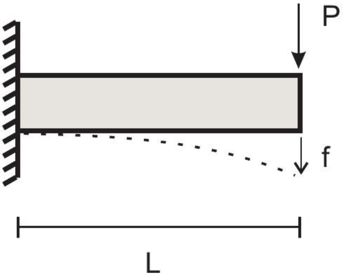
Cálculo Numérico

Métodos Numericos y Errores

Victorio E. Sonzogni

¿ Por qué cálculo numérico?

- Supóngase una tarea demandada a un ingeniero: predecir cuánto se deforma una viga



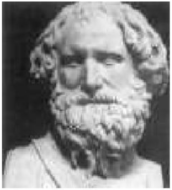
El cálculo en ingeniería

La evaluación de magnitudes que cuantifiquen el estado mecánico de estructuras, piezas industriales, recursos naturales, tejidos orgánicos, etc. puede efectuarse por algunas de las siguientes maneras:

- Experimental
- Analítica (Teórica)
- Numérica

En ese orden ha sido posible -históricamente- contar con esas herramientas.

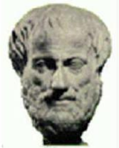
Primer gran ingeniero



Arquimedes
287-212 a.c

- principio hidrostática
- palanca
- tornillo
- π

En la antiguedad



Aristóteles
(384, 322 a.c.)

- conocimiento empírico, basado en la observación
- filosofía
- postulados de Aristóteles: $\rightarrow$ casi 2000 años
- lógica
- herramientas matemáticas: geometría

Método experimental

Galileo
(1564-1642)

- introdujo método experimental y dió paso al método científico
- resistencia de materiales, cinemática, astronomía, ...
- (limitadas posibilidades experimentales y de cálculo)
- el método experimental $\rightarrow$ permite resolver nuestro problema

Cálculo infinitesimal

Sir Isaac Newton
(1642-1727)

- matemático, físico y experimentador
- junto a (pero separadamente de) Gottfried Leibnitz (1646-1716) creador de *cálculo infinitesimal*
- importantes aportes en matemática, mecánica, óptica, ...

Solución analítica de problemas físico-mecánicos

- las magnitudes (propiedades, estado) pueden ser medidas y representadas como *variables* dependientes de otras (*funciones*)

> *Lord Kelvin (1824-1907): cuando se puede medir aquello de que se habla, y expresarlo en números, se sabe algo de ello; pero nuestro saber es deficiente e insatisfactorio mientras no podamos expresarlo en números*

- leyes o condiciones a cumplir $\rightarrow$ *ecuaciones*
- problema matemático que describe el comportamiento
  - ecuación (diferencial)
  - condiciones de contorno

Solución analítica de problemas físico-mecánicos

- importante desarrollo durante los siglos XVIII y XIX
- Euler (1707-1783); Lagrange (1736-1813); Navier (1785-1836); Gauss (1777-1855); Stokes (1819-1903); Laplace (1749-1827); Fourier (1758-1830); ...
- solución elegante
- puede obtenerse para algunos casos (sencillos)

Problemas con solución analítica

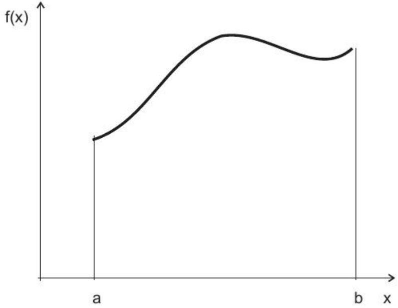
- muchos problemas pueden resolverse analíticamente
- pero ellos se limitan a geometrías sencillas, con condiciones de contorno sencillas, etc.
- en nuestro caso hay una solución analitica la problema de la viga
  $\rightarrow$ permite resolver nuestro problema
- ...
- *en la mayoría de los casos prácticos*
  no tenemos solución analítica ... !

La solución analítica será objeto de la materia *Mecánica del Contínuo*

Métodos numéricos

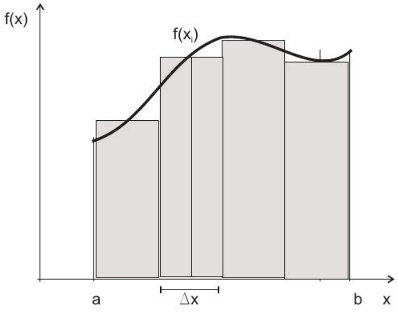
Por ejemplo: hallar una integral definida
El área bajo la curva puedo obtenerla por integración.

