Lecture 7

Differentiation and Integration

Ye Ding (丁烨)

Email: y.ding@sjtu.edu.cn

School of Mechanical Engineering

Shanghai Jiao Tong University

Differentiation and Integration

- **Motivation: from Lecture 2**

```
function y = fd_derivative(fun,x,h)
% fd_derivative(FUN,X,H) is a finite difference
% approximation to the derivative of function FUN at X
% with difference parameter H. H defaults to SQRT(EPS).
if nargin < 3
    h = sqrt(eps);
    if nargin < 2
        x = 0;
    end
end
y = (fun(x+h) - fun(x))/h;
```

Differentiation and Integration

Motivation

- How to calculate the derivative from a tabulated list?

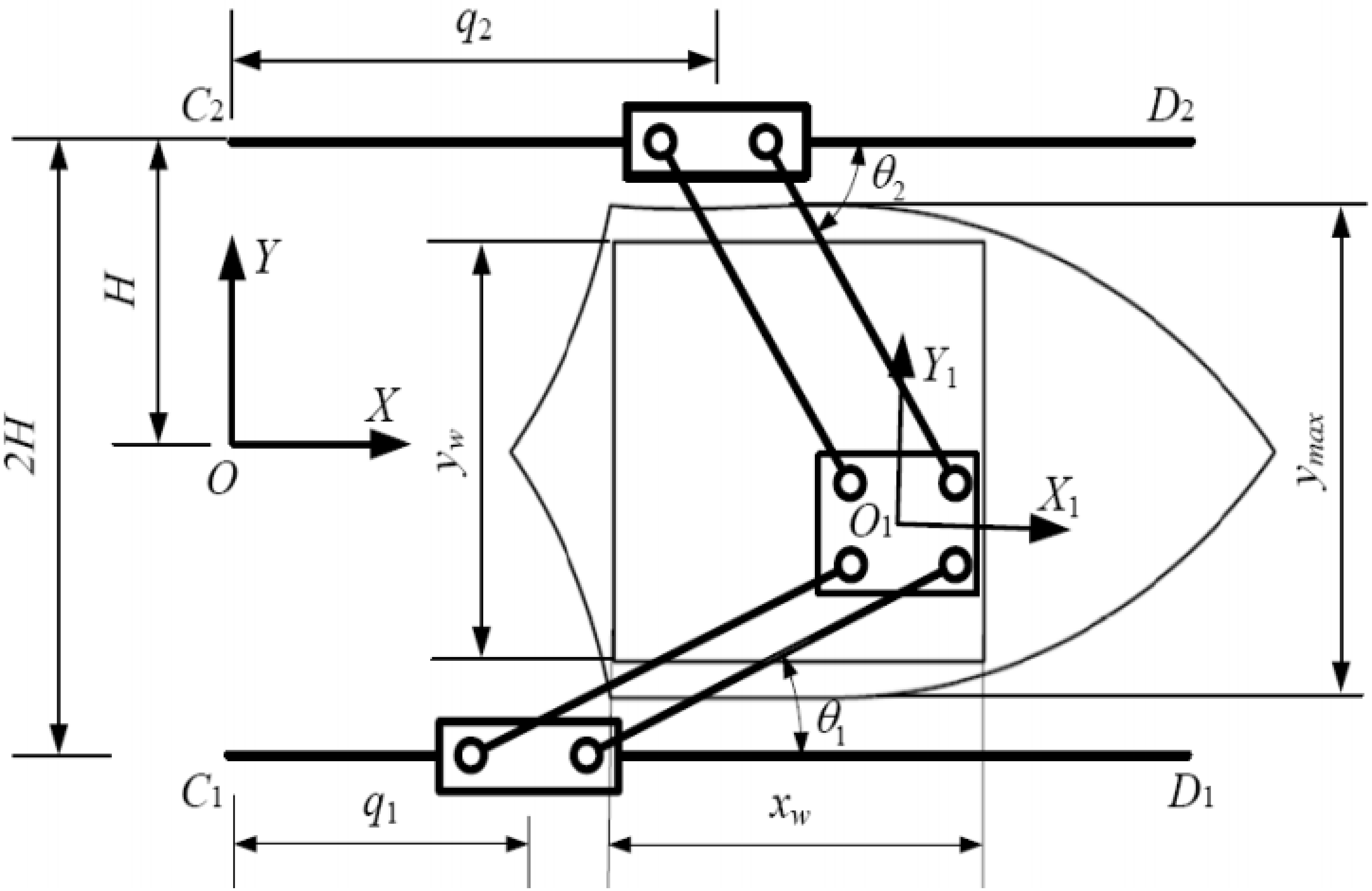
x	$f(x)$
1.8	10.889365
1.9	12.703199
2.0	14.778112
2.1	17.148957
2.2	19.855030

$$\Rightarrow \begin{cases} f'(2.0) \\ f''(2.0) \end{cases}$$

Differentiation and Integration

- **Motivation: from Robotics**

➢ **How to calculate the workspace of a robot?**



Differentiation and Integration

- **References for Differentiation and Integration**

  [1] Cleve Moler, Numerical Computing with MATLAB, Society for Industrial and Applied Mathematics, 2004. Chapter 6

  [2] Timothy Sauer, Numerical analysis (2nd ed.), Pearson Education, 2012. Chapter 5

  [3] Richard L. Burden, J. Douglas Faires, Numerical analysis (9th ed.), Brooks/Cole, 2011. Chapter 4

Differentiation and Integration

- Symbolic Computation
- Numerical Differentiation
  - Finite Difference Formulas
  - Richardson Extrapolation
- Numerical Integration
  - Newton–Cotes Methods
  - Romberg Integration

Symbolic Computation

- **Caculus**

Functions:

diff	Differentiate
int	Integrate
limit	Limit
taylor	Taylor series
jacobian	Jacobian matrix
symsum	Summation of series

Symbolic Computation

- **Caculus**

```
>> syms x t a
>> f = sin(x)/x;
>> g = limit(f)
g =
    1
```

```
>> gg = limit((1+x/t)^t,t,inf)
gg =
        exp(x)
```

Symbolic Computation

Caculus

```
>> syms x
>> g = taylor(sin(x))
g =
     x^5/120 - x^3/6 + x
```

```
>> syms x
taylor(log(x), x, 'ExpansionPoint', 1)
ans =
x - (x - 1)^2/2 + (x - 1)^3/3 - (x - 1)^4/4 + (x
- 1)^5/5 - 1
```

Symbolic Computation

- **Caculus**

```
>> syms x y z
>> f = [x*y*z; y^2; x + z];
>> v = [x, y, z];
>> R = jacobian(f, v)
```

```
R =
[ y*z, x*z, x*y]
[   0, 2*y,   0]
[   1,   0,   1]
```

Symbolic Computation

- **Caculus**

```
>>  syms a b t x y
>> f = sin(a*x)+cos(b*t);
>> g = diff(f)
g =
    a*cos(a*x)
```

Symbolic Computation

Caculus

```
>>  syms a b t x y
>> f = sin(a*x)+cos(b*t);
>> g = diff(f,t)
g =
      -b*sin(b*t)
```

```
>> g = diff(f,t,2)
g =
      -b^2*cos(b*t)
```

Symbolic Computation

- **Caculus**

```
>> syms a x
>> f = sin(a*x);
>> g = int(f)
g =
     -cos(a*x)/a
```

```
>> gg = int(f,a)
gg =
       -cos(a*x)/x
```

Symbolic Computation

- **Caculus**

```
>> syms a x
>> f = sin(a*x);
>> g = int(f,0,pi)
g =
      (2*sin((pi*a)/2)^2)/a
```

```
>> pretty(g)
           / pi a \2
  2 sin| ---- |
           \  2   /
  --------------
        a
```

Symbolic Computation

- **Caculus**

```
>> syms x a b
>> f = sin(x)/x;
>> g = int(f,x)
g =
    sinint(x)
```

```
>> g = int(f,x,a,b)
g =
    sinint(b) - sinint(a)
```

Symbolic Computation

- **Caculus**

```
>> x = sym('x')
>> f  = 1/(1+x^6);
>> g = int(f,x)
g =
    atan(x)/3 + atan(x/(3^(1/2)*i - 1) + (3^(1/2)*x*i)/(3^(1/2)*i - 1))*((3^(1/2)*i)/6 + 1/6) + atan(x/(3^(1/2)*i + 1) - (3^(1/2)*x*i)/(3^(1/2)*i + 1))*((3^(1/2)*i)/6 - 1/6)
```

Numerical Computation

- Symbolic Computation
- Numerical Differentiation
  - Finite Difference Formulas
  - Richardson Extrapolation
- Numerical Integration
  - Newton–Cotes Methods
  - Romberg Integration

Numerical Differentiation

Basic Concept

The derivative of *f*(*x*) at a value *x* is

$$f'(x) = \lim_{h \to 0} \frac{f(x+h) - f(x)}{h}$$

provided that the limit exists.

