Numerical Quadrature: Ch. 6

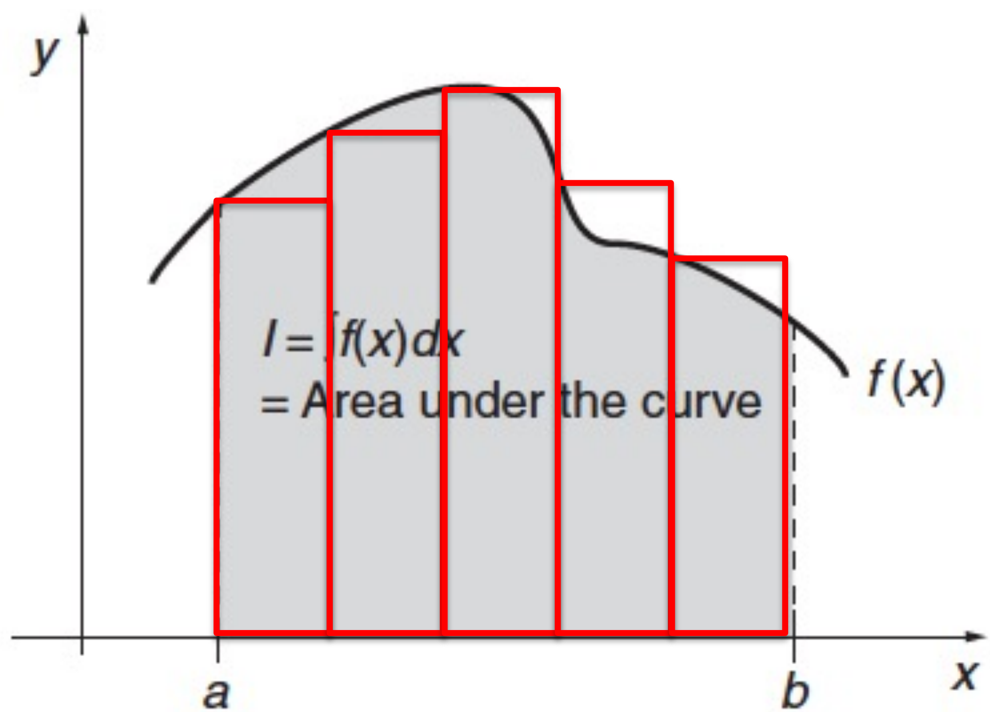
- We now look at two related topics: quadrature and integration.
- Quadrature: evaluation of integral of a known function over a specified domain.
- Integration: integration of a set of differential equations.
- Let's look at quadrature first. We want to solve:

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

Numerical Quadrature: Ch. 6

- How do we do this? We evaluate f(x) at several points in the domain [a,b] and combine them to estimate I.
- Thus:

Graphical interpretation of the process of integration.



Numerical Quadrature: Ch. 6

- We sum the area of the rectangles as an estimate of I:

$$I = \sum^{n}_{i=1} w_i f(x_i) + R_n$$

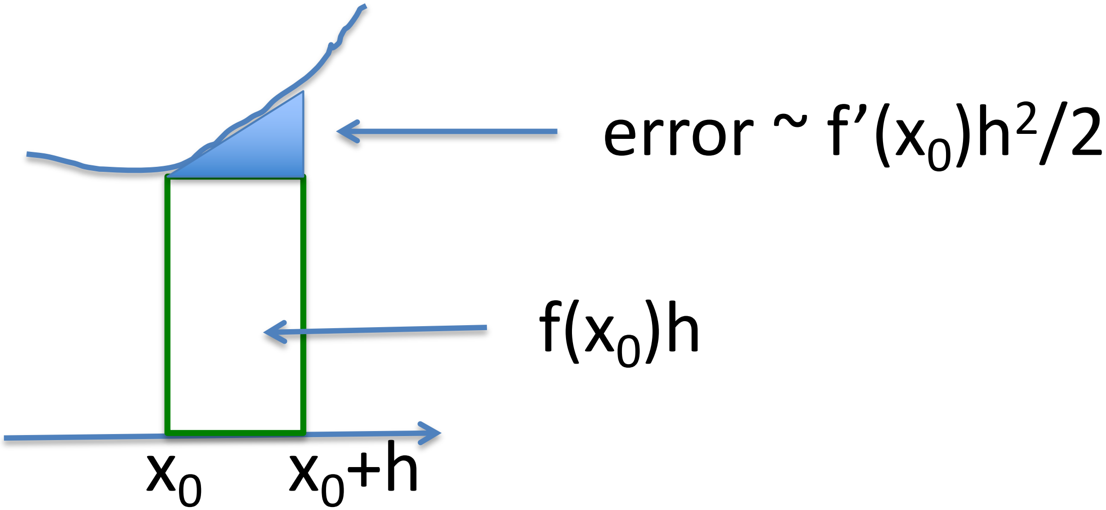
w_i = weights, x_i = nodes, R_n = error in expansion

