NUMERICAL DIFFERENTIATION and INTEGRATION

Numerical Differentiation

PROBLEM

Estimate the derivatives (slope, curvature, etc.) of a function, given a set of function values at a discrete set of points.

→ Finite Difference Formulas

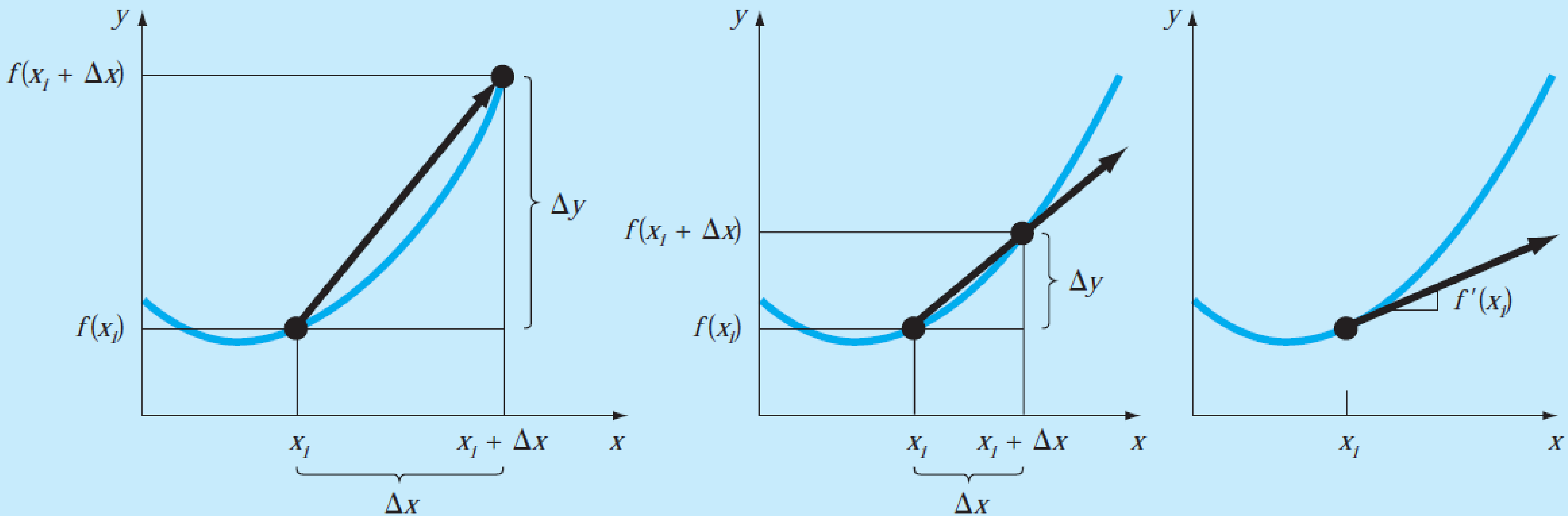
Numerical Differentiation

The simplest way to numerically compute a derivative is to mimic the formal definition:

$$u'(x) = \lim_{\Delta x \to 0} \frac{u(x+\Delta x) - u(x)}{\Delta x}$$

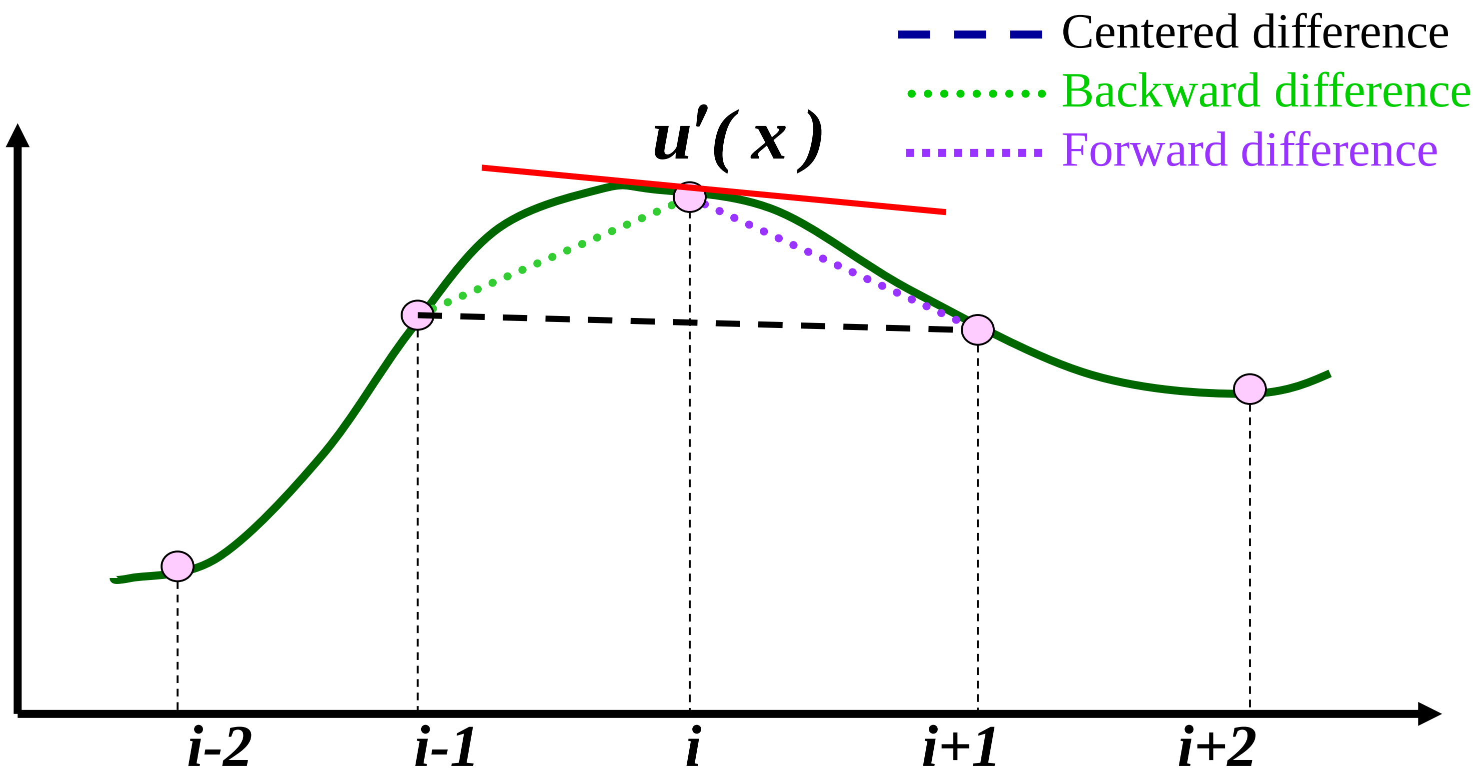
$$u'(x) \approx \frac{u(x+\Delta x) - u(x)}{\Delta x}$$

For a linear function *u(x)=ax+b* the formula is exact.



y
$f(x_1 + \Delta x)$
Δy
$f(x_1)$
x_1
$x_1 + \Delta x$
x
Δx
y
$f(x_1 + \Delta x)$
Δy
$f(x_1)$
x_1
$x_1 + \Delta x$
x
Δx
y
$f'(x_1)$
x_1
x

First Derivative at a point: finite difference schemes



First Derivative

Centered difference

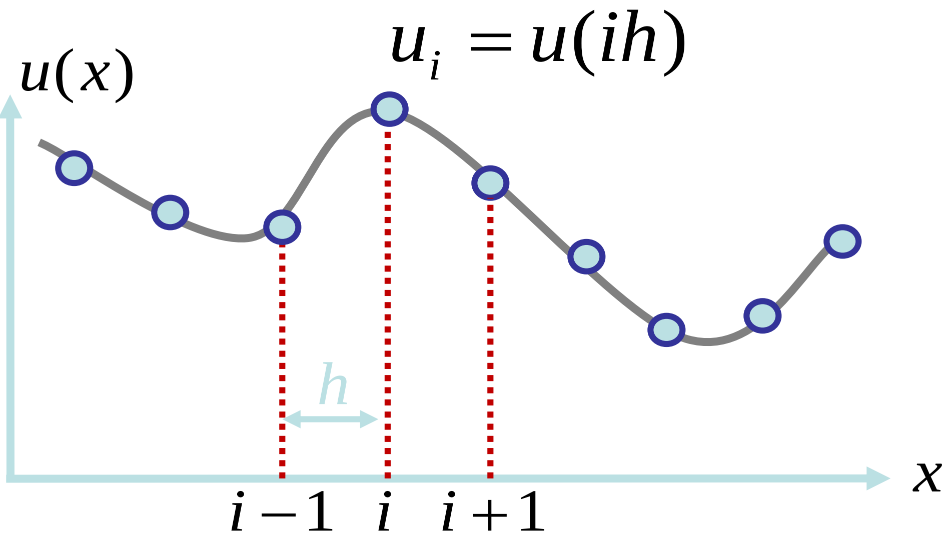
$$D_x u \equiv \frac{u_{i+1} - u_{i-1}}{2h}$$

Forward difference

$$D_x^+ u \equiv \frac{u_{i+1} - u_i}{h}$$

Backward difference

$$D_x^- u \equiv \frac{u_i - u_{i-1}}{h}$$



Local Truncation Error

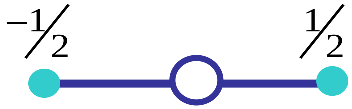
We write a Taylor expansion of u(x) about *x*=*ih*

$$u_{i+1} = u(ih+h) = u(ih) + hu'(ih) + \tfrac{1}{2!}h^2u''(ih) + O(h^3)$$

$$u_{i-1} = u(ih-h) = u(ih) - hu'(ih) + \tfrac{1}{2!}h^2u''(ih) + O(h^3)$$

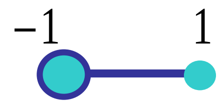
Second order error term

$$D_x u_i = u'(ih) + O(h^2)$$



First order error terms

$$D_x^+ u_i = u'(ih) + O(h)$$



$$D_x^- u_i = u'(ih) + O(h)$$

Stencils

Local Truncation Error

$$D_x u_i = u'(ih) + O(h^2)$$

Proof

$$u_{i+1} = u(ih+h) = u(ih) + hu'(ih) + \tfrac{1}{2!}h^2u''(ih) + \tfrac{1}{3!}h^3u'''(ih) + O(h^4)$$
$$u_{i-1} = u(ih-h) = u(ih) - hu'(ih) + \tfrac{1}{2!}h^2u''(ih) - \tfrac{1}{3!}h^3u'''(ih) + O(h^4)$$

Subtracting these two eqs:

$$u(ih+h) - u(ih-h) = 2hu'(ih) + 2\frac{h^3}{3!}u'''(ih) + O(h^4)$$

$$u'(ih) = \left(\frac{u(ih+h) - u(ih-h)}{2h}\right) + \frac{1}{3!}h^2u'''(ih) + O(h^3)$$

Second order error term

As the distance h tends to zero, we expect the approximation to improve

The greater the power of h the better the accuracy.

Consistency

For $h \to 0$ LTE approaches to zero

The "speed" in which the error goes to zero as h -> 0 is called the **rate of convergence**.

When the truncation error is of the order of O(h), we say that the method is a first order method. We refer to a method as a **pth-order method** if the truncation error is of the order of $O(h^p)$

Order of CONSISTENCY $O(h^p)$

Centered scheme ($O(h^2)$) is a more accurate formula than forward or backward (O(h)), that is the LTE decreases more rapidly.

Second Derivative

approximation of the second derivative by

Centered Difference formula

$$D_{xx}u \equiv \frac{u_{i+1} - 2u_i + u_{i-1}}{h^2}$$

$$D_{xx}u_i = u''(ih) + O(h^2)$$

Second order error term

Proof

$$u_{i+1} = u(ih+h) = u(ih) + hu'(ih) + \tfrac{1}{2!}h^2u''(ih) + \tfrac{1}{3!}h^3u'''(ih) + O(h^4)$$

$$u_{i-1} = u(ih-h) = u(ih) - hu'(ih) + \tfrac{1}{2!}h^2u''(ih) - \tfrac{1}{3!}h^3u'''(ih) + O(h^4)$$

Sum of these two eqns:

$$u(ih+h) + u(ih-h) = 2u(ih) + 2\left(\frac{h^2}{2!}u''(ih)\right) + 2\frac{h^4}{4!}u^{iv}(ih) + \dots$$

$$u''(ih) = \left(\frac{u(ih+h) - 2u(ih) + u(ih-h)}{h^2}\right) - \frac{1}{12}h^2u^{(iv)}(ih) + \dots$$

Finite Difference Formulas for k>1

Linear Operators

$$\Delta^1 u(x) = u(ih+h) - u(ih)$$ Forward linear operator

$$\nabla^1 u(x) = u(ih) - u(ih-h)$$ Backward linear operator

$$\delta^1 u(x) = u(ih+\frac{h}{2}) - u(ih-\frac{h}{2})$$ Centered linear operator

$$\Delta^1 u_i = u_{i+1} - u_i \qquad \nabla^1 u_i = u_i - u_{i-1} \qquad \delta^1 u_i = u_{i+\frac{1}{2}} - u_{i-\frac{1}{2}}$$

Finite Difference Formulas for k>1

We define the linear operators of order k at $\boldsymbol{x_i = ih}$ ***as***

$$\Delta^k u_i = \Delta^1 (\Delta^{k-1} u_i) = \Delta^{k-1} u_{i+1} - \Delta^{k-1} u_i$$

$$\nabla^k u_i = \nabla^1 (\nabla^{k-1} u_i) = \nabla^{k-1} u_i - \nabla^{k-1} u_{i-1}$$

$$\delta^k u_i = \delta^1 (\delta^{k-1} u_i) = \delta^{k-1} u_{i+\frac{1}{2}} - \delta^{k-1} u_{i-\frac{1}{2}}$$

Computing second order centered finite differencing

$$D_{xx} u_i = D^+ D^- u_i = D^- D^+ u_i = D_{1/2} D_{1/2} u_i$$

Finite Difference Formulas for k>1

Theorem

Let $u(x) \in C^k\left[x_0, x_n\right], h > 0,\ x_i = x_0 + ih,$

then $\exists \eta \in \left[x_i, x_i + h\right]$ such that

$$u^k(\eta) = \frac{\Delta^k u_i}{h^k}$$

Compute...

$$\delta^4 u_i = u_{i+2} - 4u_{i+1} + 6u_i - 4u_{i-1} + u_{i-2}$$

$$\Rightarrow u^{iv}(x_i) \approx \frac{u(x_{i+2}) - 4u(x_{i+1}) + 6u(x_i) - 4u(x_{i-1}) + u(x_{i-2})}{(h)^4}$$

Numerical problems

- Truncation Error:

  error due to the truncation of the Taylor expansion

- Rounding error:

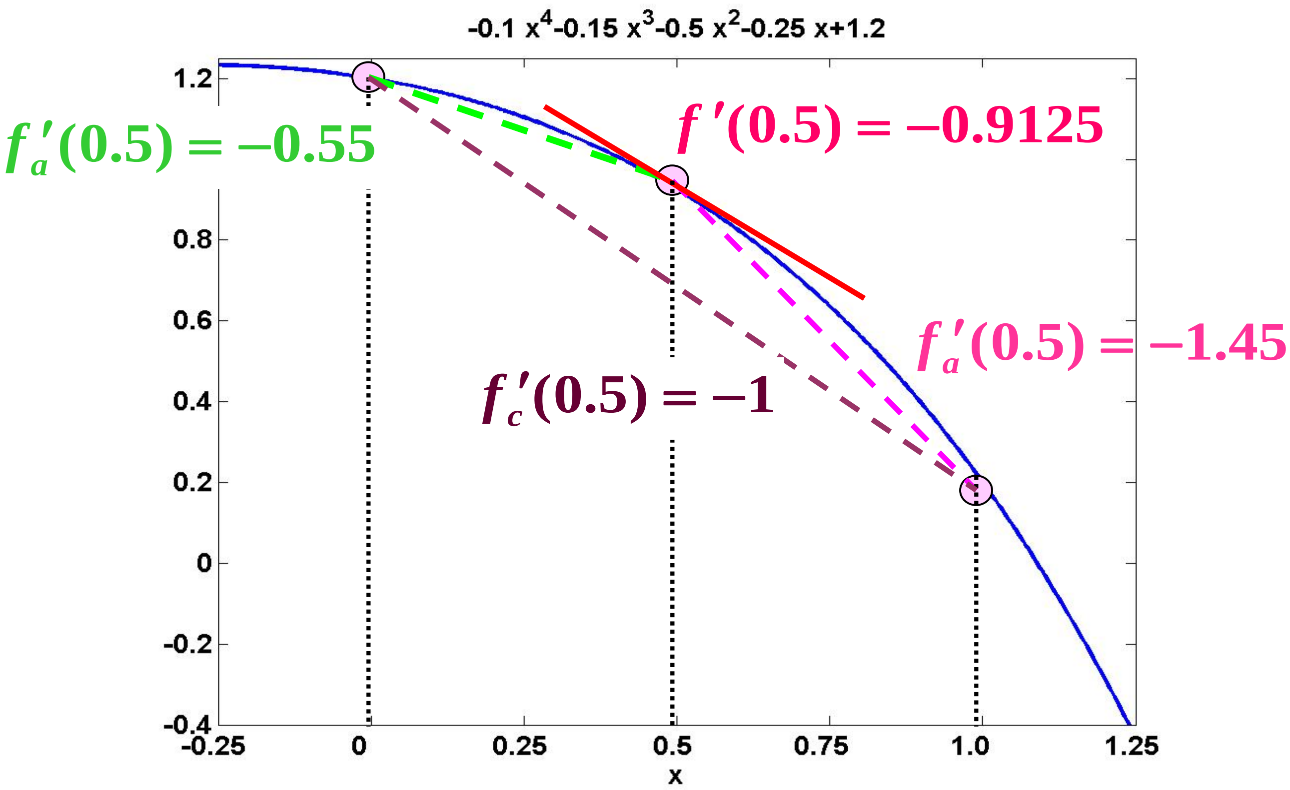
  approximation error in finite arithmetic

  In finite arithmetic a numerical evaluation which uses an arbitrarily small value of h does not lead to a reduction of total error.

Example

$$f(x) = -0.1x^4 - 0.15x^3 - 0.5x^2 - 0.25x + 1.2$$

Estimate the first derivative with backward, forward and centered differences at point x = 0.5 (with h = 0.5 and 0.25)



h=0.5

Example, h=0.5

Forward Difference

$$h = 0.5,\quad f'(0.5) = \frac{f(1) - f(0.5)}{1 - 0.5} = \frac{0.2 - 0.925}{0.5} = -1.45$$

$$\text{relative error}\quad \frac{-1.45 + 0.91250}{-0.91250} = 0.\ 58904$$

Backward Difference

$$h = 0.5,\quad f'(0.5) = \frac{f(0.5) - f(0)}{1 - 0.5} = \frac{0.925 - 1.2}{0.5} = -0.55$$

$$\text{relative error}\quad \frac{-0.55 + 0.91250}{-0.91250} = -0.39726$$

Centered Difference

$$h = 0.5,\quad f'(0.5) = \frac{f(1) - f(0)}{1 - 0} = \frac{0.2 - 1.2}{1} = -1.0$$

$$\text{relative error}\quad \frac{-1 + 0.91250}{-0.91250} = 0.09589$$

Example, h=0.25

Forward Difference

$$f_a'(0.5) = \frac{f(0.75) - f(0.5)}{0.75 - 0.5} = \frac{0.63632813 - 0.925}{0.25} = \text{-}1.1547,$$

$$\text{relative error} \quad \frac{\text{-}1.1547 + 0.91250}{-0.91250} = 0.26541$$

Backward Difference

$$f_i'(0.5) = \frac{f(0.5) - f(0.25)}{0.75 - 0.5} = \frac{0.925 - 1.10351563}{0.25} = \text{-}0.71406$$

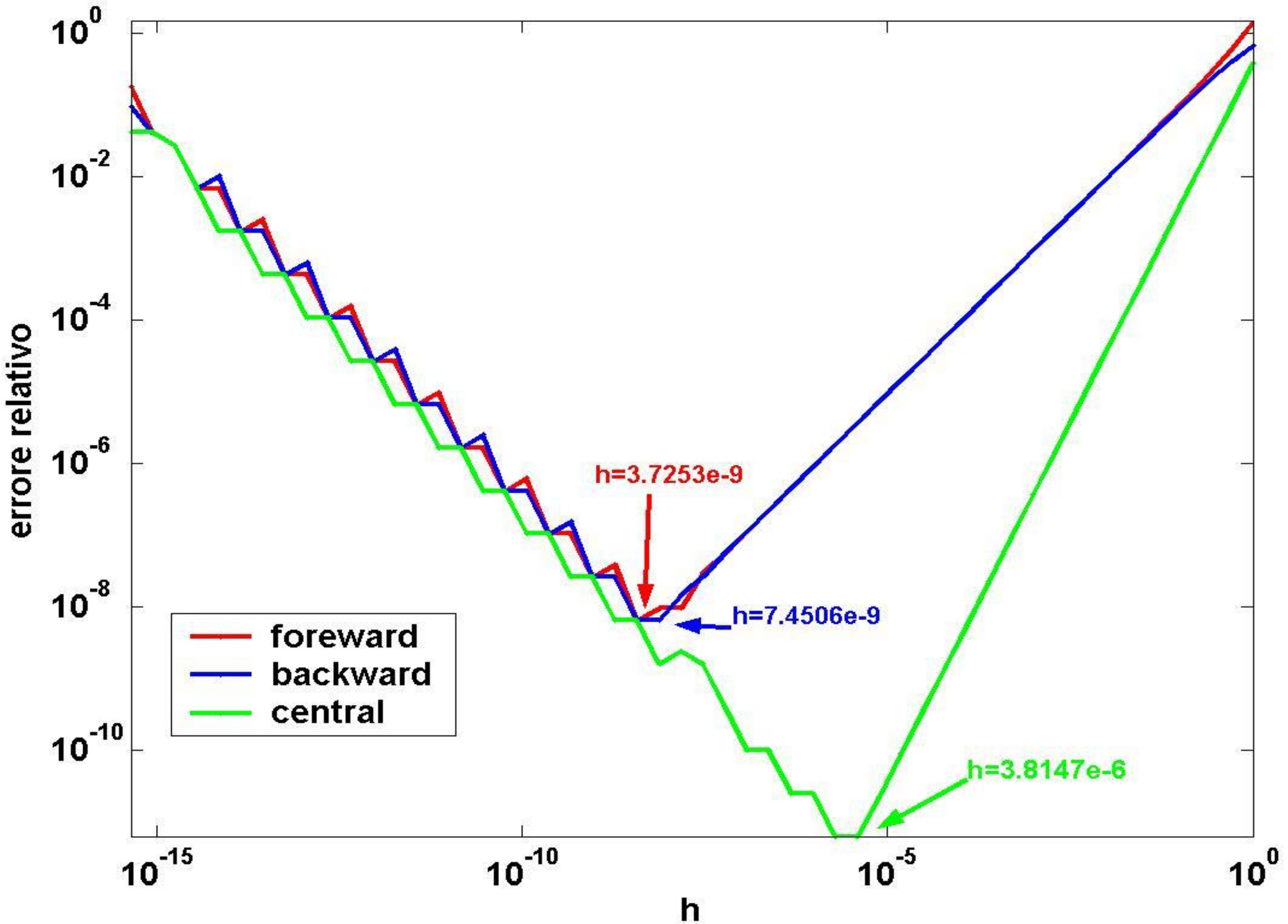
$$\text{relative error} \quad \frac{\text{-}0.71406 + 0.91250}{-0.91250} = -0.21747$$

Centered Difference

$$f_c'(0.5) = \frac{f(0.75) - f(0.25)}{0.75 - 0.25} = \frac{0.63632813 - 1.10351563}{0.5} = \text{-}0.93438$$

$$\text{relative error} \quad \frac{\text{-}0.93438 + 0.91250}{-0.91250} = 0.023973$$

Relative errors as a function of h



Remarks

- Rounding errors cause deterioration of the approximation for small values of h.
- The value of h which allows a correct evaluation of the formulas depends on the accuracy of the machine.
- If the terms $f(x_i \pm h)$ are calculated inaccurately then the errors are multiplied by a factor 1 / h, which grows very quickly for small values of h.

$$\tilde{f}(x_i + h) = f(x_i + h) + \delta \Rightarrow \hat{f}_a'(x_i) = \frac{\tilde{f}(x_i + h) - f(x_i)}{h} =$$

Computed value

$$= \frac{f(x_i + h) - f(x_i)}{h} + \frac{\delta}{h} = f_a'(x_i) + \frac{\delta}{h}$$

Approximation error

Differentiation Via Polynomial Interpolation

The first stage is to construct an interpolating polynomial from the data. An approximation of the derivative at any point can be then obtained by a direct differentiation of the interpolant.