$$A = \int_a^b f(x)dx$$



Métodos numéricos

Si no puedo integrar esa función, puedo aproximar el área computando las áreas de estos rectángulos:



$$\hat{A} = \sum_{i=1}^{n} f(x_i)\Delta x$$

Métodos numéricos

- Grecia (antes de la era cristiana)
  Reemplazo de una curva por una poligonal. Cálculo de π
- Lord Rayleigh (1870)
  Técnicas variacionales para problemas de vibración. Reemplaza la función incógnita por una serie de funciones conocidas y coeficientes incógnitas.
- Ritz (1909)
  Extendió la idea de Rayleigh a otros tipos de problemas (electrodinámica). *Método de Rayleigh-Ritz*. Precisa un funcional.

Método de Rayleigh-Ritz

Lord Rayleigh (John W. Strutt)
(1842-1919), Nobel 1904

Walter Ritz
(1978-1909)

- aproximación

$$\hat{u}(x) = \sum_{i=1}^{n} \phi_i(x)\ a_i$$

- calcula a_i por minimización del funcional

Métodos numéricos

- Galerkin (1915)
  Minimización del error: residuos ponderados.

Boris Grigorievich Galerkin
(1871-1945)

- ... *y muchos otros* ...
- *En nuestro caso el Método Numérico* $\rightarrow$ *permite resolver nuestro problema*

Los Mét. Numéricos serán objeto de la materia *Mecanica Computacional*

El cálculo numérico

La metodología para realizar una simulación numérica implica:

- Desarrollo de un *modelo matemático* $\rightarrow$ *problema matemático*
- *Solución numérica* del *problema matemático*
- Desarrollo de software para la *solución numérica*
- Verificación de los *métodos numéricos* con casos simples
- Validación del *modelo matemático* con resultados experimentales
- Utilización práctica para predecir comportamiento.

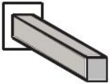
Aproximaciones

Aproximación

Aproximación

Problema físico

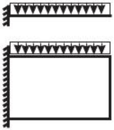
Dominio espacial
Dimensiones
Material
Vinculos
Cargas



Modelo teórico

Hipótesis:

Cinematicas	*Vigas clásicas* *Placas* *Tenisones planas, etc.*
Materiales	*Elasticidad lineal* *Isotropía* *Plasticidad, etc.*
Condiciones de borde	*Vínculos* *Desplazamientos prescriptos, etc.*
Cargas	*Fuerzas puntuales* *Momentos, etc.*



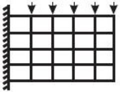
Problema matemático

Ecuaciones de campo
Condiciones de borde

Modelo numérico

Métodos de aproxim.
Discretización
Procedim. de solución
etc.

Solucion numérica del problema matemático



Errores numéricos

Errores numéricos

- Antes de continuar se pide que escriba un programita muy sencillo:
  1) guardar en una variable 1 dividido por 3
  2) restar de 1 esa variable, y repetir otras dos veces
  3) si el resultado es cero imprimir "verdadero", y sino "falso"
- Sorprendentemente el programa produce un resultado erróneo. Esto nos pone frente a los *errores numéricos*. Esto puede acarrear resultados catastróficos (al final de la clase se muestran algunos casos)
- En cálculos numéricos debemos estar preparados para manejar estos errores numéricos.

Errores numéricos

- Los errores numéricos provienen de la diferencia entre los valores calculados y los valores reales
- La palabra *error* no significa aquí *equivocación*
- En todos los cálculos con computadoras y en todos los métodos numéricos es necesario manejar el tema de *errores*

Fuentes de errores

En aproximaciones en ingeniería se introducen errores (entendidos aqui como diferencia entre el valor calculado y el valor real) de diferentes fuentes:

- Errores en la formulación del problema matemático
- Errores en la aproximación de la geometría
- Errores en los datos físicos
- Errores del método numérico de aproximación
- Errores en la resolución del sistema de ecuaciones
- Errores de redondeo

Errores numéricos

- En los métodos numéricos manejaremos dos clases de errores:
    - Errores de truncamiento
      También llamados *errores algorítmicos*.
      Son los provenientes del método numérico de aproximación. Por ejemplo: el área calculada con los rectángulos, en el ejemplo dado, no es igual al área encerrada por la curva en la integración definida.
      Se los suele llamar *errores de truncamiento*, ya que en muchos casos provienen de truncar una serie infinita.
    - Errores de redondeo
      Son los introducidos por las computadora digital que tiene un espacio finito para almacenar sus variables.
      En este caso entran tanto los denominados errores por truncamiento (hacia abajo) como por redondeo (hacia arriba). En ambos casos se engloban en el término *errores de redondeo*
- A continuación veremos errores de redondeo.
- Los errores de truncamiento serán tratados en los capítulos siguientes.

