中央精簡科技系列

數値分析

—原理及題解—

黃子琴　譯著

FRANCIS SCHEID　原著

THEORY AND PROBLEMS
OF
NUMERICAL ANALYSIS

中央圖書出版社出版

數值分析

—原理及題解—

黃子琴　譯著

FRANCIS SCHEID　原著

THEORY AND PROBLEMS OF NUMERICAL ANALYSIS

中央圖書出版社出版

數值分析

——原理及解題——

實價新台幣叁佰捌拾元整

原著者：FRANCIS SCHEID
譯著者：黃子琴
出版者：中央圖書出版社
台北市重慶南路一段一四一號
發行人：林在高
發行所：中央圖書出版社
台北市重慶南路一段一四一號
電話：三三一五七二六
三七一九八九三
郵撥：〇〇〇九一四—六
印刷所：國順文具印刷行
板橋市中正路216巷2弄13號
電話：九六七七二二六
中華民國七十五年五月初版

行政院新聞局出版事業登記證，
局版台業字第〇九二〇號

編號：3224

序

今天，**數值分析**（Numerical analysis） 在使用上層面的普遍與急劇的成長，更進一步證明了應用仍是激勵數學創造力的主要泉源。每當一種新的數學觀念發展時，通常便有一種新的應用已明確地指示出一條長遠的途徑。最顯著地，電子計算機便是一個極佳的實例。為了適應快速計算的迫切需要，不得不探索更精密的數值方法來推展出具有此種功能的機器。這就是現代數值分析發生的根源。它是應用分析中所包含的一部分。

不過，將這個主題與所謂的純（或抽象）分析間的界線區別得太精細却是一種錯誤，因為通常這種界線是無法明確劃分的，而從一方面所取得的材料往往與另一方面密切相關。過去，一個數學家精通純數學與應用數學兩者是很平常的事，但長久以來，經過細密的發展，要達到兩者精通的程度就不簡單了。即使是這樣，應用數學家（包括數值分析家）還是必須嘗試了解透過此界線所發生的狀況。對於這點，本書也常提供必要的論證。如像討論泰勒級數便是一個例子。這些級數在純分析中是很重要的，但它們對於計算函數，估計誤差等也很有價值。另外，傅立葉級數，正交多項式，與微擾級數等對此界線兩邊都有相當幫助。微分方程的古典存在理論利用“應用”方法的證明，是從“應用”最終如何導致抽象理論的美好實例。雖然在這裡主要的討論對象是數值數學，但有些相關的論題也要隨時提到，因為它們對計算有用。

本書的編寫主要是用作數值分析正式課程的教本，也可用作所有現行標準教本的補充讀物。它提供數值分析的基本概念與題解實例，對各相關科系的讀者頗有幫助。此外，由於書中所含的材料比一般同類的書本為多

，使得本書的應用更具彈性，因此，對於那些需要數値分析的工作者，它也是一本很有用的參考書，並且，更可進一步激發他們對這門學科的興趣。

每一章開始都包含了相關定義、原理、與定理的明析敍述，並列有充分的實例加以解說。隨後便是一系列按類型和等級分組的解答題與補充題。解答題是用來說明和擴大理論的應用（焦點是集中在細微的要點上），證明定理，並重複基本原理以提高學習效率。補充題是用來複習每一章中所探討的材料，且在本書末尾提供了相關的答案。

評估數値法的一種常用程序，就是把它應用到正確解已知的問題上，於是，這個問題便用作“檢定情況”。這樣的例子已列出許多。當它們是在補充題內發生時，那麼，所提供的答案即爲正確解。不用說，數値法不應期望產生此正確解，把它提出來是使得計算機可核對讀者自己所求出的結果準確到他所需要的幾位小數。

