Computació Numèrica

Notes de classe

Tema 1. Conceptes bàsics

M. Àngela Grau Gotés

10 de febrer de 2023

Índex

1 Guia d'estudi

1.1 Glosari

- Error absolut. Xifres decimals correctes .
- Error relatiu. Error relatiu percentual. Xifres significatives correctes.
- Errors d'arrodoniment.
- Errors de truncament.
- Errors de discretització: $e_d = \mathcal{O}(h^p)$.
- Propagació de l'error. Fórmula d'una variable i de dues.
- Nombre de condició / factor de propagació de l'error.
- Aritmètica de punt flotant de MATLAB® .
- Aritmètica de 64 bits en general.
- Norma `IEEE-754`.
- Epsilon de la màquina.
- Algorisme numèricament estable/inestable.
- Algorisme sensible a les condicions inicials.
- Algorisme amb pèrdua de xifres significatives.
- Algorisme amb cancel·lació.
- Método de Horner.

1.2 Referències

1.2.1 Llibres de consulta online

[1] Accès UPCommons, Càlcul numèric: teoria i pràctica

- Conceptes associats: capítol 1, de la pàgina 2 a la 30.
- Problemes resolts : intercalats en els conceptes.
- Problemes proposats: 2, 3 i 9.
- Pràctiques resoltes : capítol 1, de la pàgina 35 a la 41.
- Pràctiques proposades: de la pàgina 41 a la 46.

[2] Accès UPCommons, Cálculo numérico

[3] Accès Biblioteca, Cálculo Científico con MATLAB y Octave by A. Quarteroni, F. Saleri

[4] Accès lliure, Cleve Moler - Llibre de text i codis - MathWorks

1.2.2 Lectures de comprensió del tema

- What is Scientific Computing ?
- Scientific computing
- Whats the difference between precision and accuracy?
- Some disasters attributable to bad numerical computing
- Cleve's Corner: Cleve Moler on Mathematics and Computing,
  Floating Point Arithmetic Before IEEE 754
- Cleve's Corner: Cleve Moler on Mathematics and Computing,
  Floating points. IEEE Standard unifies arithmetic model
- William Kahan, Pàgina web
- Charles Severance, Reminiscències per a IEEE Computer, 20 de febrer de 1998. An Interview with the Old Man of Floating-Point

- Reprint of the paper "What Every Computer Scientist Should Know About Floating-Point Arithmetic", by David Goldberg, published in the March, 1991 issue of Computing Surveys. Copyright 1991, Association for Computing Machinery, Inc. What Every Programmer Should Know About Floating-Point Arithmetic
- Cleve's Corner: Cleve Moler on Mathematics and Computing, Floating Point Numbers, Floating Point Denormals, . . . ,
- Lecture Notes on the Status of IEEE Standard 754 for Binary Floating-Point Arithmetic (30 pages) IEEE754.pdf
- IEEE Std 754?-2008 (Revision of IEEE Std 754-1985). IEEE-754-2008.pdf
- Moler, C. ?A Brief History of MATLAB? Cleve's Corner Collection. The MathWorks,nc.,8 Sep 2018 A Brief History of MATLAB
- Floating point guide floating-point-guide

2 Conceptes bàsics

2.1 Introducció

Durant els segles XX i XXI models matemàtics avançats s'han aplicat en diferents àrees de coneixement com l'enginyeria, la medicina, l'economia o les ciències socials. Sovint, les aplicacions generen problemes matemàtics que per la seva complexitat no poden ser resolts de manera exacta.

La **matemàtica computacional**, s'ocupa del disseny, anàlisi i implementació d'algorismes per obtenir solucions numèriques aproximades de models físics, químics, matemàtics, estadístics, ...

2.1.1 Modelització d'un problema científic.

Davant d'un fet real, la modelització consisteix a construir un conjunt de fórmules i equacions que ens el representin de la manera més fidel possible, de manera que ens permeti fer prediccions correctes.

Els models resultants quasi mai no poden ser resolts per complet utilitzant mètodes (llapís i paper) d'anàlisi. La simulació en un ordinador ens permet interpretar els resultats i comparar-los amb les dades experimentals.

Davant d'un fet real, la modelització consisteix a construir un conjunt de fórmules i equacions que ens el representin de la manera més fidel possible, de manera que ens permeti fer prediccions correctes. Els models resultants quasi mai no poden ser resolts per complet utilitzant mètodes (llapís i paper) d'anàlisi. La simulació en un ordinador ens permet interpretar els resultats i comparar-los amb les dades experimentals.

Exemple La funció definida per

$$\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2}\, dt$$

s'anomena **funció error**, s'aplica si els resultats d'un conjunt de mesures es descriuen per una distribució normal de mitja zero i desviació estàndar σ, llavors $\operatorname{erf}\left(\frac{\epsilon}{\sigma\sqrt{2}} \right)$ és la

probabilitat que l'error en la mesura x es trobi entre $-\epsilon$ i ϵ. Però la integral definida no es pot expressar per mitja de funcions elementals. Cal obtenir aproximacions numèriques!

Una opció són les sèries de potències. Una sèrie de potències en un entorn de 0 de la funció error és:

$$erf(x) = \frac{2}{\sqrt{\pi}} \sum_{n=0}^{\infty} \frac{(-1)^n x^{2n+1}}{n!(2n+1)} = \frac{2}{\sqrt{\pi}} \left(x - \frac{x^3}{3} + \frac{x^5}{10} - \frac{x^7}{42} + \frac{x^9}{216} - \cdots \right)$$

2.1.2 Fonts d'error en la resolució numèrica d'un problema científic.

Fixarem quatre grans fonts d'error, que poden influir en una aproximació "pobre" del fet observat:

- Error de modelització. Una elecció equivocada o inapropiada de model matemàtic. Es busca que la solució del mateix no s'aparti significativament de la solució del problema original.
- Error de truncament. Aproximacions discretes i finites del model matemàtic (algorismes). La impossibilitat material de calcular els infinits termes del desenvolupament en sèrie d'una funció exigeix truncar-lo a un nombre finit de termes, per calcular amb l'ordinador.
- Error experimental. Mesures incorrectes o dolentes.
  - Errors aleatoris. Les mesures.
  - Errors sistemàtics. Cal·libració incorrecte.
  - Errors aberrants. Per descomptat també hi ha el factor humà. Un error per descuit o ignorància.
- Error d'arrodoniment. Error acumulat a causa de l'execució del nostre model en un ordinador on les operacions descrites per l'algorisme es fan amb un nombre finit de dígits.