If *f*(*x*) is twice continuously differentiable, then

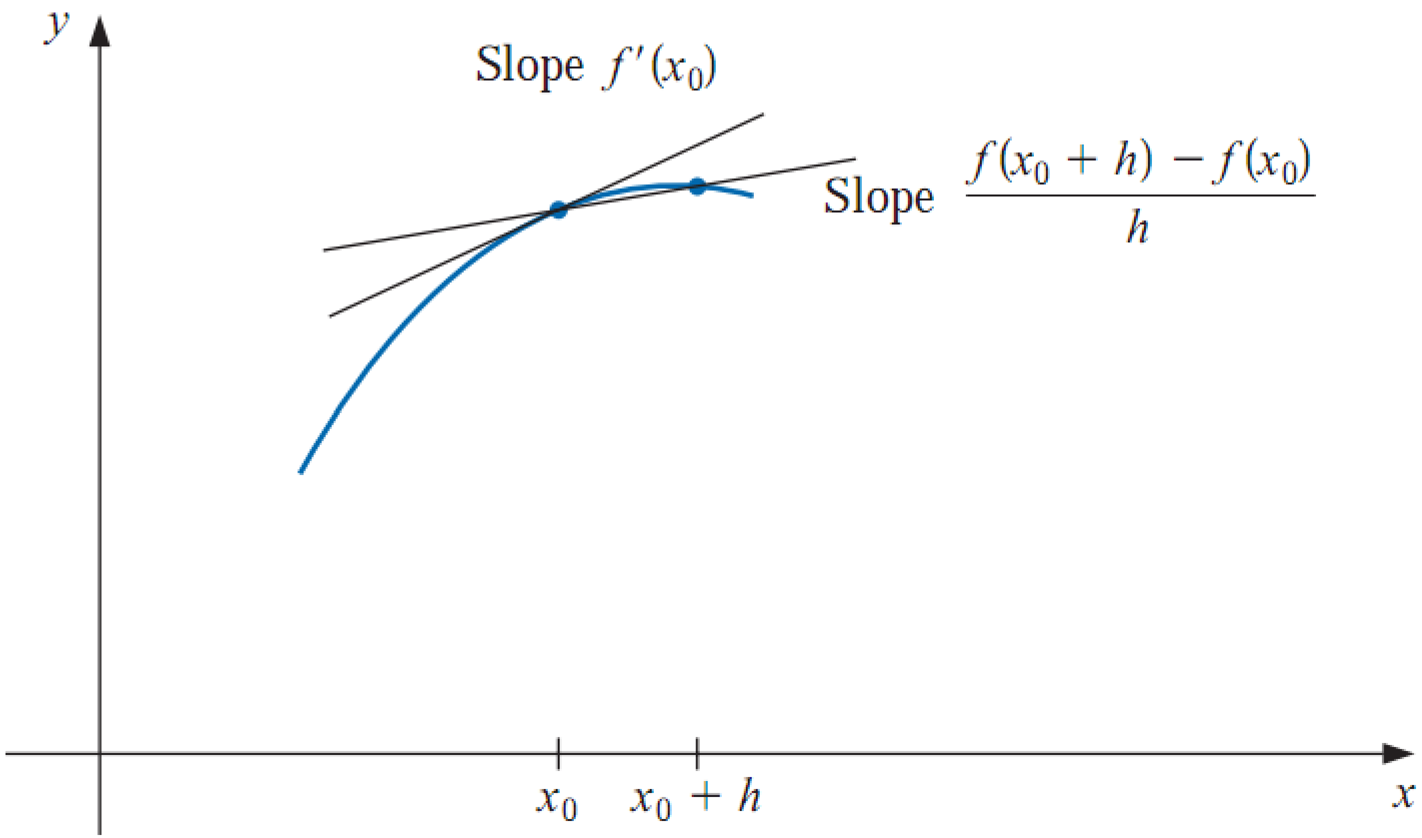
$$f(x+h) = f(x) + hf'(x) + \frac{h^2}{2} f''(c)$$

where *c* is between *x* and *x* + *h*.

Finite Difference Formulas

- **Two-point forward-difference formula**

Geometric Interpretation:



Finite Difference Formulas

- **Two-point forward-difference formula**

If $f(x)$ is twice continuously differentiable, then

$$f'(x) = \frac{f(x+h) - f(x)}{h} - \frac{h}{2} f''(c)$$

where c is between x and $x + h$.

$$\boxed{f'(x) \approx \frac{f(x+h) - f(x)}{h}}$$

Finite Difference Formulas

- **Definition: Order *n* Approximation**

If the error is $O(h^n)$,

the formula is an order *n* approximation.

The two-point-forward-difference formula

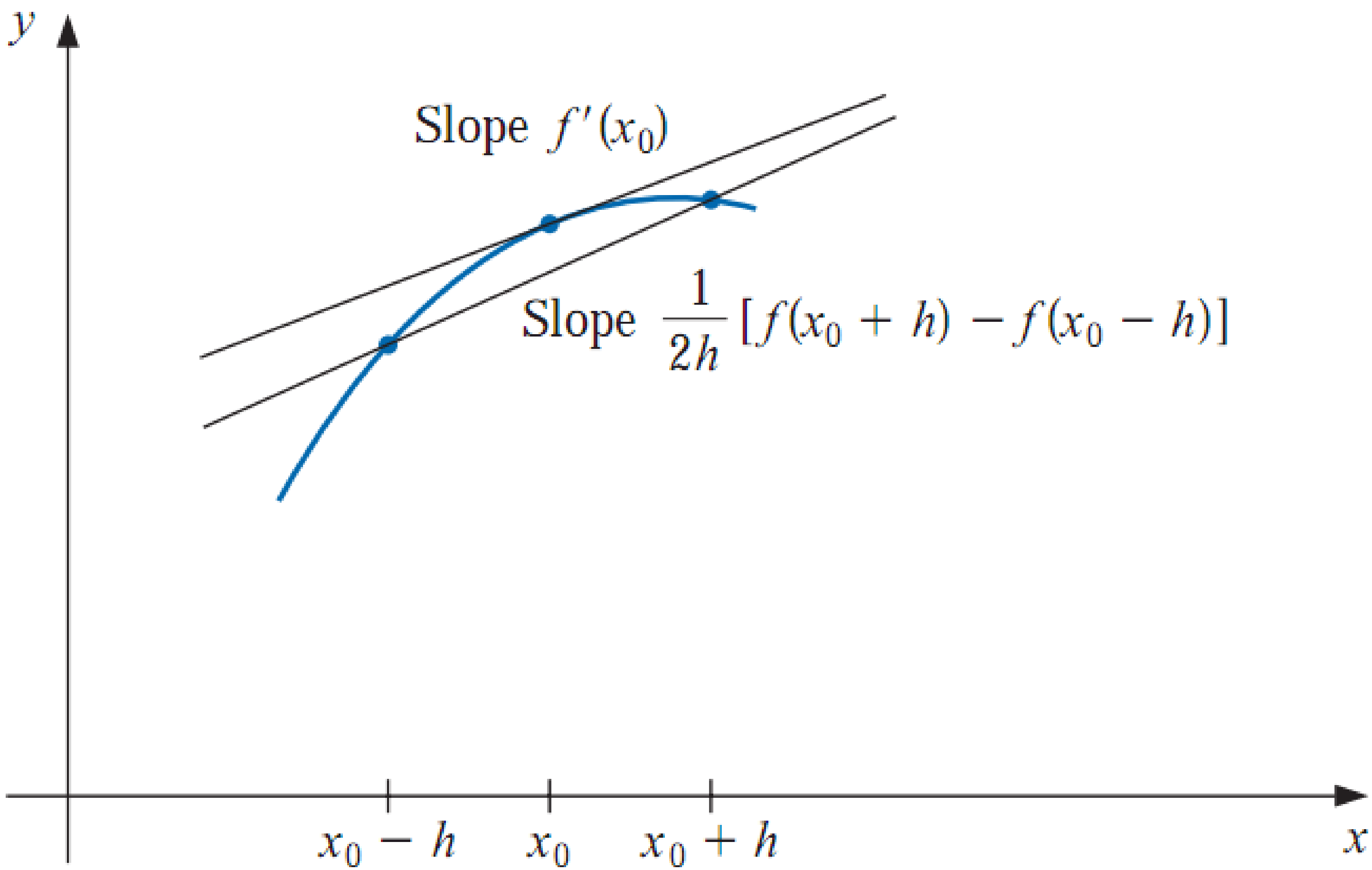
$$f'(x) \approx \frac{f(x+h) - f(x)}{h}$$