黃子琴 譯

中華民國七十五年五月

目　錄

序
第 一 章　數值分析是什麼……1
第 二 章　並列多項式……13
第 三 章　有限差分……19
第 四 章　階乘多項式……29
第 五 章　求　和……41
第 六 章　牛頓公式……47
第 七 章　算子與並列多項式……53
第 八 章　非等間隔自變數……73
第 九 章　被除差分……81
第 十 章　密切多項式……91
第十一章　泰勒多項式……99
第十二章　插值法與預測……111
第十三章　數值微分法……133
第十四章　數值積分……147
第十五章　高士積分……171

第十六章 奇異積分……203
第十七章 和與級數……211
第十八章 差分方程式……237
第十九章 微分方程式……257
第二十章 高階微分問題……293
第二十一章 最小二乘方近似……309
第二十二章 極小・極大值多項式近似……349
第二十三章 有理函數的近似……369
第二十四章 三角近似……383
第二十五章 非線性代數……405
第二十六章 線性方程組……435
第二十七章 線性規劃……469
第二十八章 過確定方程組……487
第二十九章 邊界值問題……495
第三十章 蒙特卡洛法……519
補充題答案……527

第一章

數值分析是什麼

演算法

數值分析（Numerical analysis）這個主題，可利用例子來加以說明。本章中所列出的例子，是對本書內容與行動方向所作的一個預示。將這些例子綜合起來，顯示數值分析是涉及在從已知數值資料來計算所需的數值結果的過程中，對計算方法所作的發展與評估。因此，這使數值分析成爲電腦科學中**資訊處理**（Information processing）的一部分。也就是，已知資料爲輸入資訊，所需的結果爲輸出資訊，計算方法稱爲**演算法**（Algorithm）。數值分析的這些主要部分，可用下面的流程圖來加以綜述，參看圖1-1。

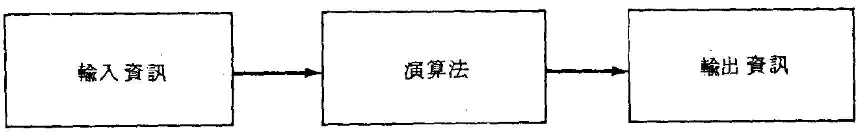

圖1.1

誤差的存在

上面的說明是由明確的應用來決定方向的，而將重點集中在演算法的尋求上。常常產生所需的輸出資訊有幾種演算法可以使用，但究竟那一種較佳就必須作個選擇。選擇演算法的依據雖然也有許多種，可是最常用的兩種準則是速率與準確度。速率顯然是一大優點，如果其它方面都相同，那麼，較快速的方法必定會獲得採用。其次要考慮的是準確度，它顯示數值分析的另一個主要特徵，即誤差的存在。輸入資訊很少是完全正確的，因爲這些資訊通常都是透過某些種儀器的量度而得到的。並且，執行計算的演算法也會進一步地引進誤差。因此，輸出資訊便包含來自這兩個部分（即誤差源）的誤差，正如像在下面第二個流程圖中所暗示的（參看圖1-2）。顯然，降低誤差成長的演算法是值得優先考慮的。

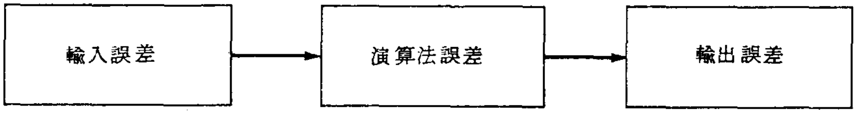

圖 1.2

支持理論

由於數値分析是由明確的應用來決定方向的，自然會涉及**支持理論**（Supporting theory）。這個理論通常爲特定的數學，而具有基本的重要性，因爲它對較佳演算法的尋求很有幫助。

解答題

1.1. 在計算乘積 45× 17 的問題中，指出輸入資訊，演算法，與輸出資訊。

$$\text{演算法}\left\{\begin{array}{l} \left.\begin{array}{r}45\\ \underline{17}\end{array}\right\}\text{輸入資訊}\\ 315\\ \underline{45}\\ 765 \leftarrow \text{輸出資訊}\end{array}\right.$$