2.1.3 Algorismes per a la resolució numèrica d'un problema científic.

En aquest curs treballarem amb dos tipus d'algorismes: directes i iteratius.

Directes El algorismes directes obtenen la solució en un nombre finit de pasos. Per exemple, *l'algorisme per resoldre les equacions de segon grau o l'algorisme d'eliminació gaussiana.* Aquests tipus d'algorismes, si fem ús d'un ordinador per calcular, presenenten errors d'arrodoniment: són deguts a l'aritmètica de punt/coma flotant de l'ordinador.

Iteratius El algorismes iteratius generen una seqüència de valors aproximats que convergeixen a la solució quan el nombre de passos tendeix a infinit. Per exemple, *els termes de la successió següent convergeixen a* $\sqrt{2}$*.*

$$x_k = \frac{1}{2}\left(x_{k-1} + \frac{2}{x_{k-1}}\right) \qquad k \geq 1 \ i \ x_0 = 3\,.$$

Aquests tipus d'algorismes, si fem ús d'un ordinador per calcular hem d'interrompre el proces infinit. Convertir un procés infinit en finit produeix errors de truncament. Fer ús de l'aritmètica de punt/coma flotant de l'ordinador produeix errors d'arrodoniment.

Característiques d'un algorisme Les solucions numèriques obtingudes amb l'ordinador són gairebé sempre solucions aproximades, fins i tot en molts casos les que es corresponen a un procediment teòricament exacte. En general, qualsevol algorisme que desenvolupem o del que fem ús ha de ser: exacte, estable, eficient i robust.

Exacte Què tan bo és l'algorisme d'aproximació de la quantitat a calcular (accuracy)

Estable La sortida de l'algorisme és sensible a petits canvis en les dades d'entrada (stability).

Eficient Quant costa (en nombre d'operacions) a obtenir una aproximació raonable (efficiency).

Robust Per a quants casos puc fer ús de l'algorisme (robustness).

En alguns casos també importa la memòria necessària (storage) i si és paral·lelizable (parallelization).

2.2 Exactitud de les solucions numèriques

Les solucions que s'obtenen pels algorismes numèrics són, en la gran majoria de les ocasions, solucions aproximades. Interessa, per tant, mesurar el grau d'aproximació assolit amb la nostra solució, o el que és el mateix, l'error d'aproximació.

2.2.1 Definició d'error: error absolut i error relatiu

Error absolut Anomenem error absolut, escrit com ϵ_a, a la diferència entre el valor exacte d'una quantitat i el valor aproximat d'aquesta quantitat, fórmula (2.2.1), sino importa el signe fem ús de la fórmula (2.2.2). Usualment es treballa amb fites de l'error absolut, notació ϵ_a. Un defecte que té el concepte és que no considera la magnitud del valor i depen de les unitats de la quantitat x.

Si notem per x, el valor exacte i per $\widetilde{x}$, un valor aproximat de x, les expressions per calcular l'error d'arrodoniment són:

$$e_a(x) = x - \widetilde{x} \tag{2.2.1}$$

$$\Delta x = |x - \widetilde{x}| \tag{2.2.2}$$

Xifres decimals correctes (accurate) Direm que $\widetilde{x}$ és una aproximació a x amb **d** xifres decimals correctes si d és el nombre natural més gran tal que

$$|x - \widetilde{x}| < 0.5 \cdot 10^{-d} \tag{2.2.3}$$

Per x, el valor exacte i per $\widetilde{x}$, un valor aproximat de x, el nombre de decimals correctes es pot calcular fent

$$\lfloor -\log_{10}(2 \cdot \epsilon_a) \rfloor \tag{2.2.4}$$

Error relatiu Anomenem error relatiu, escrit com ϵ_r, al quocient de l'error absolut amb el valor exacte d'una quantitat, fórmula (2.2.3). Usualment es treballa l'error relatiu en tant per 100, anomenat *error relatiu percentual* i amb fites de l'error relatiu, notació ϵ_r. La fórmula (2.2.4) reb el nom de *error relatiu aproximat*.

Si x és el valor exacte i $\widetilde{x}$, un valor aproximat de la quantitat x les expressions per calcular l'error d'arrodoniment són:

$$e_r(x) = \frac{\Delta x}{|x|} \tag{2.2.5}$$

$$e_r(\widetilde{x}) = \frac{\Delta x}{|\widetilde{x}|} \tag{2.2.6}$$

Xifres significatives correctes (precision) Direm que $\widetilde{x}$ és una aproximació a x amb **t** xifres significatives si t és el nombre natural més gran tal que

$$\frac{|x-\widetilde{x}|}{|x|} < 0.5 \cdot 10^{-t} \tag{2.2.7}$$

Per x, el valor exacte i per $\widetilde{x}$, un valor aproximat de x, el nombre de decimals correctes es pot calcular fent

$$\lfloor -\log_{10}(2 \cdot \epsilon_7) \rfloor \tag{2.2.8}$$

Example 2.2.1

Sigui $x = \sqrt{2} = 1.414213562\ldots$ i $\widetilde{x} = 1.414$, aleshores

$$\Delta x = 0.0002135\ldots \qquad \epsilon_x = 0.00015099\ldots$$

i les fites podrien ser

$$\epsilon_a = 0.00022, \qquad \epsilon_r = 0.00016.$$

Example 2.2.2

Per a $x = \pi$ i $\widetilde{x} = 3.141$, tenim

$$\Delta x = 0.00059265\ldots \qquad e_r(x) = 0.000188647\ldots$$

i les fites podríen ser

$$\epsilon_a = 0.6 \cdot 10^{-3}\,, \quad x = 3.141 \pm 0.6 \cdot 10^{-3}\,.$$

$$\epsilon_r = 0.2 \cdot 10^{-3} = 0.02\%\,, \quad x = 3.141 \cdot (1 \pm 0.02\%)\,.$$

