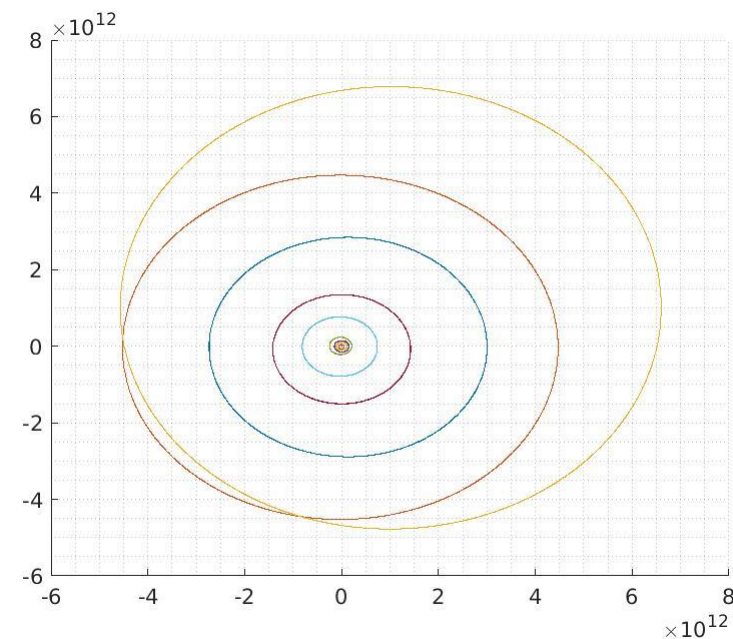


Algoritmos Numéricos por Computadora

COM - 14105



Temario

1. Introducción

1. Modelado de sistemas dinámicos
2. Truncamiento y redondeo
3. Raíces de funciones y optimización

2. Sistemas lineales

1. Valores y vectores propios
2. Eliminación de Gauss
3. Factorizaciones
4. Métodos iterativos
5. Sistemas no lineales

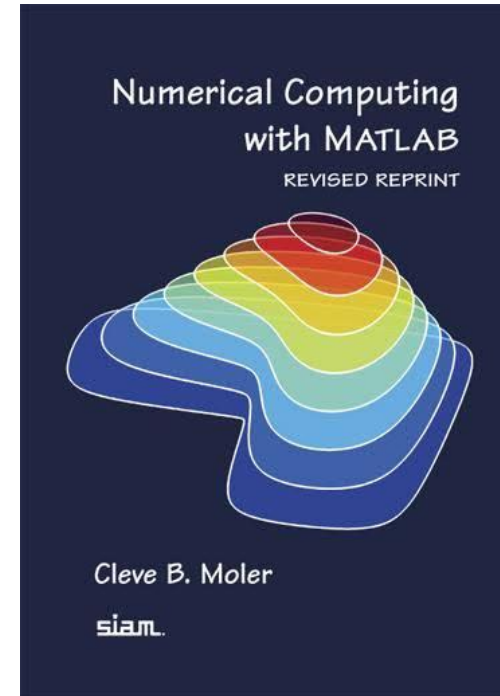
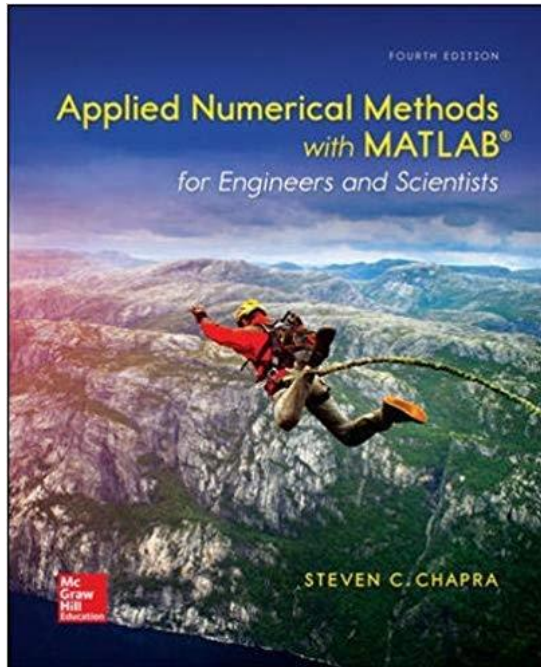
3. Ecuaciones diferenciales ordinarias

1. Interpolación e integración
2. Soluciones analíticas sencillas
3. Problemas con valor inicial
4. Sistemas de ecuaciones lineales de primer orden
5. ODE de orden superior
6. Métodos de paso variable, multipasos e implícitos
7. Problemas con valores en la frontera

4. Ecuaciones diferenciales parciales (lineales de segundo orden)*

Bibliografía

Steven Chapra, *Applied Numerical Methods with MATLAB for Engineers and Scientists*, McGraw- Hill, Fourth edition, 2018.



Cleve Moler, *Numerical Computing with MATLAB*, SIAM, 2008.

Actually, a person does not really understand something until he can teach it to a computer.

Donald Knuth

Simplicity is a great virtue, but it requires hard work to achieve it and education to appreciate it. And to make matters worse: complexity sells better.

Edsger Wybe Dijkstra

Temario

1. Introducción

1. Modelado de sistemas dinámicos
2. Truncamiento y redondeo
3. Raíces de funciones y optimización

2. Sistemas lineales

1. Valores y vectores propios
2. Eliminación de Gauss
3. Factorizaciones
4. Métodos iterativos
5. Sistemas no lineales

3. Ecuaciones diferenciales ordinarias

1. Interpolación e integración
2. Soluciones analíticas sencillas
3. Problemas con valor inicial
4. Sistemas de ecuaciones lineales de primer orden
5. ODE de orden superior
6. Métodos de paso variable, multipasos e implícitos
7. Problemas con valores en la frontera