- In this case, with n <u>panels</u>, we have:

$$w_i = (b - a)/n, \quad x_i = a + (b - a)/n*(i - 1)$$

- What is R_n? We make some error in each interval…

Numerical Quadrature: Ch. 6



- Thus, in each panel the error is $O(f'h^2/2)$
- The number of panels is $n = (b - a)/h$

  h: panel width

- Thus the total error is: $O(f'h(b - a))$

Numerical Quadrature

- So the rule is of order $h^{(1)}$

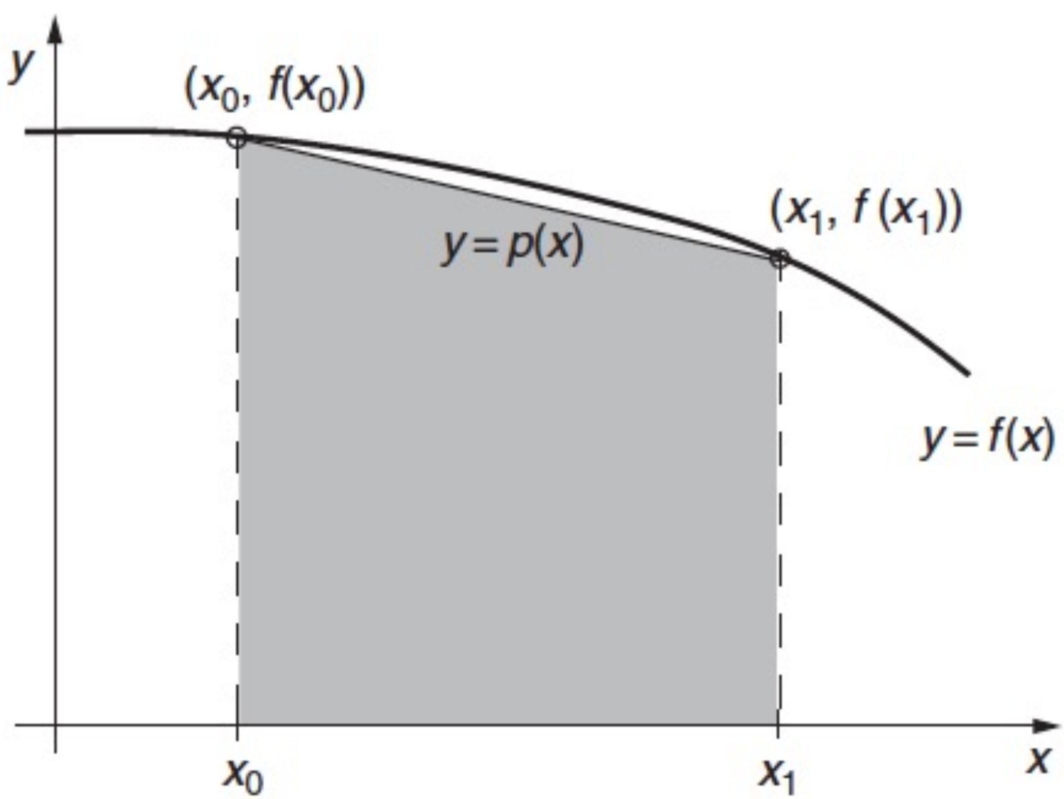
$$R_n = O((b-a)^2/n*f')$$

- If $f' = 0$ (f = constant) then the rule is exact ($R_n=0$)
- We call a quadrature rule of polynomial degree d if it gets all polynomials of degree d <u>exactly</u>, but makes errors in polynomials of degree (d+1).
- This rule was of degree <u>zero</u>.