2.2.2 Errors d'arrodoniment

El concepte error d'arrodoniment fa referència a l'error acumulat a causa de l'execució del nostre model en un ordinador on les operacions descrites en l'algorisme es realitzen amb un nombre finit de dígits. L'error d'arrodoniment s'acumula a mesura que augmenta el nombre de càlculs que fan en un nombre finit de dígits.

La representació decimal d'un **nombre real** es redueix per tal representar/usar els nombres reals a l'ordinador o en càlculs manuals. La representació decimal d'un nombre és pot reduir per tall o arrodonint del nombre de dígits.

Tallar nombres decimals Sigui x qualsevol nombre decimal positiu de la forma

$$x = 0.d_1d_2\ldots d_{n-1}d_nd_{n+1}\ldots d_m\,, \tag{2.2.9}$$

La aproximació per tall del nombre x a n dígits ($n < m$) és el nombre $\widehat{t_x}$ obtingut en descartat tots els dígits posteriors al dígit n.

$$x = 0.d_1d_2\ldots d_n\ldots d_m \xrightarrow[\text{n dígits}]{tallar} \widehat{t_x} = 0.d_1d_2\ldots d_n\,, \tag{2.2.10}$$

Estimació error tallar Si el nombre x es talla, i $\widehat{t_x}$ és el seu valor aproximat a n dígits, aleshores la fita de l'error absolut és $|x - \widehat{t_x}| \leq 10^{-n}$.

Arrodonir nombres decimals Sigui x qualsevol nombre decimal positiu de la forma

$$x = 0.d_1d_2\ldots d_{n-1}d_nd_{n+1}\ldots d_m\,, \tag{2.2.11}$$

llavors $\widetilde{r_x}$, l'arrodoniment de x a n xifres decimals ($n < m$) depèn del valor del dígit $n+1$.

$$\widetilde{r_x} = 0.d_1d_2\ldots d_{n-1}d\,, \tag{2.2.12}$$

$$d = \begin{cases} d_n & \text{si} \quad d_{n+1} \in \{0,1,2,3,4\}\,, \\ d_n + 1 & \text{si} \quad d_{n+1} \in \{5,6,7,8,9\}\,. \end{cases}$$

Estimació error arrodonir Si el nombre x s'arrodoneix, i $\widetilde{r_x}$ és el seu valor arrodonit a n dígits, aleshores la fita de l'error absolut és $|x - \widetilde{r_x}| \leq 0.5 \cdot 10^{-n}$.

2.2.3 Errors de truncament

Els errors de truncament són els que resulten en fer servir una aproximació en lloc d'un procediment matemàtic exacte.

En l'aproximació de funcions mitjançant el seu desenvolupament en sèrie de potències, la impossibilitat material de calcular els infinits termes del desenvolupament exigeix truncar-lo a un nombre finit de termes, fet que genera aquest tipus d'error. En general, com més termes es considerin, menor serà l'error de truncament.

Example 2.2.3

Leibniz va obtenir la següent sèrie matemàtica (1682):

$$\sum_{n=0}^{\infty} \frac{(-1)^n}{2n+1} = 1 - \frac{1}{3} + \frac{1}{5} - \cdots = \frac{\pi}{4}$$

De vegades, el procés numèric és una aproximació del model matemàtic obtingut com a funció d'un paràmetre de discretització, que serà notat per h, i suposarem positiu. Si, quan h tendeix a 0, el procés numèric torna la solució del model matemàtic, direm que el procés numèric és convergent.

A més, si l'error (absolut o relatiu) es pot fitar en funció de h, de la forma

$$e_d \leq Ch^p\,,\ C>0\,,\ p>0 \Longleftrightarrow e_d = \mathcal{O}(h^p)$$

es diu que *el mètode és convergent d'ordre p* i escrivim $e_d = \mathcal{O}(h^p)$. L'error de discretització serà menor si h disminueix o si l'ordre (p) augmenta.

Example 2.2.4

De la fórmula de Taylor per $f(x_0+h)$ s'obté:

$$f'(x_0) = \frac{f(x_0+h)-f(x_0)}{h} + \frac{f''(\xi)}{2}h,$$

i escrivim:

$$f'(x_0) \approx \frac{f(x_0+h)-f(x_0)}{h} \qquad e_d = \mathcal{O}(h)$$

llavors podem fer aproximacions numèriques de la derivada ($h>0$)

Example 2.2.5

La funció e^x es pot aproximar per la seva sèrie de Taylor en un entorn de $x_0=0$, per qualsevol h,

$$e^h \approx 1+h+\frac{h^2}{2!}+\frac{h^3}{3!}+\frac{h^4}{4!} \qquad e_d = \mathcal{O}(h^5)$$

llavors podem calcular e^h fent ús de funcions elementals.

2.2.4 Propagació de l'error

L'error d'arrodoniment també afecta les operacions aritmètiques que es facin amb l'ordinador; en una successió d'operacions es poden anar produint errors successius, que es van arrossegant (propagant), cosa que ocasiona una pèrdua general d'exactitud.