Aritmética de las computadoras

- Normalmente usamos una base **decimal** para escribir los números. Ej:
$$1563 = 1 \times 10^3 + 5 \times 10^2 + 6 \times 10^1 + 3 \times 10^0$$
- Un número *natural* se puede representar:
$$N = a_k \times 10^k + a_{k-1} \times 10^{k-1} + \dots a_1 \times 10^1 + a_0 \times 10^0$$
con $a_i \in \{0, 1, 2, \dots 9\}$ y se escribe:
$$N = a_k a_{k-1} \dots a_1 a_0 = (a_k a_{k-1} \dots a_1 a_0)_{(10)}$$

Aritmética de las computadoras

- Un número *real* se puede representar:

$$X = s\ a_k a_{k-1} \ldots a_1 a_0 . b_1 b_2 \ldots_{(10)}$$

$$X = s\ \left(\sum_{i=0}^{k} a_i 10^i + \sum_{i=1}^{\infty} b_i 10^{-i} \right)_{(10)}$$

donde $s = \begin{cases} + \\ - \end{cases}$

- Def:
  Dígitos significativos: cantidad de dígitos a_i y b_i.
  (Empezando por el primero no nulo)

- Def:
  Representación normalizada:

$$X = s\ (0.a_1a_2\dots)_{(10)} \times 10^m$$

donde

$$X = \underbrace{s}_{signo}\ \underbrace{(0.a_1a_2\dots)_{(10)}}_{mantisa} \times \underbrace{10}_{base}\underbrace{{}^m}_{exp}$$

Ej:

$$1563 = 0.1563 \times 10^4$$

$$2100000 = 0.21 \times 10^7$$

$$0.00017 = 0.17 \times 10{-}3$$

La mantisa es ≥ 0.1 y < 1

- Base binaria

$$1563 = 1 \times 2^{10} + 1 \times 2^{9} + 0 \times 2^{8} + 0 \times 2^{7} + 0 \times 2^{6} + 0 \times 2^{5}$$

$$+1 \times 2^{4} + 1 \times 2^{3} + 0 \times 2^{2} + 1 \times 2^{1} + 1 \times 2^{0}$$

Los a_i son ahora 0 o 1. La base es 2.

$$1563 = 1024 + 512 + 16 + 8 + 2 + 1$$

Como se hizo en base decimal, en base binaria se puede representar:

$$1563_{(10)} = 11000011011_{(2)}$$

Aritmética de las computadoras

- Representación binaria normalizada
  Ventajas:
  - un número $\rightarrow$ string de bits (no preciso el punto)
  - el dígito a_1 no preciso guardarlo (es siempre 1)
- Se denomina *floating point*
- Se destina una cantidad de bits para la mantisa (que determina la precisión); y otra cantidad para el exponente (que determina el rango que puede representar)

Aritmética de las computadoras

Representación estándar IEEE

- Precisión simple (32 bits)
  - 1 bit para el signo; 8 bits p/exponente; y 23 p/ mantisa
  - El rango que puede representar va de 2^{-126} a 2^{128} o sea 2×10^{-39} a 1.7×10^{38}. Por debajo o encima da *underflow* o *overflow*.
  - El error es $2^{-24} \simeq 6 \times 10^{-8}$
- Precisión doble (64 bits)
  - 1 bit para el signo; 11 bits p/exponente; y 52 p/ mantisa
  - El rango que puede representar va de 2^{-1022} a 2^{1023} o sea 10^{-308} a 10^{308} aprox.
  - El error es $2^{-53} \simeq 10^{-16}$

Errores de redondeo

- Representación binaria normalizada

$$x = s\ (0.a_1 a_2 \ldots a_k a_{k+1} \ldots)_{(2)} \times 2^m \ = \ \pm q \times 2^m$$

  donde la mantisa q está entre $\frac{1}{2} \leq q < 1$

- El *número de máquina* más cercano, por defecto:

$$x' = s\ (0.a_1 a_2 \ldots a_k)_{(2)} \times 2^m$$

  se retienen $k-1$ bits $(a_1 = 1)$ y descarta el resto
  Se obtiene un número por defecto. $\rightarrow$ *truncamiento* o *poda* o *cancelación*