輸入資訊是由數字 45 和 17所組成。演算法爲乘法。輸出資訊是數字 765。如果假定輸入是正確的，那麼，由於演算法來引進誤差，而輸出也是正確的。

1.2. 以“**俄農演算法**”（Russian peasant algorithm）來計算題 1.1 的乘積。

此法是連續將一個因數減半，而將另一個因數加倍，減半時有時會留下餘數。

$$\text{演算法}\left\{\begin{array}{rcrr} 45 & R & 17 \leftarrow\text{———— 輸入資訊} & \\ 22 & & & 34 \\ 11 & R & 68 & \\ 5 & R & 136 & \\ 2 & & & 272 \\ 1 & R & \underline{544} & \\ & & 765 \leftarrow\text{———— 輸出資訊} & \end{array}\right.$$

最後一個步驟是在餘數出現的各列上，將 17 的那些倍數相加。結果所得到的輸出資訊 765 是與題 1.1 中的相同。這表示計算乘積能有一個以上的演算法可以使用。

1.3. 兩個長度 X 與 Y 經過量度約爲 $X \sim 3.32$ 與 $Y \sim 5.39$，符號～代表近似等式。以“三數位加法”來計算 $X+Y$，$X+(.1)Y$，與 $X+(.01)Y$ 等的近似值。

這是一個初等數學問題，用來說明在計算所用的數學中誤差的存在。

$$\begin{array}{r} 3.32 \\ \underline{5.39} \\ X+Y \sim 8.71 \end{array} \qquad \begin{array}{r} 3.32 \\ \underline{0.54} \\ X+(.1)Y \sim 3.86 \end{array} \qquad \begin{array}{r} 3.32 \\ \underline{0.05} \\ X+(.01)Y \sim 3.37 \end{array}$$

此處所有數字都保持三個數位的等長度，每當需要時，**便捨入**(Rounding off)，或首項加零。這是電腦計算的特徵。就如像在這裡所進行的一樣，電腦將等長度的數字儲存起來並加以運算。通常電腦所使用的數字長度是六個或六個以上的數位，但爲了說明，只用了三個數位。此題的作用在下面的圖中加以綜合，參看圖 1-3 。

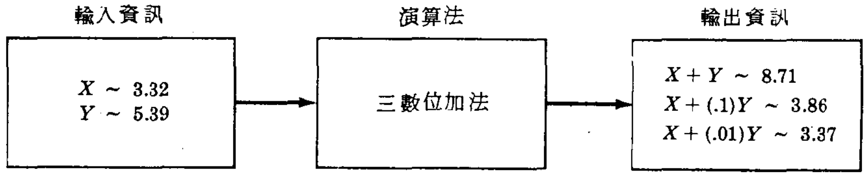

圖 1.3

1.4.　指出題 1.3 中的誤差源，並說明它們的大小。

假定輸入資訊 3.32 與 5.39 正確到三個數位，並仍以 X 與 Y 表示未知數的正確值，於是，輸入資訊內的誤差爲

$$E_1 = X - 3.32, \quad E_2 = Y - 5.39$$

兩者誤差都不超過 .005。另外還有演算法內的誤差存在。在求 (.1) Y 的近似值時，演算法將 .539 捨入爲 .54，而在求 (.01) Y 的近似值時，它將 .0539 捨入爲 .05，兩個誤差都是在 .001 的大小級。