Fórmula de la propagació de l'error absolut

Si $f:\mathbb{R}\to\mathbb{R}$ és una funció derivable, x un nombre real, $\widetilde{x}$ una aproximació de x amb fita d'error ϵ, $x=\widetilde{x}\pm\epsilon$, el teorema del valor mig diu

$$|f(x)-f(\widetilde{x})| = |f'(\xi)|\,|x-\widetilde{x}|\,, \qquad |\widetilde{x}-x|\leq\epsilon_a.$$

Així, una manera de valorar l'efecte que tenen els errors en les dades d'entrada en el càlcul de $y = f(x)$ seria:

$$|\Delta f| \approx |f'(\widetilde{x})|\,\epsilon_a. \tag{2.2.13}$$

La fórmula (2.2.13) s'anomena fórmula de la propagació de l'error.

Si introduïm el factor $|f(x)|$ als dos costats de l'expressió i substituïm l'error absolut per l'error relatiu s'obté,

$$\left|\frac{\Delta f}{f(x)}\right| \approx \left|\frac{x\,f'(\widetilde{x})}{f(x)}\right| \epsilon_r \tag{2.2.14}$$

El factor que acompanya a l'error relatiu en la fórmula (2.2.14) s'anomena nombre de condició del problema $y = f(x)$.

El teorema del valor mig per funcions de diverses variables, ens proporciona fórmules per estudiar la propagació de l'error en càlculs que impliquin més d'una dada.

Propagació de l'error per les operacions aritmètiques elementals

Si $g:\ \mathbb{R}^2 \to \mathbb{R}$ és una funció diferenciable i $x_1 = \widetilde{x}_1 \pm \epsilon_1$ i $x_2 = \widetilde{x}_2 \pm \epsilon_2$, llavors FGPE en dues variables

$$|\Delta g| \approx \left|\frac{\partial g(\widetilde{x}_1, \widetilde{x}_2)}{\partial x_1}\right| |\epsilon_1| + \left|\frac{\partial g(\widetilde{x}_1, \widetilde{x}_2)}{\partial x_2}\right| |\epsilon_2|. \tag{2.2.15}$$

Si fem ús d'una funció real de dues variables serà possible fitar l'error propagat per les operacions elementals.

Example 2.2.6

Suma, $g(x_1, x_2) = x_1 + x_2$

$$x_1 + x_2 = (\widetilde{x}_1 + \widetilde{x}_2) \pm (\epsilon_1 + \epsilon_2) \tag{2.2.16}$$

Resta, $g(x_1, x_2) = x_1 - x_2$

$$x_1 - x_2 = (\widetilde{x}_1 - \widetilde{x}_2) \pm (\epsilon_1 + \epsilon_2) \tag{2.2.17}$$

Les fites dels errors absoluts es sumen en les operacions de sumar i restar nombres reals.

Producte, $g(x_1, x_2) = x_1 \cdot x_2$

$$|\Delta g| \approx |\widetilde{x}_2||\epsilon_1| + |\widetilde{x}_1||\epsilon_2|\,, \qquad \left|\frac{\Delta g}{g}\right| \approx \left|\frac{\epsilon_1}{\widetilde{x}_1}\right| + \left|\frac{\epsilon_2}{\widetilde{x}_2}\right|. \tag{2.2.18}$$

Divisió, $g(x_1, x_2) = x_1/x_2$

$$|\Delta g| \approx \left|\frac{1}{\widetilde{x}_2}\right| |\epsilon_1| + \left|\frac{\widetilde{x}_1}{\widetilde{x}_2^2}\right| |\epsilon_2|\,, \qquad \left|\frac{\Delta g}{g}\right| \approx \left|\frac{\epsilon_1}{\widetilde{x}_1}\right| + \left|\frac{\epsilon_2}{\widetilde{x}_2}\right|. \tag{2.2.19}$$

Les cotes dels errors relatius es sumen en les operacions de multiplicar i dividir nombres reals.

Fórmula general de propagació de l'error. FGPE

Sigui $g : \mathcal{D} \to \mathbb{R}\,,\ \mathcal{D}$ una regió de $\mathbb{R}^n$, g una funció diferenciable en un entorn del vector $\widetilde{x}$ $x = \widetilde{x} \pm \Delta x$, amb $\Delta x = (\Delta x_1, \ldots, \Delta x_n)^t$. Fent ús de la de la fórmula de Taylor aplicada a $y = g(x)$ i $\widetilde{y} = g(\widetilde{x})$ obtenim l'expressió:

$$|\Delta y| = |y - \widetilde{y}| \approx \sum_{i=1}^{n} \left| \frac{\partial g(\widetilde{x})}{\partial x_i} \right| \Delta x_i \,. \tag{2.2.20}$$

La fórmula (2.2.20) s'empra per obtenir una fita superior de l'error absolut propagat.

Dividint per $\widetilde{y}$ els dos costats de l'expressió (2.2.20), multiplicant i dividint per $\widetilde{x}_i$ i substituïnt l'error absolut per l'error relatiu en segon costat, s'obté,

$$\left| \frac{\Delta y}{\widetilde{y}} \right| \approx \sum_{i=1}^{n} \left| \frac{\widetilde{x}_i}{g(\widetilde{x})} \frac{\partial g(\widetilde{x})}{\partial x_i} \right| \left| \frac{\Delta x_i}{\widetilde{x}_i} \right| \,. \tag{2.2.21}$$

L'expressió (2.2.21) pot emprar per obtenir una fita superior de l'error relatiu propagat. Els n valors

$$\left| \frac{\widetilde{x}_i}{g(\widetilde{x})} \frac{\partial g(\widetilde{x})}{\partial x_i} \right| \tag{2.2.22}$$

s'anomenen **números de condició** del problema o factors de propagació. Aquests donen una mesura de quan un problema és mal condicionat.