Example The Lagrange form of the polynomial interpolation through 3 values (x_i,y_i) is:

$$p(x) = L_1(x)y_1 + L_2(x)y_2 + L_3(x)y_3$$

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

Differentiating the interpolant

$$p'(x) \cong \frac{2x - x_2 - x_3}{(x_1 - x_2)(x_1 - x_3)} y_1 + \frac{2x - x_1 - x_3}{(x_2 - x_1)(x_2 - x_3)} y_2 + \frac{2x - x_1 - x_2}{(x_3 - x_2)(x_3 - x_1)} y_3$$

Assuming uniform x points

$$p'(x) = \frac{2x - x_2 - x_3}{2\Delta x^2} y_1 + \frac{2x - x_1 - x_3}{-\Delta x^2} y_2 + \frac{2x - x_1 - x_2}{2\Delta x^2} y_3$$

First derivative of the Lagrange interpolant:

$$p'(x)=\frac{2x-x_2-x_3}{2\Delta x^2}y_1+\frac{2x-x_1-x_3}{-\Delta x^2}y_2+\frac{2x-x_1-x_2}{2\Delta x^2}y_3$$

Evaluate the derivative at several points:

$$p'(x_1)=\frac{2x_1-x_2-x_3}{2\Delta x^2}y_1+\frac{2x_1-x_1-x_3}{-\Delta x^2}y_2+\frac{2x_1-x_1-x_2}{2\Delta x^2}y_3=\frac{-3y_1+4y_2-y_3}{2\Delta x}$$

$$p'(x_2)=\frac{2x_2-x_2-x_3}{2\Delta x^2}y_1+\frac{2x_2-x_1-x_3}{-\Delta x^2}y_2+\frac{2x_2-x_1-x_2}{2\Delta x^2}y_3=\frac{y_3-y_1}{2\Delta x}$$

We get the centered difference formula

$$p'(x_3)=\frac{2x_3-x_2-x_3}{2\Delta x^2}y_1+\frac{2x_3-x_1-x_3}{-\Delta x^2}y_2+\frac{2x_3-x_1-x_2}{2\Delta x^2}y_3=\frac{y_1-4y_2+3y_3}{2\Delta x}$$

To calculate derivatives with higher order from the Lagrange interpolating polynomial,

$$p'(x) = \frac{2x - x_2 - x_3}{2\Delta x^2} y_1 + \frac{2x - x_1 - x_3}{-\Delta x^2} y_2 + \frac{2x - x_1 - x_2}{2\Delta x^2} y_3$$

differentiate

$$p''(x) = \frac{1}{\Delta x^2} y_1 + \frac{2}{-\Delta x^2} y_2 + \frac{1}{\Delta x^2} y_3 = \frac{y_1 - 2y_2 + y_3}{\Delta x^2}$$

To obtain derivatives of order n the interpolation polynomial must be of degree greater than or equal to n.

Truncation Error

We know that the interpolation error is

$$f(x) = p(x) + \frac{\Pi_k(x)}{(k+1)!} f^{(k+1)}(\xi) \quad ; \Pi_k(x) = \prod_{i=0}^{k} (x - x_i)$$

$$f'(x) = p'(x) + \frac{\Pi'_k(x)}{(k+1)!} f^{(k+1)}(\xi) + \frac{\Pi_k(x)}{(k+1)!} \frac{d}{dx} f^{(k+1)}(\xi)$$

$$if \quad x = x_i \quad is\ one\ of\ the\ knots, then \quad \Pi_k(x_i) = 0 :$$

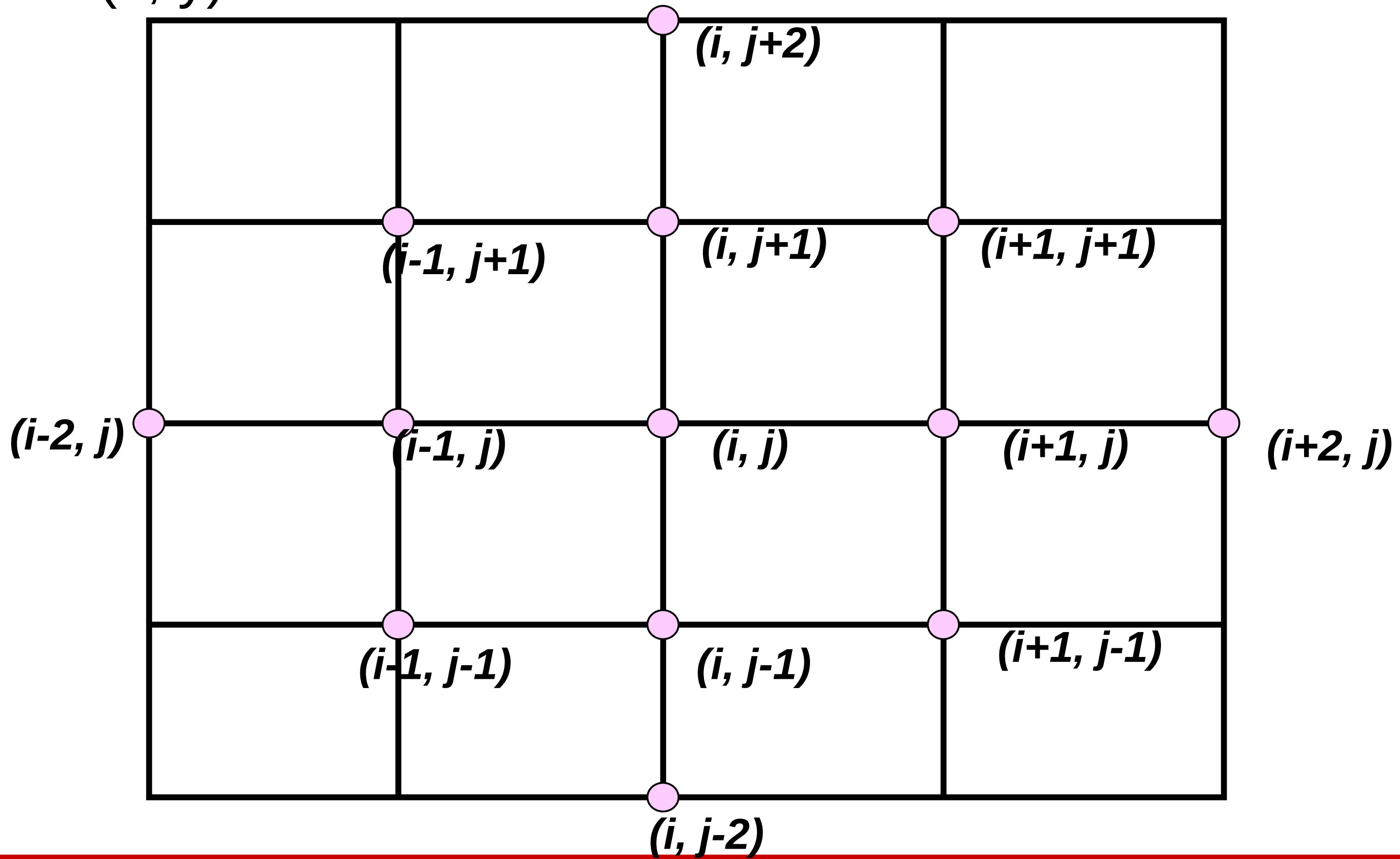
$$f'(x) = p'(x) + \frac{\Pi'_k(x_i)}{(k+1)!} f^{(k+1)}(\xi)$$

Truncation Error

Multi-dimensional Derivatives

Extension of the one-dimensional case:

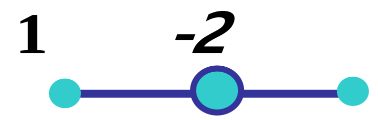
Finite difference formula to approximate partial derivatives of function u(x, y)

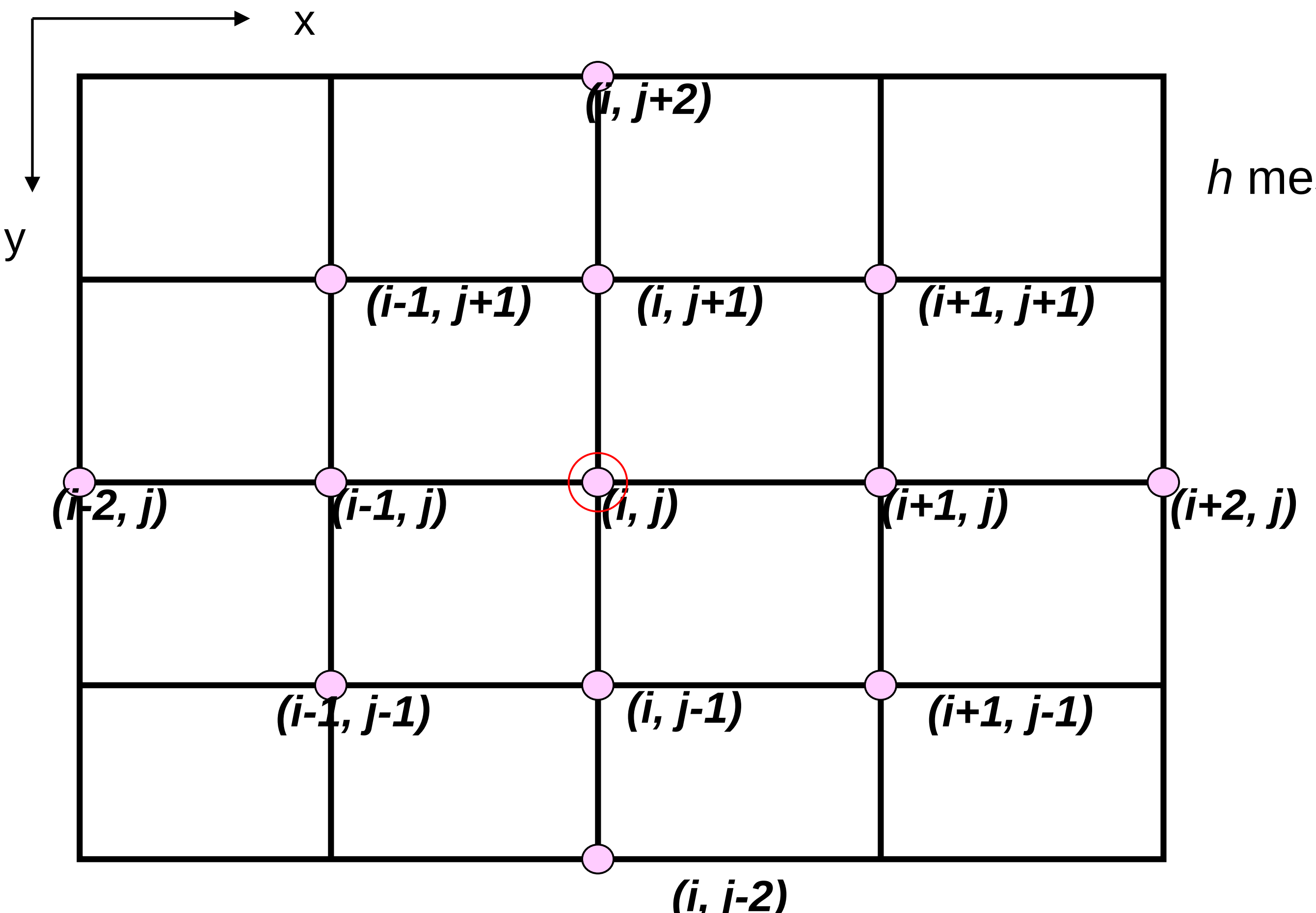


$$u_x = \frac{1}{2h}(-u(x_{i-1,}y_j) + u(x_{i+1}, y_j))$$

$$u_{xx} = \frac{1}{h^2}(u(x_{i-1,}y_j) - 2u(x_i, y_j) + u(x_{i+1}, y_j))$$

$$D_{xx}u \equiv \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h^2}$$

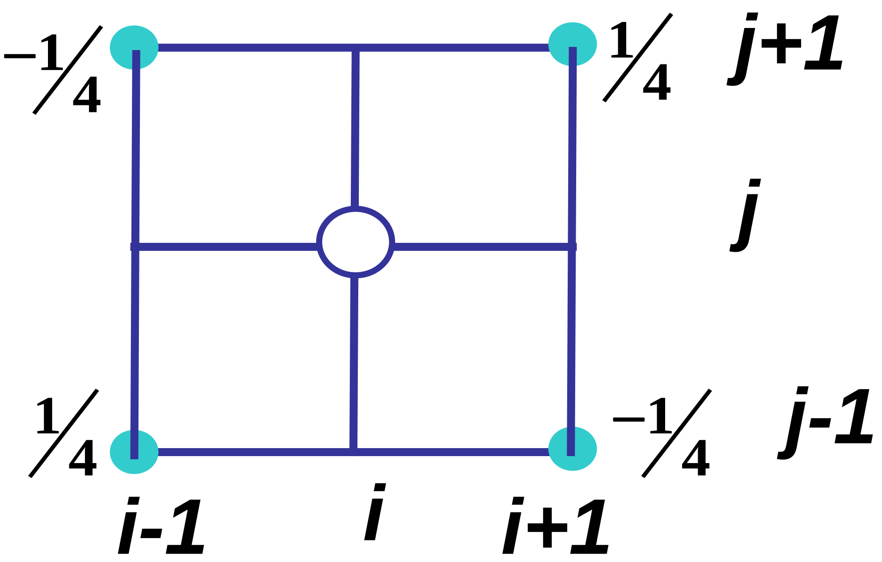




h mesh step

$$u_x = \frac{1}{2h}(u(x_{i+1}, y_j) - u(x_{i-1,} y_j))$$

$$u_{xy} = \frac{\partial}{\partial y}(u_x) = \frac{1}{4kh}(u(x_{i+1}, y_{j+1}) - u(x_{i+1}, y_{j-1}) - u(x_{i-1,} y_{j+1}) + u(x_{i-1}, y_{j-1}))$$



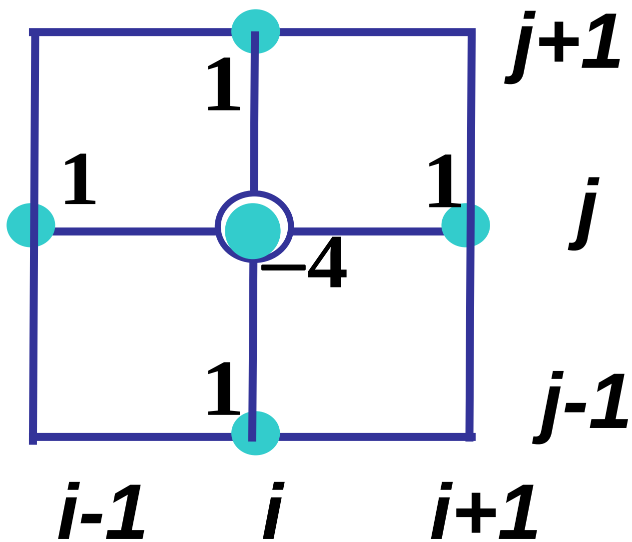
$$D_{xy}u \equiv \frac{u_{i+1,j+1} - u_{i-1,j+1} - u_{i+1,j-1} + u_{i-1,j-1}}{4h^2}$$

Laplacian Operator (5-points)

$$\nabla^2 u = u_{xx} + u_{yy} =$$

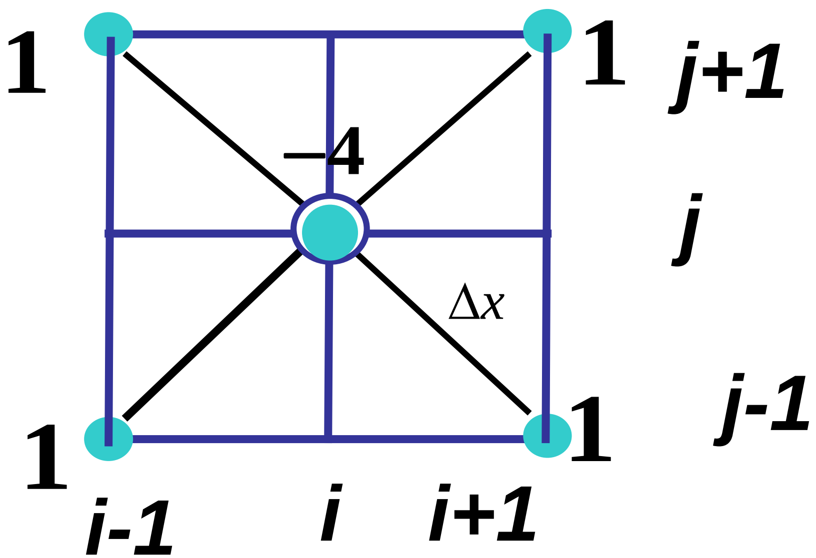
$$\approx \frac{u_{i+1,j} - 2u_{ij} + u_{i-1,j}}{h^2} + \frac{u_{i,j+1} - 2u_{ij} + u_{i,j-1}}{h^2}$$

$$\approx \frac{u_{i+1,j} + u_{i,j+1} + u_{i-1,j} + u_{i,j-1}}{h^2} - 4\frac{u_{ij}}{h^2}$$



local truncation error $O(h^2)$

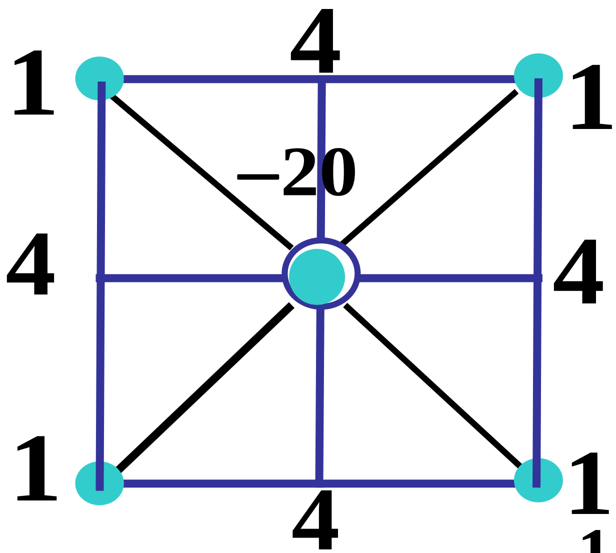
Laplacian Operator (5-points)



$$\nabla^2 u(x_i, y_j) \approx \frac{u_{i+1,j+1} + u_{i-1,j-1}}{2\Delta x^2} + \frac{u_{i+1,j-1} + u_{i-1,j+1}}{2\Delta x^2} - 4\frac{u_{ij}}{2\Delta x^2}$$

The local Truncation error for both the approximations is $O(h^2)$

Laplacian Operator (9-points)

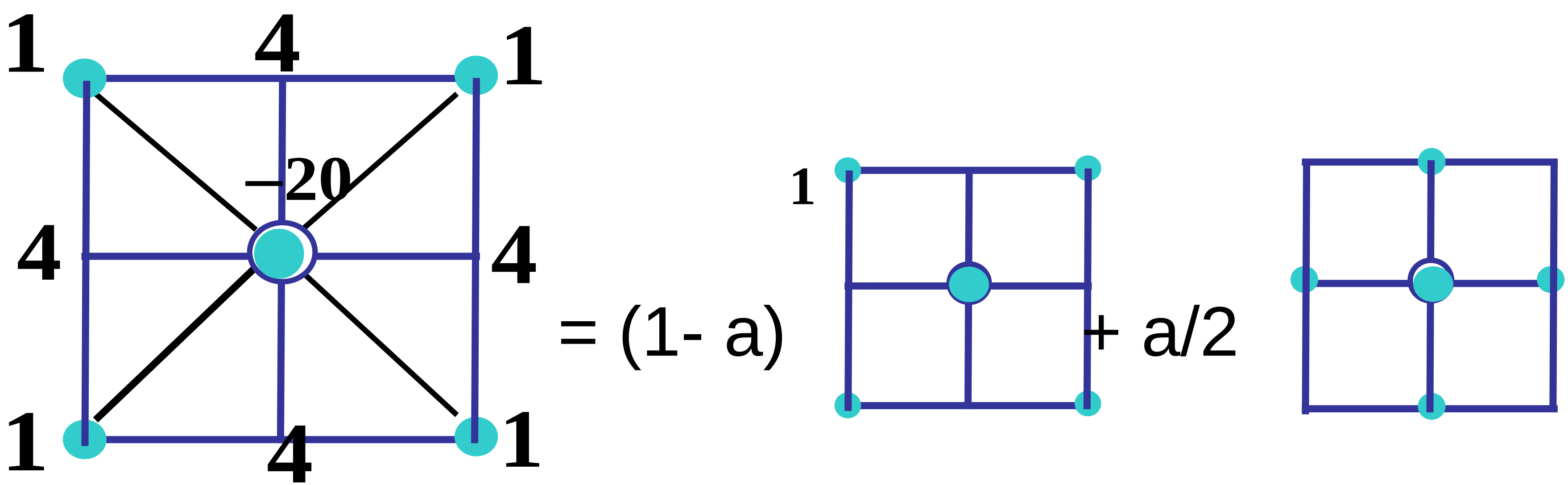


$$\nabla^2 u(x_i, y_j) \approx \frac{1}{6\Delta x^2}\Big[4u_{i+1,j} + 4u_{i-1,j} + 4u_{i,j-1} + 4u_{i,j+1}$$
$$+u_{i-1,j-1} + u_{i-1,j+1} + u_{i+1,j-1} + u_{i+1,j+1} - 20u_{ij}\Big]$$

The local Truncation Error is $O(h^2)$

$$\nabla_9^2 u(x_i, y_j) \approx \nabla^2 u + \frac{h^2}{2}(u_{xxxx} + 2u_{xxyy} + u_{yyyy}) + O(h^4)$$

Laplacian Operator (9-points)



