Second Midterm Exam: November 18 (Tuesday) 3:30pm – 5:30pm.
Lectures 6 – 20, Open-note, JH 245

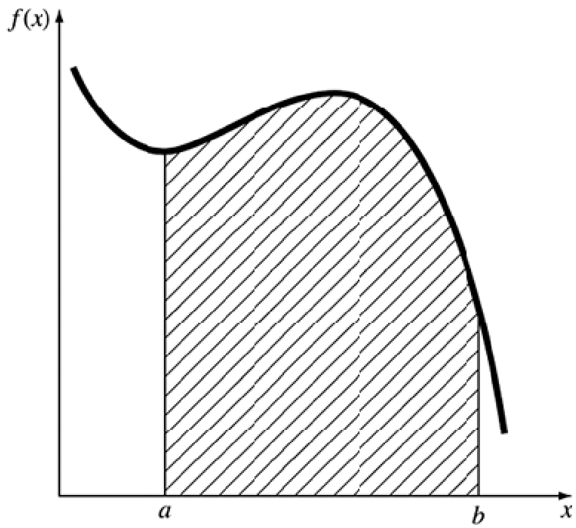
Lecture 24

Numerical Integration Formulas

- Integration

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

Is the total value, or summation, of $f(x)dx$ over the range from a to b:



The integral is the area under the curve
(The area of the shaded region)

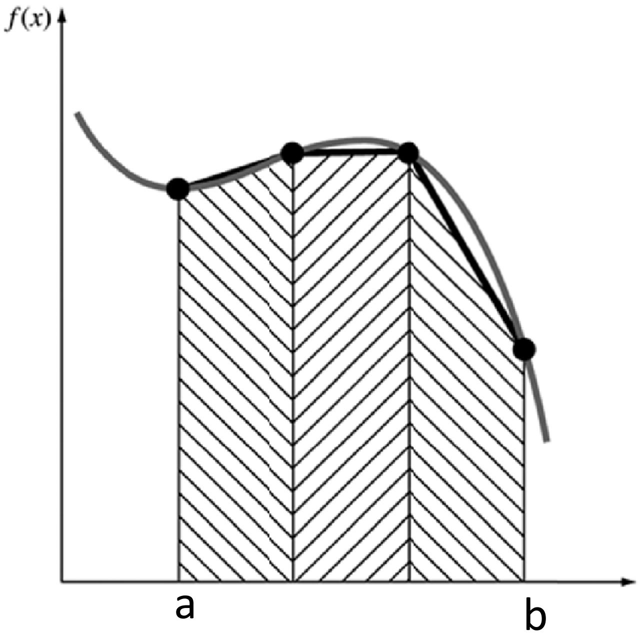
Numerical Integration Formulas

- The Newton-Cotes formulas are the most common numerical integration schemes.
- Generally, they are based on replacing a complicated function or tabulated data with a polynomial that is easy to integrate:

$$I = \int_a^b f(x)dx \cong \int_a^b f_n(x)dx$$

where $f_n(x)$ is an n-th order interpolating polynomial.

- The interpolating function f_n can be polynomials of any order
- The integral can be approximated in one step or in a series of steps to improve accuracy.



Numerical Integration Formulas

- The trapezoidal rule is the first of the Newton-Cotes integration formulas; it uses a straight-line approximation for the function (a straight line is a 1st order polynomial)