2.3 Representació de valors numèrics

Els ordinadors emmagatzemen i operen els nombre en coma flotant. La precisió d'un ordinador dependrà del fabricant i del tipus de variable que es defineixi; la unitat d'informació ve donada pel nombre de dígits binaris o longitud de la paraula (word): *1 byte = 8 bits, 1 word = 2 bytes(PC).*

La longitud de les paraules imposa una restricció sobre la precisió amb la un ordinador pot representar nombres reals. Això vol dir que l'ordinador emmagatzema no el nombre real x si no una aproximació binària a aquest nombre, usualment es designa per $fl(x)$.

2.3.1 El conjunt $\mathsf{F}(\beta, \mathsf{t}, \mathsf{L}, \mathsf{U})$

El conjunt de nombres en coma flotant representables a l'ordinador el designarem per F(β,t,L,U), on β representa la **base**, t la **precisió** (o nombre de dígits representats o significatius) i el interval $[L, U]$ és el rang de l'exponent e.

Per a tot nombre real x expressat en el conjunt $F(\beta, t, L, U)$ existeixen $\boldsymbol{t}$ xifres i un exponent $\boldsymbol{e}$ tal que

$$fl(x) = \pm \left(\frac{d_1}{\beta} + \frac{d_2}{\beta^2} + \ldots + \frac{d_t}{\beta^t} \right) \cdot \beta^e \,,$$

amb dígits $d_i \in \mathbb{N}$ tal que $0 \leq d_i < \beta$ per a tot $i = 1 \div t$; i exponent $L \leq e \leq U$. Hi pot haver $U - L + 1$ exponents diferents. La quantitat

$$\pm \left(\frac{d_1}{\beta} + \frac{d_2}{\beta^2} + \ldots + \frac{d_t}{\beta^t} \right)$$

s'anomena fracció o part fraccionària del nombre x. Si exigim $d_1 \neq 0$ per a $x \neq 0$, resulta que $\beta^{-1} \leq |f| < 1$ i es diu que la representació és normalitzada. La quantitat $m = d_1 d_2 d_3 \ldots d_t$ s'anomena **mantissa**. Hi pot haver β^t mantisses diferents.

Si l'ordinador té l'aritmètica $F(\beta, t, L, U)$, llavors

$$fl(x) = x(1 + \delta) \qquad |\delta| \leq \epsilon_M = \frac{1}{2}\beta^{1-t} \,.$$

La **precisió** d'una aritmètica de coma flotant es caracteritza per l'**èpsilon de la màquina**, no és el nombre més petit representable, però dóna una mesura relativa de fins a on dos nombres molt pròxims seran diferents. El seu valor es correspon a la meitat distància entre 1 i el següent nombre en coma flotant.

Aritmètica a $F(\beta, t, L, U)$

Les operacions aritmètiques en coma flotant són

$$\begin{array}{lcll} x + y = & \longleftrightarrow & x \oplus y = & fl(fl(x) + fl(y)) \\ x - y = & \longleftrightarrow & x \ominus y = & fl(fl(x) - fl(y)) \\ x \times y = & \longleftrightarrow & x \otimes y = & fl(fl(x) \times fl(y)) \\ x \div y = & \longleftrightarrow & x \oslash y = & fl(fl(x) \div fl(y)) \end{array}$$

En totes les operacions s'ha de complir

$$x \circledast y = fl(x \circledast \ y)(1 + \epsilon_M).$$

Example 2.3.1

Imaginem un ordinador $F(10, 5, 0, 127)$, i els nombres $x = .31426 \cdot 10^3$ i $y = .92577 \cdot 10^5$.

$$\begin{array}{lrlr} x \times y = & .2909324802 \cdot 10^8 & x \otimes y = & .29093 \cdot 10^8 \\ x + y = & .9289126 \cdot 10^5 & x \oplus y = & .92891 \cdot 10^5 \\ x - y = & -.92262740 \cdot 10^5 & x \ominus y = & -.92263 \cdot 10^5 \\ x \div y = & .3394579647 \cdot 10^{-2} & x \oslash y = & .33946 \cdot 10^{-2} \end{array}$$

En aquests resultats l'error relatiu és de $8.5 \cdot 10^{-6}$, $2.3 \cdot 10^{-6}$, $2.8 \cdot 10^{-6}$, $6.0 \cdot 10^{-6}$, respectivament, tots per sota de 10^{-5}.

2.3.2 Norma $754 - 1985$

Els anys 60 i 70 les operacions amb nombres reals tenien implementacions diferents en cada ordinador: format, precissió, arrodoniment, gestió d'execepcions, etc. D'aquesta manera era molt difícil d'escriure codi portàtil.

El 1982 l' *Institute of Electrical and Electronics Engineers* va definir l'estàndar IEEE-754 i el va implementar en els processadors intel 8087. En tots els ordinadors que el tenien implementat, el programes obtenien els mateixos resultats. L'any 2002 l'estàndar IEEE-754 es va implementar universalment en tots els ordinadors de propòsit general.

L'any 1985, l'*Institute for Electrical and Electronic Engineers (IEEE)* va publicar l'informe

Binary Floating Point Arithmetic Standard $754 - 1985$,

en el que s'especifiquen normes per representar nombres en punt/coma flotant amb precisió simple, doble i extensa. L' informe va ser revisat i actualitzat l'any 2008, *IEEE Std 754-2008*.