1.5.　估計由於題 1.4 中的誤差源在輸出資訊內所產生的誤差。

首先考慮 $X+Y$ 。從題 1.4 中的方程式很易求得

$$(X+Y) - 8.71 = (X - 3.32) + (Y - 5.39) = E_1 + E_2$$

因此，未知數正確值 $X+Y$ 與其計算的近似值 8.71 間的差爲

$$|X+Y-8.71| \leqq .005 + .005$$

或 .01 。故在 8.71 中第二位小數值得懷疑。注意，演算法誤差在此直加問題中不起作用。其次考慮 $X+(.1)Y$ 。由於

$$(.1)Y = (.1)(E_2 + 5.39) = (.1)E_2 + .539$$

很易求得

$$X + (.1)Y - 3.86 = E_1 + 3.32 + (.1)E_2 + .539 - 3.86 = E_1 + (.1)E_2 - .001$$

而未知數正確值$X + (.1)Y$ 與其計算的近似值 3.86 間的差爲

$$|X + (.1)Y - 3.86| \leqq |E_1| + |(.1)E_2| + |-.001| \leqq .005 + .0005 + .001$$

即不超過 .0065 。此式中 .005 爲輸入誤差， .0005 是當演算進行時，乘以 .1 後的輸入誤差， .001 是演算法內的捨入誤差。同樣地，求得

$$X + (.01)Y - 3.37 = E_1 + 3.32 + (.01)E_2 + .0539 - 3.37 = E_1 + (.01)E_2 + .0039$$

於是，計算的近似值 3.37 內的誤差是

$$|X + (.01)Y - 3.37| \leqq .005 + .00005 + .0039$$

或不超過 .009。在所有輸出資訊中，第二位小數似乎是值得懷疑。此題表示即使是在簡單的計算中，誤差大小的問題也是不易回答的。在這題中得到了可能的最大誤差估計。但在題 1.6 中將會發現這些估計值是太悲觀了些。

1.6.　**假設題 1.3 中的 X 與 Y 是 11 與 29 的平方根。現在不必量度這兩個長度，而可加以計算。（參看後面一章中計算平方根的方法。）所得結果正確到六個數位，即 $X \sim 3.31662$ 與 $Y \sim 5.38616$ 。重新計算題 1.3 中所需的結果，並將此題中輸出資訊的實際誤差與題 1.5 中計算的可能最大誤差加以比較。**

使用六數位算術，很易求得

$$X + Y \sim 8.70278, \quad X + (.1)Y \sim 3.85524, \quad X + (.01)Y \sim 3.37048$$

題 1.5 中的最大誤差分析，現在可證明這些結果至少正確到第四位小數。並且，在題 1.3 中計算的實際誤差也可更準確地加以估計。

	$X + Y$	$X + (.1)Y$	$X + (.01)Y$
實際誤差	.0073	.0048	.0005
最大誤差	.0100	.0065	.0090

在$X + (.1)Y$ 中的實際誤差是遠小於最大誤差。實際的誤差估計爲數值分析的最困難工作之一。常常將正確解已知的一個問題用來檢定演算法中誤差的行爲，正如像在此情況中。

1.7.　**使用三數位算術來求二次方程式 $x^2 - 20x + 1 = 0$ 的較小根。**

根據熟悉的代數定理，此式的兩個根爲 $10 \pm \sqrt{99}$ 較小的一個根具有負號。使用三數位算術，得出

$$10 - \sqrt{99} \sim 10.0 - 09.9 = 00.1$$

這可用來說明兩個近似相等的數字相減時的情況。雖然這些數字本身都可具有三數位準確度，但在相減過程中，這些數位中的一部分（或許全部）都會消失。此題的主要部分在下面的圖中加以綜合（圖 1-4）。而較佳的演算法，參看題 1.8 。