Trapezoidal Rule

- What other rules are there? How about the <u>trapezoidal rule</u>?

The trapezoidal rule for numerical integration. The shaded area is a trapezoid.



Trapezoidal Rule

$$\int_a^b f(x)dx \approx \frac{h}{2}(f(x_0) + f(x_1)).$$

- What is the error?
- We are approximating f(x) with a <u>line</u> from $f(x_0)$ to $f(x_0+h)$. The error in this is proportional to the curvature, or f'' !
- error ~ $O(h^3 f'')$
- We require h^3 for dimensions to work out.

Trapezoidal Rule

- Since the number of intervals is again:

$$n = (b - a)/h$$

- We get for the total error:

$$R_n = O((b - a)^3/n^2\, f'') = O((b - a)h^2\, f'')$$

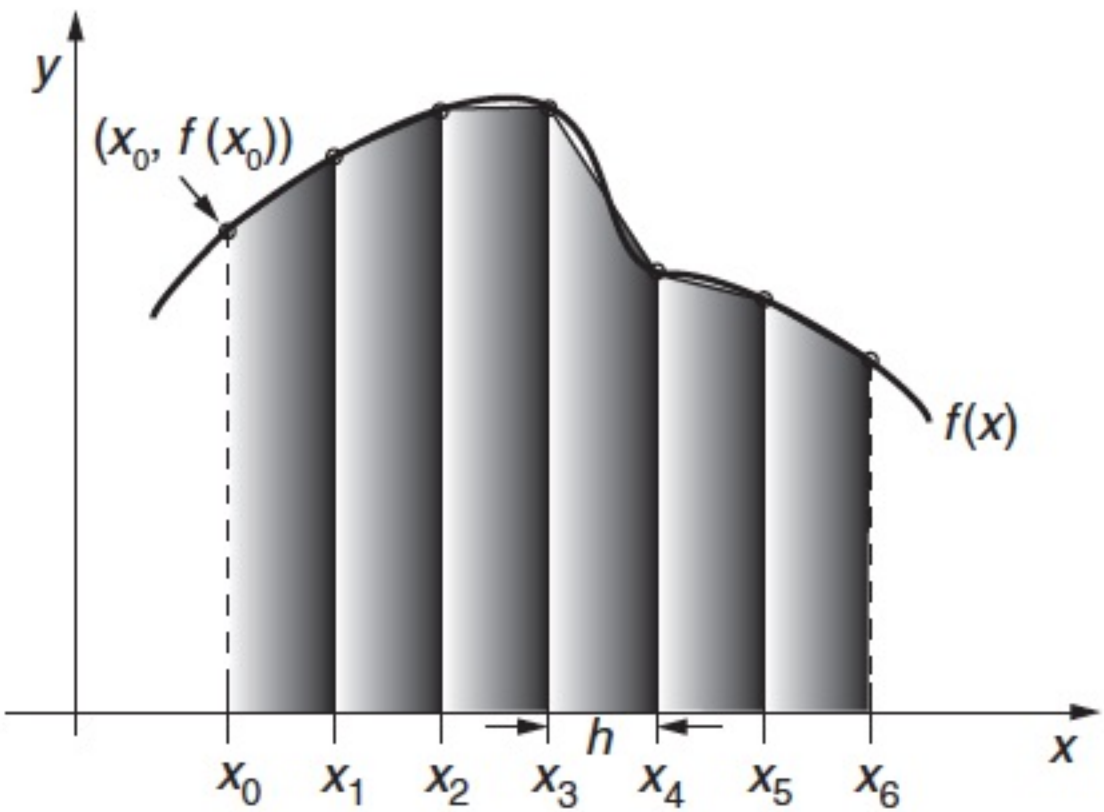
with $w_i = (b - a)/n * \begin{cases} ½ & i = 0,n \\ 1 & 1 <= I <= n - 1 \end{cases}$

$$x_i = a + (b - a)/n*i$$

Trapezoidal Rule

- The trapezoidal rule is of polynomial degree 1 because it gets lines correct but makes errors on quadratic functions.