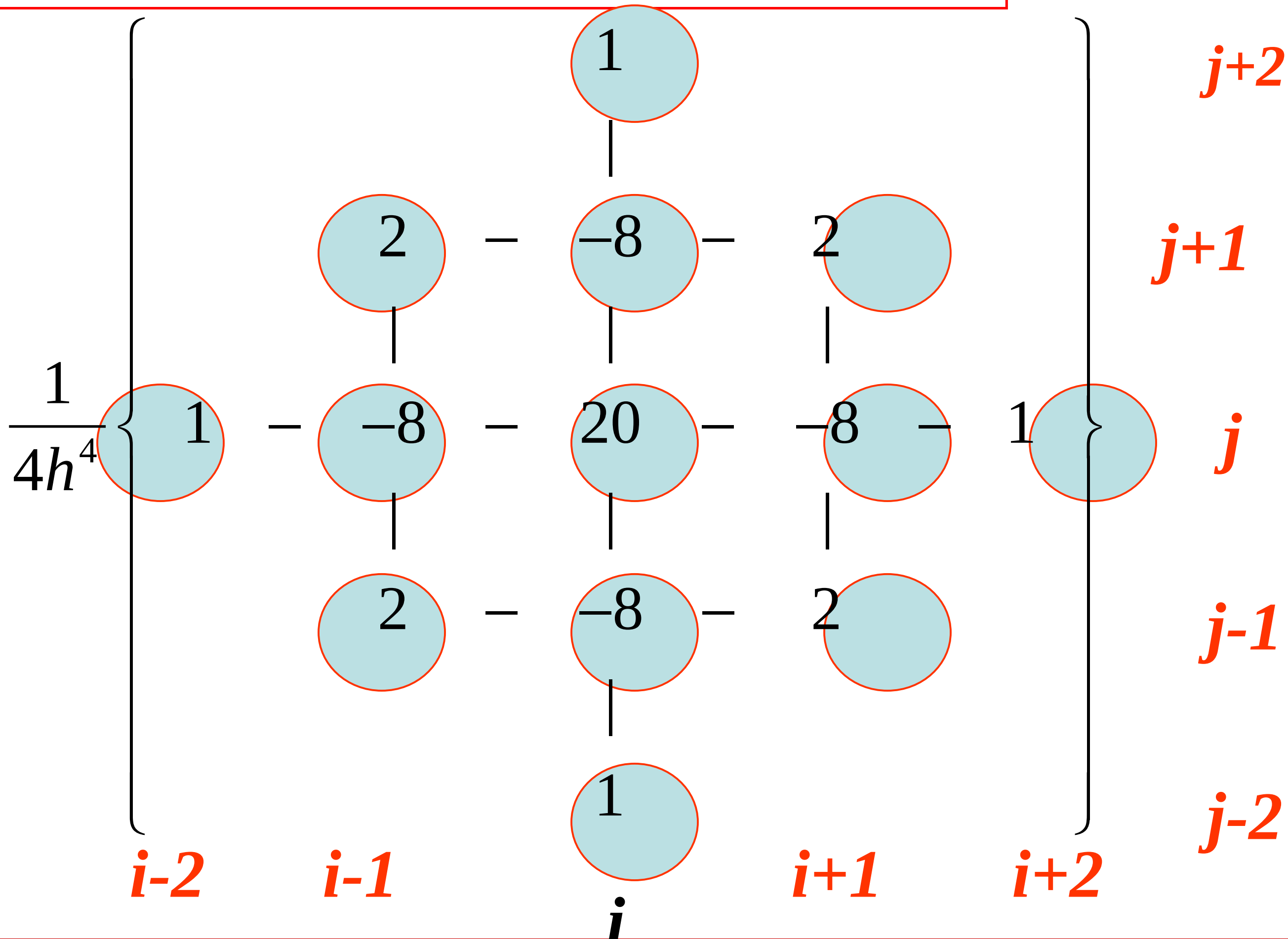
a=1/3 is the only value of a which yields a higher order of accuracy for the Laplacian

$$\nabla^2 u(x_i, y_j) \approx \frac{1}{6\Delta x^2}\Big[4u_{i+1,j} + 4u_{i-1,j} + 4u_{i,j-1} + 4u_{i,j+1} + u_{i-1,j-1} + u_{i-1,j+1} + u_{i+1,j-1} + u_{i+1,j+1} - 20u_{ij}\Big]$$

The local Truncation Error is $O(h^2)$

Biharmonic operator

$$\nabla^4 u = (\nabla^2 u)^2 = \frac{\partial^4 u}{\partial x^4} + 2\frac{\partial^2 u}{\partial x^2 \partial y^2} + \frac{\partial^4 u}{\partial y^4} \approx$$



Divergence Operator (1)

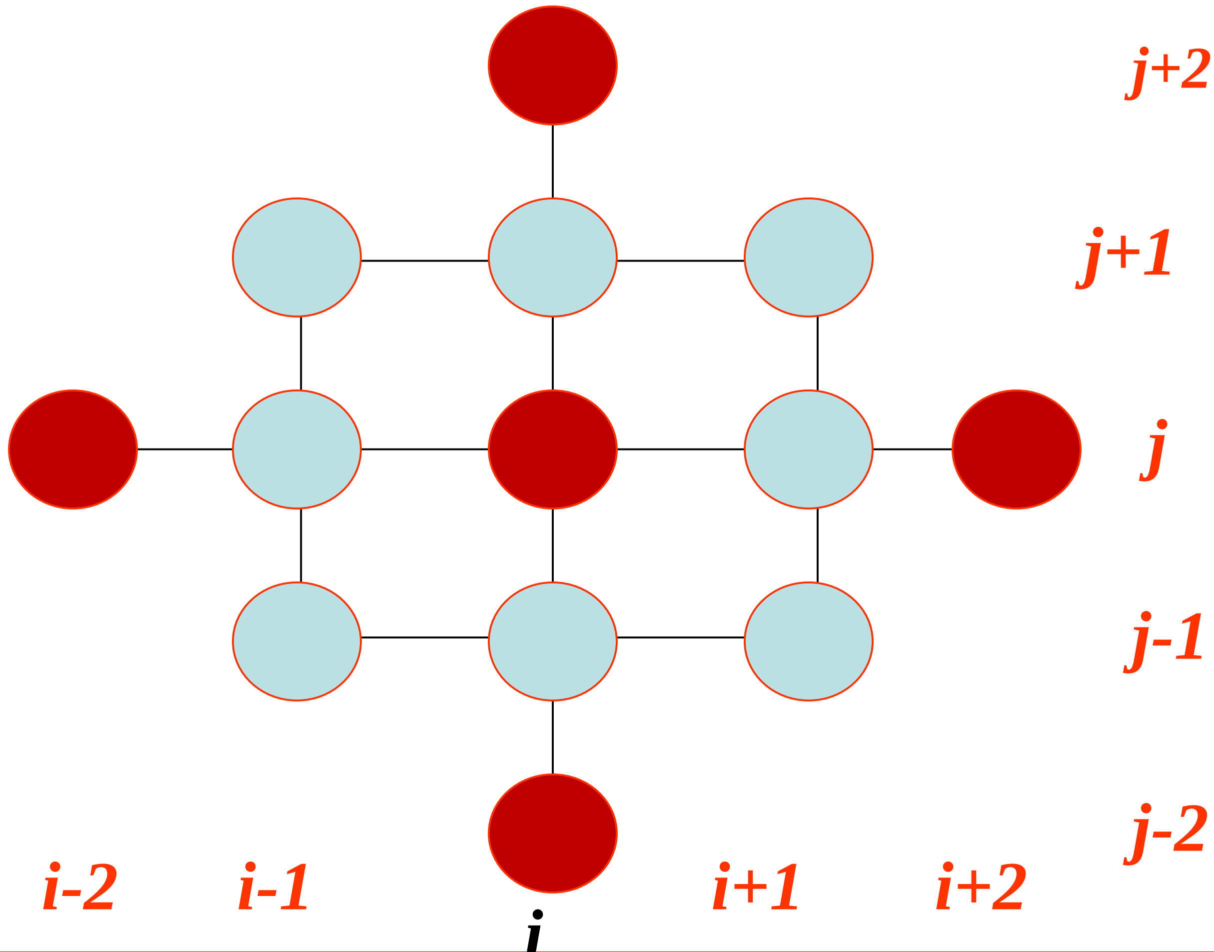
$$div(b\nabla u) = \frac{\partial}{\partial x}(b\frac{\partial u}{\partial x}) + \frac{\partial}{\partial y}(b\frac{\partial u}{\partial y})$$

$$div(b\nabla u) \approx \delta_x(b_{i,j}\delta_x u_{i,j}) + \delta_y(b_{i,j}\delta_y u_{i,j})$$

$$\approx \frac{b_{i+1,j}u_{i+2,j} + b_{i-1,j}u_{i-2,j} + b_{i,j+1}u_{i,j+2} + b_{i,j-1}u_{i,j-2}}{4h^2}$$

$$- \frac{(b_{i+1,j} + b_{i-1,j} + b_{i,j+1} + b_{i,j-1})u_{ij}}{4h^2}$$

Divergence Operator



⇨ It's not robust!

Divergence Operator (2)

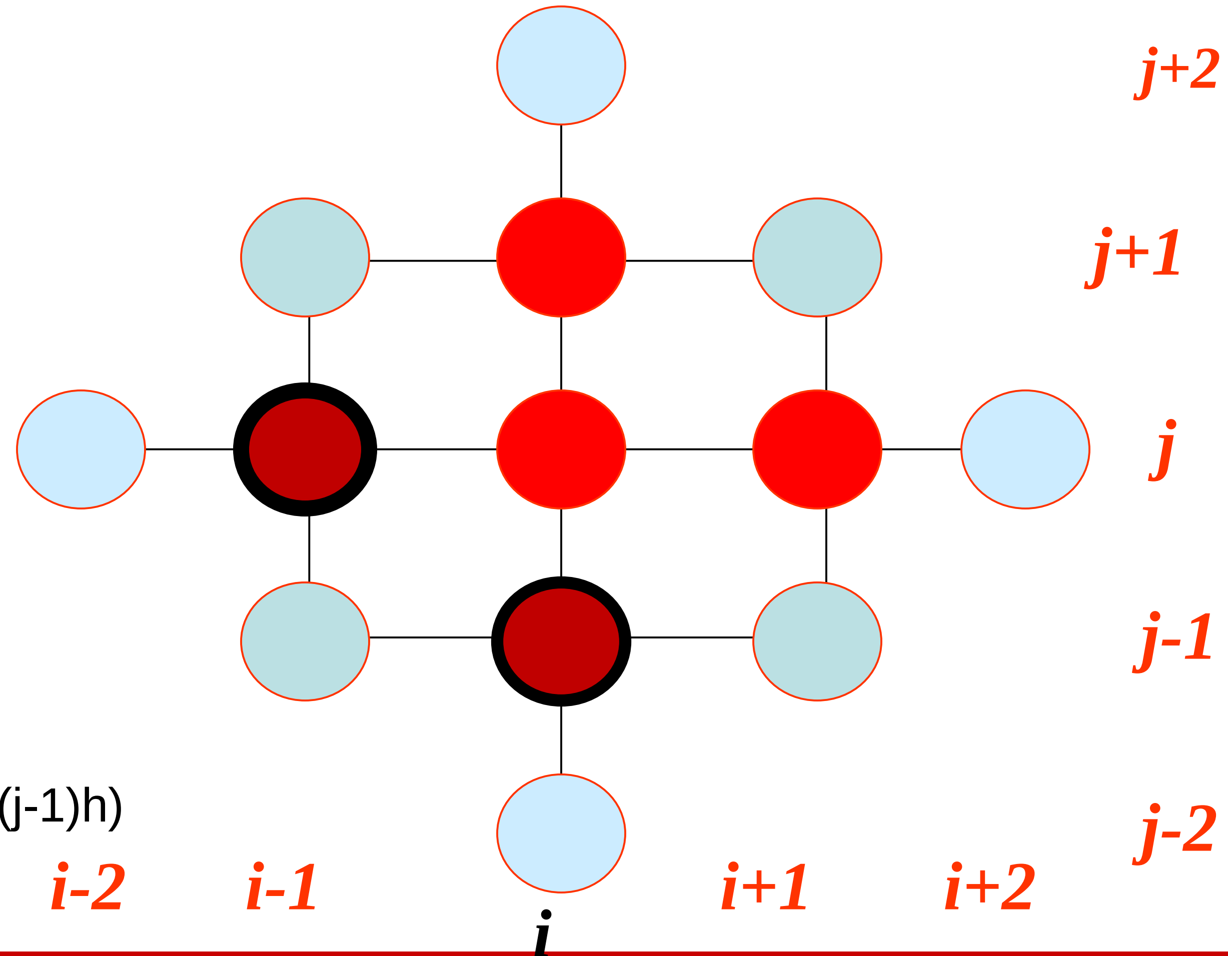
$$div(b\nabla u) = \frac{\partial}{\partial x}\left(b\frac{\partial u}{\partial x}\right) + \frac{\partial}{\partial y}\left(b\frac{\partial u}{\partial y}\right)$$

$$div(b\nabla u) \approx D_x^+(b_{i,j}D_x^- u_{i,j}) + D_y^+(b_{i,j}D_y^- u_{i,j})$$

$$\approx \frac{b_{i+1,j}u_{i+1,j} + b_{i,j}u_{i-1,j} + b_{i,j+1}u_{i,j+1} + b_{i,j}u_{i,j-1}}{h^2}$$

$$- \frac{(b_{i+1,j} + b_{i,j+1} + 2b_{i,j})u_{ij}}{h^2}$$

Divergence Operator



It does not use
b((i-1)h,jh) and b(ih,(j-1)h)

Divergence Operator (3)

$$div(b\nabla u) = \frac{\partial}{\partial x}\left(b\frac{\partial u}{\partial x}\right) + \frac{\partial}{\partial y}\left(b\frac{\partial u}{\partial y}\right)$$

$$\delta_x^* = \frac{u_{i+\frac{1}{2},j} - u_{i-\frac{1}{2},j}}{h} \qquad \delta_y^* = \frac{u_{i,j+\frac{1}{2}} - u_{i,j-\frac{1}{2}}}{h}$$

$$div(b\nabla u) \approx \delta_x^*(b_{i,j}\delta_x^* u_{i,j}) + \delta_y^*(b_{i,j}\delta_y^* u_{i,j})$$

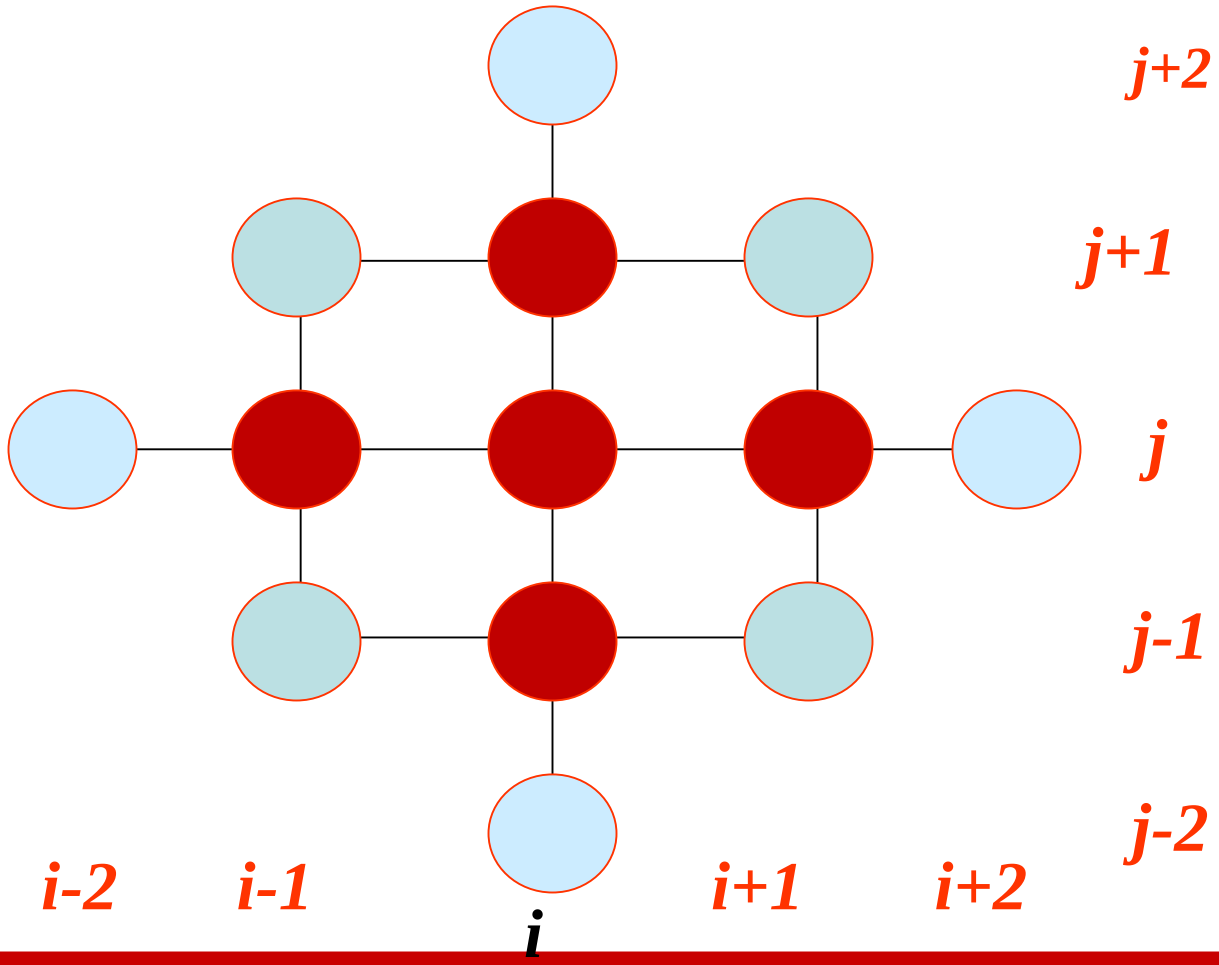
$$\approx \frac{b_{+0,j}u_{i+1,j} + b_{-0}u_{i-1,j} + b_{0+}u_{i,j+1} + b_{0-}u_{i,j-1}}{h^2} - \frac{(b_{+0} + b_{-0} + b_{0+} + b_{0-})u_{ij}}{h^2}$$

where

$$b_{\pm 0} = b_{i\pm\frac{1}{2},j} \quad e \quad b_{0\pm} = b_{i,j\pm\frac{1}{2}}$$

Interpolated Values

Divergence Operator



Second order

Remarks

- To increase the order of accuracy of the formulas is necessary to increase the number of points involved in the calculation.
- Higher precision is equivalent to a higher computational complexity.
- To achieve greater accuracy without increasing the order of the formulas we can use extrapolation techniques such as that of Richardson.

Richardson's Extrapolation

This technique uses the concept of grids with variable amplitude step procedure for improving the accuracy of approximations.

Example in which we show how to turn a **second-order approximation of the second derivative** into a fourth order approximation of the same quantity

$$f''(x_i) = \left(\frac{f(x_{i+1}) - 2f(x_i) + f(x_{i-1})}{\Delta x^2} \right) + a_1 \Delta x^2 + a_2 \Delta x^4 + \dots$$

Richardson's Extrapolation

second-order approximation of the second-derivative:

$$f''(x_i) = \left(\frac{f(x_{i+1}) - 2f(x_i) + f(x_{i-1})}{\Delta x^2} \right) + a_1 \Delta x^2 + a_2 \Delta x^4 + \ldots$$

We write the equation for grids of different sizes

$$f''(x_i) = F(\Delta x) + a_1 \Delta x^2 + a_2 \Delta x^4 + \ldots$$

$$f''(x_i) = F\left(\frac{\Delta x}{2}\right) + a_1 \left(\frac{\Delta x}{2}\right)^2 + a_2 \left(\frac{\Delta x}{2}\right)^4 + \ldots$$

Richardson's Extrapolation

This, of course, is still a second-order approximation of the derivative. However, the idea is to combine [1] with [2] such that the Δx^2 term in the error vanishes.

$$f''(x_i) = F(\Delta x) + \quad a_1 \Delta x^2 + a_2 \Delta x^4 + \quad \dots \qquad [1]$$

$$f''(x_i) = F\left(\frac{\Delta x}{2}\right) + \quad a_1 \left(\frac{\Delta x^2}{4}\right) + a_2 \left(\frac{\Delta x^4}{16}\right) + \quad \dots \qquad [2]$$

Richardson's Extrapolation

Indeed, multiplying eq.[2] by 4 and subtracting [1] from [2]

$$4f''(x_i) = 4F\left(\frac{\Delta x}{2}\right) + 4a_1\left(\frac{\Delta x^2}{4}\right) + 4a_2\left(\frac{\Delta x^4}{16}\right) + \ldots$$

$$-f''(x_i) = -F(\Delta x) - a_1\Delta x^2 - a_2\Delta x^4 + \ldots$$

$$3f''(x_i) = 4F\left(\frac{\Delta x}{2}\right) - F(\Delta x) - 12a_2\left(\frac{\Delta x^4}{16}\right) + \ldots$$

Richardson's Extrapolation

The equation can be rewritten as:

$$f''(x_i) = \left(\frac{4F\left(\frac{\Delta x}{2}\right) - F(\Delta x)}{3} \right) - a_2 \left(\frac{\Delta x^4}{4} \right) + \quad \ldots$$

The accuracy of the new estimation of $f''(x_i)$ is

$O(\Delta x^4)$ instead of $O(\Delta x^2)$

Richardson's Extrapolation

We can continue to eliminate higher order terms of the error using a grid even finer:

$$f''(x_i) = B(\Delta x) + b_1\Delta x^4 + b_2\Delta x^6 + \quad \ldots$$

$$f''(x_i) = \left(\frac{16B\left(\frac{\Delta x}{2}\right) - B(\Delta x)}{15} \right) + O\left(\Delta x^6\right) + \quad \ldots$$

Richardson's Extrapolation example

Given the function

$$f(x) = x^3 - 2x^2 + 4x - 8$$

Find the first derivative at x=1.25 using a centered difference formula and step Δh = 0.25.

$$f'(x) = 3x^2 - 4x + 4 = 3(1.25)^2 - 4(1.25) + 4 = 3.6875$$

Richardson's Extrapolation example

Given the discrete data set:

x	f(x)
1	-5
1.125	-4.607
1.25	-4.172
1.375	-3.682
1.5	-3.125

Compute first derivatives using centered differencing

$$F(\Delta x) \approx f'(1.25) = \frac{f(1.5) - f(1.0)}{2(0.25)} = \frac{-3.125 + 5}{0.5} = 3.75$$

$$F(\Delta x/2) \approx f'(1.25) = \frac{f(1.375) - f(1.125)}{2(0.125)} = \frac{-3.6816 + 4.6074}{0.25} = 3.7032$$