is a *first-order method* for approximating the first derivative.

Finite Difference Formulas

- **Second-order formula**

Geometric Interpretation:



Finite Difference Formulas

- **Second-order formula**

If *f*(*x*) is is three times continuously differentiable,

$$f(x+h) = f(x) + hf'(x) + \frac{h^2}{2} f''(x) + \frac{h^3}{6} f'''(c_1)$$

$$f(x-h) = f(x) - hf'(x) + \frac{h^2}{2} f''(x) - \frac{h^3}{6} f'''(c_2)$$

where $x - h < c_2 < x < c_1 < x + h$.

$$\boxed{f'(x) = \frac{f(x+h) - f(x-h)}{2h} - \frac{h^2}{12} f'''(c_1) - \frac{h^2}{12} f'''(c_2)}$$

Finite Difference Formulas

- **Generalized IntermediateValue Theorem**

Let f be a continuous function on the interval $[a,b]$. Let $x_1, \ldots, x_n$ be points in $[a,b]$, and $a_1, \ldots, a_n > 0$.

Then there exists a number c between a and b such that

$$(a_1 + \cdots + a_n) f(c) = a_1 f(x_1) + \cdots + a_n f(x_n)$$

Finite Difference Formulas

- **Second-order formula**

If $f(x)$ is is three times continuously differentiable,

Three-point centered-difference formula

$$f'(x) = \frac{f(x+h) - f(x-h)}{2h} - \frac{h^2}{6} f'''(c)$$

where $x - h < c < x + h$.

Finite Difference Formulas

- **First-order formula *VS*. Second-order formula**

Approximate the derivative of $f(x) = e^x$ at $x = 0$.

$$f'(x) \approx \frac{e^{x+h} - e^x}{h}$$

h =	f_d_1st_app =	error_1st_app =
0.1	1.051709180756477	-0.051709180756477
0.01	1.005016708416795	-0.005016708416795
0.001	1.000500166708385	-0.000500166708385
0.0001	1.000050001667141	-0.000050001667141
0.00001	1.000005000006965	-0.000005000006965
0.000001	1.000000499962184	-0.000000499962184
0.0000001	1.000000049433680	-0.000000049433680
0.00000001	0.999999993922529	0.000000006077471
0.000000001	1.000000082740371	-0.000000082740371

Finite Difference Formulas

- **First-order formula *VS*. Second-order formula**

Approximate the derivative of $f(x) = e^x$ at $x = 0$.

$$f'(x) \approx \frac{e^{x+h} - e^{x-h}}{2h}$$

h =	f_d_2nd_app =	error_2nd_app =
0.1	1.001667500198441	-0.001667500198441
0.01	1.000016666749992	-0.000016666749992
0.001	1.000000166666681	-0.000000166666681
0.0001	1.000000001666890	-0.000000001666890
0.00001	1.000000000012102	-0.000000000012102
0.000001	0.999999999973245	0.000000000026755
0.0000001	0.999999999473644	0.000000000526356
0.00000001	0.999999993922529	0.000000006077471
0.000000001	1.000000027229220	-0.000000027229220

Finite Difference Formulas

Rounding Error: Loss of Significance

Denote the floating point version of the inputs

$f(x+h)$ by $\hat{f}(x+h)$ **and** $f(x-h)$ by $\hat{f}(x-h)$

$$\hat{f}(x+h) = f(x+h) + \epsilon_1 \quad \hat{f}(x-h) = f(x-h) + \epsilon_2$$

$$f'(x)_{\text{correct}} - f'(x)_{\text{machine}}$$

$$= f'(x) - \frac{\hat{f}(x+h) - \hat{f}(x-h)}{2h}$$

$$= f'(x) - \frac{f(x+h) + \epsilon_1 - (f(x-h) + \epsilon_2)}{2h}$$

$$= \left(f'(x) - \frac{f(x+h) - f(x-h)}{2h} \right) + \frac{\epsilon_2 - \epsilon_1}{2h}$$

$$= \left(f'(x)_{\text{correct}} - f'(x)_{\text{formula}} \right) + \text{error}_{\text{rounding}}.$$

Finite Difference Formulas

- **Rounding Error: Loss of Significance**

The rounding error has absolute value

$$\left|\frac{\epsilon_2 - \epsilon_1}{2h}\right| \leq \frac{2\epsilon_{\text{mach}}}{2h} = \frac{\epsilon_{\text{mach}}}{h}$$

The absolute value of the error of the machine approximation of *f'*(*x*) is bounded above by

$$E(h) \equiv \frac{h^2}{6} f'''(c) + \frac{\epsilon_{\text{mach}}}{h}$$

Finite Difference Formulas

- **Rounding Error: Loss of Significance**

The absolute value of the error of the machine approximation of *f'*(*x*) is bounded above by