- El número por exceso:

$$x'' = s\ \left((0.a_1 a_2 \ldots a_k)_{(2)} + 2^{-k}\right) \times 2^m$$

Errores de redondeo

- El número de máquina más próximo a x lo llamamos $fl(x)$
- En el *redondeo*
  - si el bit $a_{k+1} = 0 \to fl(x) = x'$
  - si el bit $a_{k+1} = 1 \to fl(x) = x''$
- El error al usar $fl(x)$ en vez de x se llama *error de redondeo* (ya sea que esté podado o redondeado).
- Error absoluto

$$|x - fl(x)|$$

- Error relativo

$$\frac{|x - fl(x)|}{|x|}$$

siempre que $|x| \neq 0$

Epsilon de máquina ϵ

- La diferencia entre las representaciones por exceso y por defecto:

$$x'' - x' \; = \; 2^{m-k}$$

- La representación en punto flotante es de ellas la más cercana a x, de donde, el error absoluto:

$$|x - fl(x)| \; \leq \frac{1}{2}(x'' - x') \; = \; \frac{1}{2}2^{m-k} \; = \; 2^{m-k-1}$$

- Error relativo

$$\frac{|x - fl(x)|}{|x|} \; \leq \; \frac{2^{m-k-1}}{q \; 2^m} \; = \; \frac{2^{-k-1}}{q} \; \leq \; \frac{2^{-k-1}}{\frac{1}{2}} \; = \; 2^{-k}$$

donde q es la mantisa cuyo valor esta entre $\frac{1}{2}$ y 1.
(recordar que $(0.1)_2 = (0.5)_{10}$)

Errores de redondeo

- Puede ponerse:

$$fl(x) = x(1+\delta)$$

  con $|\delta| \leq \epsilon$
- Donde $\epsilon = 2^{-k}$
- El ϵ de máquina varía según la computadora.
- Para computadoras con palabras de 32 bits
  - simple precisión: $\epsilon \simeq 10^{-7}$
  - doble precisión: $\epsilon \simeq 10^{-15}$
- Para computadoras con palabras de 64 bits
  - simple precisión: $\epsilon \simeq 10^{-14}$
  - doble precisión: $\epsilon \simeq 10^{-28}$
- Luego

$$\frac{|x - fl(x)|}{|x|} \leq \epsilon$$

Errores de redondeo

- Cifras significativas
  Se dice que x^* aproxima a x con t cifras (o dígitos) significativas, si t es el entero, no negativo, más grande para el cual

$$\frac{|x - fl(x)|}{|x|} \leq 0.5 \times 10^{-t}$$

Ejemplos:

- Sea la aproximación para $\pi = 3.141592$. El número 3.1416 aproxima al anterior con 5 cifras significativas, ya que $\frac{|x-fl(x)|}{|x|} \simeq 2.5 \times 10^{-6} \leq 0.5 \times 10^{-5}$
- Una medición con error relativo $1‰$ contiene 2 cifras significativas, ya que $0.001 \leq 0.5 \times 10^{-2}$

Propagación de errores

- Los errores de redondeo, por la aritmética de la máquina, se pueden propagar con las operaciones, acumulándose.

Propagación de errores

En operación suma o resta

- Sea la suma de dos número x e y, cuyas representaciones en punto flotante son $fl(x)$ y $fl(y)$, y los errores de redondeo η_x y η_y:

$$x = fl(x) + \eta_x$$

$$y = fl(y) + \eta_y$$

- El error absoluto de la suma:

$$|(x + y) - (fl(x) + fl(y))| = |\eta_x| + |\eta_y|$$

- El error absoluto de la suma (o resta), es la suma de los errores absolutos de cada sumando.

En operación producto

- Sea el producto de dos número x e y, cuyas representaciones en punto flotante son $fl(x)$ y $fl(y)$, y los errores de redondeo η_x y η_y:
- El error absoluto del producto:

$$|(xy)-(fl(x)fl(y))| = |(xy)-(x-\eta_x)(y-\eta_y)| = |x\eta_y + y\eta_x - \eta_x\eta_y|$$

- El error relativo (despreciando el término de orden superior):

$$\frac{|(xy)-(fl(x)fl(y))|}{|xy|} = \frac{\eta_x}{x} + \frac{\eta_y}{y}$$

- El error relativo del producto, es la suma de los errores relativos de cada factor.