Richardson's Extrapolation example

Compute the error with the exact first derivatives:

$$F(\Delta x) \approx f'(1.25) = 3.75 \qquad \text{Error} = 1.69\%$$

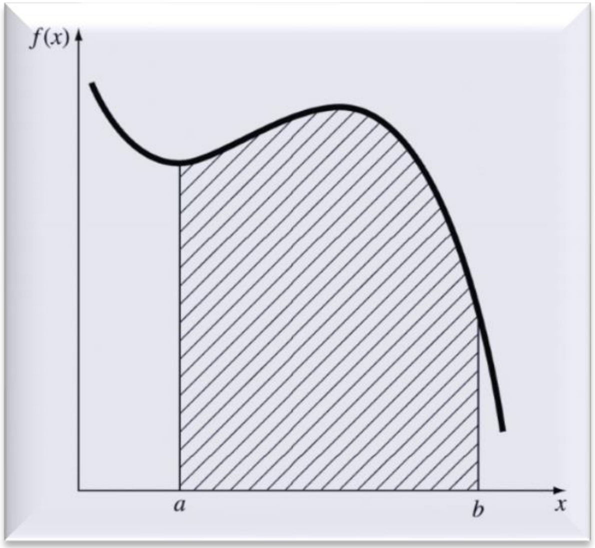
$$F(\frac{\Delta x}{2}) \approx f'(1.25) = 3.7032 \qquad \text{Error} = 0.425\%$$

Apply Richardson's extrapolation using these results to find a better solution

$$f'(1.25) = \frac{4F\left(\frac{\Delta x}{2}\right) - F(\Delta x)}{3} = \frac{4(3.7032) - 3.75}{3} = 3.6876 \qquad \text{Error} = 0.003\%$$

NUMERICAL INTEGRATION

- **Newton-Cotes quadrature rules**
  - Trapezoidal
  - Simpson
  - Rectangle
  - Composite
- **Adaptive Methods**



LUNGHEZZE

Lunghezza di una funzione

Data una funzione regolare y=f(x) x in [a,b] intervallo limitato, la lunghezza della curva che rappresenta il grafico della funzione è fornita dalla formula

$$L = \int_a^b \sqrt{(1+(f'(x))^2)}dx$$

Lunghezza di una curva

descritta parametricamente C(t)=(x(t),y(t)) , t in [0,1]

–lunghezza d'arco-:

$$L = \int_0^1 \|C'(t)\| dt = \int_0^1 \sqrt{(x'(t))^2 + (y'(t))^2}\, dt$$

AREE

Area

Data una funzione regolare y=f(x) x in [a,b] intervallo limitato, f(x)>0, l'area sottesa alla funzione f(x) è fornita dalla formula

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

Area di una curva

Data una curva in forma parametrica C(t)=(x(t),y(t)), t in [0,1], l'area che tale curva sottende con l'origine degli assi (se la curva è chiusa questo equivale all'area della regione che resta definita dalla curva), è fornita dalla formula

$$A = \pm\frac{1}{2}\int_0^1 C(t)\times C'(t)dt = \pm\frac{1}{2}\int_0^1 [x(t)y'(t) - x'(t)y(t)]dt$$

+ parametrizzazione antioraria, - oraria

VOLUMI

Data una funzione regolare z=f(x,y) x,y in D=[a,b]x[c,d] intervallo limitato, f(x,y)>0, il volume racchiuso tra la superficie f(x,y) e il dominio D è fornito dalla formula

$$V = \iint_D f(x,y)\,dxdy$$

Ipervolume

Data una funzione regolare w=f(x,y,z) x,y,z in E=[a,b]x[c,d]x[e,f] intervallo limitato, f(x,y,z)>0, l'ipervolume è fornito dalla formula

$$V = \iiint_E f(x,y,z)\,dxdydz$$

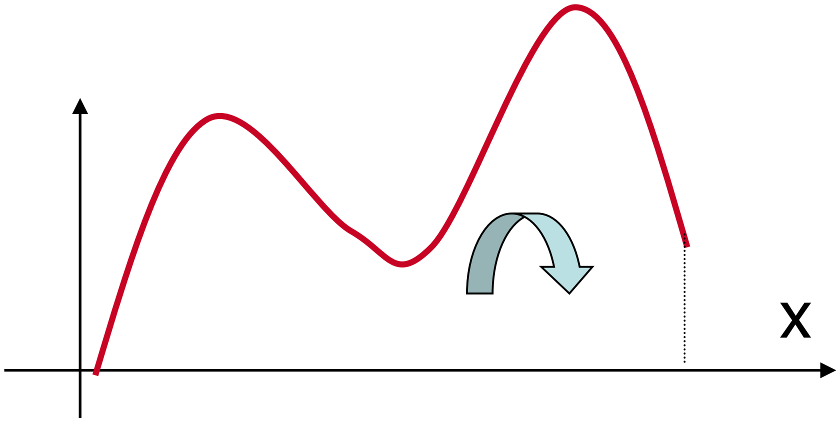
Se f(x,y,z)=1 per ogni x in E, allora: $V = \iiint_E dV$

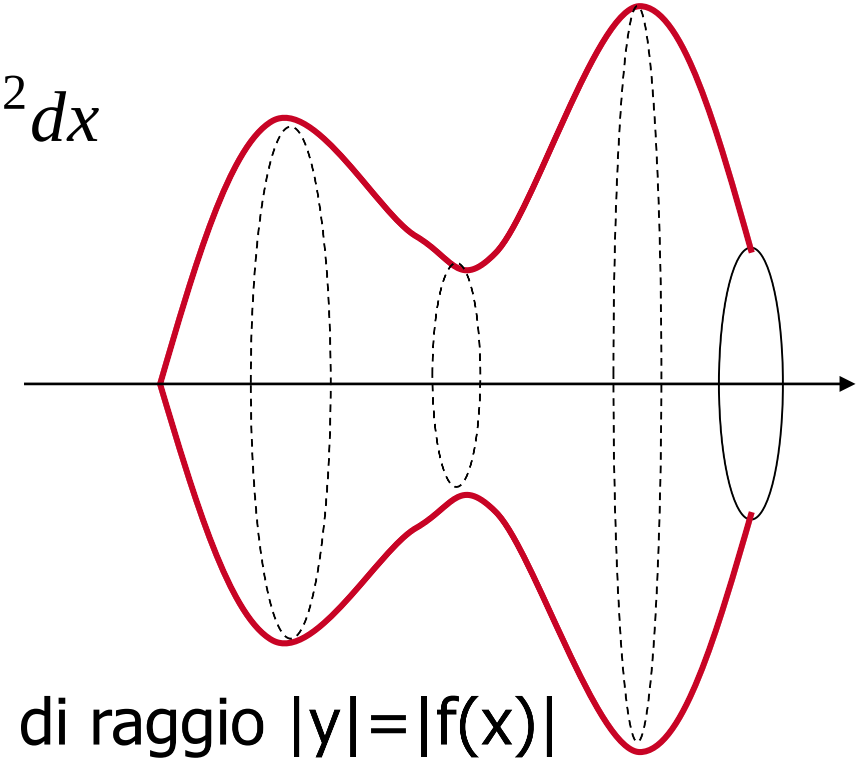
Rappresenta il volume di E

Volume di un solido di rotazione

Il solido è ottenuto facendo ruotare intorno all'asse x la curva y=f(x), x in [a,b]:

$$V = \int_a^b A(x)dx = \pi \int_a^b (f(x))^2 dx$$





La sezione in x è il disco circolare di raggio |y|=|f(x)| quindi l'area della sezione è: $A = \pi y^2 = \pi[f(x)]^2$

Massa e baricentro di un corpo bidimesionale

Una lamina che occupa una regione piana P composta di un materiale di densità superficiale m(x,y). La massa totale del corpo è data da:

$$massa = \iint_P m(x,y)\,dxdy$$

Mentre le coordinate del baricentro (x_b, y_b) sono:

$$x_b = \frac{1}{massa}\iint_P x\, m(x,y)\,dxdy \qquad y_b = \frac{1}{massa}\iint_P y\, m(x,y)\,dxdy$$

Massa e baricentro di un solido

Supponiamo di aver un oggetto solido che occupa il volume E composto di un materiale di densità superficiale m(x,y,z).
La massa totale dell'oggetto è data da:

$$massa = \iiint_E m(x, y, z)dV$$

Mentre il centro di massa è localizzato nel punto (x_b,y_b ,z_b), dove:

$$x_b = \frac{1}{massa} \iiint_E x\, m(x, y, z)dV$$

$$y_b = \frac{1}{massa} \iiint_E y\, m(x, y, z)dV$$

$$z_b = \frac{1}{massa} \iiint_E z\, m(x, y, z)dV$$

Momenti rispetto ai tre piani coordinati

Momenti di inerzia rispetto ai tre assi

$$I_x = \iiint_E (y^2 + z^2)\, m(x,y,z)\,dV$$

$$I_y = \iiint_E (x^2 + z^2)\, m(x,y,z)\,dV$$

$$I_z = \iiint_E (x^2 + y^2)\, m(x,y,z)\,dV$$

Numerical Integration

Find an approximation of the *definite* integral of the real function f(x) with respect to the independent variable x, evaluated between the limits x = a to x = b. The function f(x) is referred to as the *integrand:*

$$I(f;a,b)=\int_a^b f(x)dx$$

Why provide an approximation:

- A complicated continuous function that is difficult or impossible to integrate directly.
- A tabulated function where values of x and f(x) are given at a number of discrete points, as is often the case with experimental or field data.

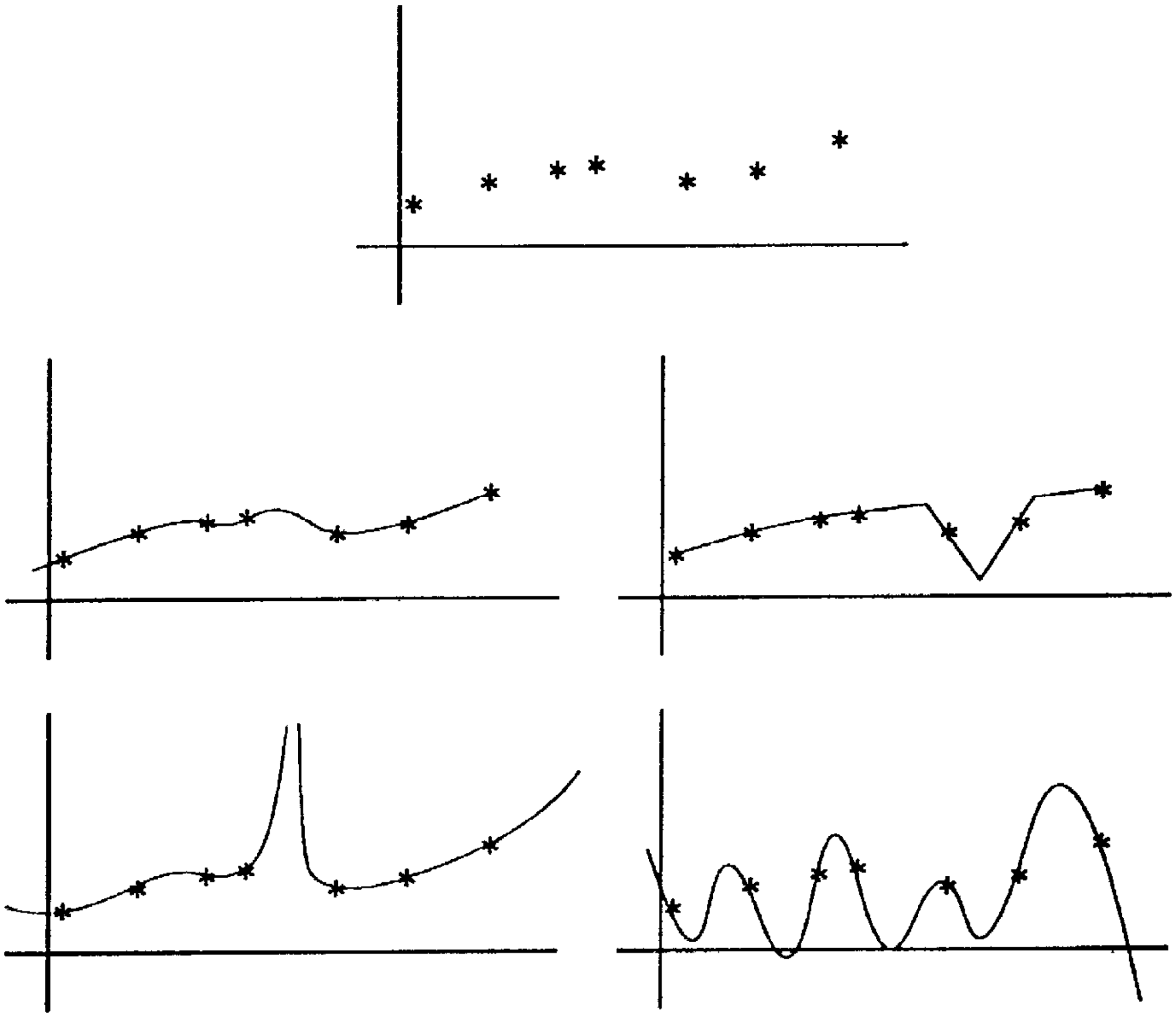
Fundamental theorem of integral calculus

Determine an integral between specified limits,

$$I(f;a,b)=\int_a^b f(x)\,dx=F(x)\Big|_a^b=F(b)-F(a)$$

F(*x*) = the integral of *f*(*x*) that is, any function such that *F'*(*x*) = *f*(*x*).

Functions for the same points



Numerical integration methods: **the quadrature formula**

Suppose we know (or can evaluate) the integrand function f(x) at points x_i (chosen or fixed), distinct in [a, b]

$$I(f;a,b)=\int_a^b f(x)dx \approx I_n(f)=\sum_{i=0}^{n} w_i f(x_i)$$

Coefficients or weights $\mathbf{c_i}$ **nodes**

Residual (Error) of the quadrature formula

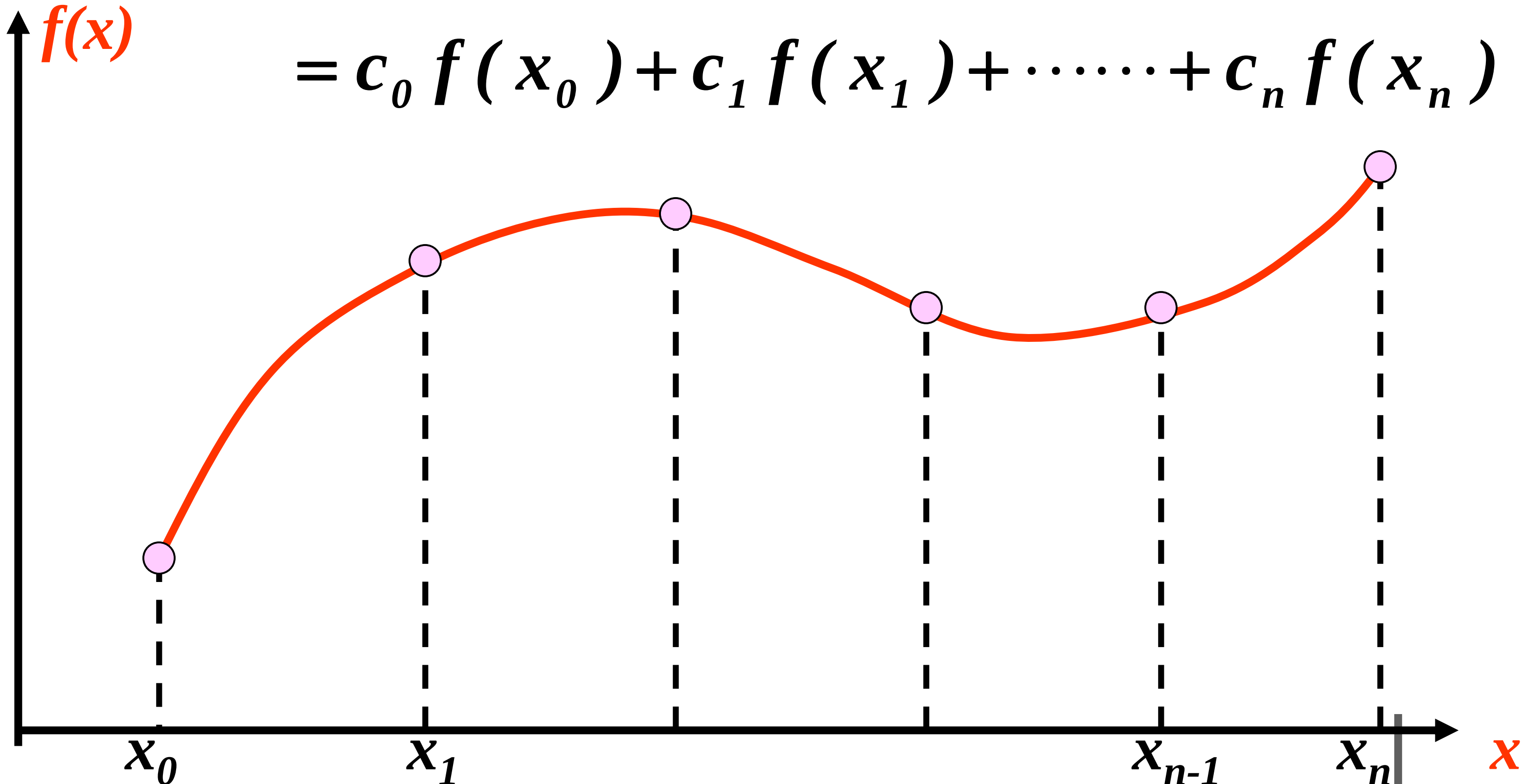
$$r_n = I - I_n$$

Quadrature formulas

Weighted sum of values of the function at appropriate nodes (points) belonging or not to the integration interval

$$\int_a^b f(x)dx \approx \sum_{i=0}^{n} c_i f(x_i)$$

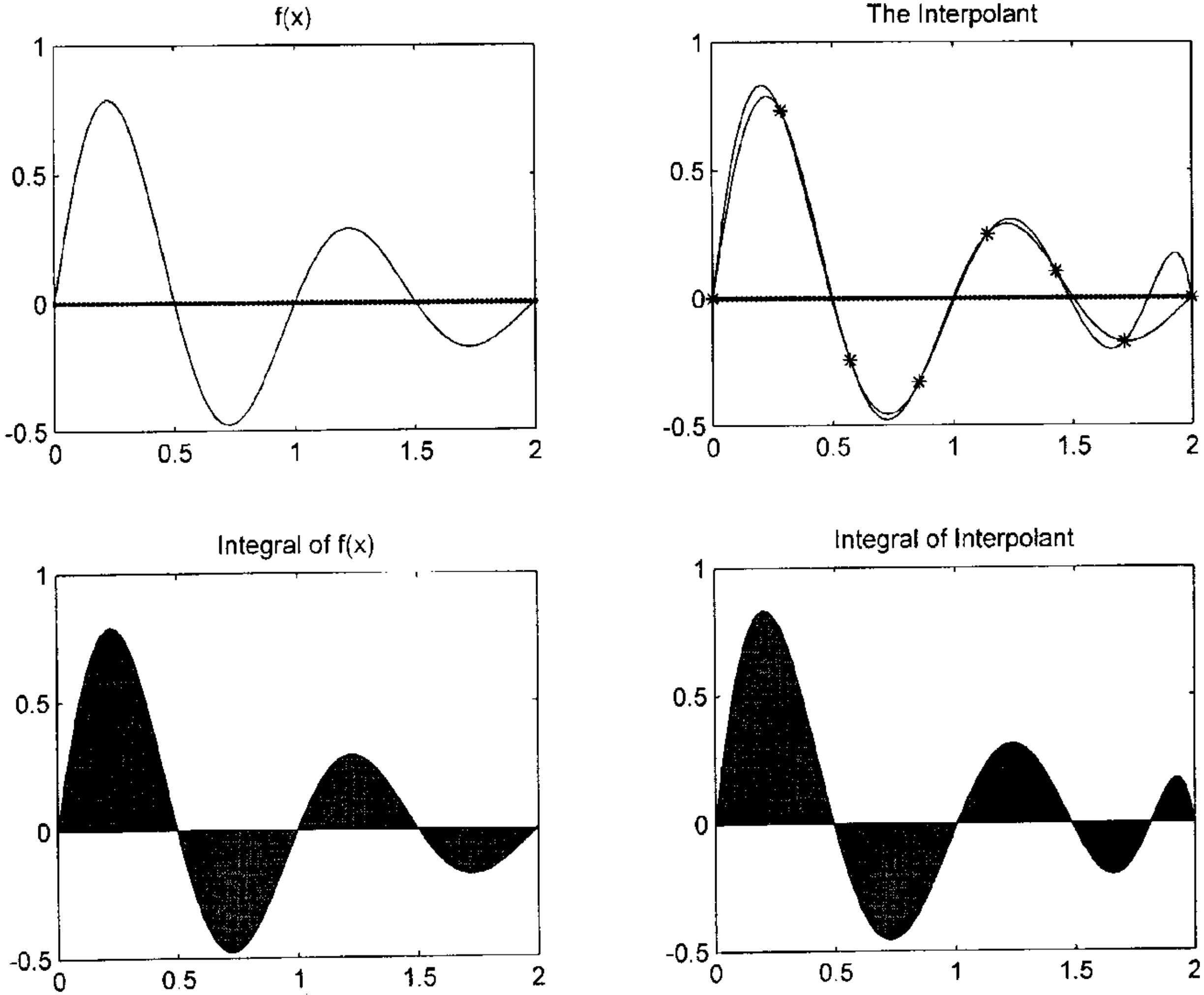
$$= c_0 f(x_0) + c_1 f(x_1) + \cdots\cdots + c_n f(x_n)$$



$$I(f;a,b)=\int_a^b f(x)dx \qquad I_n(f;a,b)=\int_a^b p_n(x)dx$$

The idea is to replace the integrand function, f(x), with an easier integration, typically a polynomial

Interpolating Polynomial of the function f(x) in n + 1 points $(x_i, f(x_i))$



Quadrature formulas

Let $p_n(x)$ be the Lagrange interpolating polynomial in n + 1 distinct points

$$p_n(x) = \sum_{i=0}^{n} f(x_i) L_i(x)$$

coefficients

$$I_n(f) = \int_a^b p_n(x)\,dx = \sum_{i=0}^{n} f(x_i) \int_a^b L_i(x)\,dx$$

nodes

The quadrature formula is accurate for construction for polynomials of degree at least n (accuracy degree at least n)

Accuracy degree

$$\underbrace{\int_a^b f(x)dx}_{I} = \underbrace{\int_a^b p(x)dx}_{I_n} + r_n \approx \int_a^b p(x)dx$$