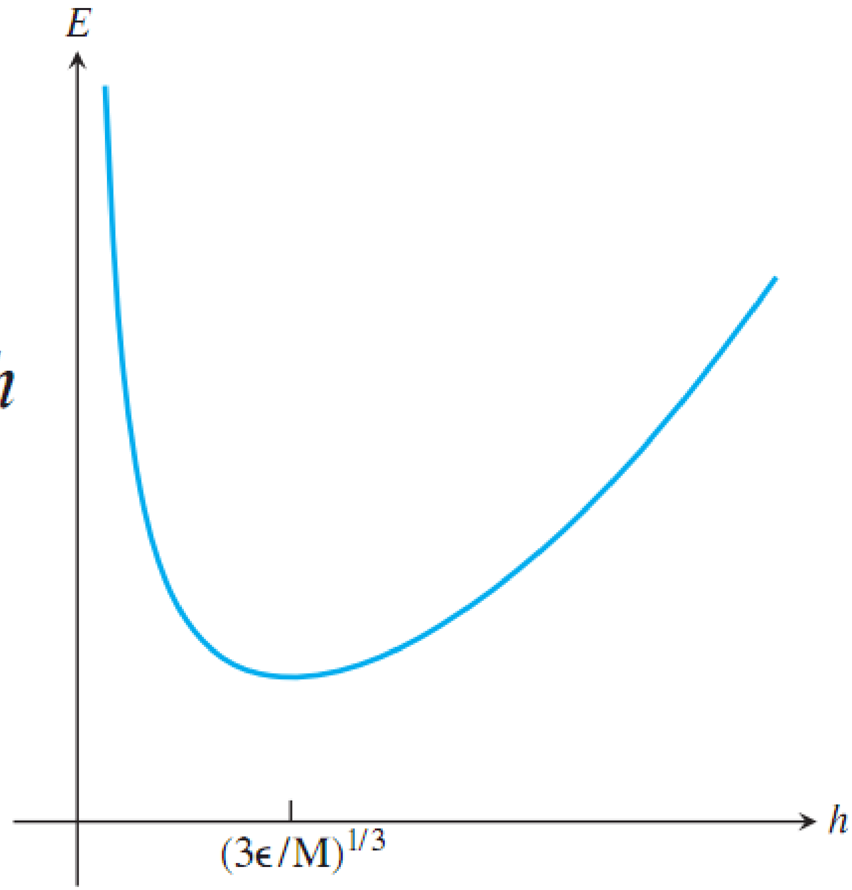
$$E(h) \equiv \frac{h^2}{6} f'''(c) + \frac{\epsilon_{\text{mach}}}{h}$$

$$\Downarrow$$

$$0 = E'(h) = -\frac{\epsilon_{\text{mach}}}{h^2} + \frac{M}{3} h$$

$$\Downarrow$$

$$h = (3\epsilon_{\text{mach}}/M)^{1/3}$$



Finite Difference Formulas

- **Approximation formula for second derivative**

$$f(x+h) = f(x) + hf'(x) + \frac{h^2}{2}f''(x) + \frac{h^3}{6}f'''(x) + \frac{h^4}{24}f^{(\mathrm{iv})}(c_1)$$

$$f(x-h) = f(x) - hf'(x) + \frac{h^2}{2}f''(x) - \frac{h^3}{6}f'''(x) + \frac{h^4}{24}f^{(\mathrm{iv})}(c_2)$$

where $x - h < c_2 < x < c_1 < x + h$

Adding them together to eliminate the first derivative terms yields:

$$f(x+h) + f(x-h) - 2f(x) = h^2 f''(x) + \frac{h^4}{24}f^{(\mathrm{iv})}(c_1) + \frac{h^4}{24}f^{(\mathrm{iv})}(c_2)$$

Finite Difference Formulas

- **Approximation formula for second derivative**

Three-point centered-difference formula for second derivative

$$f''(x) = \frac{f(x-h) - 2f(x) + f(x+h)}{h^2} - \frac{h^2}{12} f^{(\mathrm{iv})}(c)$$

for some c between $x - h$ and $x + h$.

Finite Difference Formulas

- **Approximation formula for second derivative**

Approximate the second derivative of $f(x) = e^x$ at $x = 0$.

h =	f_2d_app =	error_2d_app =
0.1	1.000833611160723	-0.000833611160723
0.01	1.000008333360558	-0.000008333360558
0.001	1.000000083406505	-0.000000083406505
0.0001	0.999999993922529	0.000000006077471
0.00001	1.000000082740371	-0.000000082740371
0.000001	0.999866855977416	0.000133144022584
0.0000001	0.999200722162641	0.000799277837359
0.00000001	0	1.000000000000000
0.000000001	0	1.000000000000000

Richardson Extrapolation

Basic Idea: Simple Case

The three-point centered-difference formula

$$f'(x) = \frac{f(x+h) - f(x-h)}{2h} - \frac{f'''(c_h)}{6}h^2$$

where the point c_h lies between x and $x + h$.

How to leverage an order 2 formula into one of higher order?

$$f'(x) = F(h) + \alpha_1 h^2 + \alpha_2 h^4 + \cdots$$

where $F(h) = \dfrac{1}{2h}\left[f(x+h) - f(x-h) \right]$

Richardson Extrapolation

Basic Idea: Simple Case

step = *h*: $$f'(x) = F(h) + \alpha_1 h^2 + \alpha_2 h^4 + \cdots \qquad \textbf{(1)}$$

step = *h*/2: $$f'(x) \approx F(h/2) + \alpha_1 \frac{h^2}{4} + \alpha_2 \frac{h^4}{16} + \cdots \qquad \textbf{(2)}$$

4×(2) − (1)

$$3f'(x) \approx [4F(h/2) - F(h)] + O(h^4)$$

$$f'(x) \approx \frac{[4F(h/2) - F(h)]}{3} + O(h^4)$$

Richardson Extrapolation

- **Basic Idea: General Case**

An order n formula $F(h)$ for approximating a given quantity Q.

$$Q \approx F(h) + Kh^n$$

Applying the formula again with $h/2$ instead of h

$$Q - F(h/2) \approx \frac{1}{2^n}(Q - F(h))$$

$$\Downarrow$$

$$Q \approx \frac{2^n F(h/2) - F(h)}{2^n - 1}$$

Richardson Extrapolation

- **Richardson extrapolation: More Details**

$$Q = F_n(h) + Kh^n + O(h^{n+1})$$

Then cutting *h* in half yields:

$$Q = F_n(h/2) + K\frac{h^n}{2^n} + O(h^{n+1})$$

$$F_{n+1}(h) = \frac{2^n F_n(h/2) - F_n(h)}{2^n - 1}$$

$$= \frac{2^n(Q - Kh^n/2^n - O(h^{n+1})) - (Q - Kh^n - O(h^{n+1}))}{2^n - 1}$$

$$= Q + \boxed{O(h^{n+1})}$$

$$\boxed{F_{n+1}(h) \text{ is (at least) an order } n+1 \text{ formula}}$$

Richardson Extrapolation

- **Five-point centered-difference formula**

$$f'(x) = \frac{f(x+h) - f(x-h)}{2h} - \frac{f'''(c_h)}{6}h^2$$

$$\Downarrow \quad F_2(h) = \frac{f(x+h) - f(x-h)}{2h}$$

$$F_4(h) = \frac{2^2 F_2(h/2) - F_2(h)}{2^2 - 1}$$

The five-point centered-difference formula

$$= \left[4\frac{f(x+h/2) - f(x-h/2)}{h} - \frac{f(x+h) - f(x-h)}{2h}\right] \Big/ 3$$

$$= \frac{f(x-h) - 8f(x-h/2) + 8f(x+h/2) - f(x+h)}{6h}.$$

Richardson Extrapolation

- **Apply extrapolation to the second derivative formula**

  Form the tableau:

$F_2(h)$			
$F_2(h/2)$	$F_4(h)$		
$F_2(h/4)$	$F_4(h/2)$	$F_6(h)$	$\cdots$
$\vdots$	$\vdots$	$\vdots$	

Richardson Extrapolation

- **Example 2:**

Find approximations of order $O(h^2)$ and $O(h^4)$, for $f'(2.0)$ when $f(x) = xe^x$ and $h = 0.2$.

- $$F_2(h) = \frac{1}{2h}[f(x_0+h) - f(x_0-h)]$$

$$F_2(0.2) = \frac{1}{0.4}[f(2.2) - f(1.8)] = 2.5(19.855030 - 10.889365) = 22.414160$$

$$F_2(0.1) = \frac{1}{0.2}[f(2.1) - f(1.9)] = 5(17.148957 - 12.703199) = 22.228786$$

- $$F_4(h) = F_2(h/2) + (F_2(h/2) - F_2(h))/3$$

$$F_4(0.2) = F_2(0.1) + (F_2(0.1) - F_2(0.2))/3$$

$$= 22.228786 + \frac{1}{3}(22.228786 - 22.414160) = 22.166995$$

Richardson Extrapolation

- **Example 2: Using the tableau**

Find approximations of order $O(h^2)$ and $O(h^4)$, for $f'(2.0)$ when $f(x) = xe^x$ and $h = 0.2$.

