摆动在 -1 与 1 之間，并在区間 $-1<x<1$ 內 $n+1$ 次为零值．由于截断誤差的性质主要的是 $T_{n+1}(x)$ 的性质，所以它具有 $n+1$ 个零值，并且它的一些极值的絕对值很接近于相等，而它們的符号交錯地为正为負．

$\sigma_n(a,x)$ 的性质，虽然不完全和經典的 (Чебышев[1]) 次数 $\leqslant n$ 的最佳逼近多項式 $P_n^*(x)$ 的性质相同，但也很相似，后者能使离开 $f(ax)$ 的离差 $\varDelta_n$ 取最小可能的数值，这个离差 $\varDelta_n$ 将定义为

$$\varDelta_n=\max_{-1\leqslant x\leqslant1}|f(ax)-P_n^*(x)|.$$

选擇 $P_n^*(x)$ 来代表 $f(ax)$ 的思想是很自然的，但是它的系数的計算是非常困难，而且費时，所以值得来探問是否有更易于求出的其他 n 次近似多項式，能代替 $P_n^*(x)$ 而产生同样的 Dg，并用同样次数 M 的运算操作？答复是肯定的．借用 $\sigma_n(a,x)$ 正象用 $P_n^*(x)$ 一样也能达到同一的实际精确度([2]，9 頁)．从实用观点来看这是很重要的，因为在

$$\sigma_n(a,x)=\frac{1}{2}c_0(a)+\sum_{m=1}^{n}c_m(a)\cdot T_m(x)=\sum_{m=0}^{n}\gamma_m(a)\cdot x^m$$