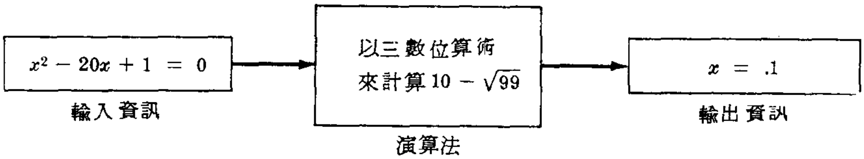

圖 1.4

1.8.　注意，理論結果 $10-\sqrt{99} = 1/(10+\sqrt{99})$ ，使用此式等式右邊的式子來計算題 1.7 中所需的根。

再使用三數位算術，

$$10 + \sqrt{99} = 10.0 + 09.9 = 19.9 \qquad \text{而} \qquad 1.00/19.9 = .0503$$

大多數電腦在乘法與除法的結果內，將首項都放置零，同時並保留其後表示機器容量的數位數目（在此情況中爲 3 ）。換句話說，對於上面的除法，可把輸出當作三數位數字，而略去首項的零。但要注意，在 10.0 與 09.9 的加法中，這個首項零是作用中的三個數位之一。在題 1.3 中也是一樣，這是電腦中典型的加法運算。圖 1-5 將此計算的主要部分加以綜合。

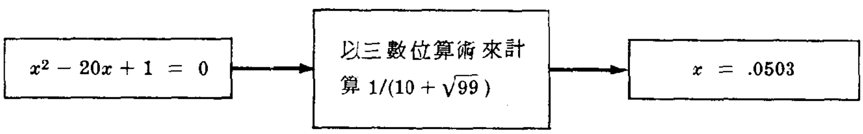

圖 1.5

支持理論對此根的計算提供了一個交替的演算法。誤差分析可予以略去，但新的結果是正確到三位小數，而使其遠比題 1.7 的爲優。新的演算法引進很小的演算誤差。

.9.　計算和 $\sqrt{1} + \sqrt{2} + \cdots\cdots + \sqrt{100}$ 。

假設首先從平方根表將所有平方根得到兩位小數，前面的幾個是 1.00, 1.41, 1.73, 2.00, 等，然後再提供計算根的演算法。於是，這一百個數字的和得出 671.27 。顯然，此和至少需要“五數位算術”才能計算。由於在演算過程中已作了一百次的捨入，因此，這個結果的準確度是不定的，但可參看其次的幾題。此計算在下面的圖中加以綜合（圖 1-6 ）。

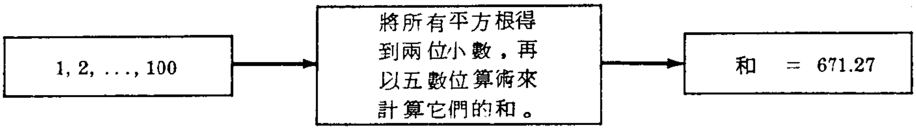

圖 1.6

1.10. 假設數字 x_1，$x_2,\cdots,x_N$ 是 X_1，X_2，$\cdots$，X_N 的近似值，並且在每一種情況中，可能的最大誤差爲 E。證明和 $x_1+x_2+\cdots+x_N$ 中的可能最大誤差爲 NE。

這個問題對支持理論提供另一個例子，由於

$$x_i - E \leqq X_i \leqq x_i + E$$

故可利用加法得到

$$\Sigma x_i - NE \leqq \Sigma X_i \leqq \Sigma x_i + NE$$

因此，$-NE \leqq \Sigma X_i - \Sigma x_i \leqq NE$, 即爲所需證明的。

1.11. 在題1.9中，將各數正確到二位小數的一百個數字求和，那麼，它們的和中可能的最大誤差是多少？

每一個數字中的誤差最多爲 .005。以 $E=.005$ 與 $N=100$ 代入題1.10 的公式中，於是，求得 $NE=.5$。這暗示此和可能正確還不到一位小數。（也參看題1.12。）

1.12. 一個統計公式也爲支持理論，暗示當 N 個數字求和時，**可能誤差（概差）**（Probable error）是 $\sqrt{N}E$，E 爲 N 個數字的可能最大誤差。應用此公式來求題1.9中所計算的和的可能誤差（概差）。