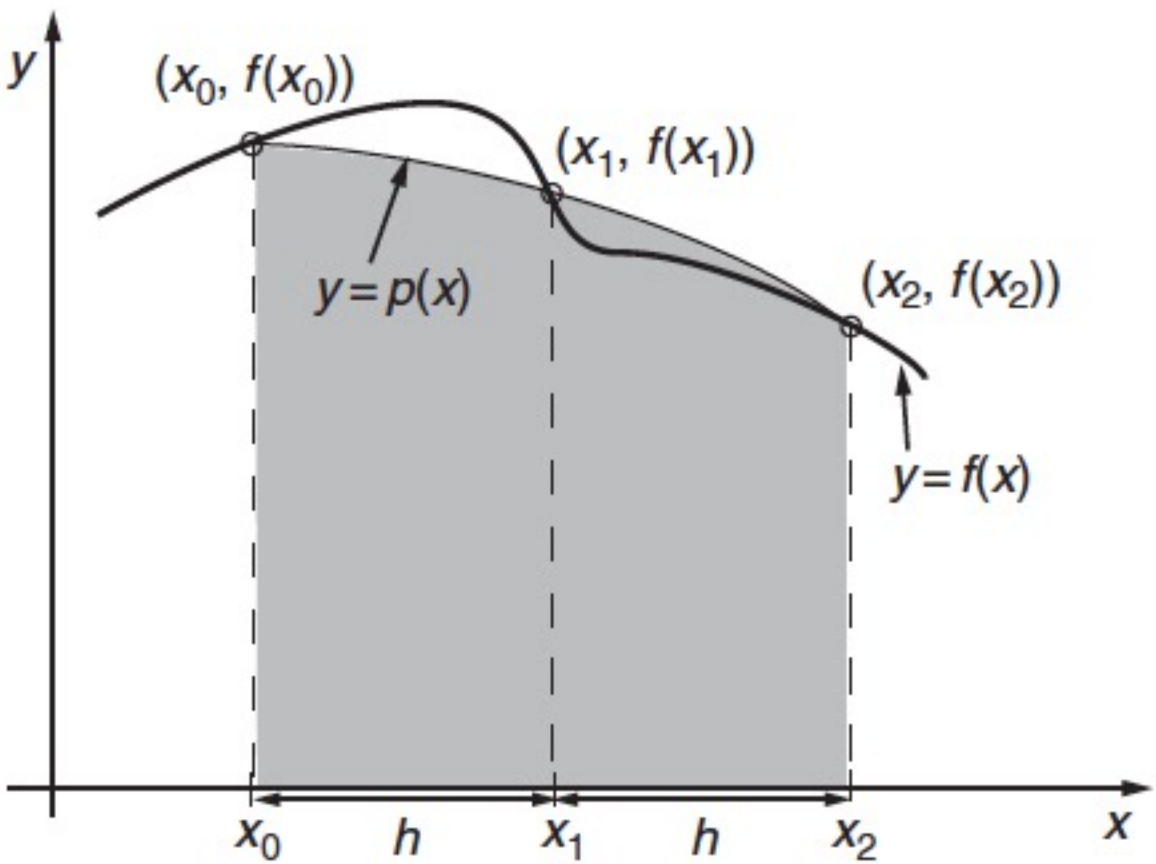
The composite trapezoidal rule. The six shaded areas each have the shape of a trapezoid.



Simpson's Rule

- We can do better using Simpson's (1/3) Rule:

Graphical description of Simpson's 1/3 rule.



- We take a <u>pair</u> of panels with <u>three</u> function evaluations.