Avui en dia, quasi tots els fabricants d'ordinadors han acceptat aquesta norma; per tant l'ordinador emmagatzema no el nombre real x si no una aproximació binària (octal o hexadecimal) en coma flotant a x. Consulteu Floating Point Numbers

El format stàndar de coma flotant de 32 bits

La representació en coma flotant amb simple precisió ocupa 1 paraula; té 32 dígits binaris, assignats de la manera següent,

signe del nombre real x	1 bit
exponent, (enter)	7 bits
mantissa, (real)	23 bits

Un nombre enter pot utilitzar tots els bits d'una paraula de l'ordinador, llevat del bit del signe. Per tant, en simple precisió els enters estan compresos entre

$$-(2^{31}-1) = -2147483647 \quad \text{i} \quad (2^{31}-1) = 2147483647.$$

Un nombre real x en simple precisió es representa en 32 bits binari per

$$fl(x) = (-1)^s \times M \times 2^{c-127}.$$

amb $M = 1.m_{22}m_{21}\ldots m_1m_0 = 1+f$

El format stàndar de coma flotant de 64 bits

La representació en coma flotant amb doble precisió

$$fl(x) = (-1)^s \times M \times 2^{c-1023}$$

ocupa 2 paraules; té 64 dígits binaris, assignats de la manera següent

s signe del nombre real x	1 bit
c exponent, (enter)	11 bits
m mantissa, (real)	52 bits

La condició de 11 bits per l'exponent significa que $|e| \leq 2^{11}-1 = 2047$, restant 1023 obtenim el rang $(-1023, 1024)$ per a l'exponent.

Si s'acorda la representació en que el primer dígit binari de la mantissa sigui 1, $M = 1.m_{51}m_{50}\ldots m_1m_0 = 1+f$, podem tindre matises de 53 dígits binaris, que es corresponen almenys 15 xifres decimals de precisió.

El nombre positiu més petit és $s=0$, $c=1$ i $f=0$ que es correspon al nombre decimal $2^{-1022}\cdot(1+0) \approx 0.2225\cdot 10^{-307}$.

El nombre positiu més gran és $s=0$, $c=2046$ i $f=1-2^{-52}$ que es correspon al nombre decimal $2^{1023}\cdot(1+f) \approx 0.17977\cdot 10^{309}$.

Consulteu el document *Aritmètica en coma flotant* by Cleve Moler per una descripció amb més detall l'aritmètica de 64 bits de MATLAB® .

2.4 Estabilitat numèrica i problemes ben condicionats

2.4.1 Error numèric total

Si f representa a l'algoritme *real* i $f^\star$ a l'algoritme *computacional*, x el nombre *real* i x^* el nombre *computacional*, aleshores l'error en el resultat final es pot definir com:

$$|f(x) - f^*(x^*)| \leq \underbrace{|f(x) - f(x^*)|}_{\text{condició}} + \underbrace{|f(x^*) - f^*(x)|}_{\text{estabilitat}} + \underbrace{|f^*(x) - f^*(x^*)|}_{\text{truncament}}$$

L'error numèric total és la suma de tots els errors comesos. Limitant-nos a analitzar els errors, des del model matemàtic fins que s'obté una solució numèrica, les dues fonts principals d'error són l'error d'arrodoniment i l'error de truncament. Aquests errors poden i han de ser controlats amb mesures de caràcter general.

- Per reduir l'error de truncament del mètode cal millorar de l'exactitud del mètode numèric emprat, fet que sol comportar un increment en el nombre d'operacions aritmètiques, fet que fa augmentar l'error d'arrodoniment, a més d'incrementar el temps de càlcul.
- Per reduir l'error d'arrodoniment es presta atenció a l'ordre i la forma com es realitzen les operacions, i s'incrementa el nombre de xifres significatives amb què opera l'ordinador ja que l'error d'arrodoniment s'acumula a mesura que augmenta el nombre de càlculs que fan en un nombre finit de dígits.
- Per tal de reduir o evitar la propagació dels errors es recomana, minimitzar el nombre d'operacions, reordenar les operacions i replantejar el problema en altres termes. P.e. fer ús de la **regla de Horner** per avaluar polinomis.

Estabilitat numèrica

Un algorisme el classificarem com **numèricament estable** si un error, no creix gaire en el procès de càlcul. L'estabilitat numèrica es veu afectada pel nombre de xifres significatives, poques xifres o la pèrdua en pasos intermitjos del càlcul disminueix la fiabilitat dels resultats obtinguts.

Example 2.4.1

L'expressió $\left(1 + 10^{-30} - 1\right) \cdot 10^{30}$ operada en l'ordre que s'indica, dóna a la majoria dels ordinadors com a resultat 0, en contrast amb el resultat correcte que és 1.

Algorismes amb cancel·lació La **pèrdua de xifres significatives** per cancel·lació, es produeix en restar dos nombres molt propers. La situació es pot resumir en

$$g(x+\delta) - g(x) \quad \text{amb } |\delta| \ll 1\,; \tag{2.4.1}$$

Example 2.4.2

Les solucions de $x^2 - 18x + 1 = 0$ són $x_{1,2} = 9 \pm \sqrt{80}$.
Si $\sqrt{80} = 8.9443 \pm 0.5 \cdot 10^{-4}$ llavors $x_1 = 17.9443 \pm 0.5 \cdot 10^{-4}$, té 6 xifres significactives, mentre que $x_2 = 0.0557 \pm 0.5 \cdot 10^{-4}$, només en té 3.

Inestabilitat numèrica Sense rigor, diem que un procés numèric és inestable quan els petits errors que es produeixen en un dels seus estadis s'agranda en etapes posteriors, fins a tal punt que no podem fiar-nos del càlcul global.

Example 2.4.3

Per calcular les integrals $I_n = \int_0^1 x^n e^{x-1} dx\,, \quad n \geq 1\,$, disposem de dos mètodes iteratius diferents:

$$\text{a)} \quad I_{n-1} = \frac{1 - I_n}{n}, \qquad n \geq 2 \qquad \text{on } I_{50} = 0\,,$$

$$\text{b)} \quad I_n = 1 - nI_{n-1}, \quad n \geq 2 \quad \text{on } I_1 = 1/e\,.$$

Sensibles a les condicions inicials Molts problemes són especialment sensibles a les dades inicials, independentment dels errors d'arrodoniment i de l'algorisme emprat.

Són problemes on la solució depèn de manera molt sensible de les dades. Si petites variacions de les dades provoquen grans variacions en la solució, es diu que el problema està mal condicionat.