以 $N=100$，與 $E=.005$，可能誤差（概差）$=\sqrt{N}E=10(.005)=.05$。這比題1.11中所求得的 .5 的最大誤差估計較樂觀。但究竟那一種估計更接近眞值？

1.13. 對於題1.9中的和，可利用一種新的計算法，將所有平方根最先都求得五位小數而非二位小數，於是，得出的和爲671.36385。顯然，這需要八數位算術來計算。使用題1.10來證明此和中的誤差最多爲 .0005，而使其至少正確到三位小數。再將所得的結果的實際誤差與題1.11和1.12 的最大誤差和可能誤差估計加以比較。

以 $N=100$，與 $E=.000005$，得出 $NE=.0005$，正如像所需求的。於是，各種誤差爲

實際誤差 $\sim 671.36-671.27=.09$

可能的最大誤差 $=.50$

可能誤差（概差）$=.05$

其中一種估計是太悲觀，另一種估計是太樂觀。在此題中，可利用能含八數位算術的電腦來核對這種計算法的準確度與探討誤差的發展。

1.14. 已知

$$A_n = 1 - \frac{1}{2} + \frac{1}{3} - \frac{1}{4} + \cdots + (-1)^{n+1} \cdot \frac{1}{n}$$

計算 $\lim A_n$ 正確到三個數位。

下面的基本分析定理是常用的一種支持理論。即“在無限序列 $A_1, A_2, A_3, \ldots$ 中，如果 A_n 是交錯增加與減少的，差 $|A_n - A_{n+1}|$ 爲單調減至零，並且，$|(\lim A_n) - A_n| \leq |A_n - A_{n+1}|$，那麼，此序列便是收斂序列。”對於上面已知的序列，這暗示收斂，也就是由於 $\lim A_n$ 的存在，與 A_n 和 $\lim A_n$ 間的差不能大於 $1/n$ 的事實。三數位準確度可容許最多 .0005的誤差，而使 $1/n \leq .0005$ 與 $n \geq 2000$ 便可達到此準確度。這表示 A_{2000} 即爲足夠準確的近似值。但如何計算此數字？假設有八數位電腦可以使用，於是，各倒數都可表示成八數位小數，而大部分都需要捨入。將 2000 個這樣的數字求和可能產生誤差

$$NE = (2000)(.000\,000\,005) = .00001$$

這似乎是可以略去。因此，可容許八數位電腦來計算此冗長的和。當捨入到三個數位後，所得的結果爲：計算的和 = .693.

注意，在此題中，輸入資訊無誤差。並且，已知 A_n 的正確公式。所有誤差都是演算誤差。首先決定計算 A_{2000} 而不是 $\lim A_n$。這可當作是將無限級數第 2000 項以後的各項捨去的結果，而是稱爲**捨項誤差**（Truncation error）的一個例子。捨項誤差是當無限過程以有限過程代替時所構成的。在此題中，

$$\text{捨項誤差} = \lim A_n - A_{2000}$$

並已使其保持小於 .0005。其次，使所含倒數近似八數位小數與將那些小數求和時，也引進演算誤差。換句話說，並不是計算 A_{2000}，而只是其近似值。這種誤差稱爲**捨入誤差**（Roundoff error）.