- We fit a parabola through these three points…

Simpson's Rule

$$Q(x) = f(x_0 – h)(x – x_0)(x – x_0 – h)/(2h^2)$$
$$+ f(x_0)(x – x_0 – h)(x – x_0 + h)/(-h^2)$$
$$+ f(x_0+h)(x – x_0 + h)(x – x_0)/(2h^2)$$

- So we integrate $Q(x)$ over $x_0–h$ to x_0+h to get the quadrature rule.
- $I \sim f(x_0 – h)w_1 + f(x_0)w_2 + f(x_0+h)w_3$

  where $w_1 = w_3 = h/3 \quad w_2 = 4h/3$

Simpson's Rule

- Thus, in general,

$$\int_a^b f(x)dx \approx \frac{h}{3}\left(f(x_0) + 2\sum_{i=1}^{(n/2)-1} f(x_{2i}) + 4\sum_{i=1}^{n/2} f(x_{2i-1}) + f(x_n)\right).$$

- The error is:

$$E = -\frac{h^4(b-a)}{180}\overline{f^{(4)}},$$

- So Simpson's rule is of polynomial degree <u>3</u>.
- It gets cubics right even though we fit it with a parabola. This occurs because of the <u>symmetry</u> of the rule.

Simpson's Rule

OK, let's try this rule out:

$$\int_0^2 x^3 dx = \frac{1}{4}(2^4 - 0) = \frac{16}{4} = 4$$

We use a 2 panel approximation:

$$I \approx \frac{2-0}{6}(f(0) + 4f(1) + f(2)) = \frac{24}{6} = 4$$ no error

This is because Simpson's rule gets all cubics exactly!

$f^{IV}(x) = 0$

Now let's try integrating x^4:

$$\int_0^2 x^4 dx = \frac{1}{5}(2^5 - 0) = \frac{32}{5} = 6.4$$

So:

$$I \cong \frac{2-0}{6}(f(0) + 4f(1) + f(2)) = \frac{40}{6} = 6.67$$

The error is thus $0.266\ldots = \frac{4}{15}$

Let's cut h in half:

$$\int_0^2 x^4 dx \approx \frac{1}{6}\left(0 + 4\left(\tfrac{1}{2}\right)^4 + 2(1)^4 + 4\left(\tfrac{3}{2}\right)^4 + (2)^4\right)$$

$$= \frac{77}{12} = 6.417$$

The error is now only $0.0166\ldots = \frac{1}{60}$ which is much smaller.

Note that when we halved the interval we decreased the error by a factor of 16. This was because the error was $O(h^4)$

Gaussian Quadrature

- Suppose we want to integrate

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

  using only two points.
- Where do we put them?
- IF we use the trapezoidal rule we put them at a and b. This is <u>not</u> the optimum choice!
- Instead we let:

$$I \sim w_1 f(x_1) + w_2 f(x_2)$$

- And choose all <u>four</u> parameters such that we can integrate the highest degree polynomial possible!

Gaussian Quadrature

- We have <u>4</u> parameters, so we can integrate an arbitrary cubic polynomial with 4 constants!

  Let $m = (a+b)/2$

- We want to pick x_1, x_2, w_1, w_2 such that we integrate without error!

  $f(x) = (x - m)^0 \quad (=1)$

  $f(x) = (x - m)^1$

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

  $f(x) = (x - m)^3$

- Any cubic is a linear combination of these our functions. If we get these right, we get them all right!

Box 8.2A Optimization of a fermentation process: maximization of profit

A first-order irreversible reaction A $\rightarrow$ B with rate constant k takes place in a well-stirred fermentor tank of volume V. The process is at steady state, such that none of the reaction variables vary with time. W_B is the amount of product B in kg produced per year, and p is the price of B per kg. The total annual sales is given by \$$pW_B$.

The mass flowrate of broth through the processing system per year is denoted by W_{in}. If Q is the flowrate (m^3/hr) of broth through the system, ρ is the broth density, and the number of hours of operation annually is 8000, then $W_{in} = 8000\rho Q$. The annualized capital (investment) cost of the reactor and the downstream recovery equipment is \$$4000W_{in}^{0.6}$.

The operating cost c is \$15.00 per kg of B produced. If the concentration of B in the exiting broth is b_{out} in molar units, then we have $W_B = 8000Qb_{out}MW_B$.

A steady state material balance incorporating first-order reaction kinetics gives us

$$b_{out} = a_{in}\left(1 - \frac{1}{1 + kV/Q}\right),$$

where V is the volume of the fermentor and a_{in} is the inlet concentration of the reactant. Note that as Q increases, the conversion will decrease.