4. Ecuaciones diferenciales parciales (lineales de segundo orden)*

Introducción

Modelado de sistemas dinámicos

1. Salto en bungee - caída libre



Introducción

Modelado de sistemas dinámicos

2. Un satélite orbitando la Tierra



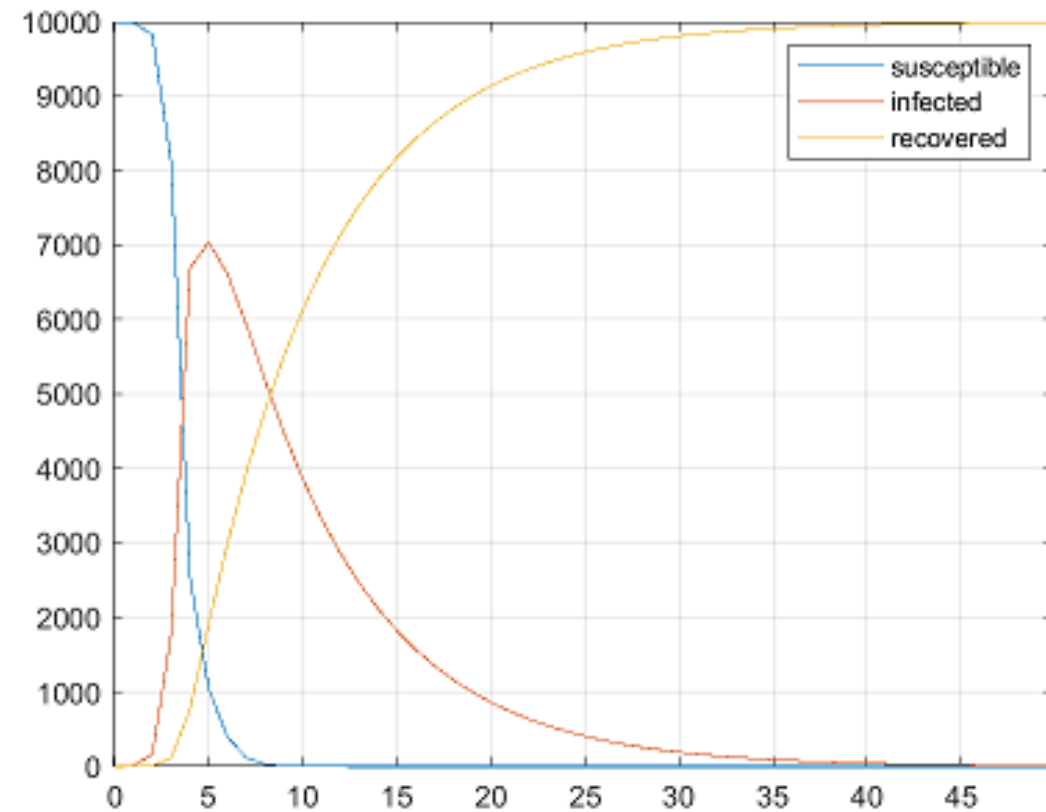
<https://www.gob.mx/telecomm/acciones-y-programas/mexsat>



Introducción

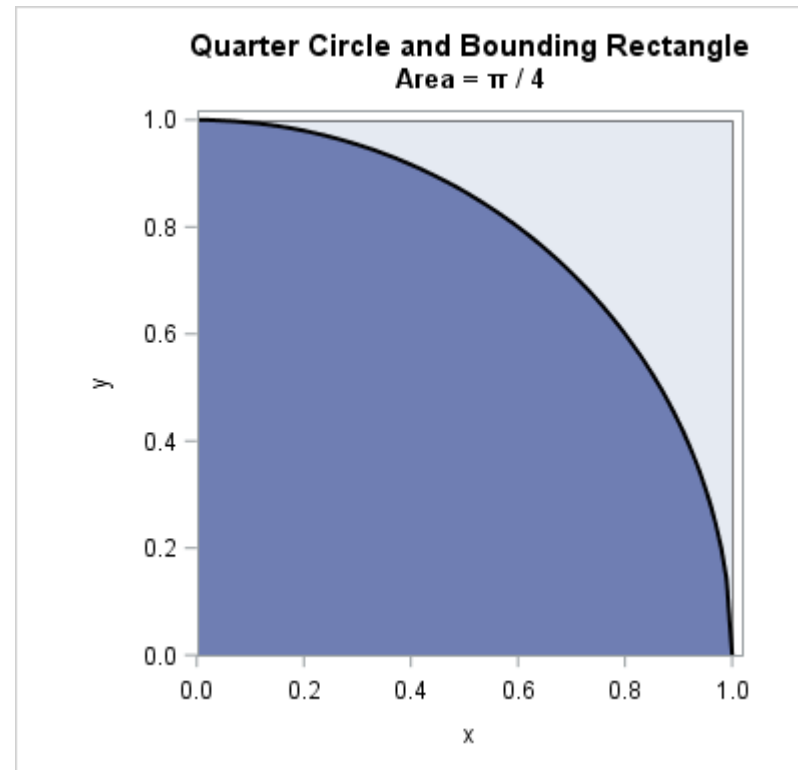
Modelado de sistemas dinámicos

3. Modelo SIR de epidemias



Introducción

Montecarlo



<https://blogs.sas.com/content/iml/2016/03/14/monte-carlo-estimates-of-pi.html>

Temario

1. Introducción

1. Modelado de sistemas dinámicos
2. Truncamiento y redondeo
3. Raíces de funciones y optimización

2. Sistemas lineales

1. Valores y vectores propios
2. Eliminación de Gauss
3. Factorizaciones
4. Métodos iterativos
5. Sistemas no lineales

3. Ecuaciones diferenciales ordinarias

1. Interpolación e integración
2. Soluciones analíticas sencillas
3. Problemas con valor inicial
4. Sistemas de ecuaciones lineales de primer orden
5. ODE de orden superior
6. Métodos de paso variable, multipasos e implícitos
7. Problemas con valores en la frontera

4. Ecuaciones diferenciales parciales (lineales de segundo orden)*

Introducción

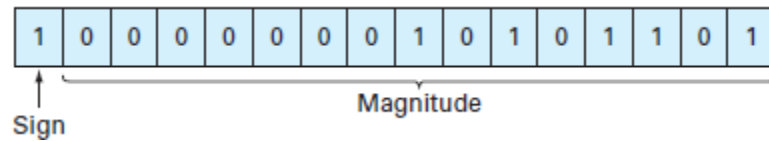
Redondeo

Double-precision floating-point numbers

ANSI/IEEE Standard 754

Introducción

¿-173 en binario?



¿Números decimales?