$F_2(h)$ 22.414160		
$F_2(h/2)$	$F_4(h)$ 22.166995	
$F_2(h/4)$	$F_4(h/2)$	$F_6(h)$ 22.16716831

Exact solution: 22.167168296791950

Numerical Computation

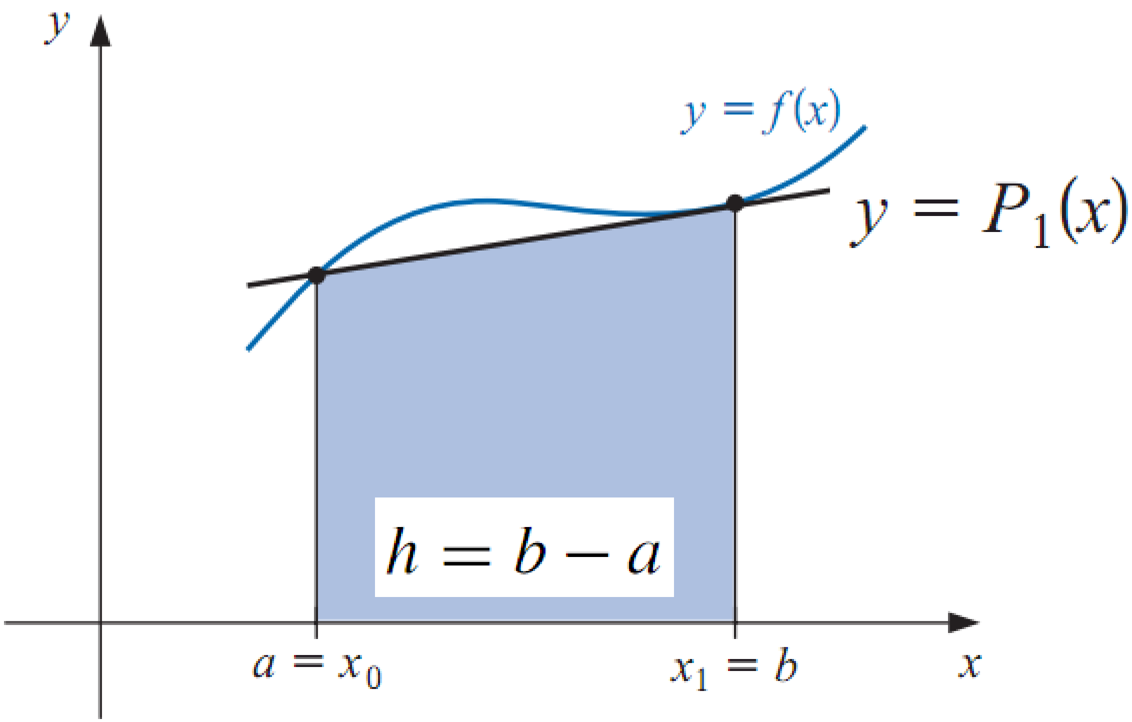
- Symbolic Computation
- Numerical Differentiation
  - Finite Difference Formulas
  - Richardson Extrapolation
- Numerical Integration
  - Newton–Cotes Methods
  - Romberg Integration

Newton–Cotes Methods

- **Trapezoidal Rule: Geometric Method**

The Trapezoidal rule for approximating $\int_a^b f(x)\,dx$

$$\int_a^b f(x)\,dx = \frac{h}{2}[f(x_0) + f(x_1)]$$

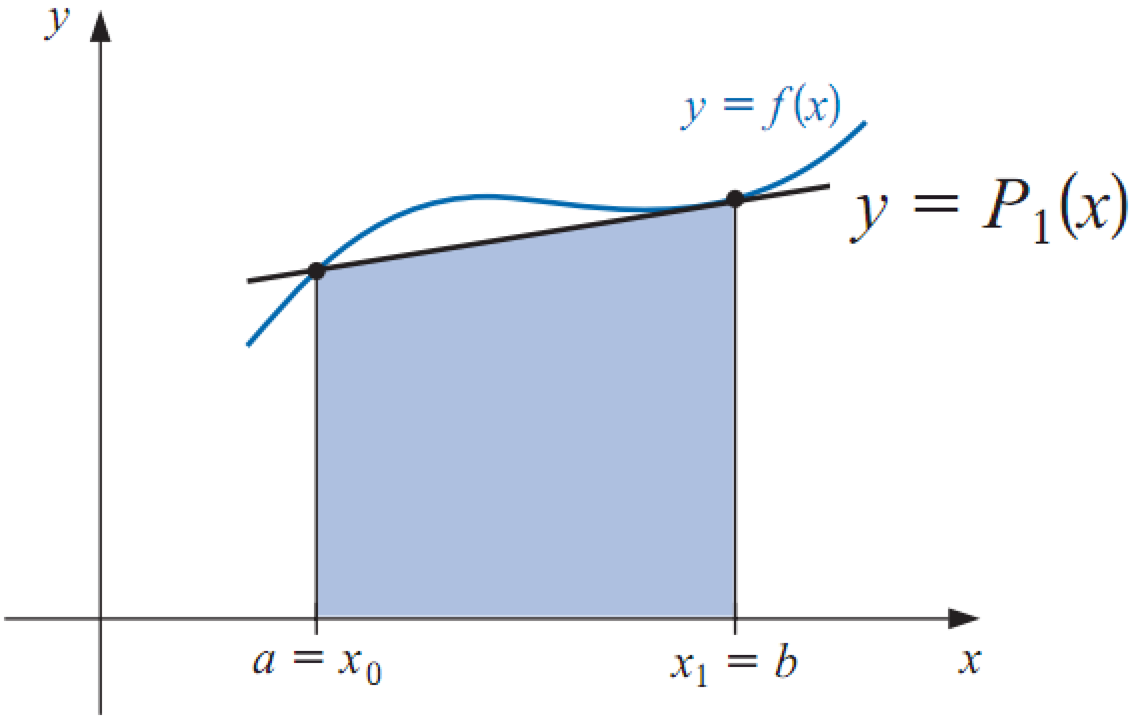


Newton–Cotes Methods

- **Lagrange Interpolation Based Method**

The linear Lagrange polynomial:

$$P_1(x) = \frac{(x - x_1)}{(x_0 - x_1)} f(x_0) + \frac{(x - x_0)}{(x_1 - x_0)} f(x_1)$$



Newton–Cotes Methods

- **Lagrange Interpolation Based Method**

The linear Lagrange polynomial:

$$P_1(x) = \frac{(x - x_1)}{(x_0 - x_1)} f(x_0) + \frac{(x - x_0)}{(x_1 - x_0)} f(x_1)$$

$$\int_a^b f(x)\,dx = \int_{x_0}^{x_1} \left[\frac{(x - x_1)}{(x_0 - x_1)} f(x_0) + \frac{(x - x_0)}{(x_1 - x_0)} f(x_1) \right] dx$$

$$+ \frac{1}{2} \int_{x_0}^{x_1} f''(\xi(x))(x - x_0)(x - x_1)\,dx$$

Newton–Cotes Methods

- **Lagrange Interpolation Based Method**

$$\int_a^b f(x)\,dx = \int_{x_0}^{x_1} \left[\frac{(x-x_1)}{(x_0-x_1)} f(x_0) + \frac{(x-x_0)}{(x_1-x_0)} f(x_1) \right] dx$$

$$+ \frac{1}{2} \int_{x_0}^{x_1} f''(\xi(x))(x-x_0)(x-x_1)\,dx$$

Newton–Cotes Methods

Weighted Mean Value Theorem for Integrals

Suppose $f \in C[a, b]$, the Riemann integral of g exists on $[a, b]$, and $g(x)$ does not change sign on $[a, b]$. Then there exists a number c in (a, b) with