Sustracción de números próximos

- Sean x e y números próximos:

$$fl(x) = (0.a_1 a_2 \dots a_p a_{p+1} \dots a_k)10^n$$

$$fl(y) = (0.a_1 a_2 \dots a_p b_{p+1} \dots b_k)10^n$$

- Restando:

$$fl(x) - fl(y) = (0.00 \dots 0 c_{p+1} \dots c_k)10^n$$

$$fl(x) - fl(y) = (0.c_{p+1} \dots c_k)10^{n-p}$$

  donde se designa $c_{p+1} = a_{p+1} - b_{p+1}$, etc.

- El resultado tiene a lo sumo $k - p$ cifras significativas.

Sustracción de números próximos

Ejemplo:

$$x = 0.3721478693$$
$$y = 0.3720230572$$
$$x - y = 0.0001248121$$

En una computadora de 5 dígitos:

$$fl(x) = 0.37215$$
$$fl(y) = 0.37202$$
$$fl(x - y) = 0.00013$$

El error relativo es grande ($\simeq 4\%$)
El resultado se expresa 0.13000×10^{-3}, pero los tres últimos dígitos (ceros) carecen de valor.

Evaluación de funciones

- Sea evaluar

$$f(x) = x^3 \; - \; 6.1\; x^2 \; + \; 3.2\; x \; + \; 1.5$$

en $x = 4.71$ usando aritmética de 3 dígitos

- El valor exacto:

$$f(4.71) = 104.487111 \; - \; 135.32301 \; + \; 15.072 \; + \; 1.5 \; = \; -14.263899$$

- El valor con tres dígitos (truncado):

$$f(4.71) = 104. \; - \; 135. \; + \; 15.0 \; + \; 1.5 \; = \; -13.5$$

- El error relativo:

$$\left| \frac{-14.263899 + 13.5}{-14.263899} \right| \simeq 0.05$$

Evaluación de funciones

- La misma función puede reescribirse en forma anidada:

$$f(x) = x^3 - 6.1\,x^2 + 3.2\,x + 1.5 = ((x - 6.1)\,x + 3.2)\,x + 1.5$$

- Evaluando esta expresión con tres dígitos (truncado):

$$f(4.71) = ((4.71 - 6.1)\,4.71 + 3.2)\,4.71 + 1.5 = -14.2$$

- El error relativo:

$$\left| \frac{-14.263899 + 14.2}{-14.263899} \right| \simeq 0.0045$$

diez veces menor que en el caso anterior.

Evaluación de funciones

- las funciones deberían escribirse siempre en forma anidada antes de evaluarlas numéricamente
- en este caso hemos visto como se ha disminuido el error de truncamiento
- tambien disminuye la cantidad de operaciones: en el primer caso se precisan 4 multiplicaciones y 3 adiciones; en el segundo 2 multiplicaciones y tres adiciones.

Problemas matemáticos y solución numérica

Problema matématico

- Un problema matemático puede escribirse:

$$\boxed{\begin{aligned} &\text{hallar } x \text{ tal que} \\ &F(x,d) \;=\; 0 \end{aligned}} \qquad (1)$$

donde

- F : relación funcional entre x y d;
- x : incógnita (escalar, vector, funciones, etc.)
- d : datos (escalar, vector, funciones, etc.)

Problema matématico

Ejemplos:

hallar el vector $\mathbf{x}$ tal que

$$\mathbf{A}\ \mathbf{x}\ =\ \mathbf{b}$$

donde

$$\mathbf{A} = \begin{bmatrix} 4 & 3 & 0 \\ 3 & 4 & -1 \\ 0 & -1 & 4 \end{bmatrix} \quad \text{y} \quad \mathbf{b} = \begin{bmatrix} 24 \\ 30 \\ -24 \end{bmatrix}$$

Este es un problema de resolver un sistema de ecuaciones algebraicas lineales.

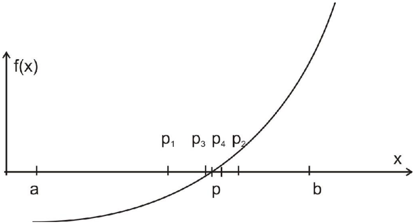
Será tratado en otro capítulo.

Problema matématico

Ejemplos:

hallar x tal que

$$f(x) = 0$$



Este es un problema de hallar el cero o raiz de una función.
Será tratado en otro capítulo.

Problemas bien planteados

- El problema (1) se dice **bien planteado** si
  - *tiene* una solución
  - la solución es *única*
  - la solución *depende con continuidad de los datos*.
- Si no, el problema se dice que es **mal planteado**

Problema mal planteado

Ejemplo:

- El problema de hallar la raiz de

$$p(x) \; = \; x^4 - x^2(2a-1) + a(a-1)$$

  es mal planteado.
- En efecto:
  - Si $a \geq 1$ tiene 4 raices reales
  - Si $a \in [0,1)$ tiene 2 raices reales
  - Si $a < 0$ no tiene raices reales
- La solución varía en forma discontinua con el dato a.