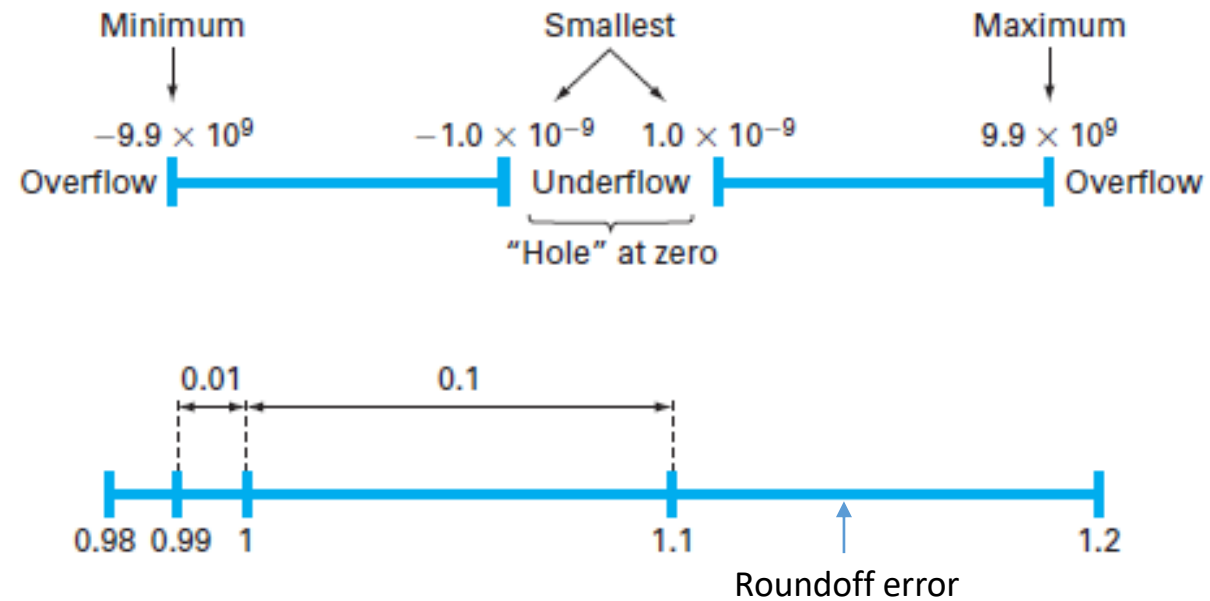
Punto flotante

Normalización

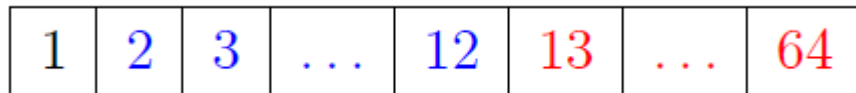
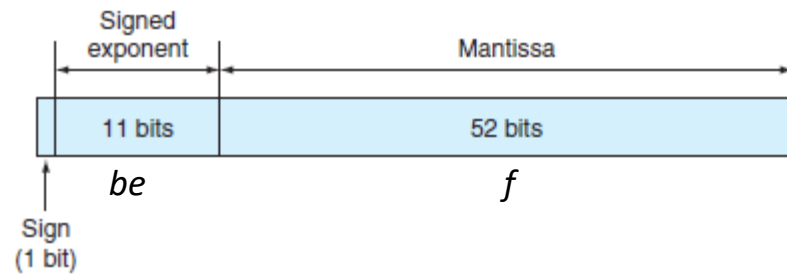
Ejemplo

Base-10 floating-point computer with 5-digit word size

$$s_1 d_1 d_2 \times 10^{s_0 d_0}$$



IEEE double-precision format



$$x = \pm(1 + f) \cdot 2^e$$

$$0 \leq f < 1$$

$$f = (\text{integer} < 2^{52}) / 2^{52}$$

$$-1022 \leq e \leq 1023$$

$$e = be - 1023$$

Finite f implies finite *precision*.

Finite e implies finite *range*

Floating point numbers have discrete spacing,
a maximum and a minimum.

Name	Binary	Decimal
eps	2^{-52}	2.2204e-16
realmin	2^{-1022}	2.2251e-308
realmax	$(2-\text{eps}) \cdot 2^{1023}$	1.7977e+308

Temario

1. Introducción

1. Modelado de sistemas dinámicos
2. Redondeo y truncamiento
3. Raíces de funciones y optimización

2. Sistemas lineales

1. Valores y vectores propios
2. Eliminación de Gauss
3. Factorizaciones
4. Métodos iterativos
5. Sistemas no lineales

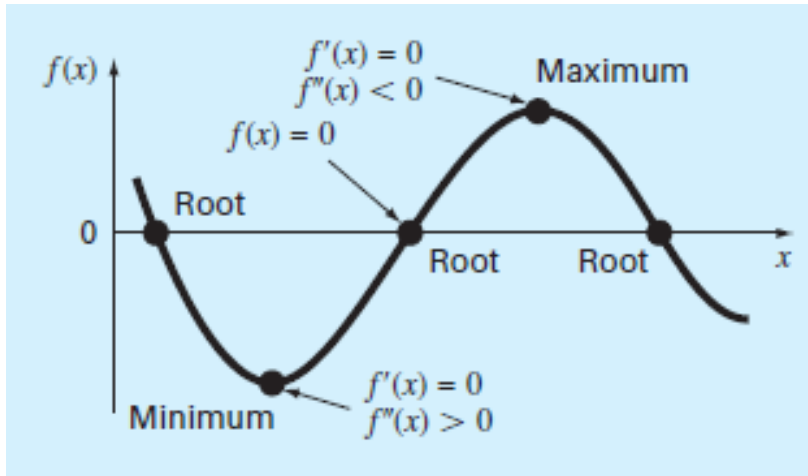
3. Ecuaciones diferenciales ordinarias

1. Interpolación e integración
2. Soluciones analíticas sencillas
3. Problemas con valor inicial
4. Sistemas de ecuaciones lineales de primer orden
5. ODE de orden superior
6. Métodos de paso variable, multipasos e implícitos
7. Problemas con valores en la frontera

4. Ecuaciones diferenciales parciales (lineales de segundo orden)*

Introducción

Raíces y optimización

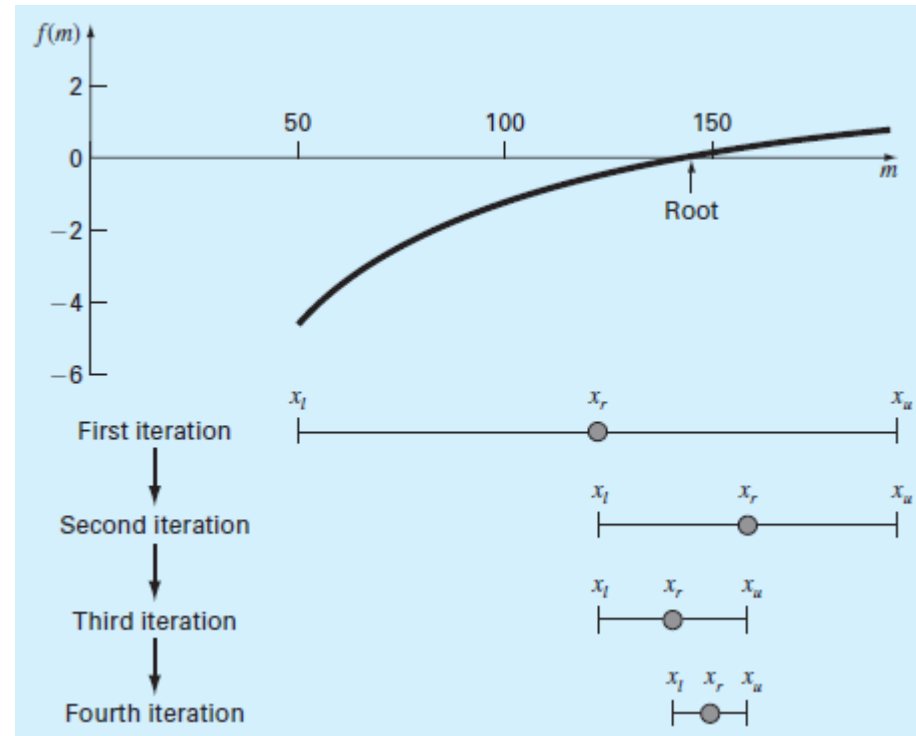


Raíces:

Métodos cerrados: bisección, interpolación lineal

Métodos abiertos: Newton-Raphson, secante

Bisección



Convergencia lineal

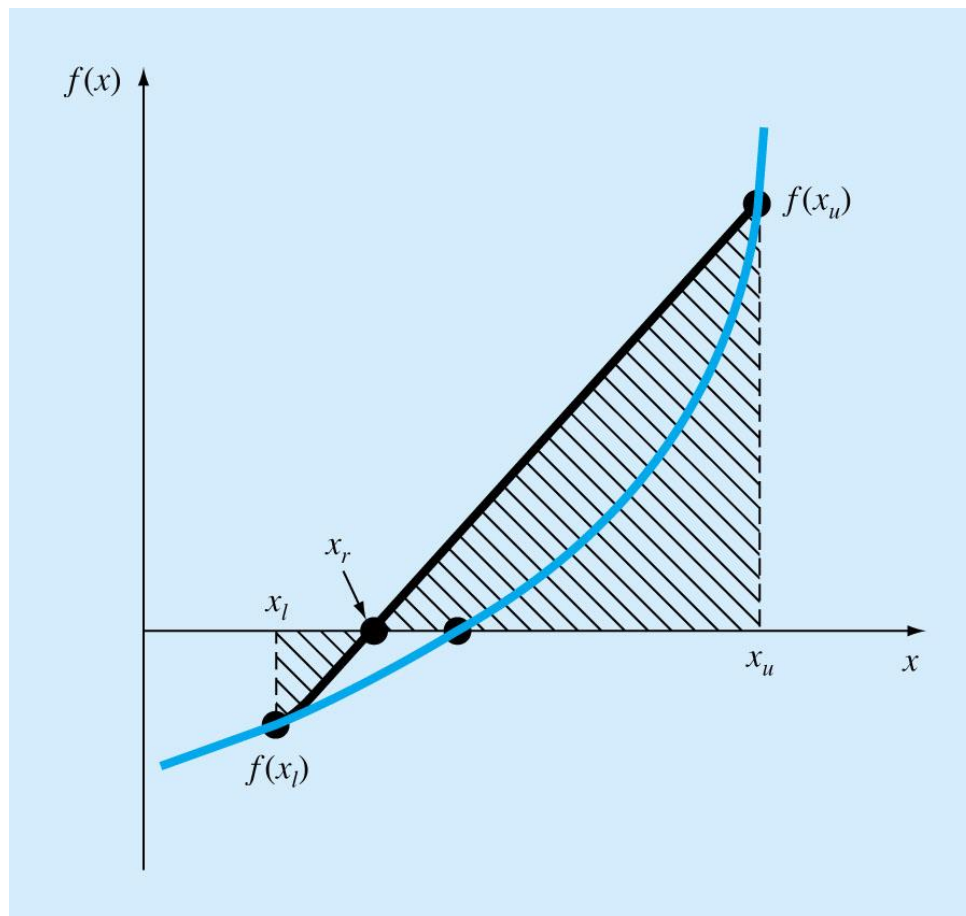
Rapidez de convergencia

La sucesión p_n converge a p con orden α y una constante de error asintótica λ si

$$\lim_{n \rightarrow \infty} \frac{|p_{n+1} - p|}{|p_n - p|^\alpha} = \lambda.$$

Interpolación lineal

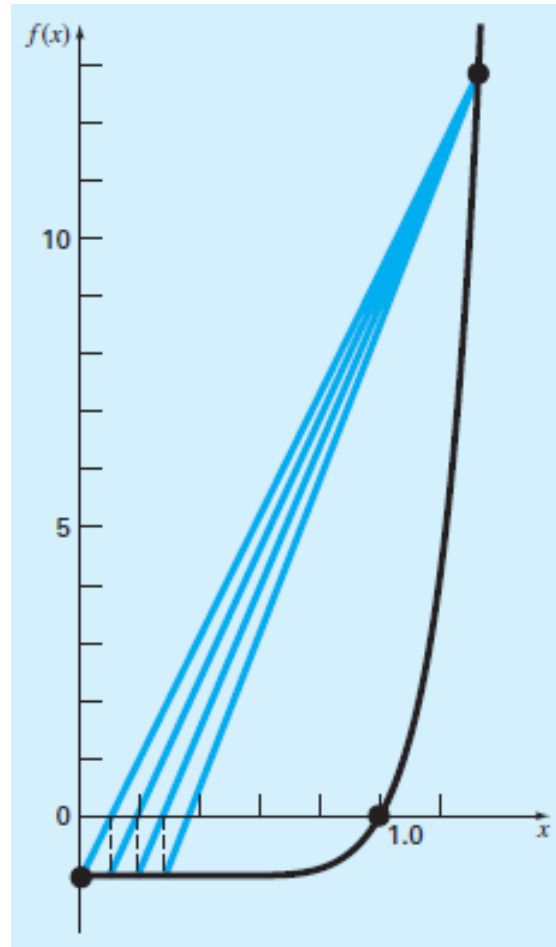
(falsa posición)