$$\int_a^b f(x)g(x)\,dx = f(c)\int_a^b g(x)\,dx$$

Newton–Cotes Methods

Lagrange Interpolation Based Method

$$\int_a^b f(x)\,dx = \int_{x_0}^{x_1}\left[\frac{(x-x_1)}{(x_0-x_1)}f(x_0)+\frac{(x-x_0)}{(x_1-x_0)}f(x_1)\right]dx$$

$$+\frac{1}{2}\underline{\int_{x_0}^{x_1} f''(\xi(x))(x-x_0)(x-x_1)\,dx}$$

$$\int_{x_0}^{x_1} f''(\xi(x))(x-x_0)(x-x_1)\,dx$$

$$= f''(\xi)\int_{x_0}^{x_1}(x-x_0)(x-x_1)\,dx$$

$$= f''(\xi)\left[\frac{x^3}{3}-\frac{(x_1+x_0)}{2}x^2+x_0x_1x\right]_{x_0}^{x_1}$$

$$= -\frac{h^3}{6}f''(\xi).$$

Newton–Cotes Methods

Lagrange Interpolation Based Method

$$\int_a^b f(x)\,dx = \int_{x_0}^{x_1}\left[\frac{(x-x_1)}{(x_0-x_1)}f(x_0)+\frac{(x-x_0)}{(x_1-x_0)}f(x_1)\right]dx$$

$$+\frac{1}{2}\underline{\int_{x_0}^{x_1} f''(\xi(x))(x-x_0)(x-x_1)\,dx}$$

$$\Downarrow \qquad -\frac{h^3}{6}f''(\xi)$$

$$\int_a^b f(x)\,dx = \left[\frac{(x-x_1)^2}{2(x_0-x_1)}f(x_0)+\frac{(x-x_0)^2}{2(x_1-x_0)}f(x_1)\right]_{x_0}^{x_1} - \frac{h^3}{12}f''(\xi)$$

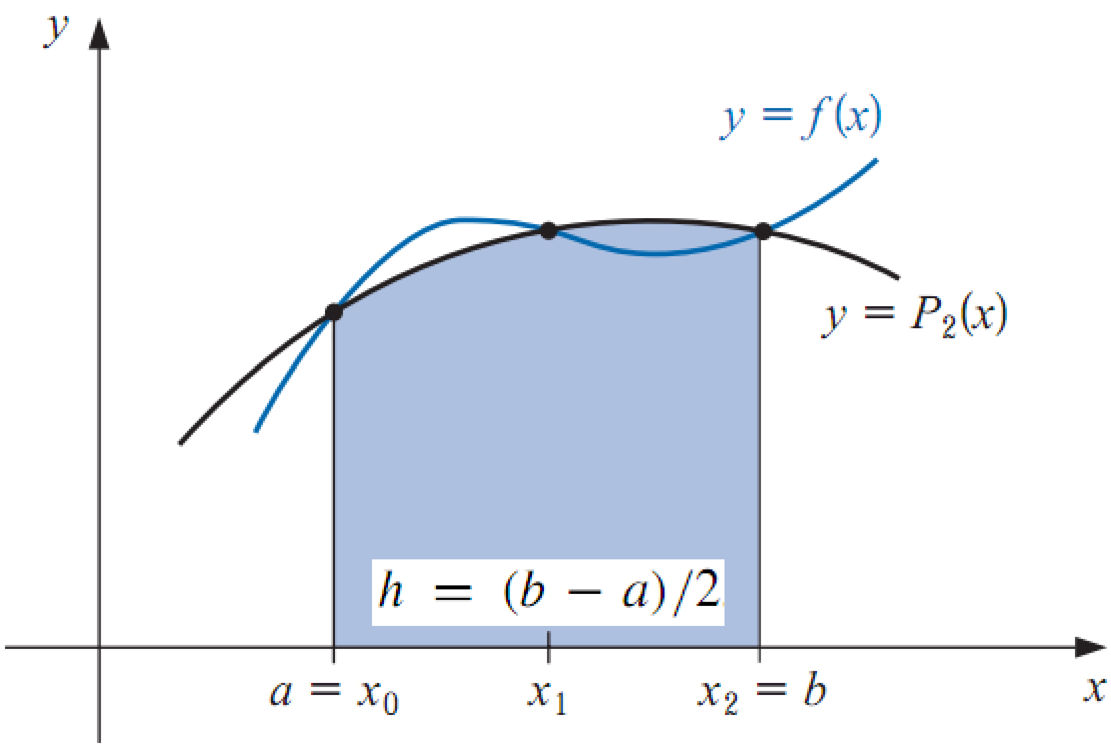
$$= \frac{(x_1-x_0)}{2}[f(x_0)+f(x_1)] \boxed{-\frac{h^3}{12}f''(\xi)}$$

the error term

Newton–Cotes Methods

Simpson's Rule

Simpson's rule results from integrating over $[a, b]$ the second Lagrange polynomial with equally-spaced nodes $x_0 = a$, $x_2 = b$, and $x_1 = a + h$, where $h = (b - a)/2$.



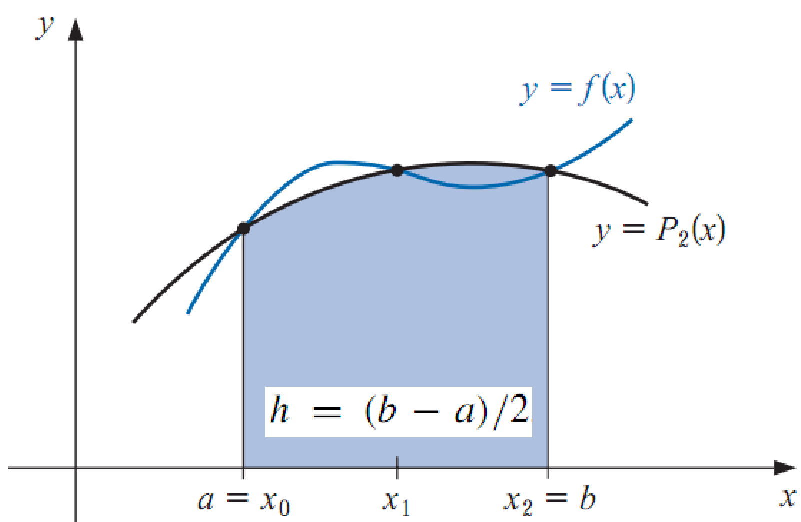
Newton–Cotes Methods

Simpson's Rule

$$\int_a^b f(x)\,dx = \int_{x_0}^{x_2}\left[\frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)}f(x_0) + \frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)}f(x_1)\right.$$

$$\left. + \frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)}f(x_2)\right]dx$$

$$+ \int_{x_0}^{x_2}\frac{(x-x_0)(x-x_1)(x-x_2)}{6}f^{(3)}(\xi(x))\,dx.$$



Newton–Cotes Methods

Simpson's Rule

$$\int_{x_0}^{x_2} f(x)\,dx = \frac{h}{3}[f(x_0) + 4f(x_1) + f(x_2)] - \frac{h^5}{90} f^{(4)}(\xi)$$

Note that the error term in Simpson's rule involves the fourth derivative of *f*(*x*), so it gives exact results when applied to any polynomial of degree three or less.

Newton–Cotes Methods

- **The degree of precision (or accuracy):**

The degree of precision of a quadrature formula is the largest positive integer n such that the formula is exact for x^k , for each $k = 0, 1, \ldots , n$.

The Trapezoidal and Simpson's rules have degrees of precision one and three, respectively.

Newton–Cotes Methods

- **Example 3:** **Apply the Trapezoid to approximate**

$$\int_1^2 \ln x \, dx$$

and find an upper bound for the error.