Error of the quadrature formula

$$r_n = I - I_n$$

$$I_n(f) = \sum_{i=0}^{n} w_i f(x_i)$$

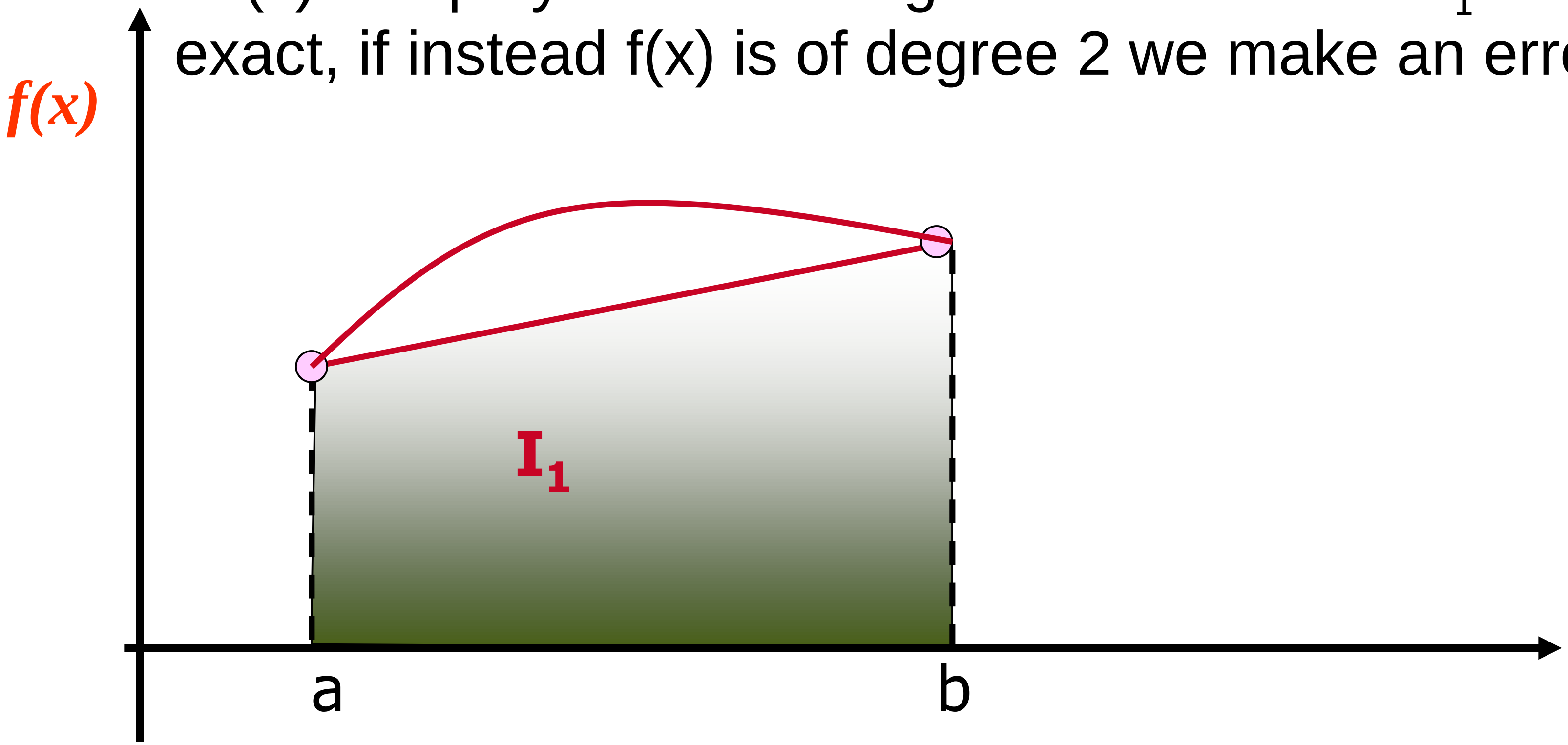
Accuracy degree

A quadrature formula I_n has degree of precision k if it is exact ($r_n = 0$) when the integrand function is any polynomial p(x) of degree less than or equal to k.

$$I(p) = I_n(p) \qquad \forall p \in \mathbf{P}_k$$

Accuracy degree

If f(x) is a polynomial of degree 1 the formula I_1 is exact, if instead f(x) is of degree 2 we make an error



I_1 has accuracy degree 1

Newton-Cotes quadrature formulas (equispaced nodes)

- **Newton-Cotes Closed forms**

  the data points at the beginning and end of the limits of integration are included as nodes

  - Trapezoidal rule: Linear
  - Simpson 1/3 formula: Quadratic
  - Simpson 3/8 formula: Cubic

- **Newton-Cotes Open forms**

  the data points at the beginning and end of the limits of integration are not included

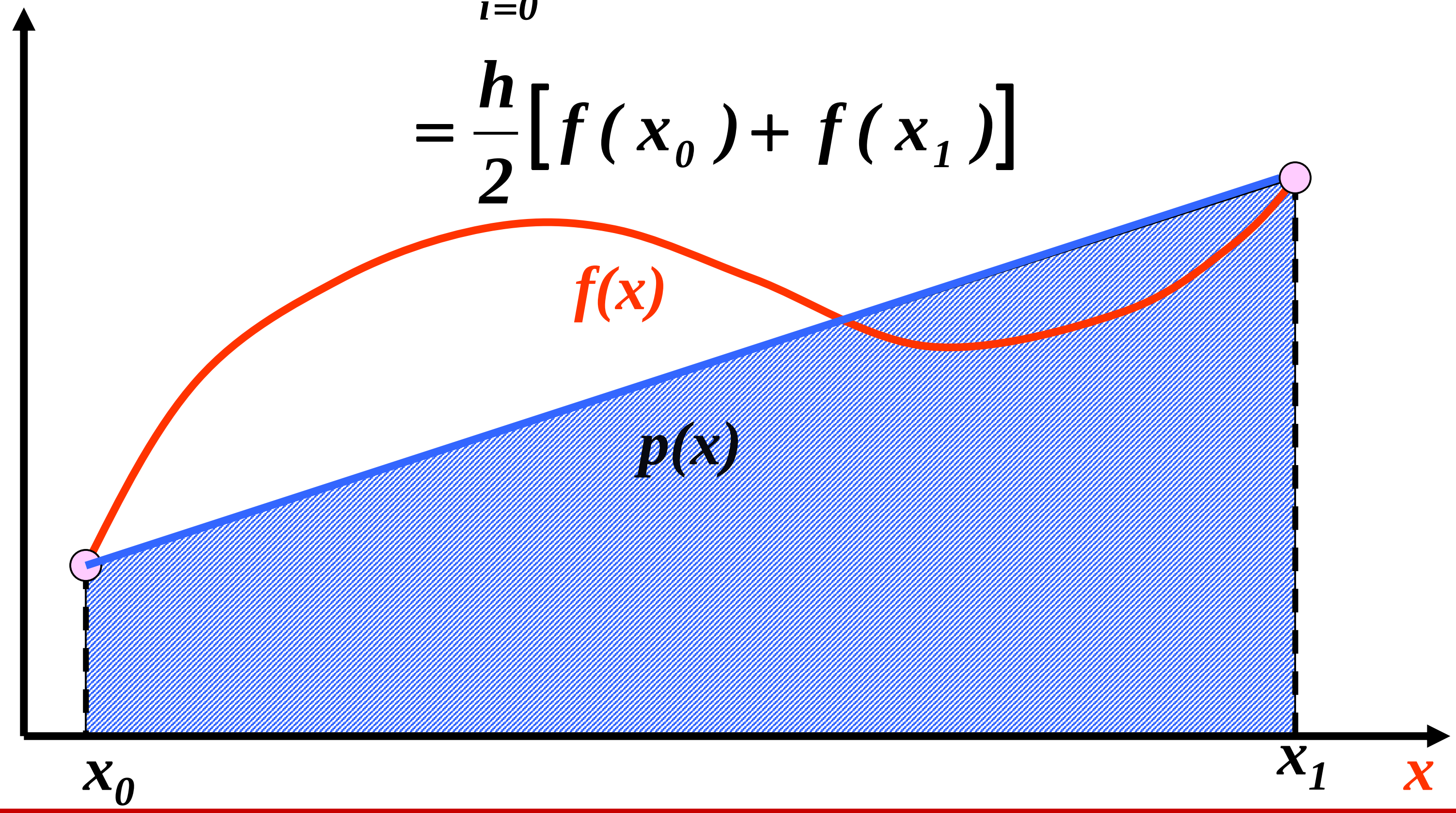
  - Rectangle rule

Trapezoidal rule

Linear approximation of f(x)

$$\int_a^b f(x)dx \approx \sum_{i=0}^{1} c_i f(x_i) = c_0 f(x_0) + c_1 f(x_1)$$

$$= \frac{h}{2}\left[f(x_0) + f(x_1)\right]$$



Trapezoidal rule

Lagrange interpolation polynomial, n = 1

$$p(x) = L_0(x) f(x_0) + L_1(x) f(x_1)$$

$$p(x) = \frac{x - x_1}{x_0 - x_1} f(x_0) + \frac{x - x_0}{x_1 - x_0} f(x_1)$$

Change of variable

nodes $\mathrm{a} = \mathrm{x}_0,\ \mathrm{b} = \mathrm{x}_1,\ \ \mathrm{x} \in [\mathrm{a,b}]\ \ \ \xi \in [0,1]$

$$\xi = \frac{x - a}{b - a},\ \ \mathrm{d}\xi = \frac{\mathrm{dx}}{\mathrm{h}};\ \ h = b - a$$

$$x = a \ \Rightarrow \xi = 0 \qquad\qquad x = b \ \Rightarrow \xi = 1$$

$$p(\xi) = (1 - \xi) f(a) + (\xi) f(b)$$

Trapezoidal rule

Integrating

$$\int_a^b f(x)dx \approx \int_a^b p(x)dx = h\int_0^1 p(\xi)d\xi$$

$$= f(a)h\int_0^1 (1-\xi)d\xi + f(b)h\int_0^1 \xi d\xi$$

$$= f(a)h(\xi - \frac{\xi^2}{2})\Big|_0^1 + f(b)h\frac{\xi^2}{2}\Big|_0^1 = \frac{h}{2}[f(a)+f(b)]$$

Example: Trapezoidal rule

Compute the integral

$$\int_0^4 xe^{2x}dx$$

Exact solution

$$\int_0^4 xe^{2x}dx = \left[\frac{x}{2}e^{2x} - \frac{1}{4}e^{2x}\right]_0^4$$

$$= \frac{1}{4}e^{2x}(2x-1)\Big|_0^1 = 5216.926477$$

Trapezoidal rule

$$I = \int_0^4 xe^{2x}dx \approx \frac{4-0}{2}\left[f(0)+f(4)\right] = 2(0+4e^8) = 23847.66$$

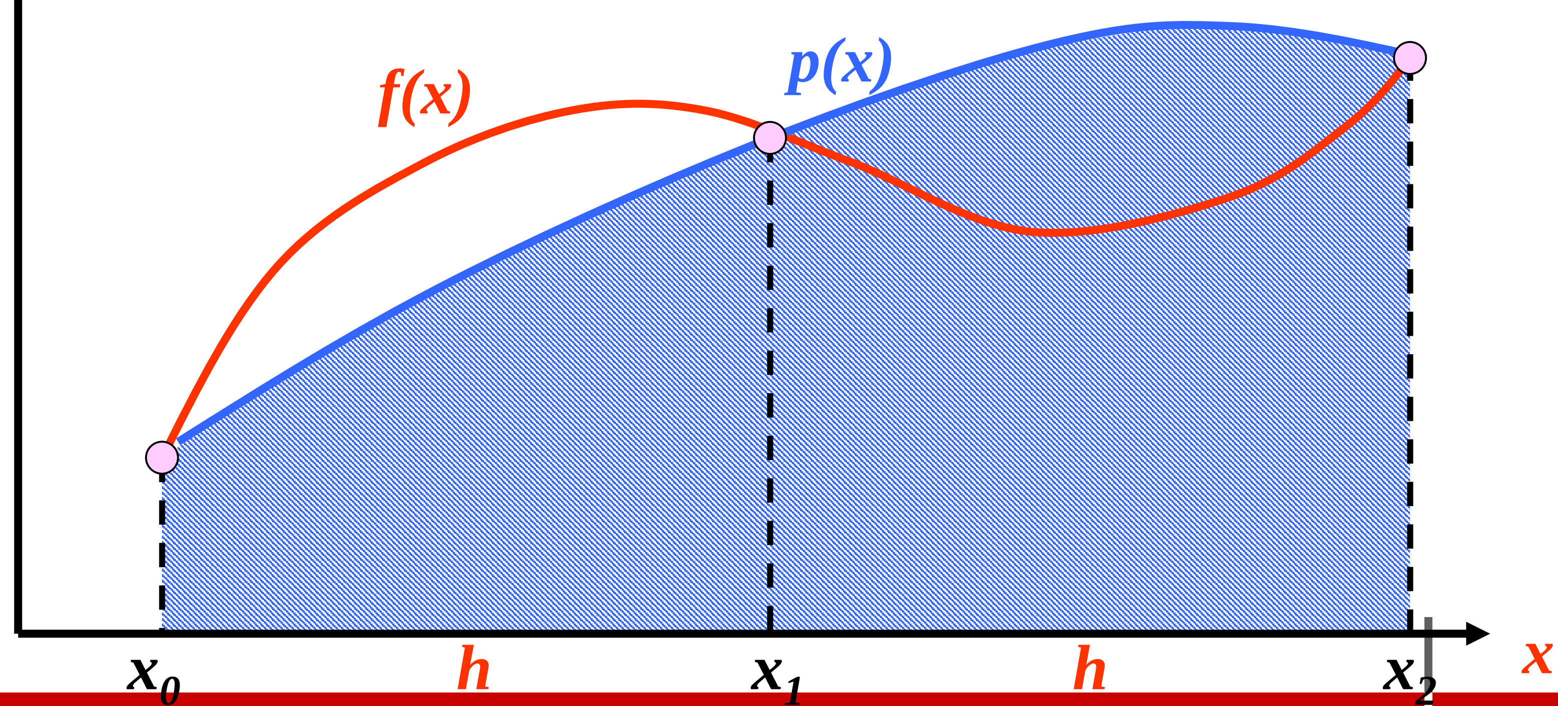
$$\varepsilon = \frac{5216.926 - 23847.66}{5216.926} = -357.12\%$$

Simpson’s 1/3 rule

Approximates the function f (x) with a parabola p(x), n = 2

$$\int_a^b f(x)dx \approx \sum_{i=0}^{2} c_i f(x_i) = c_0 f(x_0) + c_1 f(x_1) + c_2 f(x_2)$$

$$= \frac{h}{3}\left[f(x_0) + 4 f(x_1) + f(x_2) \right]$$



Simpson's 1/3 rule

$$p(x) = \frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)} f(x_0) + \frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)} f(x_1) + \frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)} f(x_2)$$

nodes $x_0 = a,\ x_2 = b,\ x_1 = \frac{a+b}{2}$

$$x \in [a,b] \qquad \xi \in [-1,1] \quad h = \frac{b-a}{2},\ \xi = \frac{x-x_1}{h},\ d\xi = \frac{dx}{h}$$

$$\begin{cases} x = x_0 \Rightarrow \xi = -1 \\ x = x_1 \Rightarrow \xi = 0 \\ x = x_2 \Rightarrow \xi = 1 \end{cases}$$

$$p(\xi) = \frac{\xi(\xi-1)}{2} f(x_0) + (1-\xi^2) f(x_1) + \frac{\xi(\xi+1)}{2} f(x_2)$$

Simpson's 1/3 rule

We integrate the Lagrange quadratic interpolating polynomial

$$\int_a^b f(x)dx \approx h\int_{-1}^{1} p(\xi)d\xi =$$

$$= f(x_0)\frac{h}{2}\int_{-1}^{1}\xi(\xi-1)d\xi + f(x_1)h\int_{-1}^{1}(1-\xi^2)d\xi + f(x_2)\frac{h}{2}\int_{-1}^{1}\xi(\xi+1)d\xi$$

$$= f(x_0)\frac{h}{2}\left(\frac{\xi^3}{3}-\frac{\xi^2}{2}\right)\Bigg|_{-1}^{1} + f(x_1)h\left(\xi-\frac{\xi^3}{3}\right)\Bigg|_{-1}^{1} + f(x_2)\frac{h}{2}\left(\frac{\xi^3}{3}+\frac{\xi^2}{2}\right)\Bigg|_{-1}^{1}$$

Accuracy degree at least 2

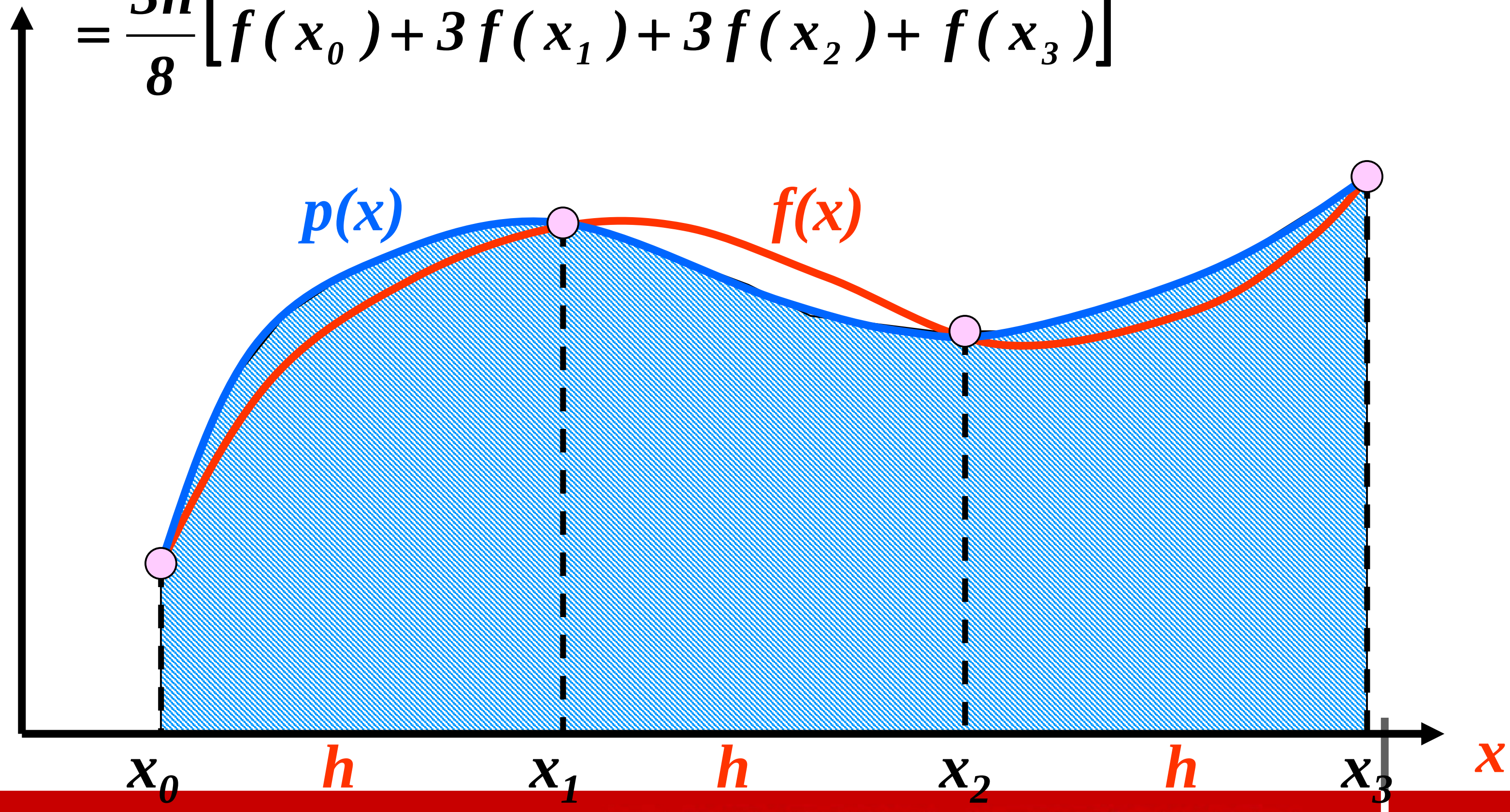
$$\int_a^b f(x)dx = \frac{h}{3}\left[f(x_0)+4f(x_1)+f(x_2)\right]$$

Simpson's 3/8 rule

We approximate with a cubic polynomial, n = 3

$$\int_a^b f(x)dx \approx \sum_{i=0}^{3} c_i f(x_i) = c_0 f(x_0) + c_1 f(x_1) + c_2 f(x_2) + c_3 f(x_3)$$

$$= \frac{3h}{8}\left[f(x_0) + 3f(x_1) + 3f(x_2) + f(x_3)\right]$$



Example: Simpson's rules

Compute the integral $\int_0^4 xe^{2x}dx$

- Simpson's 1/3

$$I = \int_0^4 xe^{2x}dx \approx \frac{h}{3}\left[f(0) + 4f(2) + f(4) \right]$$

$$= \frac{2}{3}\left[0 + 4(2e^4) + 4e^8\right] = 8240.411$$

$$\varepsilon = \frac{5216.926 - 8240.411}{5216.926} = -57.96\%$$

- Simpson's 3/8

$$I = \int_0^4 xe^{2x}dx \approx \frac{3h}{8}\left[f(0) + 3f(\frac{4}{3}) + 3f(\frac{8}{3}) + f(4) \right]$$

$$= \frac{3(4/3)}{8}\left[0 + 3(19.18922) + 3(552.33933) + 11923.832\right] = 6819.209$$

$$\varepsilon = \frac{5216.926 - 6819.209}{5216.926} = -30.71\%$$

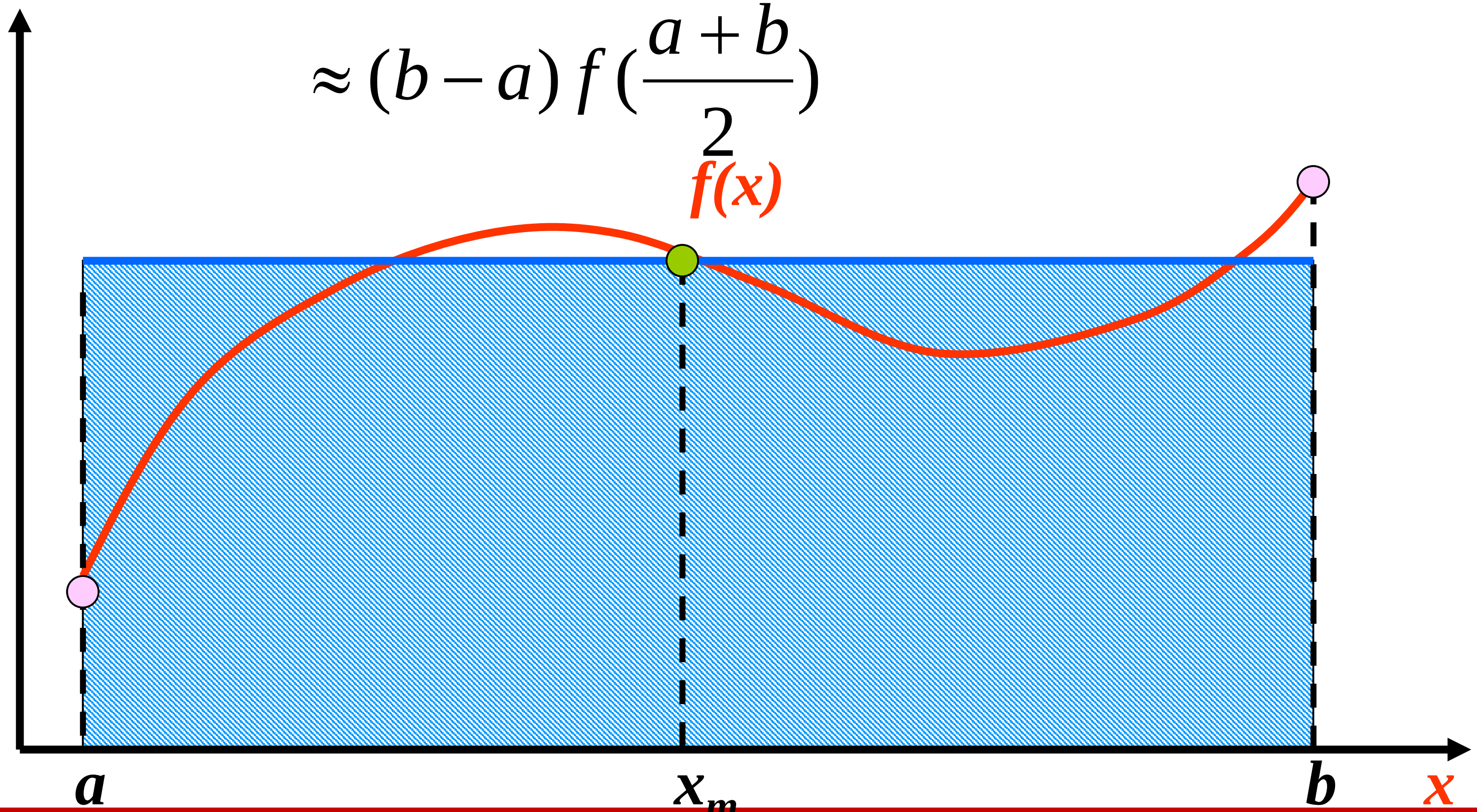
Rectangle rule

Newton-Cotes open formula

n=0

$$\int_a^b f(x)dx \approx (b-a)f(x_m)$$

$$\approx (b-a)f(\frac{a+b}{2})$$



Truncation error r_n

Polynomial Interpolation Error

$$E(x) = f(x) - p_n(x) = \frac{1}{(n+1)!}\Pi_n(x) f^{(n+1)}(\xi)$$

$$dove \quad \Pi_n(x) = \Pi_{i=0}^{n}(x - x_i)$$

Quadrature formula Error

$$\int_a^b f(x)dx = \int_a^b p(x)dx + r_n \cong \int_a^b p(x)dx$$

$$r_n = \frac{1}{(n+1)!}\int_a^b f^{(n+1)}(\xi)\Pi_n(x)dx, \quad f \in C^{n+1}\textbf{([a, b])} \;, \xi \in (a, b)$$