$$\text{捨入誤差} = A_{2000} - \text{計算的和}$$

並已使其保持小於 .00001。由於所構成的實際誤差爲

$$\lim A_n - \text{計算的和} = (\lim A_n - A_{2000}) + (A_{2000} - \text{計算的和})$$

可以看出

$$\text{實際誤差} = \text{捨項誤差} + \text{捨入誤差}$$

這並不奇怪，而使 |實際誤差| ≦ .00051，暗示三數位結果是幾乎正確的。此題的主要部分在下面的圖中加以綜合（圖 1-7）。

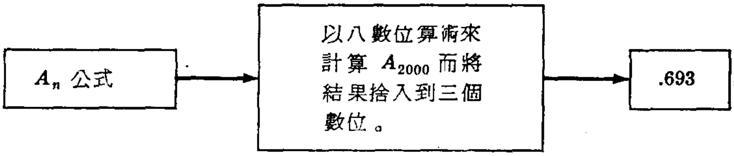

圖 1.7

1.15. 證明如果下列級數

$$a_1 - a_2 + a_3 - a_4 + \cdots$$

是收斂的，其中所有 a_k 數字都爲正值，那麼，級數

$$\tfrac{1}{2}a_1 + \tfrac{1}{2}(a_1 - a_2) - \tfrac{1}{2}(a_2 - a_3) + \tfrac{1}{2}(a_3 - a_4) - \cdots$$

也是收斂的，而表示相同的數字。

這是支持理論的另一個例子。使用 A_n 與 B_n 以表示兩個級數的第 n 個部分和，於是，很容易求得 $A_n - B_n = \pm\tfrac{1}{2}a_n$。由於第一個級數是收斂的，$\lim a_n = 0$，因此，$\lim A_n = \lim B_n$，正如像上面所說的。

1.16. 應用題 1.15 的定理來計算題 1.14 的 $\lim A_n$ 到三位小數。

新的演算法使 $\lim B_n$ 近似 B_n 數字之一。很容易求得 $B_1 = \tfrac{1}{2}$，且當 $n > 1$ 時，

$$B_n = \frac{1}{2}a_1 + \frac{1}{2}(a_1 - a_2) + \cdots + (-1)^n\frac{1}{2}(a_{n-1} - a_n) = \frac{1}{2} + \sum_{k=2}^{n}(-1)^k\frac{1}{2}\frac{1}{k(k-1)}$$

對於此序列，題 1.14 的定理確保 B_n 與 $\lim B_n$ 間的差不大於 $1/2n(n+1)$。而對於三數位準確度，需要

$$\frac{1}{2n(n+1)} \leqq .0005$$

使 $2n(n+1) \geqq 2000$，或 $n \geqq 32$。這表示 B_{32} 是足夠準確的近似值。差 $\lim B_n - B_{32}$ 不大於 .0005。並且，也引進捨入誤差。其分析在這裡比在題 1.14 中更難，而予以略去。

如果像在題 1.14 中一樣使用八數位算術，那麼，便可期望捨入誤差不會影響第三位小數。即使是這樣，但由於實際的誤差是捨項誤差與捨入誤差的和，以及需要 |實際誤差| $\leqq$ .0005，因此，使捨項誤差降低到 B_{32} 所確保的 .0005 以下，而留些微餘地來容納捨入誤差，似乎是較好的。對於新的演算法，假設以八數位算術來計算 B_{40}，而當捨入到三個數位以後，所得結果變成：計算的和＝ .693。這與前面的結果一致。但此題的重點是 B_{40} 正與 A_{2000} 一樣合用！而新的演算法是比前一個演算法快速。此計算在圖 1-8 中加以綜合。

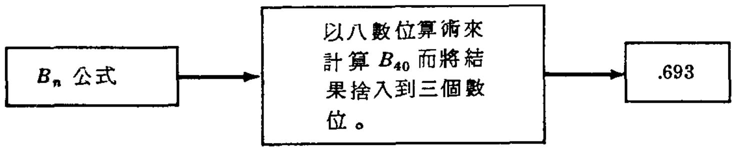

圖1.8

1.17. **證明如果 x_1，x_2 是 X_1，X_2 的近似值，而含有誤差 E_1，E_2，使得 $X_1 = x_1 + E_1$ 與 $X_2 = x_2 + E_2$，那麼，**