$$I = \int_a^b f_1(x)dx$$

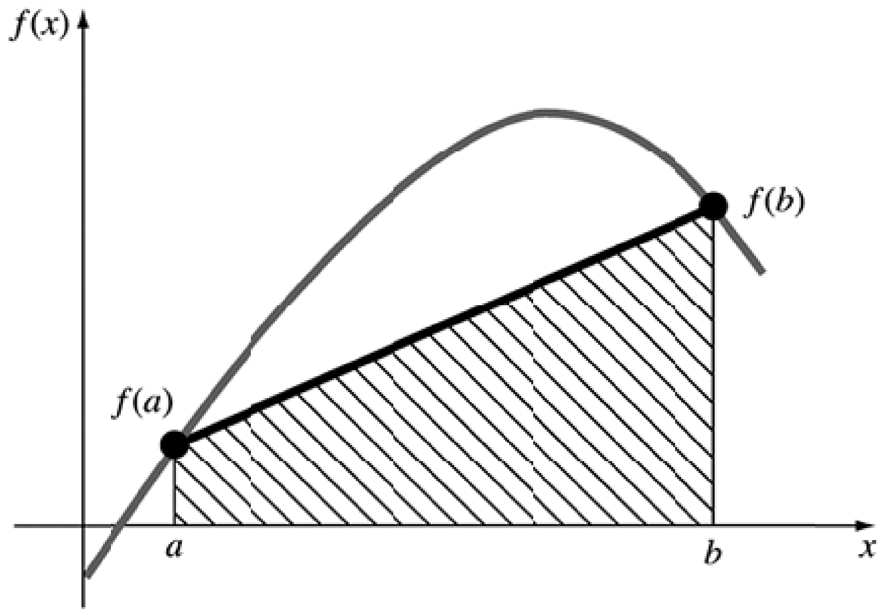
The straight line is

$$y = f(a) + \frac{f(b) - f(a)}{b - a}(x - a)$$

Substitute y expression into I

$$I = \int_a^b \left[f(a) + \frac{f(b) - f(a)}{b - a}(x - a) \right] dx$$

$$\boxed{I = (b - a)\frac{f(b) + f(a)}{2}}$$

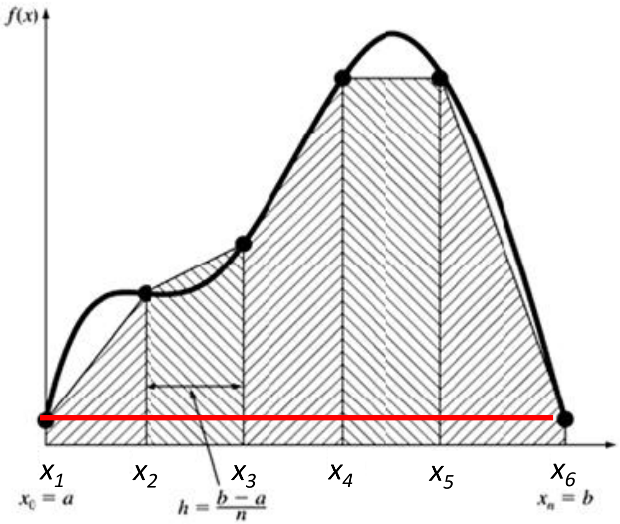


Numerical Integration Formulas

- An estimate for the local truncation error of a single application of the trapezoidal rule is

$$E_t = -\frac{1}{12} f''(\xi)(b-a)^3$$

- where ξ is somewhere between a and b.
- The formula indicates that the error is dependent upon the curvature of the actual function as well as the distance between the points
- Error can thus be reduced by breaking the curve into parts. When this is done the integral is obtained by summing the area under multiple trapezoids



Numerical Integration Formulas

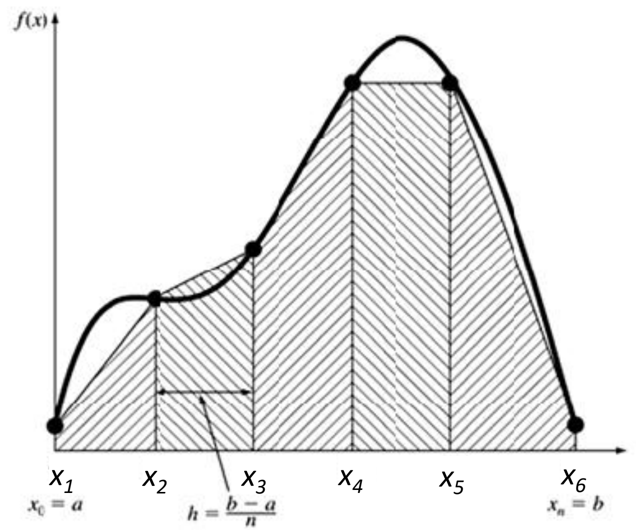
- Assuming n data points are evenly spaced, there will be n-1 intervals over which to integrate.
- The total integral can be calculated by integrating each subinterval and then adding them together