The quadrature formula is exact for construction for polynomials of degree at least n (accuracy degree at least n)

Truncation Error r_n

Since the nodes x_i are equidistant the expression of the rest is simplified

THEOREM

Formula with n even open or closed

If $f \in C^{n+2}([a,b])$ and $\xi \in (a,b)$

$$r_n = \frac{h^{n+3} f^{(n+2)}(\xi)}{(n+2)!} \int_0^n t^2 (t-1)\cdots(t-n)\,dt$$

$\int_0^n$ for closed formulas $\int_{-1}^{n+1}$ for open formula

Accuracy Degree n+1

Formula con n dispari aperte o chiuse

Se $f \in C^{n+1}([a,b])$ **e** $\xi \in (a,b)$

$$r_n = \frac{h^{n+2} f^{(n+1)}(\xi)}{(n+1)!} \int_{\mathbf{0}}^{\mathbf{n}} \mathbf{t}(t-1)\cdots(t-n)\,dt$$

$\int_0^n$ for closed formulas $\int_{-1}^{n+1}$ for open formula

Accuracy Degree n

The Newton-Cotes formulas have accuracy degree **n + 1 if n is EVEN** and **n if n is ODD**

Truncation Error in Trapezoidal rule

$$n = 1; \quad h = b - a$$

$$r_1 = \frac{h^3}{2!} f^{(2)}(\eta) \int_{0}^{1} t(t-1)dx$$

$$\int_0^1 t(t-1)dt = \left[\frac{t^3}{3} - \frac{t^2}{2}\right]_0^1 = -\frac{1}{6}$$

$$r_1 = -\frac{1}{12} h^3 f^{(2)}(\eta)$$

Truncation Errors in Newton-Cotes rules

$$r_1(f) = -\frac{h^3}{12} f''(\xi) \qquad h = b - a$$

Trapezoidal Rule

$$r_2(f) = -\frac{h^5}{90} f^{(4)}(\xi) \qquad h = \frac{b-a}{2}$$

Simpson's 1/3

$$r_3(f) = -\frac{3h^5}{80} f^{(4)}(\xi) \qquad h = \frac{b-a}{3}$$

Simpson's 3/8

$$r_0(f) = \frac{h^3}{3} f''(\xi) \quad h = \frac{b-a}{2}$$

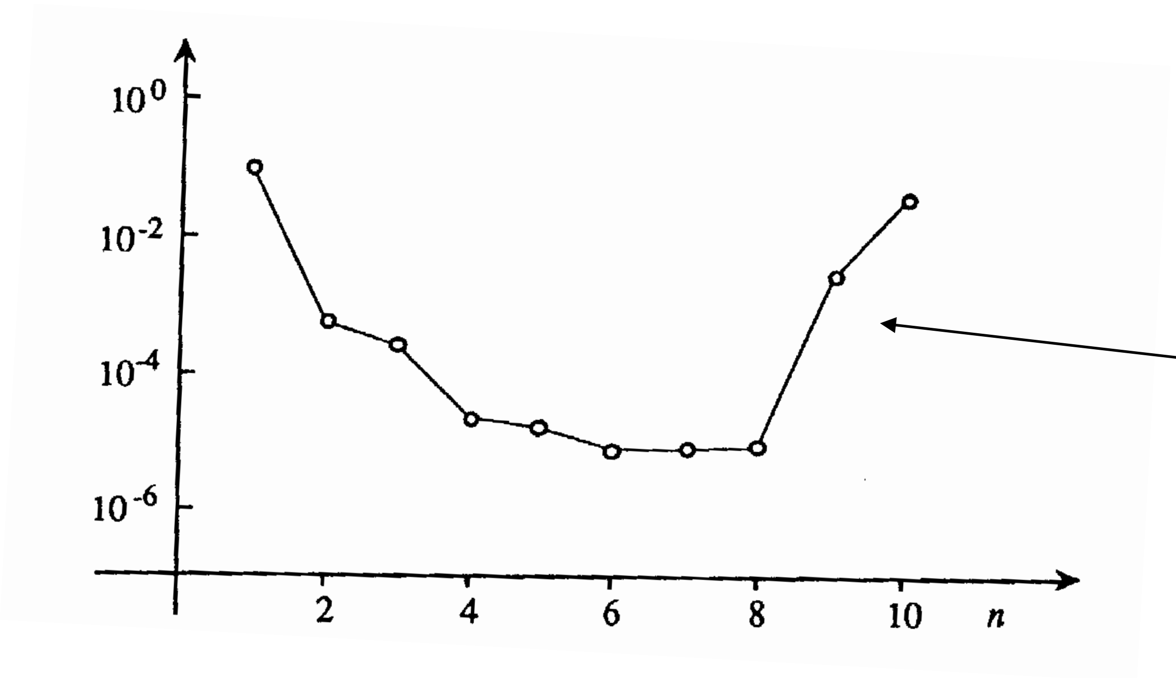
Rectangle Rule

It can be shown that the coefficients or weights w_i of a quadrature formula depend only on n,
but not on the integration interval *[a, b]* or from *f(x)* and can therefore be calculated a priori.

Weights for closed formulas of Newton-Cotes with n + 1 points

n	c_0	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
1	1/2	1/2							
2	1/3	4/3	1/3						
3	3/8	9/8	9/8	3/8					
4	14/45	64/45	24/45	64/45	14/45				
5	95/ 288	375/ 288	250/ 288	250/ 288	375/ 288	95/ 288			
6	41/ 840	216/ 840	27/ 840	272/ 840	27/ 840	216/ 840	41/ 840		
7	751/ 17280	3577/1 7280	1323/ 17280	2989/ 17280	2989/ 17280	1323/ 17280	3577/ 17280	751/ 17280	
8	989/ 28350	5888/2 8350	-928/ 28350	10496/2 8350	-4540/ 28350	10496/ 28350	-928/ 28350	5888/ 28350	989/ 28350

Newton-Cotes quadrature formulas equispaced nodes



Numerical instability of quadrature formulas for n> 8

Relative error in the calculation of the Newton-Cotes formulas for the approximation of

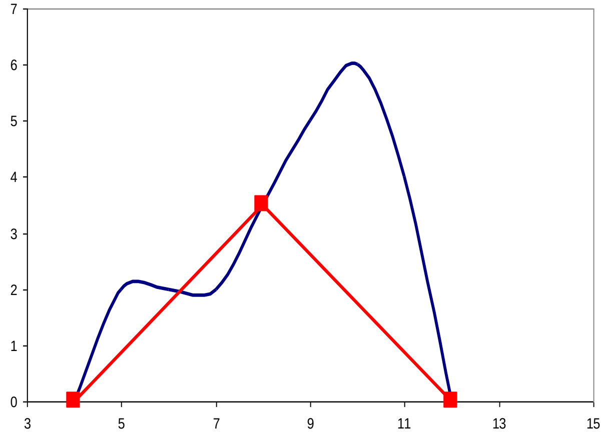
$$I = \int_0^1 e^{-x^2}\,dx$$

To improve the accuracy:

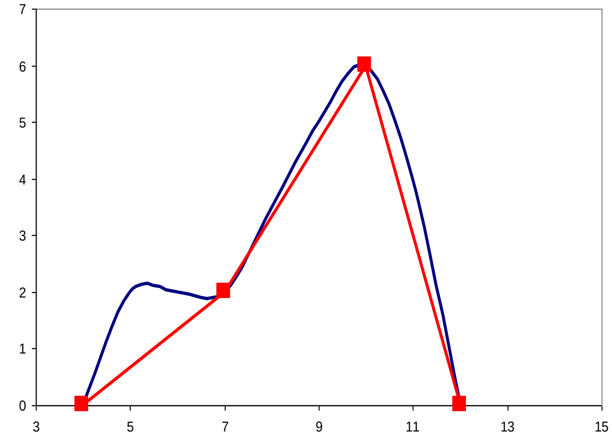
- **composite, integration formulas:**
  - Composite Trapezoidal rule
  - Composite Simpson’s rule
- **Richardson’s extrapolation**
- **Romberg’s integration**

Multiple-application trapezoidal rule

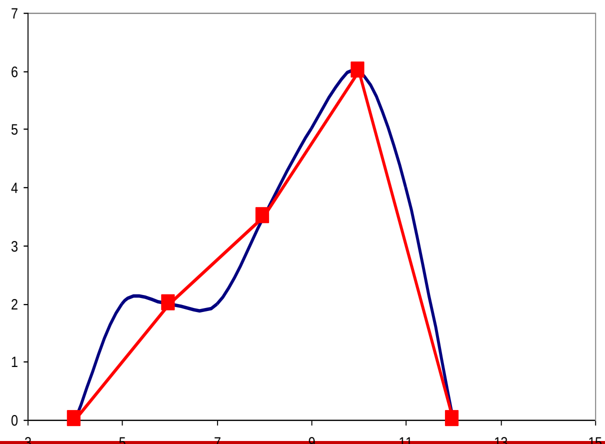
Two subintervals



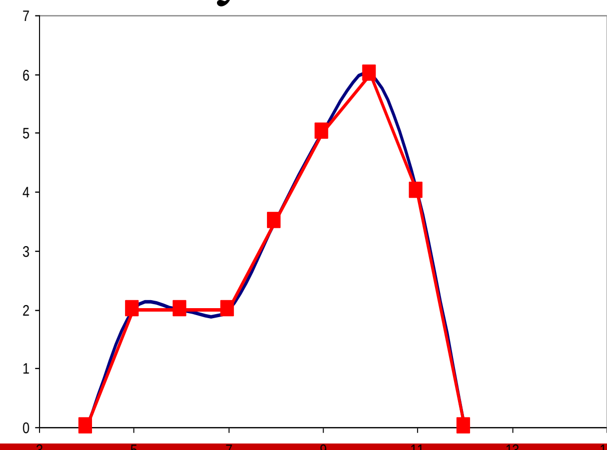
Three subintervals



Four subintervals



…many subintervals



Composite, integration formulas

- Divide the integration interval [a,b] into a number N of segments

$$[x_i, x_{i+1}] \quad i = 0,1,..,N-1$$

- Sum the areas of individual segments to yield the integral for the entire interval

$$\int_a^b f(x)dx = \sum_{i=0}^{N-1} \int_{x_i}^{x_{i+1}} f(x)dx$$

- Apply the quadrature formula to each segment

$$\int_{x_i}^{x_{i+1}} f(x)dx \quad replaced\ with \quad \int_{x_i}^{x_{i+1}} p(x)dx$$

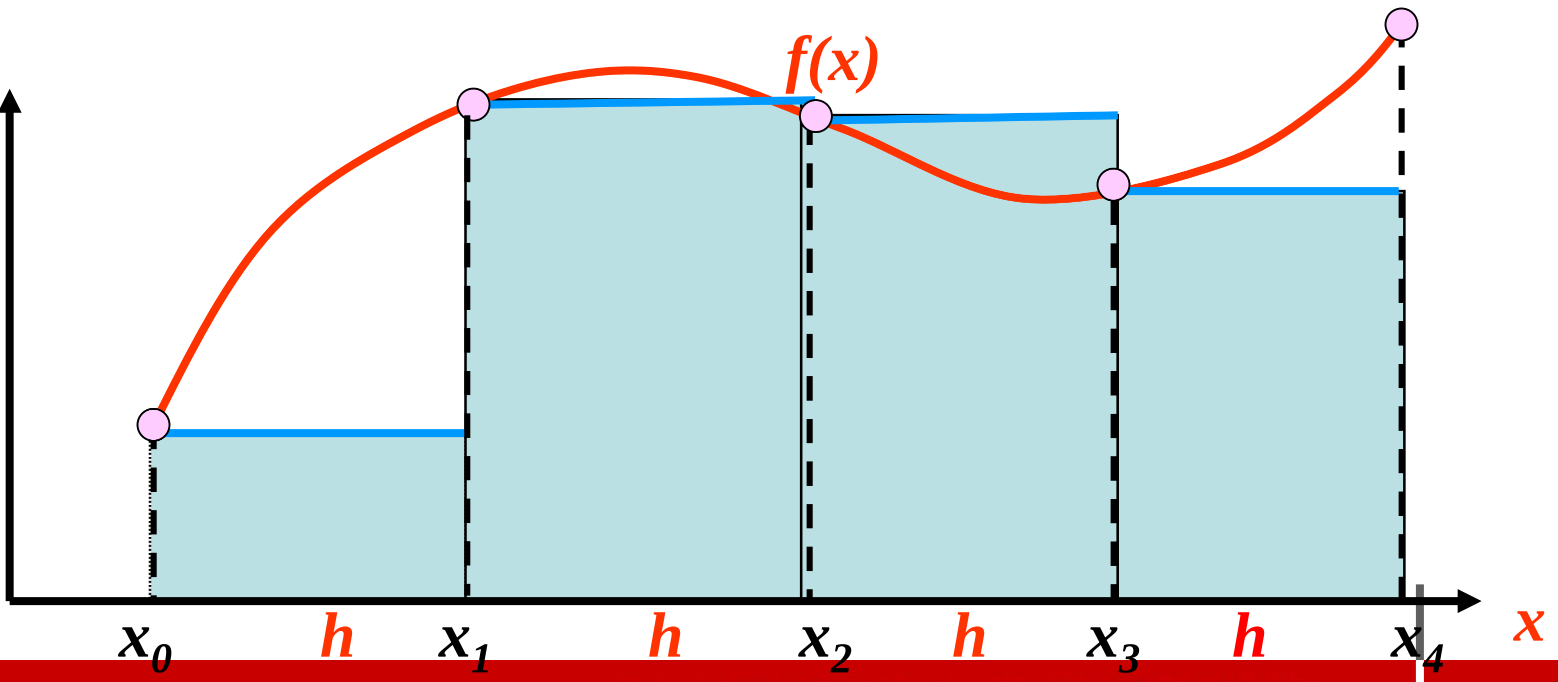
Composite Rectangles Formula

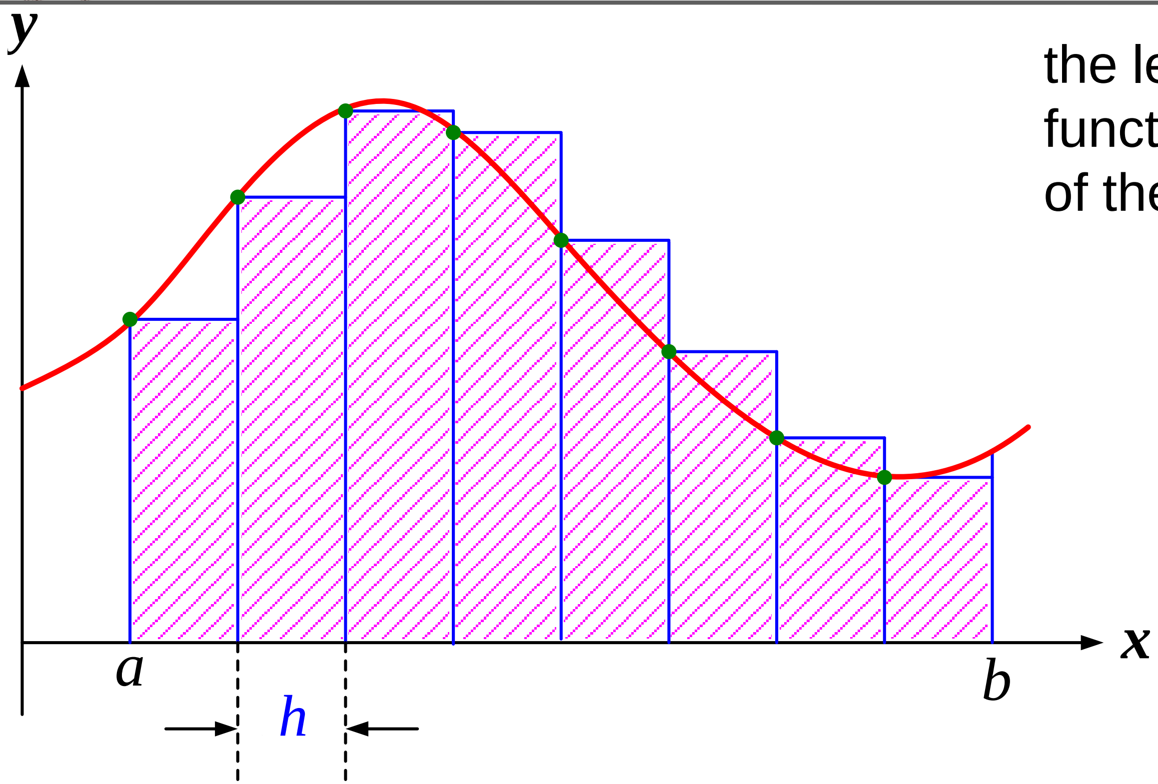
$$\int_a^b f(x)dx = \int_{x_0}^{x_1} f(x)dx + \int_{x_1}^{x_2} f(x)dx + \cdots\cdots + \int_{x_{N-1}}^{x_N} f(x)dx$$

$$= h\left[f(x_0) + f(x_1) + \cdots + f(x_i) + \cdots + f(x_{N-1})\right] + r$$

N
Subintervals of equal width

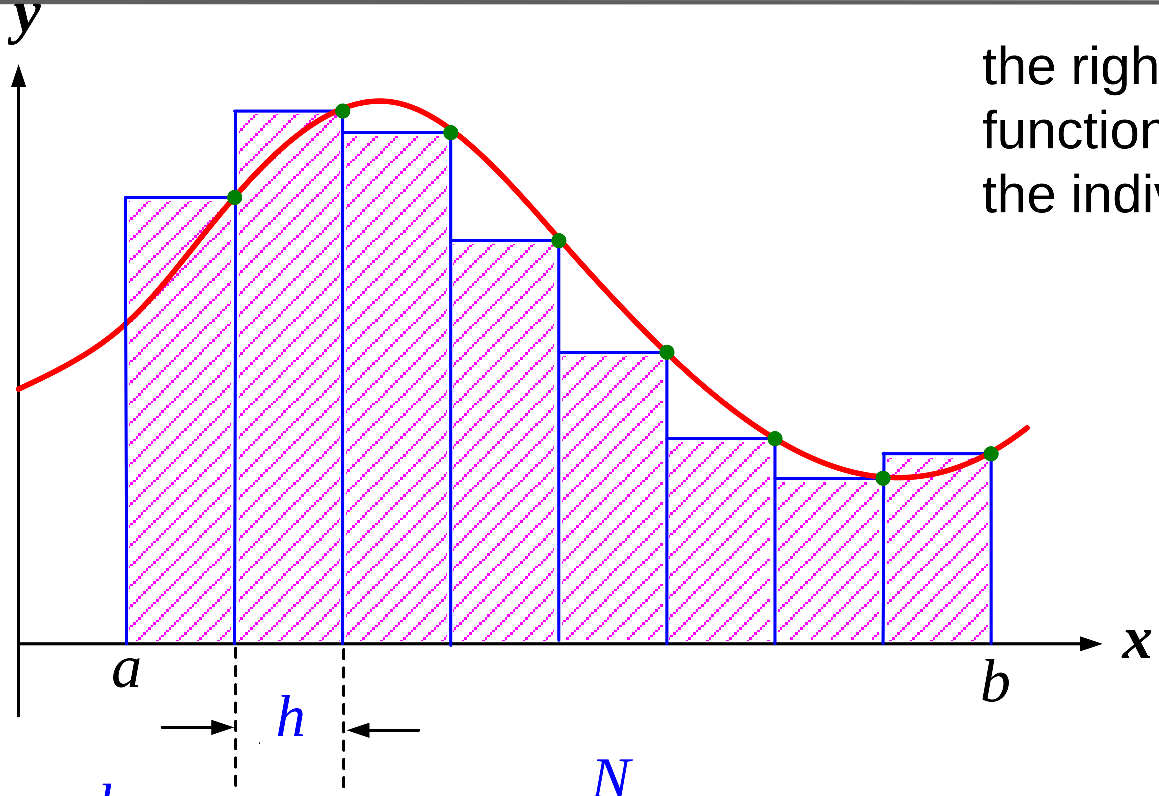
$$h = \frac{b-a}{N}$$





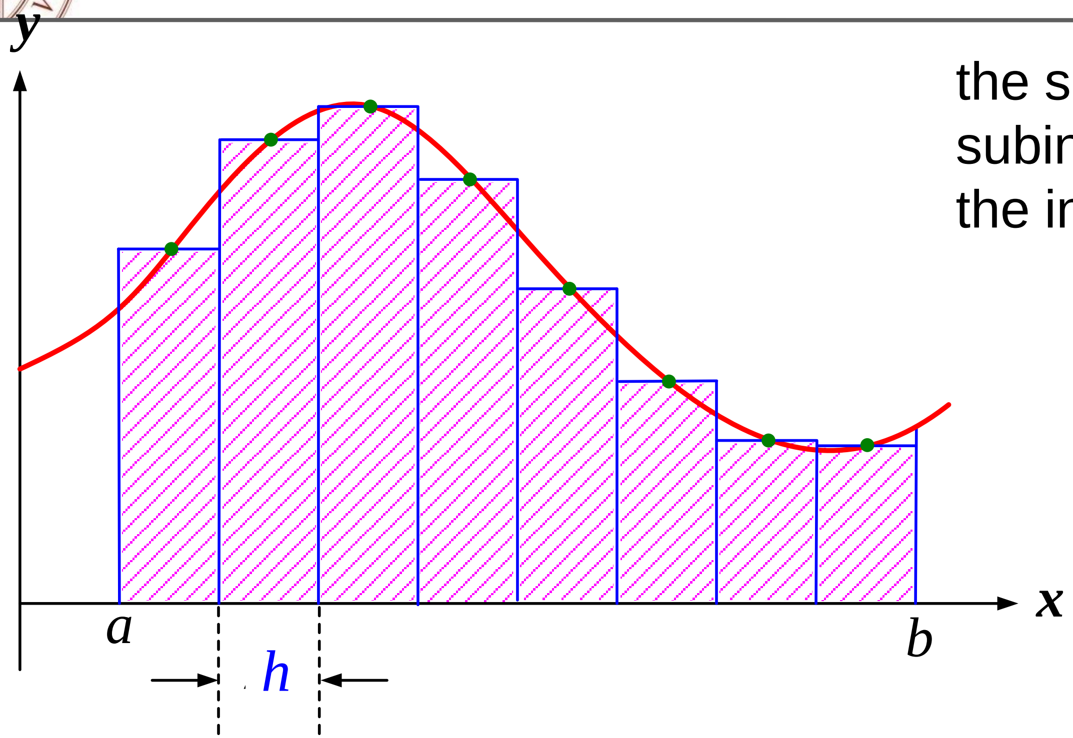
the left-side sample of the function is used as the height of the individual rectangle.