$$\frac{X_1X_2 - x_1x_2}{X_1X_2} \sim \frac{E_1}{X_1} + \frac{E_2}{X_2}$$

即乘積的相對誤差是近似各個因子的相對誤差的和。

由於 E_1E_2 與 E_1 或 E_2 比較是很小的，因此

$$X_1X_2 - x_1x_2 = E_1x_2 + E_2x_1 + E_1E_2 \sim E_1x_2 + E_2x_1$$

從這式當除以 X_1X_2 後，便得出所需的結果。

1.18. **正確（或有效）數位的數目是與相對誤差密切相關的。在一個乘積內正確數位的數目與各個因子的對應數目比較如何？**

對於相對誤差大約相同的兩個因子，前一題暗示，它們的乘積便大約具有該相對誤差的兩倍。乘積與各因子的正確數位的數目幾乎是相同的。例如，考慮 2 與 3 的平方根的乘積。對於含 2，3，與 4 個正確數位的因子，求得

$$1.4 \times 1.7 = 2.38, \quad 1.41 \times 1.73 = 2.4393, \quad 1.414 \times 1.732 = 2.449048$$

在每一種情況中，乘積與其各因子的準確度是近似相同的。因子增加，相對誤差也增加，很像和的實際誤差增加一樣。

補充題

1.19. 將 45×17 按下列演算法用心算：$17 \times 9 \times 10 \times \frac{1}{2}$.（從左到右做乘法。）

1.20. 將 45×17 按俄農演算法以 45 加倍與 17 減半來計算。（這是題 1.2 中所用的演算法的相反方式。）

1.21. 將 1/.982 以長除法計算到三位小數。

1.22. 將 1/.982 使用支持理論

$$\frac{1}{1-x} = 1 + x + x^2 + \cdots$$

以 $x = .018$ 來計算。這種演算法與題1.21中的演算法，那一種較快？

1.23. 如果 $X \sim 3.32$ 與 $Y \sim 5.39$ 正確到兩位小數，那麼，$X+Y$ 與 XY 實際可爲多大與多小？又 $3.32+5.39$ 和 $(3.32)(5.39)$ 與這些可能的極值比較如何？

1.24. 當數字的誤差不小於 .005 時，數字便準確到兩位小數。下列平方根是從表中得到。將各數捨入到兩位小數並記錄捨入的數。又這些捨入誤差與 .005 的最大誤差比較，如何？

n	11	12	13	14	15	16	17	18	19	20
將$\sqrt{n}$捨入到三位小數	3.317	3.464	3.606	3.742	3.873	4.000	4.123	4.243	4.359	4.472
將$\sqrt{n}$捨入到兩位小數	3.32	3.46								
近似捨入的數	+.003	−.004								

理論上總捨入誤差可從 $10(-.005)$ 到 $10(.005)$。實際上總捨入誤差是多少？又它與 $\sqrt{10}(.005)$ 的可能誤差（概差）比較如何？

1.25. 假設將正確到一已知數位小數的N個數字求和。那麼，當計算的和的最後一個數位是可能無意義時，N的大小是多少？又當最後兩個數位是可能無意義時，N是多少？使用可能誤差（概差）公式來計算。

1.26. 使用三數位算術來求二次方程式$x^2 - 20x + 2 = 0$的較小根。首先以 $10-\sqrt{98}$ 的演算法來計算，然後再以支持理論 $10-\sqrt{98} = 2/(10+\sqrt{98})$ 的演算法來計算。並將兩個結果代入二次方程式加以核對。那一個結果較準確？

1.27. 將題1.14的定理應用到序列

$$A_n = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \cdots \pm \frac{1}{2n-1}$$

如果要 $\lim A_n$ 正確到三位小數，那麼，必須將n選擇多大才能使 A_n 爲可接受的近似值？證明這需要

$$|A_n - A_{n+1}| = \frac{1}{2n+1} \leq .0005$$

來得到n近似1000。不要實際計算 A_{1000}。

1.28. 將題 1.15 的定理應用到題 1.27 中的序列，以得到較快速的收斂序列。由於 $a_n = 1/(2n-1)$，證明