The Trapezoid Rule:

$$\int_1^2 \ln x \, dx \approx \frac{h}{2}(y_0 + y_1) = \frac{1}{2}(\ln 1 + \ln 2) = \frac{\ln 2}{2} \approx 0.3466$$

The error for the Trapezoid Rule is $-h^3 f''(c)/12$ **where** $1 < c < 2$

Since $f''(x) = -1/x^2$, **the magnitude of the error is at most**

$$\frac{1^3}{12c^2} \leq \frac{1}{12} \approx 0.0834$$

Newton–Cotes Methods

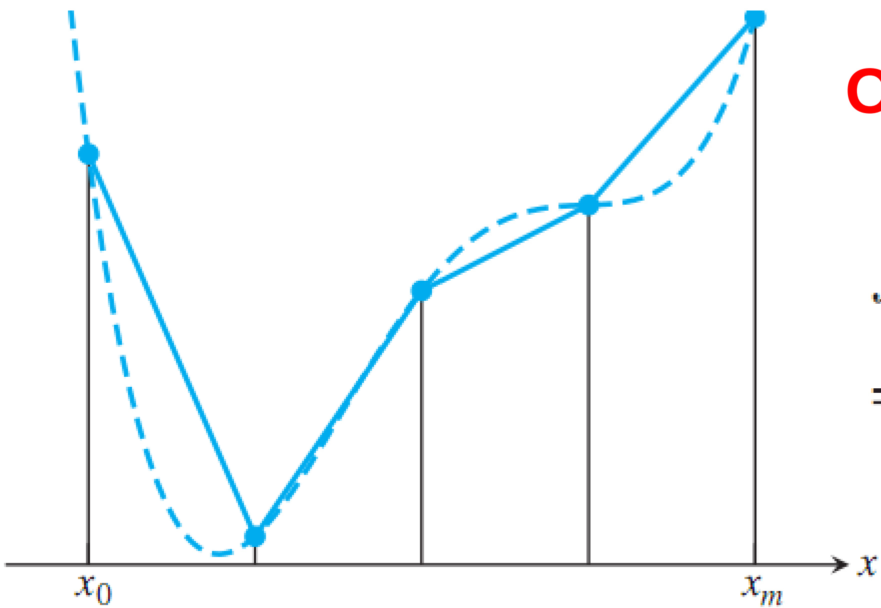
- **The composite Trapezoid Rule**

To approximate $\int_a^b f(x)\,dx$

consider an evenly spaced grid

$$a = x_0 < x_1 < x_2 < \cdots < x_{m-2} < x_{m-1} < x_m = b$$

along the horizontal axis, where $h = x_{i+1} - x_i$ for each i



On each subinterval,

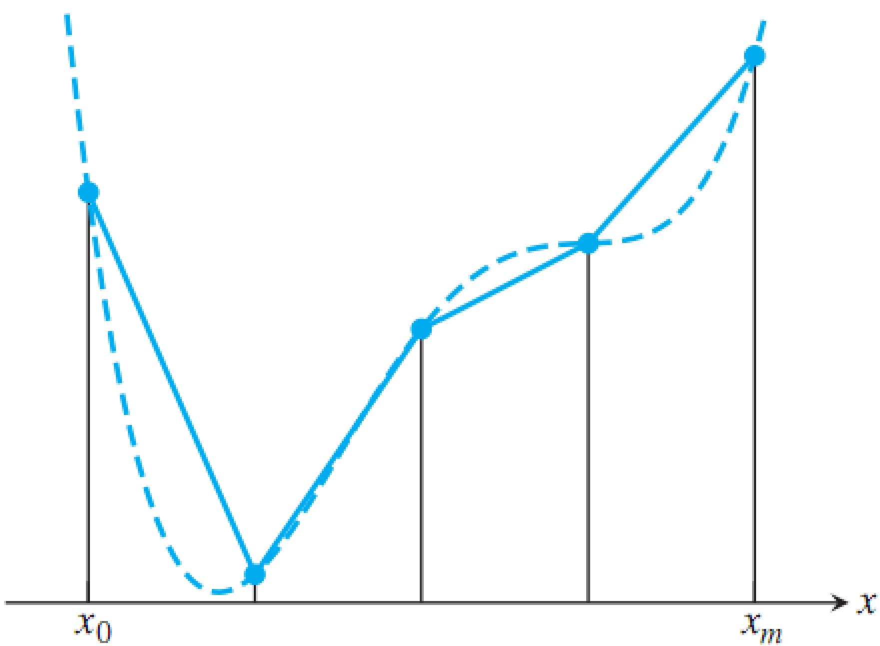
$$\int_{x_i}^{x_{i+1}} f(x)\,dx$$
$$= \frac{h}{2}(f(x_i) + f(x_{i+1})) - \frac{h^3}{12} f''(c_i)$$

Newton–Cotes Methods

- **The composite Trapezoid Rule**

Adding up over all subintervals yields:

$$\int_a^b f(x)\,dx = \frac{h}{2}\left[f(a) + f(b) + 2\sum_{i=1}^{m-1} f(x_i) \right] - \underline{\sum_{i=0}^{m-1} \frac{h^3}{12} f''(c_i)}$$



$$\frac{h^3}{12} m f''(c)$$

for some $a<c<b$.

Newton–Cotes Methods

- **The composite Trapezoid Rule**

$$\int_a^b f(x)\,dx = \frac{h}{2}\left(y_0 + y_m + 2\sum_{i=1}^{m-1} y_i\right) - \frac{(b-a)h^2}{12} f''(c)$$

where $h = (b-a)/m$ and c is between a and b.

Newton–Cotes Methods

- **Example 3 revisited:** **Carry out four-panel approximations of**

$$\int_1^2 \ln x \, dx$$

using the composite Trapezoid Rule.

$$\int_1^2 \ln x \, dx \approx \frac{1/4}{2}\left[y_0 + y_4 + 2\sum_{i=1}^{3} y_i\right]$$

$$= \frac{1}{8}[\ln 1 + \ln 2 + 2(\ln 5/4 + \ln 6/4 + \ln 7/4)]$$

$$\approx 0.3837$$

The error is at most

$$\frac{(b-a)h^2}{12}|f''(c)| = \frac{1/16}{12}\frac{1}{c^2} \leq \frac{1}{(16)(12)(1^2)} = \frac{1}{192} \approx 0.0052$$

Newton–Cotes Methods

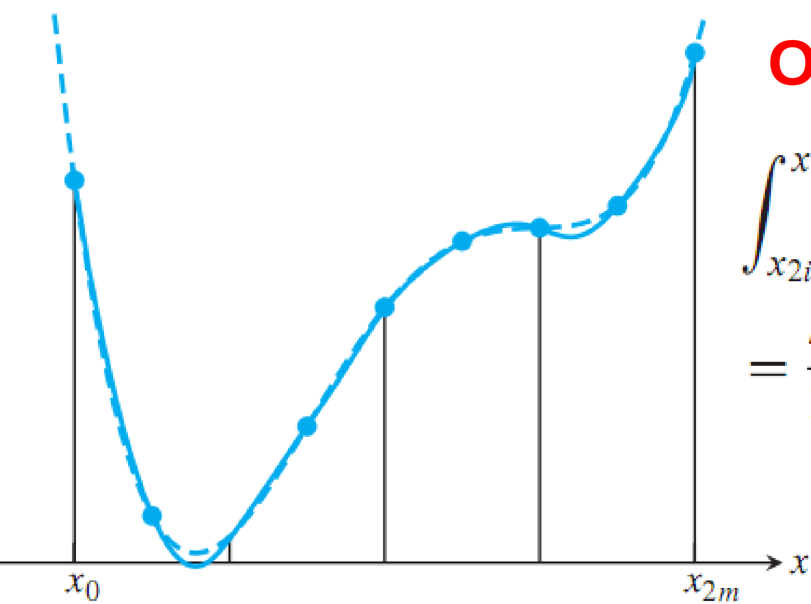
- **The composite Simpson's Rule**

To approximate $\int_a^b f(x)\,dx$

consider an evenly spaced grid

$$a = x_0 < x_1 < x_2 < \cdots < x_{2m-2} < x_{2m-1} < x_{2m} = b$$

along the horizontal axis, where $h = x_{i+1} - x_i$ for each i



On each subinterval $[x_{2i}, x_{2i+2}]$ **,**

$$\int_{x_{2i}}^{x_{2i+2}} f(x)\,dx = \frac{h}{3}[f(x_{2i}) + 4f(x_{2i+1}) + f(x_{2i+2})] - \frac{h^5}{90} f^{(iv)}(c_i)$$