$$I = \int_{x_1}^{x_n} f(x)dx = \int_{x_1}^{x_2} f(x)dx + \int_{x_2}^{x_3} f(x)dx + \cdots + \int_{x_{n-1}}^{x_n} f(x)dx$$

$$I = (x_2 - x_1)\frac{f(x_1) + f(x_2)}{2} + (x_3 - x_2)\frac{f(x_2) + f(x_3)}{2} + \cdots + (x_n - x_{n-1})\frac{f(x_{n-1}) + f(x_n)}{2}$$

$$\boxed{I = \frac{h}{2}\left[f(x_1) + 2\sum_{i=2}^{n-1} f(x_i) + f(x_n)\right]}$$

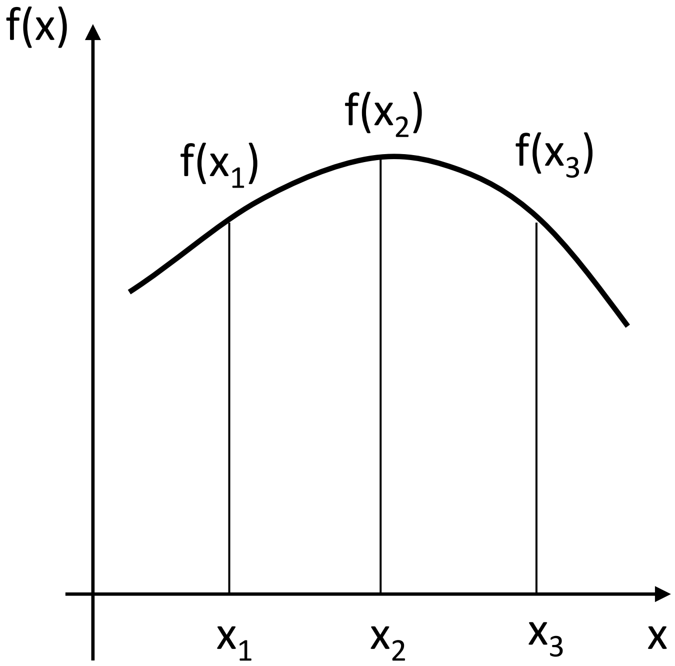
- More complicated approximation formulas can improve the accuracy for curves – these include using 2nd and 3rd order polynomials
- The formulas that result from taking the integrals under these polynomials are called Simpson's rules



Numerical Integration Formulas

- Simpson’s 1/3 rule corresponds to using second-order polynomials. Using the Lagrange form for a quadratic fit of three points

$$f_2(x) = \frac{(x-x_2)(x-x_3)}{(x_1-x_2)(x_1-x_3)} f(x_1) + \frac{(x-x_1)(x-x_3)}{(x_2-x_1)(x_2-x_3)} f(x_2) + \frac{(x-x_1)(x-x_2)}{(x_3-x_1)(x_3-x_2)} f(x_3)$$



Three points are needed for a 2^{nd} order polynomial

Numerical Integration Formulas

- Simpson's 1/3 rule corresponds to using second-order polynomials. Using the Lagrange form for a quadratic fit of three points

$$f_2(x) = \frac{(x-x_2)(x-x_3)}{(x_1-x_2)(x_1-x_3)}f(x_1) + \frac{(x-x_1)(x-x_3)}{(x_2-x_1)(x_2-x_3)}f(x_2) + \frac{(x-x_1)(x-x_2)}{(x_3-x_1)(x_3-x_2)}f(x_3)$$

- Integration over the three points simplifies to

$$\boxed{I = \int_{x_1}^{x_3} f_2(x)dx = \frac{h}{3}[f(x_1) + 4f(x_2) + f(x_3)]}$$