$$x_r = x_u - \frac{f(x_u)(x_l - x_u)}{f(x_l) - f(x_u)}$$

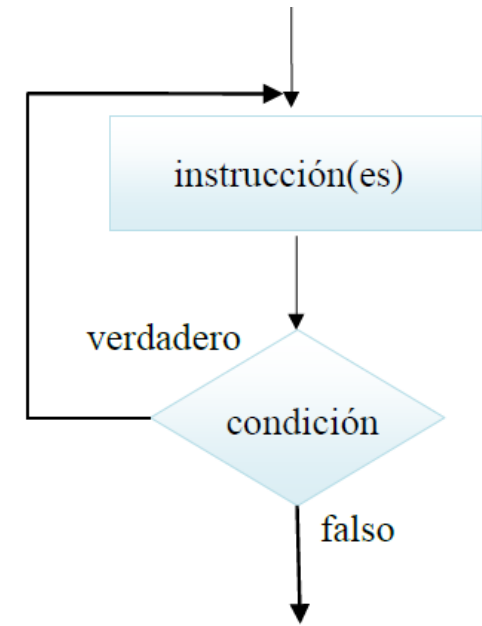
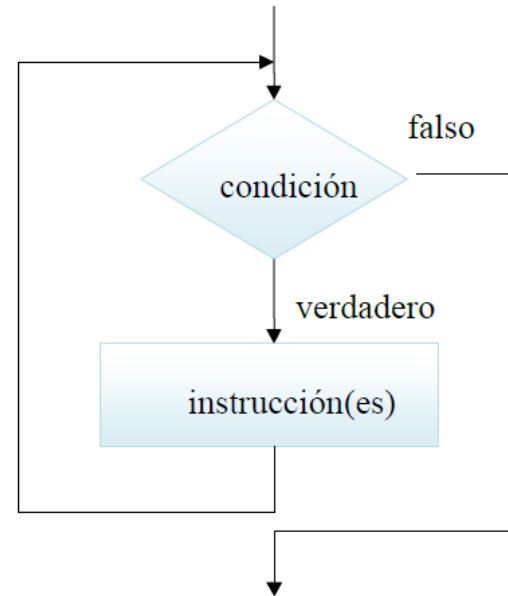
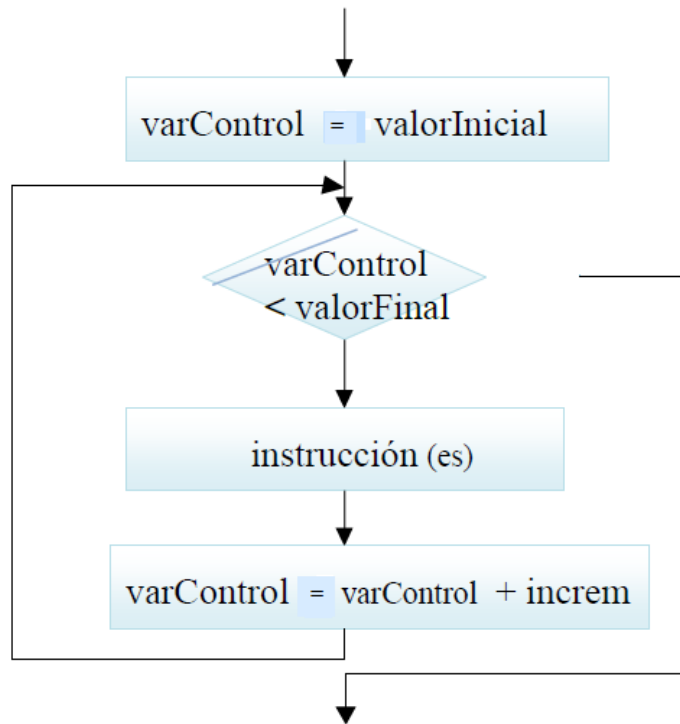
Interpolación lineal

(falsa posición)

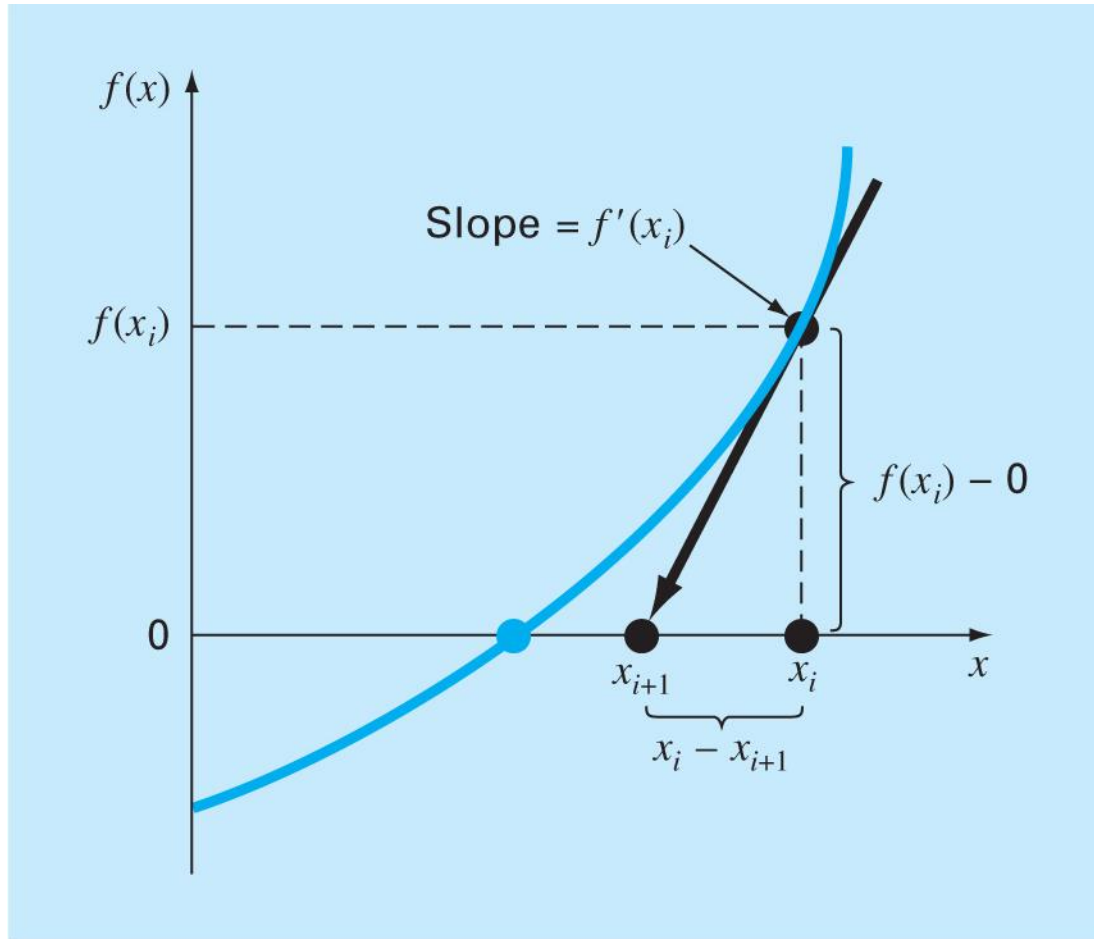


$$f(x) = x^{10} - 1$$

Estructuras algorítmicas de control iterativas



Newton-Raphson



$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

Newton-Raphson (raíz cuadrada de M)

$$f(x) = x^2 - M$$

$$\begin{aligned}x_{n+1} &= x_n - \frac{x_n^2 - M}{2x_n} \\ &= \frac{1}{2} \left(x_n + \frac{M}{x_n} \right)\end{aligned}$$

Divide y promedia

Newton-Raphson (convergencia cuadrática)