The profit function is given by

$$(p - c)W_B - 4000(W_{in})^{0.6},$$

and in terms of the unknown variable Q, it is as follows:

$$f(Q) = 8000(p - c)MW_B Q a_{in}\left(1 - \frac{1}{1 + (kV/Q)}\right) - 4000(8000\rho Q)^{0.6}. \tag{8.2}$$

Initially, as Q increases the product output will increase and profits will increase. With subsequent increases in Q, the reduced conversion will adversely impact the product generation rate while annualized capital costs continue to increase.

You are given the following values:

p = \$400 per kg of product,
$\rho = 2.5$ kg/m^3,
$a_{in} = 10$ μM (1 M $\equiv$ 1 kmol/m^3),
$k = 2$/hr,
$V = 12$ m^3,
$MW_B = 70\,000$ kg/kmol.

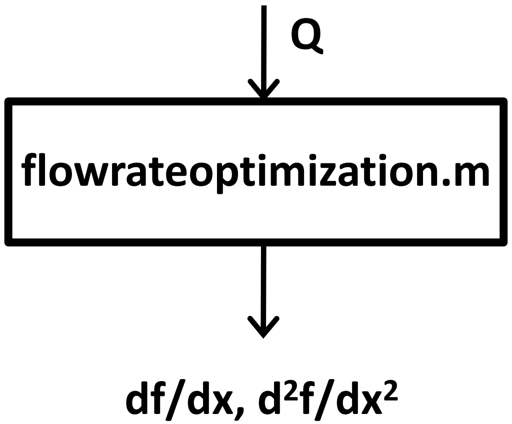
Determine the value of Q that maximizes the profit function. This problem was inspired by a discussion on reactor optimization in Chapter 6 of Nauman (2002).

Q1: If the derivatives of an objective function are difficult to calculate, which method is NOT a good choice to solve the fermentation optimization?

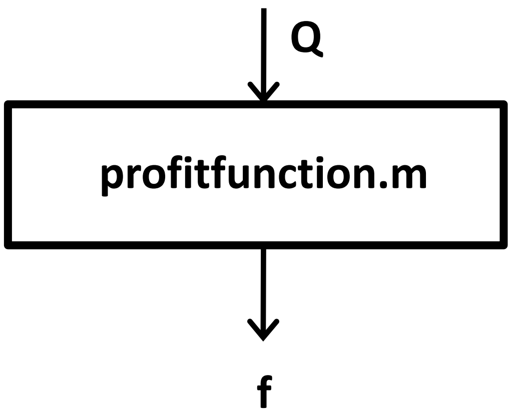
A. Newton’s Method
B. Successive Parabolic Interpolation
C. The Golden Search

Q2: Which function should be utilized when applying the method of successive parabolic interpolation?

A



B



Q3. Here is the convergence using Newton's Method:

```
>>
newtons1Doptimization('flowrateoptimizatio
n',50,.01)
 i    x(i)      f'(x(i))      f''(x(i))
 1  57.757884  5407.076741  -3295.569935
 2  59.398595  178.749716  -3080.622008
 3  59.456619  0.211420  -3073.338162
 4  59.456687  0.000000  -3073.329539
```

How many iterations would it take the Golden Search to converge on this solution to within 0.01, based on an initial interval of [40, 70] ?

A. 4

B. 10

C. 17

D. infinite (no convergence)

Q4. Here is the convergence using Golden Search:

```
>>
goldensectionsearch('profitfunction',[40
70],.01)
a = 59.453920 b = 59.462321 f(a) = -
19195305.986695 f(b) = -19195305.949685
number of iterations = 17
```

How many iterations would it take successive parabolic interpolation to converge on this solution to within 0.01, based on an initial interval of [40, 70] ?

- **A. 4**
- **B. 10**
- **C. 17**
- **D. infinite (no convergence)**

Here is the convergence using successive parabolic interpolation:

```
>>
paraboliciinterpolation('profitfunction',[4
0 70],.01)
x0 = 59.456689 x2 = 59.456699 f(x0) = -
19195305.998460 f(x2) = -19195305.998460
x_min = 59.456689 f(x_min) = -
19195305.998460
number of iterations = 10
```