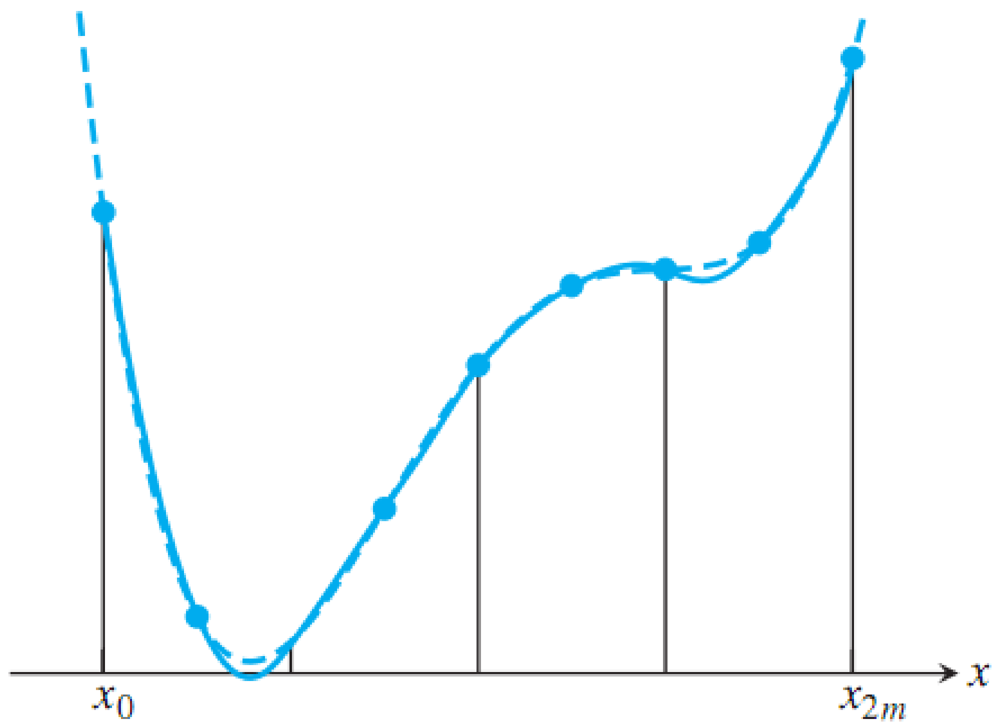
for $i = 0, \ldots, m-1$

Newton–Cotes Methods

- **The composite Simpson's Rule**

Adding up over all subintervals yields:

$$\int_a^b f(x)\,dx = \frac{h}{3}\left[f(a) + f(b) + 4\sum_{i=1}^{m} f(x_{2i-1}) + 2\sum_{i=1}^{m-1} f(x_{2i})\right] - \sum_{i=0}^{m-1} \frac{h^5}{90} f^{(iv)}(c_i)$$



$$= \frac{h^5}{90} m f^{(iv)}(c)$$

$$= \frac{(b-a)h^4}{180} f^{(iv)}(c)$$

for some $a < c < b$

Romberg Integration

- **The composite Trapezoid Rule: Review**

$$\int_a^b f(x)\,dx = \frac{h}{2}\left(y_0 + y_m + 2\sum_{i=1}^{m-1} y_i\right) - \frac{(b-a)h^2}{12} f''(c)$$

where $h = (b-a)/m$ and c is between a and b.

$$\int_a^b f(x)\,dx = \frac{h}{2}\left(y_0 + y_m + 2\sum_{i=1}^{m-1} y_i\right) + c_2h^2 + c_4h^4 + c_6h^6 + \cdots$$

Romberg Integration is the result of applying extrapolation to the composite Trapezoid Rule.

Romberg Integration

- **Basic Idea: Recursion**

The quantity being approximated is $M = \int_a^b f(x)\,dx$

Step size:

$$h_1 = b - a$$

$$h_2 = \frac{1}{2}(b - a)$$

$$h_j = \frac{1}{2^{j-1}}(b - a)$$

Composite Trapezoid Rule:

$$R_{11} = \frac{h_1}{2}(f(a) + f(b))$$

$$R_{21} = \frac{h_2}{2}\left(f(a) + f(b) + 2f\left(\frac{a+b}{2}\right)\right)$$

$$= \frac{1}{2}R_{11} + h_2 f\left(\frac{a+b}{2}\right)$$

$$R_{j1} = \frac{1}{2}R_{j-1,1} + h_j \sum_{i=1}^{2^{j-2}} f(a + (2i-1)h_j)$$

Romberg Integration

Basic Idea: Extrapolation

$$\begin{array}{llll} R_{11} & & & \\ R_{21} & \boxed{R_{22}} & & \\ R_{31} & R_{32} & R_{33} & \\ R_{41} & R_{42} & R_{43} & R_{44} \\ \vdots & & & \ddots \end{array}$$

$$R_{jk} = \frac{4^{k-1} R_{j,k-1} - R_{j-1,k-1}}{4^{k-1} - 1}$$

$$R_{22} = \frac{2^2 R_{21} - R_{11}}{3} \qquad R_{33} = \frac{4^2 R_{32} - R_{22}}{4^2 - 1}$$

$$R_{32} = \frac{2^2 R_{31} - R_{21}}{3} \qquad R_{43} = \frac{4^2 R_{42} - R_{32}}{4^2 - 1}$$

$$R_{42} = \frac{2^2 R_{41} - R_{31}}{3} \qquad R_{53} = \frac{4^2 R_{52} - R_{42}}{4^2 - 1}$$

Romberg Integration

- **Romberg Integration: General Procedure**

$$R_{11} = (b-a)\frac{f(a)+f(b)}{2}$$

for $j = 2, 3, \ldots$

$$h_j = \frac{b-a}{2^{j-1}}$$

$$R_{j1} = \frac{1}{2}R_{j-1,1} + h_j \sum_{i=1}^{2^{j-2}} f(a + (2i-1)h_j)$$

for $k = 2, \ldots, j$

$$R_{jk} = \frac{4^{k-1}R_{j,k-1} - R_{j-1,k-1}}{4^{k-1}-1}$$

end

end

Romberg Integration

- **Example 3 revisited: Apply Romberg Integration to approximate**

$$\int_1^2 \ln x \, dx$$

```
>> vpa(RombergTab,9)
ans =
[  0.34657359,           0,           0,           0]
[ 0.376019349, 0.385834602,           0,           0]
[ 0.383699509, 0.386259563, 0.386287894,           0]
[  0.38564391, 0.386292043, 0.386294209, 0.386294309]
```

```
int_log_ex =   0.386294361119891
error_abs  =   5.203364250583320e-08
```

MATLAB Built-in Functions

- **MATLAB Built-in Functions for integrations**
  - ✓ **Numerically evaluate integral, adaptive Simpson quadrature: *quad***
  - ✓ **Numerically evaluate integral, adaptive Lobatto quadrature: *quadl***
  - ✓ **Numerically evaluate double integral over planar region: *quad2d***
  - ✓ **Numerically evaluate double integral over rectangle: *dblquad***

Summary

- Symbolic Computation
- Numerical Differentiation
  - Finite Difference Formulas
  - Richardson Extrapolation
- Numerical Integration
  - Newton–Cotes Methods
  - Romberg Integration

Thank You !