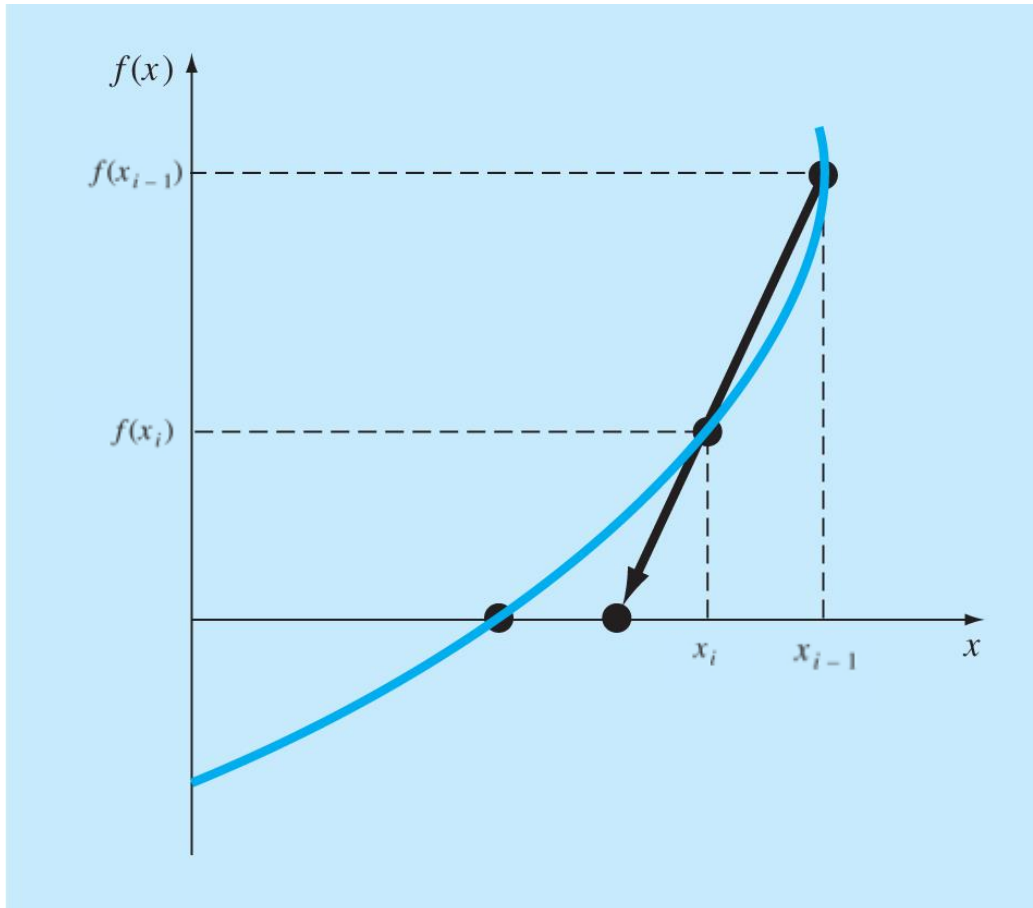
$$f(x_r) = f(x_n) + f'(x_n)(x_r - x_n) + \frac{f''(\xi)}{2} (x_r - x_n)^2 = 0$$

$$e_{n+1} = -\frac{1}{2} \frac{f''(\xi)}{f'(x_n)} e_n^2$$

$$e_{n+1} = O(e_n^2)$$

Secante

La derivada se aproxima con una diferencia finita hacia atrás:



$$s_n = \frac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}}$$
$$x_{n+1} = x_n - \frac{f(x_n)}{s_n}$$

Problema

6.22 You are designing a spherical tank (Fig. P6.22) to hold water for a small village in a developing country. The volume of liquid it can hold can be computed as

$$V = \pi h^2 \frac{3R - h}{3}$$

where V = volume [m^3], h = depth of water in tank [m], and R = the tank radius [m].

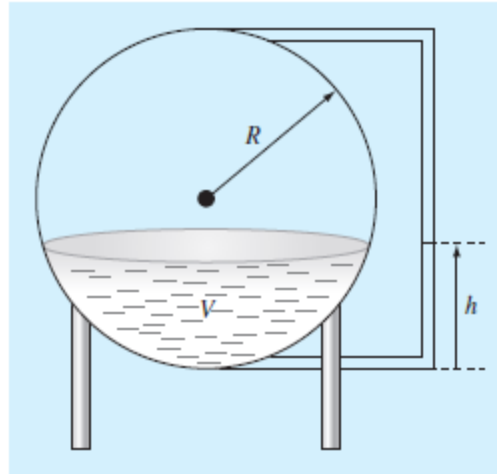


FIGURE P6.22

If $R = 3$ m, what depth must the tank be filled to so that it holds 30 m^3 ? Use three iterations of the most efficient numerical method possible to determine your answer. Determine the approximate relative error after each iteration. Also, provide justification for your choice of method. Extra information: **(a)** For bracketing methods, initial guesses of 0 and R will bracket a single root for this example. **(b)** For open methods, an initial guess of R will always converge.

Diferenciación numérica

First Derivative

$$f'(x_i) = \frac{f(x_{i+1}) - f(x_{i-1}))}{2h}$$

Error

$$O(h^2)$$

$$f'(x_i) = \frac{-f(x_{i+2}) + 8f(x_{i+1}) - 8f(x_{i-1}) + f(x_{i-2}))}{12h}$$

$$O(h^4)$$

Second Derivative

$$f''(x_i) = \frac{f(x_{i+1}) - 2f(x_i) + f(x_{i-1}))}{h^2}$$

$$O(h^2)$$

$$f''(x_i) = \frac{-f(x_{i+2}) + 16f(x_{i+1}) - 30f(x_i) + 16f(x_{i-1}) - f(x_{i-2}))}{12h^2}$$

$$O(h^4)$$

Third Derivative

$$f'''(x_i) = \frac{f(x_{i+2}) - 2f(x_{i+1}) + 2f(x_{i-1}) - f(x_{i-2}))}{2h^3}$$

$$O(h^2)$$

$$f'''(x_i) = \frac{-f(x_{i+3}) + 8f(x_{i+2}) - 13f(x_{i+1}) + 13f(x_{i-1}) - 8f(x_{i-2}) + f(x_{i-3}))}{8h^3}$$

$$O(h^4)$$

Fourth Derivative

$$f^{(4)}(x_i) = \frac{f(x_{i+2}) - 4f(x_{i+1}) + 6f(x_i) - 4f(x_{i-1}) + f(x_{i-2}))}{h^4}$$

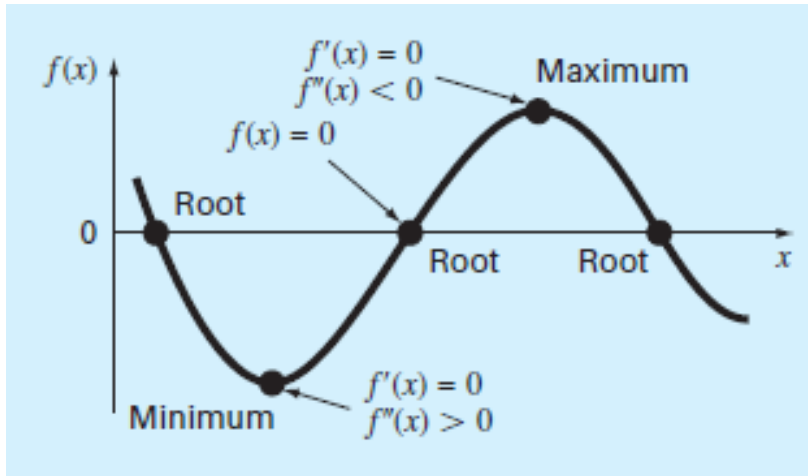
$$O(h^2)$$

$$f^{(4)}(x_i) = \frac{-f(x_{i+3}) + 12f(x_{i+2}) - 39f(x_{i+1}) + 56f(x_i) - 39f(x_{i-1}) + 12f(x_{i-2}) - f(x_{i-3}))}{6h^4}$$

$$O(h^4)$$

Introducción

Raíces y optimización

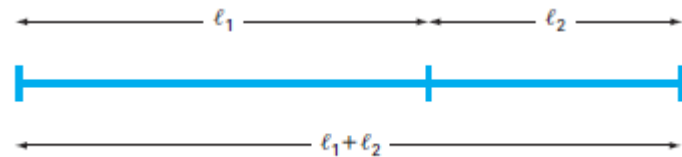


Optimización:

Proporción áurea (golden ratio)

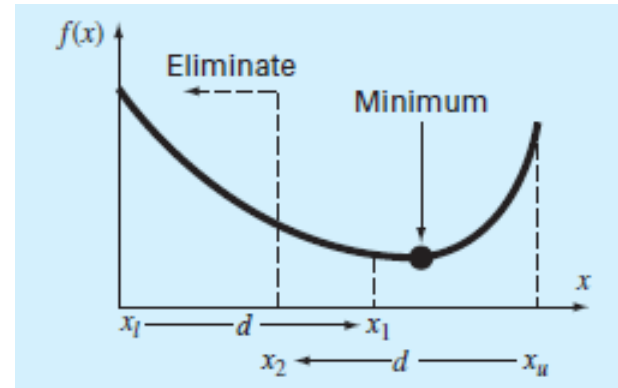
Interpolación parabólica

Proporción áurea



$$\frac{l_1 + l_2}{l_1} = \frac{l_1}{l_2} = \varphi$$

$$l_1 = (\varphi - 1)(l_1 + l_2)$$

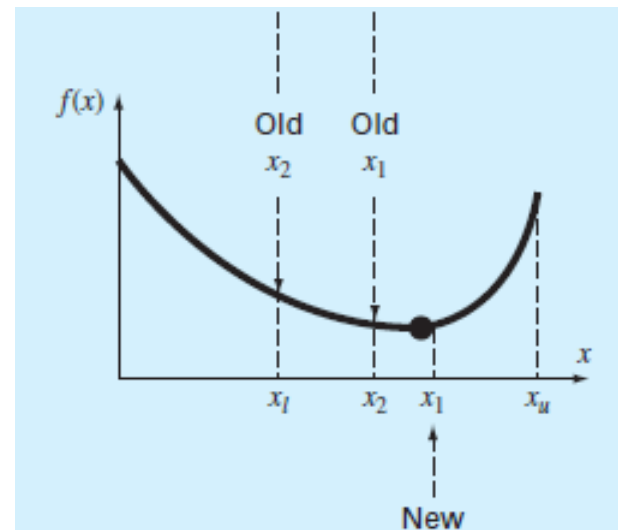


$$d = (\varphi - 1)(x_u - x_l)$$

$$d = g(x_u - x_l)$$

$$x_1 = x_l + d$$

$$x_2 = x_u - d$$

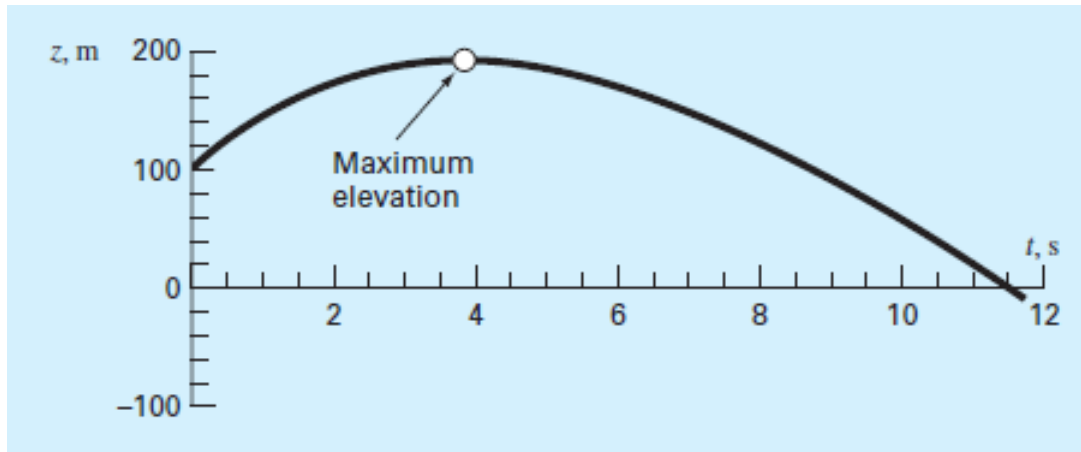


$$d_{i+1} = g d_i$$

$$x_l < x_2 < x_1 < x_u$$

Problema

Elevación de un objeto proyectado inicialmente hacia arriba con una velocidad inicial (resistencia lineal)



$$g = 9.81 \text{ m/s}^2$$

$$z_0 = 100 \text{ m}$$

$$v_0 = 55 \text{ m/s}$$

$$m = 80 \text{ kg}$$

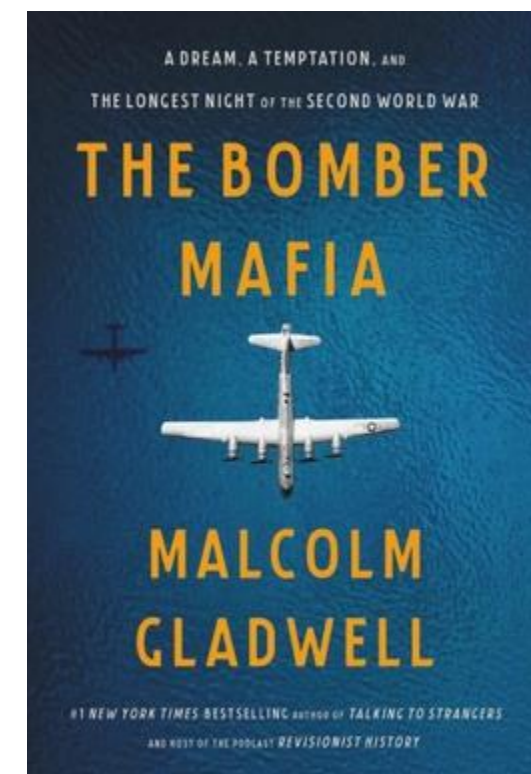
$$c = 15 \text{ kg/s}$$

$$z = z_0 + \frac{m}{c} \left(v_0 + \frac{mg}{c} \right) (1 - e^{-(c/m)t}) - \frac{mg}{c} t$$