Example 2.4.4

$$\begin{cases} 2x - 4y = 1 \\ -2.998x + 6.001y = 2 \end{cases} \qquad \begin{cases} 2x - 4y = 1 \\ -3x + 6.001y = 2 \end{cases}$$

2.4.2 Mètodes iteratius

Un mètode iteratiu es un algorisme que genera la successió de valors aproximats

$$x_0\,, x_1\,, x_2\,, \ldots, x_n\,, \ldots \tag{2.4.2}$$

tals que el límit de la successió és la solució α exacte del problema. No és possible calcular infinits termes de la successió x_n, cal limitar el nombre màxim d'ietracions a fer i cal establir una condició de parada de l'algorisme.

Mètode iteratiu convergent Definim l'error en l'iteració k per la fórmula $e_k = |x_k - \alpha|$.

El métode iteratiu és convergent si $\lim_{k\to\infty} e_k = 0$. La solució exacta α no es conneix, aixíque tampoc es conneix el valor de l'error e_k. Per aturar l'algorisme controlarem l'error absolut o l'error relatiu i finalment comprovarem la validesa de la solució obtinguda amb un criteri adient.

Controlant l'error absolut Fixat ϵ_a, prenem x_n com a valor aproximat α si

$$|x_n - x_{n-1}| \leq \epsilon_a$$

Controlant l'error relatiu Fixat ϵ_r, prenem x_n com a valor aproximat α si

$$\frac{|x_n - x_{n-1}|}{|x_n|} \leq \epsilon_r$$

Example 2.4.5

El mètode iteratiu següent convergeix a $\sqrt{2}$.

$$x_n = \frac{1}{2}\left(x_{n-1} + \frac{2}{x_{n-1}}\right) \qquad n \geq 1 \; i \; x_0 = 3\,.$$

n	x_n	$\lvert x_n - \sqrt{2}\rvert$
0	3.000000000000000	$1.58578643762690 \times 10^{0}$
1	1.833333333333333	$4.19119770960238 \times 10^{-1}$
2	1.462121212121212	$4.79076497481170 \times 10^{-2}$
3	1.414998429894803	$7.84867521707922 \times 10^{-4}$
4	1.414213780047198	$2.17674102520604 \times 10^{-7}$
5	1.414213562373112	$1.66533453693773 \times 10^{-14}$
6	1.414213562373095	$2.22044604925031 \times 10^{-16}$

3 Per pràcticar

3.1 Exercicis

1 Calculeu l'error absolut, l'error relatiu i l'error relatiu aproximat de les quantitats:

$$\begin{aligned} x &= 9234.567, \quad \widetilde{x} = 9234.564; \\ x &= 0.634, \qquad \widetilde{x} = 0.631\,. \end{aligned}$$

Què s'observa?

2 Calculeu l'error absolut, l'error relatiu, les xifres correctes de les quantitats:

$$\begin{aligned} x &= 1/3, \quad \widetilde{x} = 0.3333\,, \\ x &= 1/3, \quad \widetilde{x} = 0.3334\,, \end{aligned}$$

3 Calculeu les xifres significatives de les quantitats:

$$\begin{aligned} x &= 10000, \qquad \widetilde{x} = 9998\,, \\ x &= 10000, \qquad \widetilde{x} = 9999.99998\,, \\ x &= 0.0000025, \quad \widetilde{x} = 0.0000018\,, \end{aligned}$$

4 Calculeu: $\displaystyle\sum_{k=1}^{6} \frac{1}{3^k}$ i $\displaystyle\sum_{k=1}^{6} \frac{1}{3^{(7-k)}}$

a) Fent ús de l'aritmètica de tres xifres arrodonint.

b) Fent ús de l'aritmètica de quatre xifres arrodonint.

c) Per què donen diferent? Calculeu en cada cas l'error relatiu percentual.

5 Per a calcular el punt mig de dos punts a i b a la recta real, podem utilitzar les dues expressions següents:

$$0.5(a+b) \quad \text{i} \quad a+0.5(b-a)$$

Calculeu les dues quan $a = 0.982$ i $b = 0.987$, amb una aritmètica de tres xifres bo i tallant. Repetiu els càlculs ara arrodonint. Comenteu els resultats obtinguts.

Propagació de l'error

6 Calculeu $\dfrac{1}{\left(\sqrt{3}+2\right)^4}$ tenint accés al valor aproximat de 1.7321 per $\sqrt{3}$. Calculeu l'error comès si es fa el càlcul directe o avaluant l'expressió $97-56\sqrt{3}$.

7 Fent ús de la fórmula (2.2.15) deduïu les expresssions (2.2.16), (2.2.17), (2.2.18) i (2.2.19)

8 Determineu l'error màxim en el càlcul de $y = \dfrac{x_1 x_2^2}{\sqrt{x_3}}$ amb $x_1 = 2.0 \pm 0.1$, $x_2 = 3.0 \pm 0.2$ i $x_3 = 1.0 \pm 0.1$.

Quina de les dades contribueix més a l'error en y? Per què?

9 Sigui $p(x) = (x-1)(x-2)(x-3)...(x-10)$, el polinomi amb arrels els deu primers nombres naturals, definim el polinomi $q(x) = p(x) + \dfrac{1}{2^{13}}\, x^9$, modificant lleugerament el coeficient de x^9 respecte de $p(x)$. Com haurien de ser les arrels del polinomi $q(x)$? Calculeu-les.

10 Equació de segon grau

Resoldre l'equació $x^2 + 62.10x + 1 = 0$ treballant amb quatre dígits i arrodonint. Quantes xifres decimals correctes s'obtenen?

11 Regla de horner

Avaluar el polinomi $P(x) = x^3 - 6.1x^2 + 3.2x + 1.5$ per $x = 4.71$ fent ús d'una aritmètica de tres dígits. Quantes xifres significatives correctes s'obtenen?