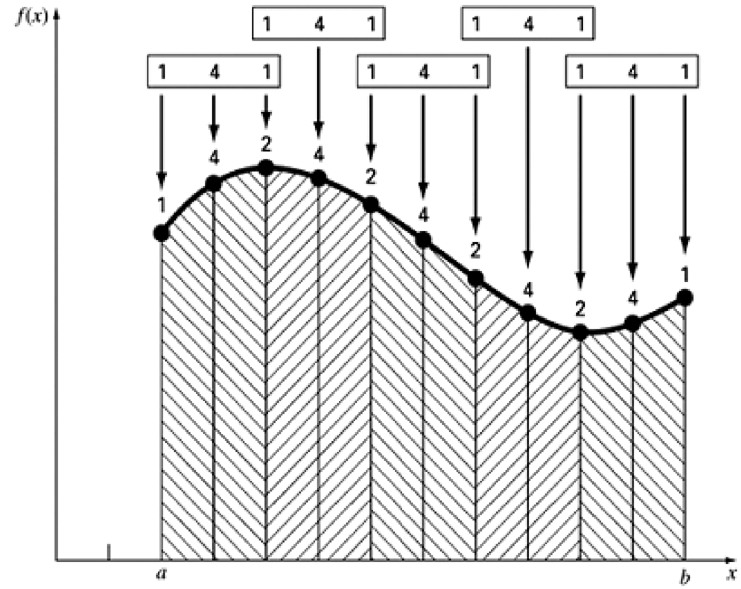
- An estimate for the local truncation error of a single application of Simpson's 1/3 rule is

$$E_t = -\frac{1}{2880}f^{(4)}(\xi)(b-a)^5$$

where ξ is somewhere between a and b.

Numerical Integration Formulas

- Simpson's 1/3 rule can be used on a set of subintervals in the same way the trapezoidal rule was, except there must be an odd number of points.
- Because of the heavy weighting of the internal points, the formula is a little more complicated than for the trapezoidal rule:



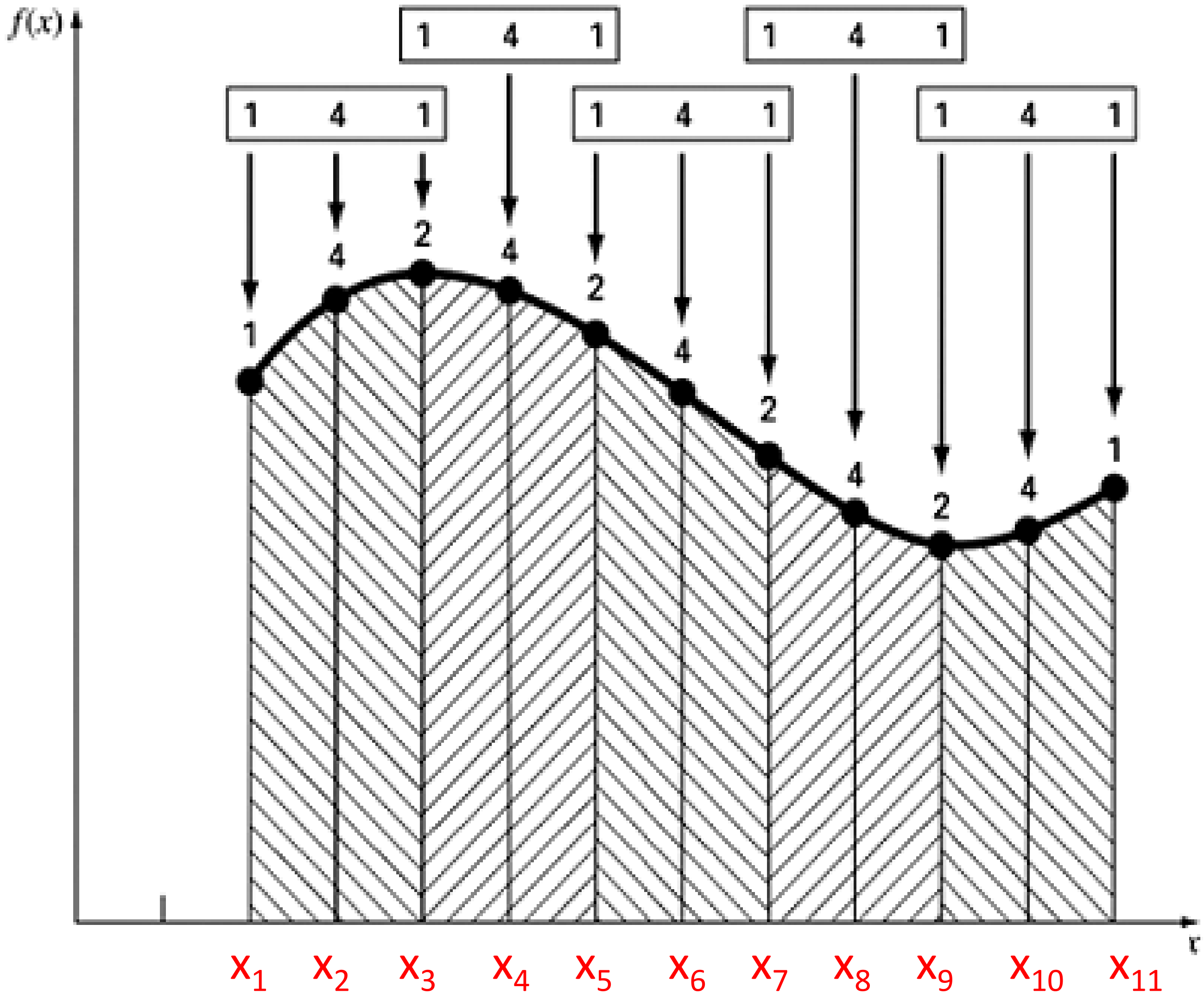
$$I = \int_{x_1}^{x_n} f(x)dx = \int_{x_1}^{x_3} f(x)dx + \int_{x_3}^{x_5} f(x)dx + \cdots + \int_{x_{n-2}}^{x_n} f(x)dx$$

$$I = \frac{h}{3}[f(x_1) + 4f(x_2) + f(x_3)] + \frac{h}{3}[f(x_3) + 4f(x_4) + f(x_5)] + \ldots$$

$$+ \frac{h}{3}[f(x_{n-2}) + 4f(x_{n-1}) + f(x_n)]$$

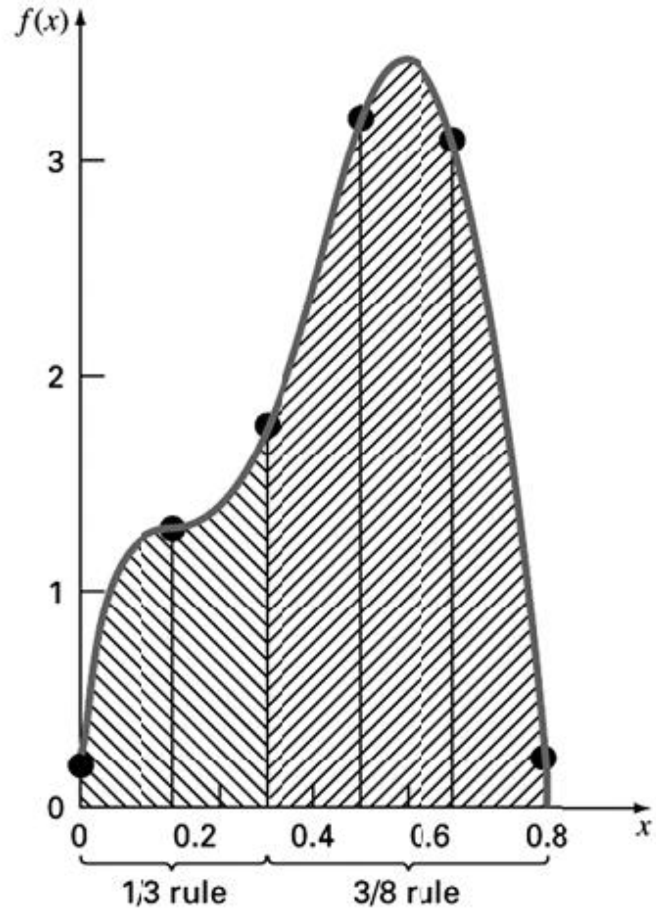
$$I = \frac{h}{3}\left[f(x_1) + 4\sum_{\substack{i=2 \\ i,even}}^{n-1} f(x_i) + 2\sum_{\substack{j=3 \\ j,odd}}^{n-2} f(x_j) + f(x_n) \right]$$