Condicionamiento

- Si con δd se indica una variación en los datos de entrada, y con δx la variación acorde de la solución, se puede definir un *numero de condición*:

$$\kappa \ = \ \sup_{\delta d} \frac{\|\delta x\|/\|x\|}{\|\delta d\|/\|d\|}$$

  donde $\|.\|$ es una *norma* (medida escalar).
- Si $\kappa >> 1$ el problema está *mal condicionado*.
- Si $\kappa \sim 1$ el problema está *bien condicionado*.
- Ejemplo:

  Para el problema de resolver un sistema de ecuaciones $\mathbf{Ax} = \mathbf{b}$ *se puede definir un número de condición para la matriz:* $\kappa = \|\mathbf{A}\|\|\mathbf{A}^{-1}\|$

Condicionamiento

Ejemplo:

- Se desea hallar un polinomio cúbico

$$y = c_1 \; x^3 + c_2 \; x^2 + c_3 \; x + c_4$$

  que pase por los 4 puntos: (2,8), (3,27), (4,64). (5,125). (evidentemente este polinomio es $y = x^3$).

- La técnica para hallarlo se vera más adelante, pero conduce a resolver el SEAL (Sistema de Ecuaciones Algebraicas Lineales):

$$\begin{bmatrix} 20514 & 4424 & 978 & 224 \\ 4424 & 978 & 224 & 54 \\ 978 & 224 & 54 & 14 \\ 224 & 54 & 14 & 4 \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{bmatrix} = \begin{bmatrix} 20514 \\ 4424 \\ 978 \\ 224 \end{bmatrix}$$

Condicionamiento

Ejemplo: (cont.)

- La solución exacta del sistema de ecuaciones es: $[1 \quad 0 \quad 0 \quad 0]$.
- Una computadora con 9 cifras da:
  $[1.000004 \quad -0.000038 \quad 0.000126 \quad -0.000131]$
  que es cercano a la solución.
- Pero si el numero a_{11} de la matriz, en lugar de ser 20514 fuese 20515, la misma computadora da: $[0.642857 \quad 3.75000 \quad -12.3928 \quad 12.7500]$
- El resultado es muy sensible a los datos, y no es confiable. Esto se dió porque el número de condición es muy alto.
- El número de condición de la matriz en este caso es $\kappa = 3.1875e + 07$

Solución numérica

- Sea el problema (1) bien planteado

$$\boxed{\begin{array}{l}\text{hallar } x \text{ tal que} \\ F(x,d) \;=\; 0\end{array}} \qquad (1)$$

- Hay métodos numéricos que se basan en construir una secuencia de problemas aproximados:

$$\boxed{\begin{array}{l}\text{hallar } x^{(k)} \text{ tal que} \\ F^{(k)}(x^{(k)},d^{(k)}) \;=\; 0 \qquad k \geq 1\end{array}} \qquad (2)$$

  con la expectativa que $x^{(k)} \to x^*$ para $k \to \infty$, donde x^* es la solución exacta de (1).
- Para que se dé esto debe ser $d^{(k)} \to d$ y $F^{(k)} \to F$ cuando $k \to \infty$.

Solución numérica

Ejemplo:
El metodo de Newton-Raphson para hallar cero de una función, resuelve una sucesión de problemas aproximados del tipo:

$$f^{(k)}(x^{(k)}) \;=\; x^{(k)} - x^{(k-1)} - \frac{f(x^{(k-1)})}{f'(x^{(k-1)})} \;=\; 0$$

Consistencia

- El problema aproximado (2) se dice **consistente** si

$$F^{(k)}(x^*, d) \rightarrow F(x^*, d)$$

para $k \rightarrow \infty$

Estabilidad

- Un método numérico se dice que es **estable**, si para cada iteración k existe una solución unica $x^{(k)}$ para los datos $d^{(k)}$; y si esa solución depende continuamente de los datos.
- Es decir que para pequeños cambios $\delta d^{(k)}$ en los datos se producen pequeños cambios en los resultados $\delta x^{(k)}$
- Este es un concepto análogo al de problema *bien planteado*
- Puede evaluarse un *número de condición* para el método numérico
- Los conceptos de *bien planteado*, *bien condicionado*, y *estable*, se usan como sinónimos.