$$\int_a^b f(x)\,dx \approx \sum_{i=0}^{N-1} f(x_i)\,h$$



the right-side sample of the function is used as the height of the individual rectangle.

$$\int_a^b f(x)\,dx \approx \sum_{i=1}^{N} f(x_i)h$$



the sample at the middle of the subinterval is used as the height of the individual rectangle.

$$\int_a^b f(x)\,dx \approx \sum_{i=1}^{N} f\left(\overline{x_i}\right)h$$

Composite Trapezoidal rule

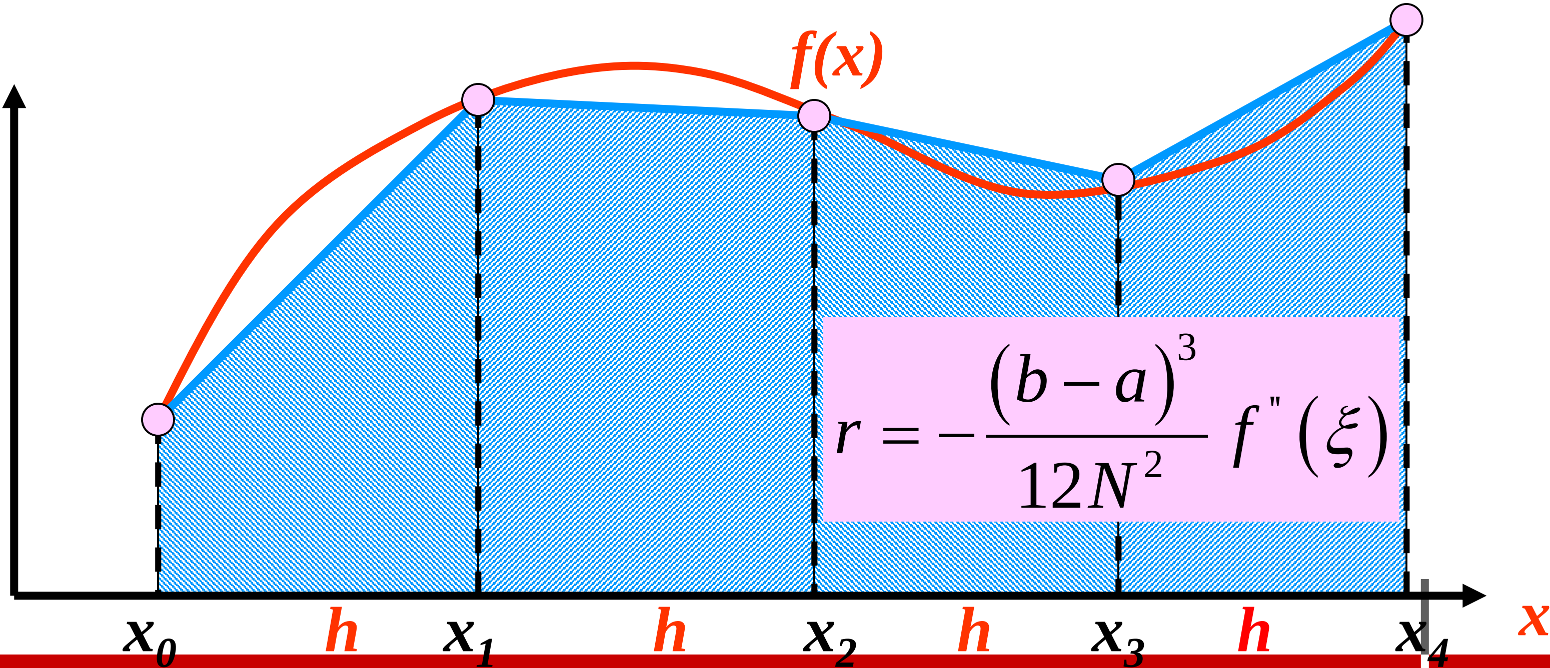
$$\int_a^b f(x)dx = \int_{x_0}^{x_1} f(x)dx + \int_{x_1}^{x_2} f(x)dx + \cdots\cdots + \int_{x_{N-1}}^{x_N} f(x)dx$$

$$= \frac{h}{2}\left[f(x_0) + f(x_1)\right] + \frac{h}{2}\left[f(x_1) + f(x_2)\right] + \cdots + \frac{h}{2}\left[f(x_{N-1}) + f(x_N)\right] + r$$

$$= \frac{h}{2}\left[f(x_0) + 2f(x_1) + \cdots + 2f(x_i) + \cdots + 2f(x_{N-1}) + f(x_N)\right] + r$$

N
Subintervals
of equal width

$$h = \frac{b-a}{N}$$



Composite Trapezoidal rule

Compute the integral $I = \int_0^4 xe^{2x}dx$

$$N = 1, h = 4 \Rightarrow I = \frac{h}{2}\left[f(0) + f(4)\right] = 23847.66 \qquad \varepsilon = -357.12\%$$

$$N = 2, h = 2 \Rightarrow I = \frac{h}{2}\left[f(0) + 2f(2) + f(4)\right] = 12142.23 \qquad \varepsilon = -132.75\%$$

$$N = 4, h = 1 \Rightarrow I = \frac{h}{2}\left[f(0) + 2f(1) + 2f(2) + 2f(3) + f(4)\right] = 7288.79 \qquad \varepsilon = -39.71\%$$

$$N = 8, h = 0.5 \Rightarrow I = \frac{h}{2}\left[f(0) + 2f(0.5) + 2f(1) + 2f(1.5) + 2f(2) + 2f(2.5) + 2f(3) + 2f(3.5) + f(4)\right] = 5764.76 \qquad \varepsilon = -10.50\%$$

$$N = 16, h = 0.25 \Rightarrow I = \frac{h}{2}\left[f(0) + 2f(0.25) + 2f(0.5) + \cdots + 2f(3.5) + 2f(3.75) + f(4)\right] = 5355.95 \qquad \varepsilon = -2.66\%$$

Composite Trapezoidal rule with unequal subintervals

Compute the integral $$I = \int_0^4 xe^{2x}dx$$

$h_1 = 2$, $h_2 = 1$, $h_3 = 0.5$, $h_4 = 0.5$

$$I = \int_0^2 f(x)dx + \int_2^3 f(x)dx + \int_3^{3.5} f(x)dx + \int_{3.5}^4 f(x)dx$$

$$= \frac{h_1}{2}\left[f(0) + f(2)\right] + \frac{h_2}{2}\left[f(2) + f(3)\right]$$

$$+ \frac{h_3}{2}\left[f(3) + f(3.5)\right] + \frac{h_4}{2}\left[f(3.5) + f(4)\right]$$

$$= \frac{2}{2}\left[0 + 2e^4\right] + \frac{1}{2}\left[2e^4 + 3e^6\right] + \frac{0.5}{2}\left[3e^6 + 3.5e^7\right]$$

$$+ \frac{0.5}{2}\left[3.5e^7 + 4e^8\right] = 5971.58 \qquad \Rightarrow \varepsilon = -14.45\%$$

Global Truncation Error of the Trapezoidal rule

- Divide the integration interval [a,b] into a number N of segments $[x_i, x_{i+1}] \quad i = 0,1,..,N-1$

- The global truncation error is given by summing the individual errors for each subinterval

$$\int_a^b f(x)dx = \sum_{i=0}^{N-1} \int_{x_i}^{x_{i+1}} f(x)dx = \sum_{i=0}^{N-1} [\frac{h}{2}(f(x_i) + f(x_{i+1}))] - \frac{1}{12} h^3 f^{(2)}(\eta_i)$$

Local truncation error

global truncation error

$$R = -\frac{(b-a)^3}{12N^3} \sum_{i=0}^{N-1} f^{(2)}(\eta_i)$$

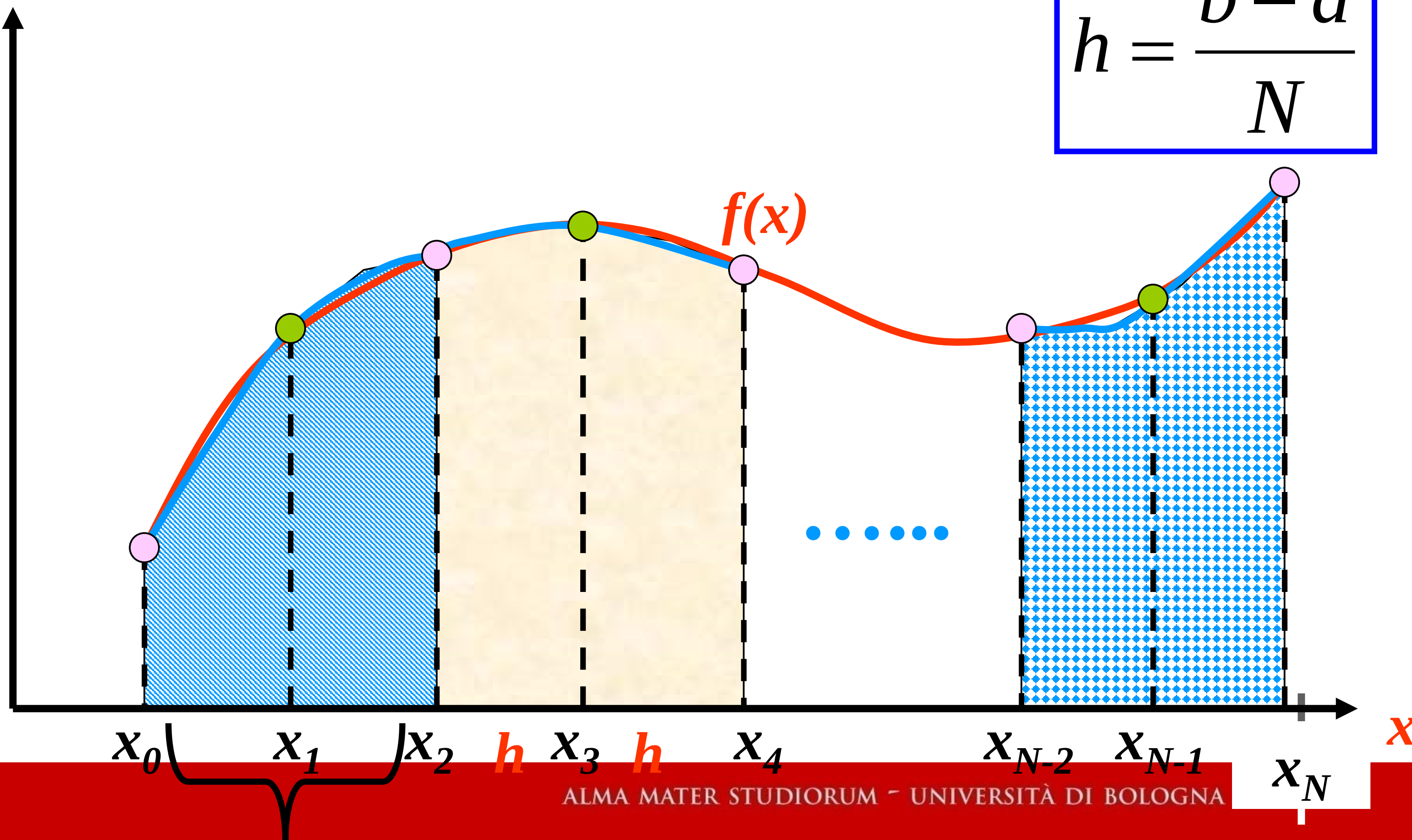
$$h = \frac{b-a}{N}$$

Composite Simpson's rule

Piecewise quadratic interpolation

Subdivision of [a, b] in k = N / 2 subintervals

$$h = \frac{b-a}{N}$$



Composite Simpson's rule

$$\int_a^b f(x)dx = \int_{x_0}^{x_2} f(x)dx + \int_{x_2}^{x_4} f(x)dx + \cdots + \int_{x_{N-2}}^{x_N} f(x)dx$$

$$= \frac{h}{3}\left[f(x_0) + 4f(x_1) + f(x_2)\right] + \frac{h}{3}\left[f(x_2) + 4f(x_3) + f(x_4)\right]$$

$$+ \cdots + \frac{h}{3}\left[f(x_{N-2}) + 4f(x_{N-1}) + f(x_N)\right] + r$$

$$= \frac{h}{3}\left[f(x_0) + 4f(x_1) + 2f(x_2) + 4f(x_3) + 2f(x_4) + \cdots \right.$$

$$+ 4f(x_{2i-1}) + 2f(x_{2i}) + 4f(x_{2i+1}) + \cdots$$

$$\left. + 2f(x_{N-2}) + 4f(x_{N-1}) + f(x_N)\right] + r$$

$$r = -\frac{(b-a)^5}{2880N^4} f^{(4)}(\xi)$$

Composite Simpson's rule

Compute the integral $I = \int_0^4 x e^{2x} dx$

n = 2, h = 2

$$I = \frac{h}{3}\left[f(0) + 4f(2) + f(4)\right]$$
$$= \frac{2}{3}\left[0 + 4(2e^4) + 4e^8\right] = 8240.411 \quad \Rightarrow \varepsilon = -57.96\%$$

n = 4, h = 1

$$I = \frac{h}{3}\left[f(0) + 4f(1) + 2f(2) + 4f(3) + f(4)\right]$$
$$= \frac{1}{3}\left[0 + 4(e^2) + 2(2e^4) + 4(3e^6) + 4e^8\right]$$
$$= 5670.975 \quad \Rightarrow \varepsilon = -8.70\%$$

Composite Simpson's rule with unequal subintervals

Compute the integral

$$I = \int_0^4 x e^{2x} dx$$

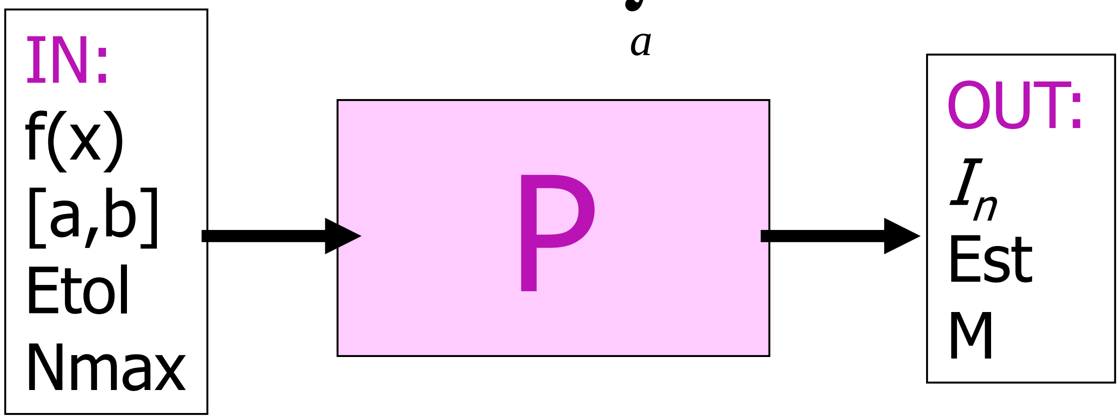
$h_1 = 1.5$, $h_2 = 0.5$

$$
\begin{aligned}
I &= \int_0^3 f(x)dx + \int_3^4 f(x)dx \\
&= \frac{h_1}{3}\left[f(0) + 4f(1.5) + f(3) \right] + \frac{h_2}{3}\left[f(3) + 4f(3.5) + f(4) \right] \\
&= \frac{1.5}{3}\left[0 + 4(1.5e^3) + 3e^6 \right] + \frac{0.5}{3}\left[3e^6 + 4(3.5e^7) + 4e^8 \right] \\
&= 5413.23 \qquad \Rightarrow \varepsilon = -3.76\%
\end{aligned}
$$

Automatic Quadrature

The program P calculates the integral value

$$I(f;a,b)=\int_a^b f(x)dx$$



with an estimated error Est < Etol and with a number of function evaluations M < Nmax

If this is not possible, program P is interrupted.

P consists in:

1. A succession of quadrature formulas that involve a growing number of evaluations of f(x)
2. We need a criterion to determine I_n
3. We need a criterion to determine an automatic estimate of the truncation error Est

Automatic quadrature schemes P:

- Romberg method (non-adaptive)
- Composite Simpson’s adaptive method (implemented in MATLAB function quad())

Composite Simpson's **adaptive** method

Adaptive quadrature methods adjust the step size so that small intervals are used in regions of rapid variations and larger intervals are used where the function changes gradually.

- Two composite quadrature formulas with steps h and h / 2 are applied to an interval = [a, b] (S(h) and S(h/2))
- **Estimation of the truncation error with Richardson's extrapolation;**
- **If the error is less than or equal to the tolerance** S(h/2) or a combination of the two approximations is taken as the value of I on that interval,
- **If the error is larger than the tolerance**, the interval [a, b] is divided in half and the process is repeated on each of the two sub-intervals.

DEMO

Richardson's Extrapolation

Richardson's Extrapolation method uses two estimates of an integral to compute a third, more accurate approximation.

Make two separate estimates using step sizes h and h/2:

I_N, I_{2N} *approximations*

$$r_N = I - I_N = \frac{\delta_N}{N^s}, \quad r_{2N} = I - I_{2N} = \frac{\delta_{2N}}{2^s N^s},$$

where δ_N *and* δ_{2N} *differs for* $f^{(s)}(\xi)$

$$r_N = -\frac{(b-a)^5}{2880 N^4} f^{(4)}(\xi)$$

$$r_{2N} = -\frac{(b-a)^5}{2880 N^4 2^4} f^{(4)}(\xi)$$

Under the assumption that $f^{(s)}(\xi)$ varies little by varying ξ, we have : $\delta_N \approx \delta_{2N} = \delta$ then $r_N - r_{2N}$ is given by

$$I_{2N} - I_N \approx \frac{\delta}{2^s N^s}(2^s - 1)$$

Richardson's Extrapolation

Estimate the residual

$$I_{2N} - I_N \approx \frac{\delta}{2^s N^s}(2^s - 1)$$

$$\frac{I_{2N} - I_N}{2^s - 1} \approx \frac{\delta}{2^s N^s} = r_{2N}$$

We can proceed with subsequent doublings of N until

$$\left| \frac{I_{2N} - I_N}{2^s - 1} \right| \leq Etol$$

The approximation of the integral will be:

$$I = I_{2N} + r_{2N} \simeq I_{2N} + \frac{I_{2N} - I_N}{2^s - 1}$$

Compute the integral $I = \int_0^1 e^{-x^2} dx$

By applying the Trapezoidal's rule (s=2), Etol=0.5×10^{-3}

N	I_N
2	0.7313700
4	0.7429838
8	0.7458653
16	0.7465825

$$\left| \frac{I_{2N} - I_N}{2^s - 1} \right| \le Etol$$

Final value (I_8 exact) = 0.7468214

Estimated Error (Est) about 0.273×10^{-5}

Compute the integral $I = \int_0^1 e^{-x^2} dx$

By applying the Simpson's rule (s=4), Etol=0.5x10^{-3}

N	I_N
2	0.7468553
4	0.7468255

$$\left| \frac{I_{2N} - I_N}{2^s - 1} \right| \leq Etol$$

Final value (I_2 exact) =0.7468235

Estimated Error (Est) about 0.633x10^{-6}

Gauss quadrature formulas

- **Newton-Cotes Formulas**

  -Evaluate functions at n+1 equispaced nodes

  -Degree of precision n (n odd) or n+1 (n even)

  (n degree of the interpolating polynomial)

What are the coefficients c_i and the nodes x_i such that the quadrature formula

$$\int_a^b f(x)dx \approx \sum_{i=0}^{n} c_i f(x_i)$$

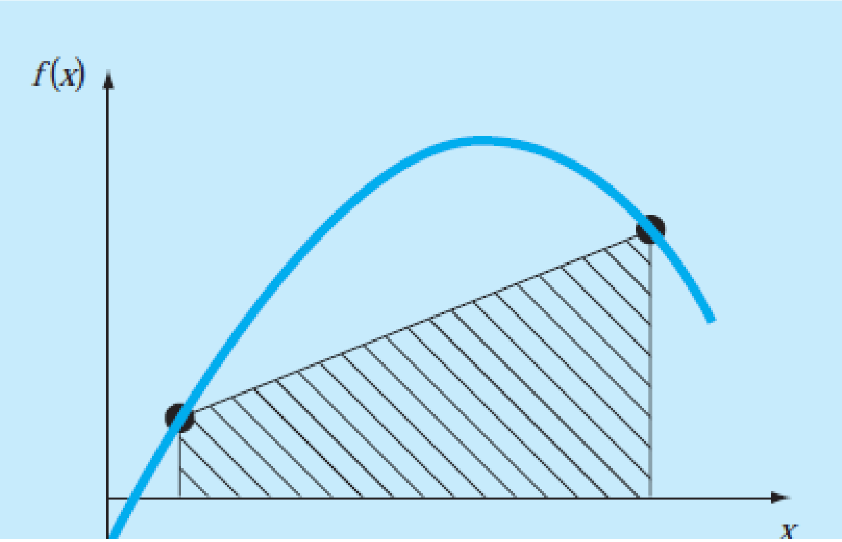
is exact for a polynomial with the highest possible degree?

- **Gauss Formulas**

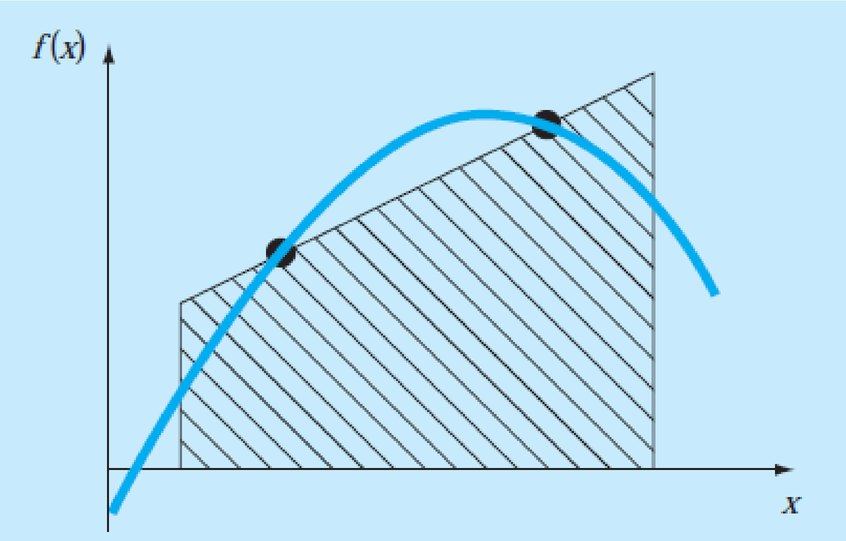
  The nodes $x_0, x_1, ..$ are not fixed, nodes and coefficients are derived in such a way to maximize the degree of precision

  Degree of Precision $\leq 2n+1$ **(n+1 number of nodes)**

Gauss Quadrature Formula



Graphical depiction of the trapezoidal rule as the area under the straight line joining fixed end points.