Numerical Integration Formulas



Numerical Integration Formulas

- Simpson's 3/8 rule corresponds to using third-order polynomials to fit four points. When this is done we get the following:
- $I = \int_{x_1}^{x_4} f(x)dx$
- $I = \frac{3h}{8}[f(x_1) + 3f(x_2) + 3f(x_3) + f(x_4)]$
- Simpson's 3/8 rule is generally used in concert with Simpson's 1/3 rule when the number of segments is odd (number of points is even).



Numerical Integration Formulas

➢ Integration with Unequal Segments

- Previous formulas were simplified based on equally spaced data points. When this is not the case we need to apply the basic formula over each segment.
- When the trapezoidal rule is applied over data containing unequal segments we get the following:

$$I = \int_{x_1}^{x_n} f(x)dx$$

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

Basic formula for each interval

$$I = \int_{x_1}^{x_2} f(x)dx + \int_{x_2}^{x_3} f(x)dx + \ldots + \int_{x_{n-1}}^{x_n} f(x)dx$$

$$I = (x_2 - x_1)\frac{f(x_1)+f(x_2)}{2} + (x_3 - x_2)\frac{f(x_2)+f(x_3)}{2} + \ldots + (x_n - x_{n-1})\frac{f(x_{n-1})+f(x_n)}{2}$$

Numerical Integration Formulas

- **MATLAB Functions**

- MATLAB has built-in functions to evaluate integrals based on the trapezoidal rule

z = trapz(y)

z = trapz(x, y)

The values of x do not have to be evenly spaced

Produces the integral of y with respect to x. If x is omitted, the program assumes h=1.

z = cumtrapz(y)

z = cumtrapz(x, y)

Produces the comulative integral of y with respect to x. If x is omitted, the program assumes h=1.

- Example: Evaluate the integration $\int_{-2}^{4}(1-x-4x^3+2x^5)dx$

```
>> f = @(x) 1-x-4*x.^3+2*x.^5
>> x = linspace(-2,4,10);
>> y = f(x);
>> trapz(x,y)
>> cumtrapz(x,y)
```

Numerical Integration Formulas

- Example: Evaluate the following integration $\int_{-2}^{4}(1 - x - 4x^3 + 2x^5)dx$

```
clc
clear
f = @(x) 1-x-4*x.^3+2*x.^5;            % Function f(x)

fi = @(x) x-x.^2/2-x.^4+x.^6/3;        % Exact integration of f(x)

n = input('Enter the number of points to use:  ');

if n<= 1
    n = 2
end

a = -2;
b = 4;

IE = fi(b) – fi(a);                     % Exact value of integration

x = linspace(a,b,n);
y = f(x);

h = x(2) – x(1);                        %  Distance between points

if n==2
   I = h*(y(1)+y(2))/2;
else
   I = (h/2)*(y(1)+2*sum(y(2:n-1))+y(n));
end

fprintf('The calculated area is equal to = %.4f\n', I);
fprintf('The exact area is equal to = %.4f\n', IE);
```

- Two data points:

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

- More points:

$$I = \frac{h}{2}\left[f(x_1) + 2\sum_{i=2}^{n-1} f(x_i) + f(x_n)\right]$$

- **Homework on Canvas**