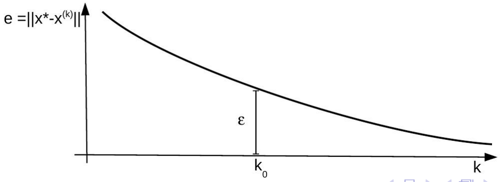
Convergencia

- El metodo numérico (2) se dice **convergente** si

$$\forall \epsilon > 0\ \exists\ k_0(\epsilon),\ \ \exists\ \delta(k_0, \epsilon) \text{ tal que}$$

$$\forall k > k_0(\epsilon),\ \forall \|\delta d^{(k)}\| < \delta(k_0, \epsilon) \quad \Rightarrow \|x(d) - x^{(k)}(d + \delta d^{(k)})\| \leq \epsilon$$

- Si un método numérico es *consistente* y *estable*, entonces es *convergente*



Orden de convergencia

- En procedimientos iterativos se construye una sucesión $[x^{(k)}]$ que se espera tienda a la solución x^*.
- Para referirse a la rapidez con que $[x^{(k)}]$ tiende a x^*. se habla de *tasa*, o *razón*, o *velocidad* de convergencia.
- Se dice que la convergencia es **lineal** si:

$$\exists\ c < 1 \text{ y } K \in \mathbb{Z} \ \text{ tal que}$$

$$|x^{(k+1)} - x^*| \ \leq \ c\,|x^{(k)} - x^*| \qquad \text{para} \ \ k \geq K$$

- Se dice que la convergencia es **superlineal** si:

$$\exists\ [\epsilon_k] \to 0 \text{ y } K \in \mathbb{Z} \ \text{ tal que}$$

$$|x^{(k+1)} - x^*| \ \leq \ \epsilon^{(k)}\ |x^{(k)} - x^*| \qquad \text{para} \ \ k \geq K$$

Orden de convergencia

- Se dice que la convergencia es **cuadrática** si:

$$\exists\ C(\text{no necesariamente} < 1)\ \text{y}\ K \in \mathbb{Z}\ \ \text{tal que}$$

$$|x^{(k+1)} - x^*| \ \leq\ C\ |x^{(k)} - x^*|^2 \qquad \text{para}\ \ k \geq K$$

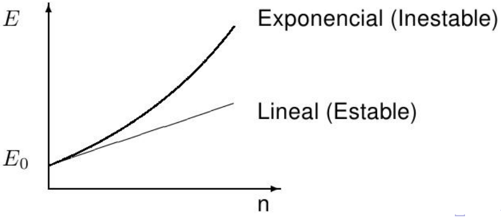
- Se dice que la convergencia es de **orden** α si:

$$\exists\ C\ \text{y}\ \alpha\ \text{constantes y}\ K \in \mathbb{Z}\ \ \text{tal que}$$

$$|x^{(k+1)} - x^*| \ \leq\ C\ |x^{(k)} - x^*|^\alpha \qquad \text{para}\ \ k \geq K$$

Estabilidad

- Un método numérico se dice estable si pequeños cambios en los datos de entrada producen pequeños cambios en los resultados.
- En algoritmos donde se evalúa una solución en varios instantes en el tiempo (historia), hay una acumulación de errores de cada paso. Si se introduce un error o perturbación E_0 en alguna etapa del cálculo, y se designa con E_n el error luego de n pasos (iteraciones), se dice que:
  - El error crece *linealmente* si $E_n = n\,C\,E_0$
  - El error crece *exponencialmente* si $E_n = C^n\,E_0 \quad , \quad C > 1$



Algoritmos

- Los métodos numéricos se describen a través de *algoritmos*
- Un algoritmo es un procedimiento que describe una serie finita de pasos, en un orden determinado, que hay que realizar para resolver un problema dado.
- El algoritmo se puede describir con un seudocódigo. Este especifica la forma de entrada de datos; de salida de resultados; y los pasos a realizar.