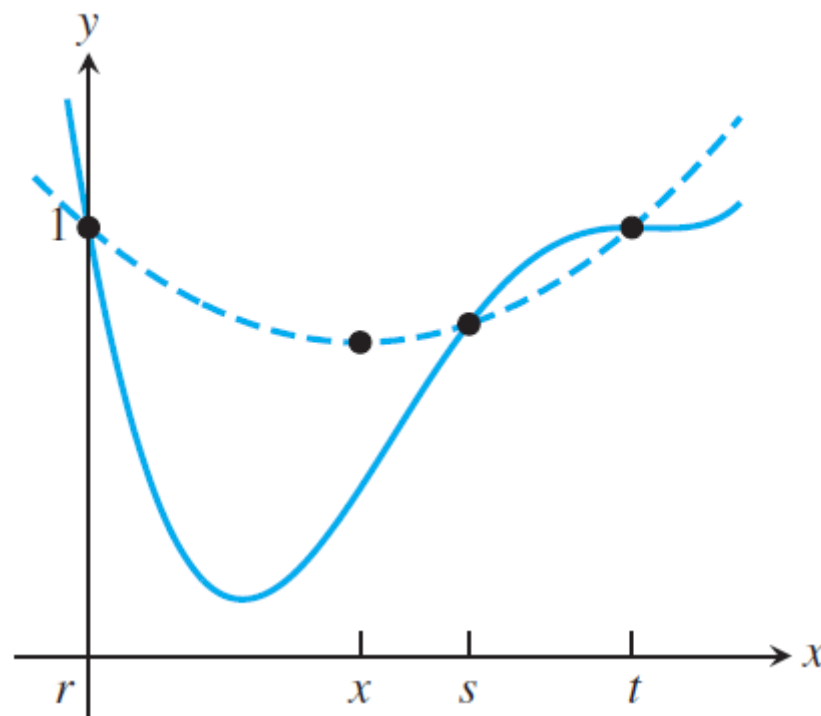
Visor de bombardeo



<https://en.wikipedia.org/wiki/Bombsight>



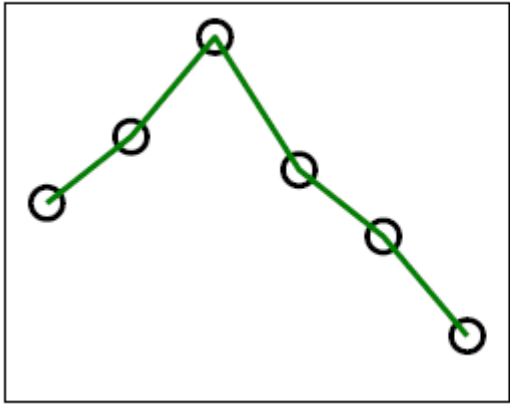
Interpolación parabólica (sucesiva)



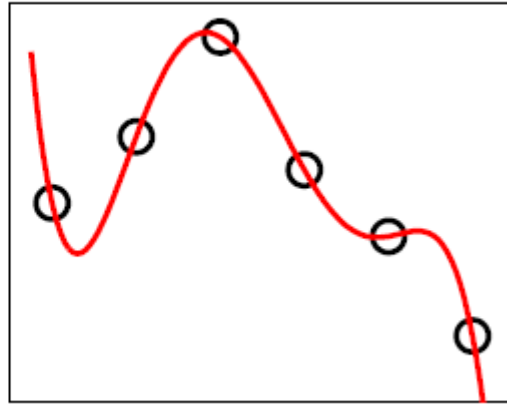
$$x = \frac{r + s}{2} - \frac{(f(s) - f(r))(t - r)(t - s)}{2[(s - r)(f(t) - f(s)) - (f(s) - f(r))(t - s)]}$$

Interpolación

Piecewise linear interpolation



Full degree polynomial interpolation



Lagrange

$$P_n(x_k) = y_k, \quad k = 1, \dots, n$$

$$P_n(x) = \sum_k \left(\prod_{j \neq k} \frac{x - x_j}{x_k - x_j} \right) y_k$$

$$P_2(x) = y_1 \frac{(x - x_2)(x - x_3)}{(x_1 - x_2)(x_1 - x_3)} + y_2 \frac{(x - x_1)(x - x_3)}{(x_2 - x_1)(x_2 - x_3)} + y_3 \frac{(x - x_1)(x - x_2)}{(x_3 - x_1)(x_3 - x_2)}$$

Interpolación

Newton

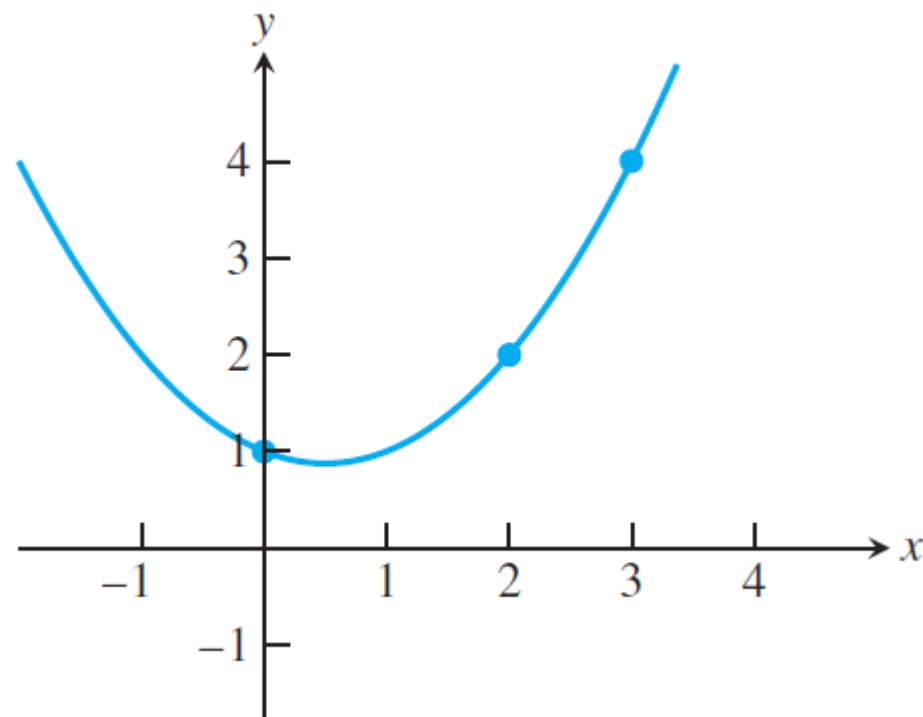
Diferencias divididas

$$P_2(x) = f[x_1] + f[x_1 \ x_2](x - x_1) \\ + f[x_1 \ x_2 \ x_3](x - x_1)(x - x_2)$$

x_1	$f[x_1]$		
		$f[x_1 \ x_2]$	
x_2	$f[x_2]$		$f[x_1 \ x_2 \ x_3]$
		$f[x_2 \ x_3]$	
x_3	$f[x_3]$		

$$f[x_k] = f(x_k) \\ f[x_k \ x_{k+1}] = \frac{f[x_{k+1}] - f[x_k]}{x_{k+1} - x_k} \\ f[x_k \ x_{k+1} \ x_{k+2}] = \frac{f[x_{k+1} \ x_{k+2}] - f[x_k \ x_{k+1}]}{x_{k+2} - x_k}$$

Interpolación



¿ $P_2(x)$?