3.1 Pràctiques i problemes

MATLAB®

1 Operacions

1. Calcula les quantitats següents:

$$r = \sqrt{1 - \frac{2}{\pi^5}}\,, \quad r = e^2 \ln 5\,, \quad r = \sin^2 2 + \cos^2 4\,.$$

2. Evalueu les següents expressions:

$$b = (5 >= 5.5)\ , \quad b = (\sin(\pi) == 0)\ , \quad b = ((7 <= 8) == (3/2 == 1))\ .$$

2 Fer una gràfica per cada una de les funcions següents:

⋆ $f(x) = x^5 e^{-x^2} - \dfrac{\sin x}{x^2+1}$ per $x \in [-2\pi, 2\pi]$.

⋆ $f(x) = \dfrac{x^2 - 4x - 7}{x^2 - x - 6}$ per $-6 \leq x \leq 6$ (presenta dues assímptotes).

⋆ $y = \cos(x^3/20)$ prenent 100 punts equiepaiats a l'interval $[-6, 6]$.

Preneu un mínim de 50 punts en l'interval; representeu les gràfiques amb títol, quadrícula.

3 Representeu gràficament la corba $y = f(x)$ definida per

$$f(x) = \begin{cases} -x^2, & x < -10\,, \\ x^3, & -10 \leq x < 2\,, \\ 0, & x \geq 2\,. \end{cases}$$

4 Scripts

1. Escriviu un script que transformi la part entera d'un nombre en base 10 a binari. Aplica-ho al nombre $x = 123.785$.

2. Escriviu un script que transformi la part decimal d'un nombre en base 10 a binari. Aplica-ho al nombre $x = 123.785$.

3. Escriviu un script que transformi un nombre binari a base 10.

4. Escriviu un script per a resoldre les equacions de segon grau $ax^2 + bx + c = 0$, on $a,\ b,\ c$ són nombres reals. Cal distingir els casos trivials i els casos $a = 0$, $b^2 - 4ac < 0$ i $b^2 - 4ac > 0$. Feu un joc de proves.

5 Funcions

1. Escriviu una funció que retorni l'error absolut i l'error relatiu de les dades x, $\tilde{x}$.
2. Escriviu una funció que retorni els decimals exactes i les xifres significatives de les dades x, $\tilde{x}$.
3. Escriviu una funció que transformi la part entera d'un nombre en base 10 a binari. Feu un joc de proves
4. Escriviu una funció que transformi la part decimal d'un nombre en base 10 a binari. Feu un joc de proves

6 Avalueu les funcions

$$f(x) = \sqrt{x^2+1} - 1\,, \quad g(x) = x^2/(\sqrt{x^2+1} + 1)$$

per a la successió de valors de $x_n = 8^{-n}\,,\ n \geq 1\,.$ Encara que $f(x) = g(x)$, l'ordinador dóna resultats diferents. Quins resultats són de fiar i quins no? Per què? Justifiqueu la vostra resposta.

Algorismes

7 Calcular el valor x_{10} del mètode iteratiu següent:

$$x_k = \frac{1}{2}\left(x_{k-1} + \frac{a}{x_{k-1}}\right) \quad k \geq 1 \ i \ x_0 = a\,.$$

Comparar el resultat obtingut amb el valor $\sqrt{a}$, quants decimals correctes s'obtenen?

8 Escriviu una **funció** d'argument m que calculi

$$\sqrt{12}\sum_{n=0}^{m}\frac{(-1/3)^n}{2n+1}$$

Calculeu el valor per $m = 5,\ m = 10$, i $m = 20$ i compareu el resultat amb π.

9 Definim el nombre e com $e = \sum_{k=0}^{\infty} \frac{1}{k!}$. Per calcular-ne una aproximació considerem el mètode iteratiu definit per

$$x_k = x_{k-1} + \frac{1}{k!}, \quad k \geq 1, \quad x_0 = 1$$

Calculeu els 20 primers termes de la recurrència, compareu els vostres resultats amb el valor $exp(1)$ retornat per MATLAB® .

10 Escriviu una `function` que calculi e^x per a tot x a partir de la sèrie de Taylor en $x = 0$ de la funció exponencial,

$$e^x = \sum_{n=0}^{\infty} \frac{x^n}{n!}.$$

Feu un joc de proves per a diferents valors de n i de x; compareu el vostre resultat amb el resultat que retorna la funció `exp` de MATLAB® . (Feu un joc de proves).

11 Per calcular les integrals $I_n = \int_0^1 x^n e^{x-1} dx$, $n \geq 1$, disposem de dos mètodes iteratius diferents:

$$\text{a)} \quad I_{n-1} = \frac{1 - I_n}{n}, \qquad n \geq 2 \qquad \text{on } I_{50} = 0,$$

$$\text{b)} \quad I_n = 1 - nI_{n-1}, \quad n \geq 2 \quad \text{on } I_1 = 1/e.$$

Discutiu la estabilitat de la recurrència.

12 Per calcular $\sin(x)$ a partir del seu desenvolupament en sèrie es considera la succesió de sumes parcials

$$S_k(x) = \sum_{n=1}^{k} \frac{(-1)^{n+1} x^{2n-1}}{(2n-1)!},$$

Feu una taula per $k = 5, 15, 25, \ldots, 85$ i $x = 0, \pi, 2\pi, 8\pi$ i calculeu $S_k(x)$ (Joc de proves).

13 La succesió

$$x_1 = 2\sqrt{2}, \quad x_2 = 2^2\sqrt{2 - \sqrt{2}}, \quad x_3 = 2^3\sqrt{2 - \sqrt{2 + \sqrt{2}}}, \quad x_4 = 2^4\sqrt{2 - \sqrt{2 + \sqrt{2 + \sqrt{2}}}}, \ \ldots$$

convergeix a π. Escriu un **script** de MATLAB® que calculi x_k i $|x_k - \pi|$ per $k = 1, 2, \ldots, 15$.