Algunos desastres a causa de errores numéricos

Algunos desastres

o inesperados resultados debido a errores numéricos

- Falla del misil Patriot
- Explosión del Ariane 5
- Hundimiento de la plataforma Sleipner A
- Elecciones parlamentarias alemanas
- Bolsa de valores de Vancouver

Falla del misil Patriot

Falla del misil Patriot

- El 25/2/1991, durante la Guerra del Golfo, un misil Patriot no pudo interceptar a un misil Scud iraquí que alcanzó su objetivo produciendo 28 muertes.
- Para seguir su objetivo, el sistema debía determinar el intervalo de tiempo, restando dos valores de tiempo medidos.
- Los tiempos, en 1/10 de segundos estaban en registros de enteros
- Para calcular el incremento de tiempo, los valores del registro (entero) eran convertidos a valores de punto flotante multiplicándolos por 0.1

Falla del misil Patriot

- Pero 0.1 en expansión binaria no es representado exáctamente con un numero finito de dígitos.

$$0.1_{(10)} = 0.0001100110011..._{(2)}$$

  Hay un error de redondeo (truncamiento). Los registros tenían 24 bits. El error introducido entre la representación (en 24 bits) y el número 0.1 es 0.95E-07

- Después de 100 hs el error acumulado es:

$$0.95E\text{-}07 \times 100 \times 3600 \times 10 = 0.34 seg$$

- La velocidad del misil Scud es aprox. 1676 m/s. En 0.34 seg. recorre mas de $\frac{1}{2}$ km. Quedó fuera del alcance del Patriot.

Explosión del Ariane 5

Explosión del Ariane 5

- El 4/6/1996 el cohete Ariane 5 , de la Agencia Espacial Europea fue lanzado desde la base de Kourou. Durante 36 segundos voló normalmente. Al segundo 37 salió de su curso y se autodestruyó.

  explosion

- Era el primer viaje del Ariane 5 , luego de una década de desarrollo, a un costo de 7 mil millones de U$S. El cohete y su carga estaban valuados en 500 millones de U$S.
- El problema estuvo en un error de software en el Sistema de Referencia Inercial (SRI)

Explosión del Ariane 5

- Un número de punto flotante de 64 bits, que relacionaba la velocidad horizontal con respecto a la plataforma, fue tratado de convertir a un entero de 16 bits.
- Este número llegó a ser mayor que 32768: no entraba en 16 bits! El sistema devolvió un mensaje de error, que fue interpretado como un dato....
- La ironía es que el programa que produjo la falla había sido heredado del Ariane 4, y que no se precisaba en el Ariane 5 !

Hundimiento de la plataforma Sleipner A

- La plataforma marina para extracción de petróleo Sleipner A está apoyada en una base de hormigón consistente en 24 celdas. Cuatro celdas se prolongan hacia arriba y sotienen la plataforma. Funciona en el Mar del Norte.
- La primera plataforma Sleipner A tuvo filtraciones de agua por fisuraciones y se hundió en Noruega el 23/8/1991. La pérdida económica resultante fue del orden de $700 millones.

Hundimiento de la plataforma Sleipner A

Hundimiento de la plataforma Sleipner A

- Las investigaciones posteriores atribuyen el accidente a errores en la aproximación numérica (por el método de elementos finitos), que subestimaron las tensiones tangenciales en un 47%.
- Eso condujo a que los espesores de las paredes de hormigón fuesen insuficientes. Lo mismo que las armaduras de refuerzo.

Elecciones parlamentarias alemanas

- En el sistema electoral alemán un partido con menos de 5% de los votos no puede tener representantes en el parlamento.
- En las elecciones de Schleswig-Holstein, los resultados impresos mostraban que el Partido Verde consiguió el 5.0% de los votos.
- Se dice que el Partido Social Demócrata consiguió un representante extra y obtuvo una mayoría en el Parlamento.
- Más tarde se comprobó que el porcentaje de votos del Partido Verde había sido 4.97% : la impresión se había hecho redondeando a dos cifras significativas lo cual arrojó 5.0%.

Bolsa de valores de Vancouver

- En 1982 la Bolsa de Valores de Vancouver (Canadá) introdujo un índice con un valor nominal de 1000.000
- Luego de cada transacción se recalculaba el índice, truncándose al tercer decimal después de la coma.
- Después de 22 meses el índice fue 524.881
- El valor real debía haber sido 1098.811

Resumen

En este capítulo hemos visto:

- Una motivación para el uso de métodos numéricos.
- La existencia de errores numericos.
- Cómo se producen los *errores de redondeo*, al trabajar en máquinas de aritmética finita.

  En otros capítulos veremos otro tipo de errores: los errores de truncamiento *o* algorítmicos*.*

- Hemos visto cómo estos errores se propagan.
- Finalmente hemos definido problemas *bien y mal planteados* y los métodos numéricos para resolverlos, introduciendo los conceptos de *consistencia, estabilidad y convergencia* y los órdenes de convergencia.
- Hemos terminado mostrando algunos casos en que la no observación de estos errores (que hubiese sido muy sencillo evitar) ha derivado en importantes daños materiales y pérdida de vidas.