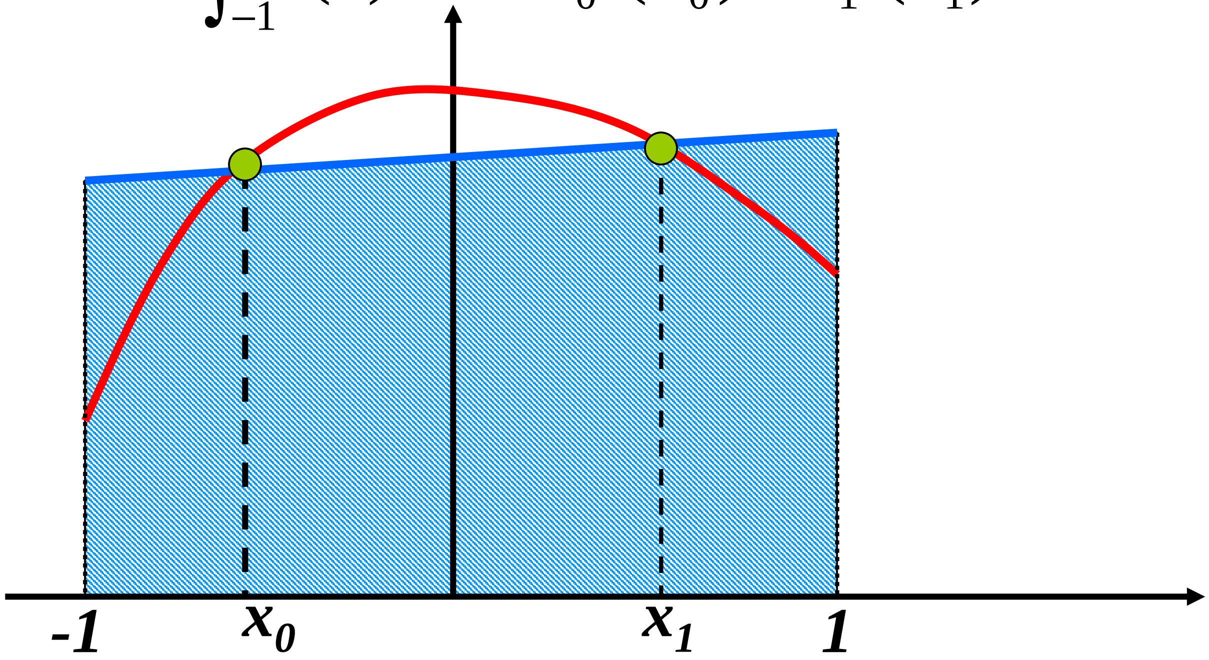


An improved integral estimate obtained by taking the area under the straight line passing through two intermediate points.

Method 1: Undetermined Coefficients

Gauss Quadrature Formula in [-1, 1]:

$$n = 1: \quad \int_{-1}^{1} f(x)dx \approx c_0 f(x_0) + c_1 f(x_1)$$



Derive **nodes** and **coefficients** *(c_0, c_1, x_0, x_1) in order to maximize the degree of precision*, by imposing that the formula is exact for $f = x^0, x^1, x^2, x^3$

Method 1: Undetermined Coefficients

Gauss Quadrature Formula in [-1, 1]:

$$n = 1: \quad \int_{-1}^{1} f(x)dx = c_0 f(x_0) + c_1 f(x_1)$$

Derive nodes and coefficients by imposing that the formula is exact for $f = x^0, x^1, x^2, x^3$

$$\begin{cases} f = 1 & \Rightarrow \int_{-1}^{1} 1dx = 2 = c_0 + c_1 \\ f = x & \Rightarrow \int_{-1}^{1} xdx = 0 = c_0 x_0 + c_1 x_1 \\ f = x^2 & \Rightarrow \int_{-1}^{1} x^2 dx = \frac{2}{3} = c_0 x_0^2 + c_1 x_1^2 \\ f = x^3 & \Rightarrow \int_{-1}^{1} x^3 dx = 0 = c_0 x_0^3 + c_1 x_1^3 \end{cases} \Rightarrow \begin{cases} c_0 = 1 \\ c_1 = 1 \\ x_0 = \frac{-1}{\sqrt{3}} \\ x_1 = \frac{1}{\sqrt{3}} \end{cases}$$

Nonlinear System of 4 eqs and 4 unknowns

$$I = \int_{-1}^{1} f(x)dx = f\left(-\frac{1}{\sqrt{3}}\right) + f\left(\frac{1}{\sqrt{3}}\right)$$

Degree of precision 2n+1 = 3

Method 2: Two-Point Gauss-Legendre Formula

Gauss Quadrature Formula in [-1, 1]

$$\mathrm{n}=1: \quad \int_{-1}^{1} \mathrm{f}(\mathrm{x})\mathrm{dx} = \mathrm{c}_0 \mathrm{f}(\mathrm{x}_0) + \mathrm{c}_1 \mathrm{f}(\mathrm{x}_1)$$

Alternatively, the nodes of a Gauss formula are obtained using orthogonal polynomials.

The **nodes** are the roots of orthogonal polynomials with respect to appropriate weight functions w(x) in [a,b]:

$$\int_a^b w(x) f(x) dx \approx \sum_{i=0}^{n} c_i f(x_i)$$

Gauss-Legendre polynomials:

$$\mathrm{n}=1: \quad \text{interval } [-1,1];\ w(x)=1$$

$$p_0(x)=1; \quad p_1(x)=x; \quad p_2(x)=\frac{1}{2}(3x^2-1),$$

Method 2: Two-Point Gauss-Legendre Formula

$$\int_a^b w(x) f(x) dx \approx \sum_{i=0}^{n} c_i f(x_i)$$

Gauss-Legendre roots in [-1,1]: $x_0 = -1/\sqrt{3}, x_1 = 1/\sqrt{3},$

$$\int_{-1}^{1} \mathrm{f(x)dx} = \mathrm{c_0 f}(-1/\sqrt{3}) + \mathrm{c_1 f}(1/\sqrt{3})$$

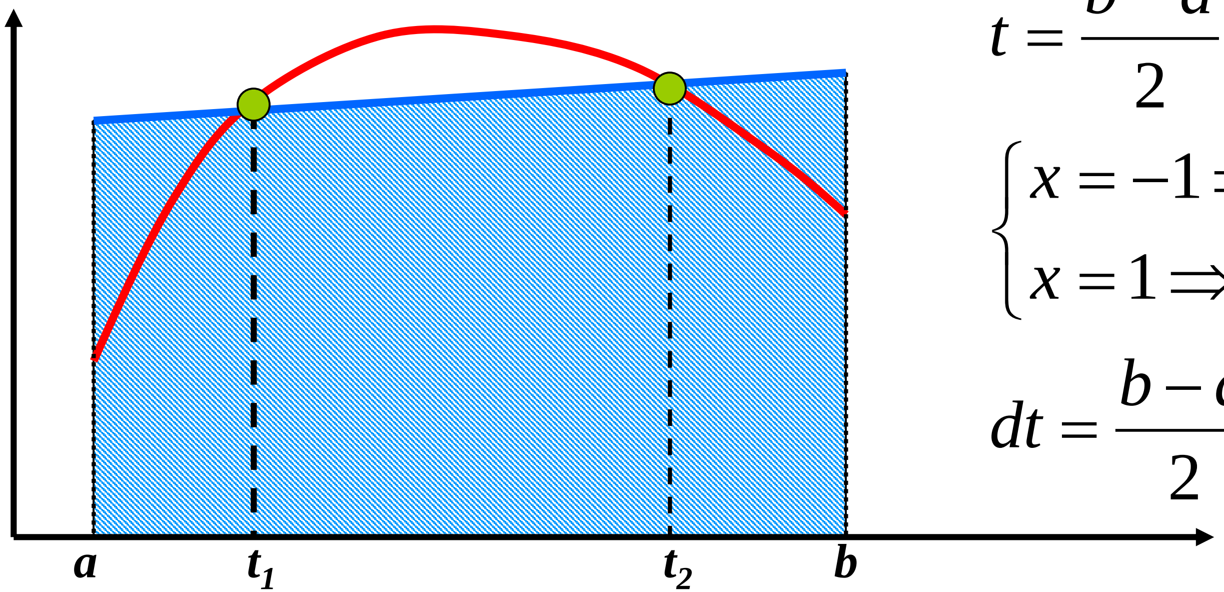
- Determine c_0 and c_1 so that the formula is exact for polynomials of degree < n + 1 (by solving a 2x2 linear system)

$$I = \int_{-1}^{1} f(x) dx = f(-\frac{1}{\sqrt{3}}) + f(\frac{1}{\sqrt{3}})$$

Degree of precision 2n+1=3

Method 2: Two-Point Gauss-Legendre in [a, b]

Change of variable from t in [a,b] to x in [-1,1]



$$t = \frac{b-a}{2}x + \frac{b+a}{2}$$

$$\begin{cases} x = -1 \Rightarrow t = a \\ x = 1 \Rightarrow \quad t = b \end{cases}$$

$$dt = \frac{b-a}{2}dx$$

$$\int_a^b f(t)dt = \int_{-1}^{1} f\left(\frac{b-a}{2}x + \frac{b+a}{2}\right)\left(\frac{b-a}{2}\right)dx = \int_{-1}^{1} g(x)dx$$

Example

- Compute $$I=\int_0^4 te^{2t}dt = 5216.926477$$
- Change of variable

$$t=\frac{b-a}{2}x+\frac{b+a}{2}=2x+2;\quad dt=2dx$$

$$I=\int_0^4 te^{2t}dt=\int_{-1}^{1}(4x+4)e^{4x+4}dx=\int_{-1}^{1}f(x)dx$$

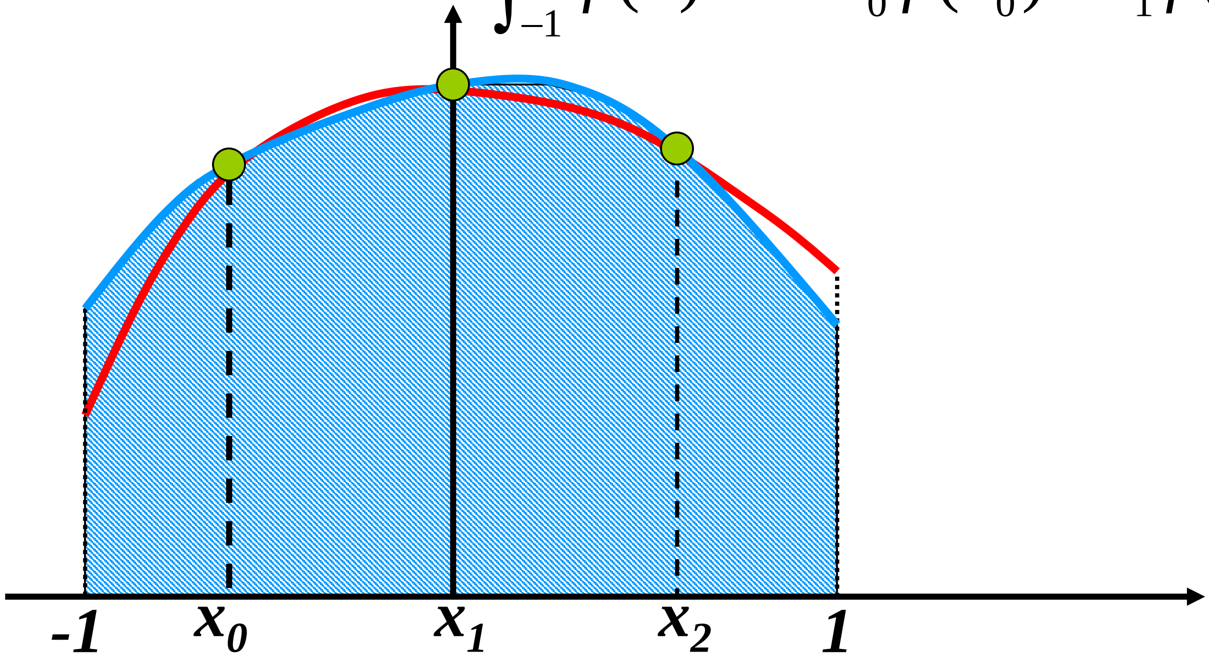
- **Gauss 2 points formula**

$$I=\int_{-1}^{1}f(x)dx=f(\frac{-1}{\sqrt{3}})+f(\frac{1}{\sqrt{3}})=(4-\frac{4}{\sqrt{3}})e^{4-\frac{4}{\sqrt{3}}}+(4+\frac{4}{\sqrt{3}})e^{4+\frac{4}{\sqrt{3}}}$$
$$=9.167657324+3468.376279=3477.543936\qquad(\varepsilon=33.34\%)$$

Method 1: Undetermined Coefficients

Gauss Quadrature Formula in [-1, 1]:

$$n = 2: \quad \int_{-1}^{1} f(x)dx = c_0 f(x_0) + c_1 f(x_1) + c_2 f(x_2)$$



Compute ***(c_0, c_1, c_2, x_0, x_1, x_2)*** in order to maximize the degree of precision, that is, we impose that the integral is exact for $f(x) = x^0, x^1, x^2, x^3, x^4, x^5$

Method 1: Undetermined Coefficients

Gauss Quadrature Formula in [-1, 1]:

$$n = 2: \quad \int_{-1}^{1} f(x)dx = c_0 f(x_0) + c_1 f(x_1) + c_2 f(x_2)$$

$$\begin{cases} f = 1 & \Rightarrow \int_{-1}^{1} 1dx = 2 = c_0 + c_1 + c_2 \\ f = x & \Rightarrow \int_{-1}^{1} xdx = 0 = c_0x_0 + c_1x_1 + c_2x_2 \\ f = x^2 & \Rightarrow \int_{-1}^{1} x^2dx = \dfrac{2}{3} = c_0x_0^2 + c_1x_1^2 + c_2x_2^2 \\ f = x^3 & \Rightarrow \int_{-1}^{1} x^3dx = 0 = c_0x_0^3 + c_1x_1^3 + c_2x_2^3 \\ f = x^4 & \Rightarrow \int_{-1}^{1} x^4dx = 0 = c_0x_0^3 + c_1x_1^3 + c_2x_2^3 \\ f = x^5 & \Rightarrow \int_{-1}^{1} x^5dx = 0 = c_0x_0^5 + c_1x_1^5 + c_2x_2^5 \end{cases} \Rightarrow \begin{cases} c_0 = 5/9 \\ c_1 = 8/9 \\ c_2 = 5/9 \\ x_0 = -\sqrt{3/5} \\ x_1 = 0 \\ x_2 = \sqrt{3/5} \end{cases}$$

Nonlinear System 6x6

$$I = \int_{-1}^{1} f(x)dx = \frac{5}{9} f\left(-\sqrt{\frac{3}{5}}\right) + \frac{8}{9} f(0) + \frac{5}{9} f\left(\sqrt{\frac{3}{5}}\right)$$

Method 2: Three-Point Gauss-Legendre in [-1,1]

n = 2; degree of precision 2n+1=5

The nodes are the roots of Gauss-Legendre orthogonal polynomials with respect to appropriate weight functions w(x)=1 in [-1, 1]:

$$x_0 = -\sqrt{\frac{3}{5}} \quad x_1 = 0 \quad x_2 = \sqrt{\frac{3}{5}}$$

The coefficients can be determined by the method of indeterminate coefficients

$$I = \int_{-1}^{1} f(x)dx = \frac{5}{9} f\left(-\sqrt{\frac{3}{5}}\right) + \frac{8}{9} f(0) + \frac{5}{9} f\left(\sqrt{\frac{3}{5}}\right)$$

Example $I = \int_0^4 te^{2t}dt = \int_{-1}^{1}(4x+4)e^{4x+4}dx = 5216,92$

3 point Formula n = 2

$$I = \int_{-1}^{1} f(x)dx = \frac{5}{9} f(-\sqrt{0.6}) + \frac{8}{9} f(0) + \frac{5}{9} f(\sqrt{0.6})$$

$$= \frac{5}{9}(4 - 4\sqrt{0.6})e^{4-\sqrt{0.6}} + \frac{8}{9}(4)e^{4} + \frac{5}{9}(4 + 4\sqrt{0.6})e^{4+\sqrt{0.6}}$$

$$= \frac{5}{9}(2.221191545) + \frac{8}{9}(218.3926001) + \frac{5}{9}(8589.142689)$$

$$= 4967.106689 \qquad (\varepsilon = 4.79\%)$$

4 point Formula n = 3

$$I = \int_{-1}^{1} f(x)dx = 0.34785\left[f(-0.861136) + f(0.861136)\right]$$

$$+ 0.652145\left[f(-0.339981) + f(0.339981)\right]$$

$$= 5197.54375 \qquad (\varepsilon = 0.37\%)$$

THEOREM

Gauss quadrature formula

$$\int_a^b w(x) f(x) dx \approx \sum_{i=0}^{n} w_i f(x_i) \qquad (1)$$

has degree of precision 2n + 1 if and only if

-the n + 1 **nodes** $\{x_0, x_1, .., x_n\}$ coincide with the n+1 zeros of poly of degree n+1, orthogonal in [a,b] to the weight function w(x).

-the n+1 **coefficients** $\{w_0, w_1, .., w_n\}$ are computed by imposing that formula (1) is exact for each polynomial of degree ≤ n

Proof

Let $f \in \mathbf{P}_{2n+1}$ polynomial division $f(x)/\ p_{n+1}(x)$

$$f(x) = p_{n+1}(x)q(x) + r(x) \quad q(x), r(x) \in \mathbf{P}_n$$

consequently $f(x_i) = r(x_i), \quad x_i$ *are* $n+1$ zeros of p_{n+1}

$$\int_a^b f(x)w(x)dx = \int_a^b (p_{n+1}q + r)w(x)dx =$$

$$= \underbrace{\int_a^b p_{n+1}(x)q(x)w(x)dx}_{0} + \int_a^b r(x)w(x)dx =$$

Exact formula for functions in $\mathbf{P}_n$

$$= \sum_{i=0}^{n} w_i r(x_i) = \sum_{i=0}^{n} w_i f(x_i)$$

then formula is exact for $f \in \mathbf{P}_{2n+1}$ #

Weighting factors (coefficients) and nodes used in Gauss-Legendre formulas

Gauss weights w are all positive.

Number of points, n	Points, x_i	Weights, w_i
1	0	2
2	$\pm 1/\sqrt{3}$	1
3	0	$8/9$
	$\pm\sqrt{3/5}$	$5/9$
4	$\pm\sqrt{\left(3-2\sqrt{6/5}\right)/7}$	$\frac{18+\sqrt{30}}{36}$
	$\pm\sqrt{\left(3+2\sqrt{6/5}\right)/7}$	$\frac{18-\sqrt{30}}{36}$
5	0	$128/225$
	$\pm\frac{1}{3}\sqrt{5-2\sqrt{10/7}}$	$\frac{322+13\sqrt{70}}{900}$
	$\pm\frac{1}{3}\sqrt{5+2\sqrt{10/7}}$	$\frac{322-13\sqrt{70}}{900}$

```
function intf = guasslengendre(f,a,b)
% Approximate the integral of f from a to b using a
       6-point Gauss-Legendre rule, n=5.
% f is the name of a function file
nodes = [-0.9324695142031520; -0.6612093864662645; -0.2386191860831969;
         0.2386191860831969; 0.6612093864662645; 0.9324695142031520];
weights =[0.1713244923791703; 0.3607615730481386; 0.4679139345726910;
          0.4679139345726910; 0.3607615730481386; 0.1713244923791703];

% change of variables from [-1,1] to [a,b]
ab_nodes = a + (b-a)*(nodes+1)/2;
ab_weights = weights*(b-a)/2;

% apply Guass-Legendre rule
intf = sum(ab_weights.*feval(f,ab_nodes));
% requires f to work for vectors
% exact for polynomials of degree 2n+1=11
```

Gauss Quadrature Formulas

$$I_{w,f} = \int_a^b w(x) f(x) dx$$

$w(x)$ weight function

$w(x) \geq 0, \quad x \in [a,b]$

Interval	ω(x)	Orthogonal polynomials
[−1, 1]	1	Legendre polynomials
(−1, 1)	$(1-x)^\alpha (1+x)^\beta, \quad \alpha, \beta > -1$	Jacobi polynomials
(−1, 1)	$\frac{1}{\sqrt{1-x^2}}$	Chebyshev polynomials (first kind)
[−1, 1]	$\sqrt{1-x^2}$	Chebyshev polynomials (second kind)
[0, ∞)	e^{-x}	Laguerre polynomials
(−∞, ∞)	e^{-x^2}	Hermite polynomials

Formulas

Gauss-Legendre
Gauss-Jacobi
Gauss-Chebyshev
Gauss-Chebyshev

Gauss-Laguerre
Gauss-Hermite

$$\int_{-1}^{1} f(x)(1-x^2)^{-1/2}\, dx \approx \frac{\pi}{k+1} \sum_{i=0}^{k} f\left(\cos \frac{(2i+1)\pi}{2(k+1)}\right)$$

Gauss-Chebyshev

$$\int_a^b f(x)\, dx = \left(\frac{b-a}{2}\right) \int_{-1}^{1} f\left(\frac{a+b+t(b-a)}{2}\right) dt$$

Gauss-Legendre

Gauss Quadrature rules

A family of orthogonal polynomilas generates a quadrature Gauss formula.
For some classical polynomials nodes and weights are stored in tables.

File MATLAB	Formula di quadratura	File
Gauss.m	Formule gaussiane con assegnata funzione peso	-
GaCe.m	Formule di Gauss-Chebicev	GCe.dat
GaHe.m	Formule di Gauss-Hermite	GHe.dat
GaLa.m	Formule di Gauss-Laguarre	GLa.dat
GaLe.m	Formule di Gauss-Legendre	GLe.dat
Lobatto.m	Formule di Lobatto	Lobatto.dat
Radau.m	Formule di Radau	Radau.dat

Tabella 5.5 Elenco degli script MATLAB per il calcolo di nodi e pesi delle formule di quadratura di Gauss più comuni.

Summarizing..Quadrature Rules

Newton-Cotes formulas

- n+1 equispaced nodes
- Degree of precision n or n+1
- Coefficients positive only for n <=7
- They can also not converge

Gauss formulas

- Degree of precision 2n+1
- Coefficients always positive
- Always convergent also increasing the number of nodes (the degree of precision tends to infinity for number of nodes which tends to infinity)

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Serena Morigi
Department of Mathematics
serena.morigi@unibo.it
http://www.dm.unibo.it/~morigi