Math 128A Spring 2002 Handout # 24
Sergey Fomel April 23, 2002

Answers to Homework 9: Numerical Integration

---

1. (a) Suppose that the function

$$f(x) = \frac{2}{1+x^2} \tag{1}$$

is known at three points: $x_1 = -1$, $x_2 = 0$, and $x_3 = 1$. Interpolate the function with a natural cubic spline and approximate the integral

$$\int\limits_{-1}^{1} \frac{2\,dx}{1+x^2} = \pi \tag{2}$$

by the integral of the spline. Is the result more accurate than the result of Simpson's rule?

Answer:

Let the spline on the second interval $x \in [0,1]$ be

$$S_2(x) = 2 + b\,x + c\,x^2 + d\,x^3\;.$$

By the symmetry of the boundary conditions, the corresponding spline on the first interval is

$$S_1(x) = S_2(-x) = 2 - b\,x + c\,x^2 - d\,x^3\;.$$

The interpolation conditions require

$$S_1(-1) = S_2(1) = 2 + b + c + d = 1\;.$$

The natural spline boundary conditions require

$$S_1''(-1) = S_2''(1) = 2\,c + 6\,d\; = 0\;.$$

The continuity conditions require

$$-b = S_1'(0) = S_2'(0) = b\;,$$

which leads to $b = 0$. The condition

$$2\,c = S_1''(0) = S_2''(0)$$

is satisfied automatically. Solving the linear equations for c and d, we obtain $c = -3/2$ and $d = 1/2$. Therefore

$$S_2(x) = 2 - \frac{3}{2}\,x^2 + \frac{1}{2}\,x^3$$

The integral approximation is then

$$\int\limits_{-1}^{1} \frac{2\,dx}{1+x^2} \approx 2\int\limits_{0}^{1} S_2(x)\,dx = 2\left(2 - \frac{1}{2} + \frac{1}{8}\right) = \frac{13}{4} = 3.25$$

This value is more accurate that Simpson's result ($10/3 = 3.3333\ldots$)

(b) Let $Sp[f]$ be the approximation of the integral

$$\int_a^b f(x)\,dx \tag{3}$$

by the integral of the natural cubic spline defined at the knots $a = x_1 < x_2 < \ldots < x_n = b$. Prove that

$$Sp[f] = CT[f] - \sum_{k=2}^{n-1} c_k \, \frac{h_{k-1}^3 + h_k^3}{12} \,, \tag{4}$$

where $h_k = x_{k+1} - x_k$, c_k are the coefficients in the spline equation

$$S_k(x) = f(x_k) + b_k\,(x - x_k) + c_k\,(x - x_k)^2 + d_k\,(x - x_k)^3 \quad \text{for} \quad x_k \le x < x_{k+1} \,, \tag{5}$$

and $CT[f]$ is the composite trapezoidal rule.

Answer:

The integral of the spline is the sum of the local integrals:

$$Sp[f] = \int_a^b S(x)\,dx = \sum_{k=1}^{n-1} \int_{x_k}^{x_{k+1}} S_k(x)\,dx \,.$$

Integrating the natural cubic spline (5) on the interval $[x_k, x_{k+1}]$, we obtain

$$\int_{x_k}^{x_{k+1}} S_k(x)\,dx = f_k\,h_k + b_k\,\frac{h_k^2}{2} + c_k\,\frac{h_k^3}{3} + d_k\,\frac{h_k^4}{4} \,.$$

With the help of the recursive relationships

$$d_k = \frac{c_{k+1} - c_k}{3\,h_k}$$

and

$$b_k = \frac{f_{k+1} - f_k}{h_k} - \frac{h_k}{3}\,(2\,c_k + c_{k+1}) \,,$$

the integral transforms to

$$\begin{aligned} \int_{x_k}^{x_{k+1}} S_k(x)\,dx \quad &= \quad f_k\,h_k + \frac{h_k}{2}\,(f_{k+1} - f_k) - \frac{h_k^3}{6}\,(2\,c_k + c_{k+1}) + c_k\,\frac{h_k^3}{3} + \frac{h_k^3}{12}\,(c_{k+1} - c_k) \\ &= \quad \frac{h_k}{2}\,(f_{k+1} + f_k) - \frac{h_k^3}{12}\,(c_k + c_{k+1}) \,. \end{aligned}$$

The first term in the last expression is exactly the trapezoidal rule on the interval $[x_k, x_k + 1]$. Therefore,

$$Sp[f] = CT[f] - \sum_{k=1}^{n-1} \frac{h_k^3}{12}\,(c_k + c_{k+1}) \,.$$

Recalling that the natural cubic spline has $c_1 = c_n = 0$, we can transform the last sum as follows:

$$\sum_{k=1}^{n-1} \frac{h_k^3}{12}\,(c_k + c_{k+1}) = \sum_{k=1}^{n-1} \frac{h_k^3}{12}\,c_k + \sum_{k=1}^{n-1} \frac{h_k^3}{12}\,c_{k+1} = \sum_{k=2}^{n-1} \frac{h_k^3}{12}\,c_k + \sum_{k=2}^{n-1} \frac{h_{k-1}^3}{12}\,c_k = \sum_{k=2}^{n-1} c_k\,\frac{h_{k-1}^3 + h_k^3}{12} \,.$$

The final expression takes the form (4).

2. (a) Suppose that function (1) is known at three points: $x_1 = -1$, $x_2 = 0$, and $x_3 = 1$ together with its derivatives. Interpolate the function using Hermite interpolation and approximate integral (2) by the integral of the interpolation polynomial. Is the result more accurate than the result of Simpson's rule?

Answer:

The derivative of $f(x)$ is

$$f'(x) = -\frac{4x}{(1+x^2)^2} \, .$$

The divided-difference table for Hermite interpolation is then

x	$f[\,]$	$f[\,,\,]$	$f[\,,\,,\,]$	$f[\,,\,,\,]$	$f[\,,\,,\,,\,]$	$f[\,,\,,\,,\,,\,]$
-1	1					
-1	1	$f'(-1) = 1$				
0	2	1	0			
0	2	$f'(0) = 0$	-1	-1		
1	1	-1	-1	0	$1/2$	
1	1	$f'(1) = -1$	0	1	$1/2$	0

which leads to the Hermite interpolation polynomial

$$\begin{aligned} P(x) &= 1 + (x+1) + 0 \times (x+1)^2 - (x+1)^2\, x + \frac{1}{2}\,(x+1)^2\, x^2 + 0 \times (x+1)^2\, x^2\, (x-1) \\ &= 2 - \frac{3}{2}\,x^2 + \frac{1}{2}\,x^4 \end{aligned}$$

Integrating this polynomial, we obtain

$$\int\limits_{-1}^{1} P(x)\,dx = 2\int\limits_{0}^{1} P(x)\,dx = 2\,(2 - \frac{1}{2} + \frac{1}{10}) = \frac{16}{5} = 3.2$$

This result is more accurate than the result of Simpson's integration $(10/3 = 3.3333\ldots)$

(b) Let $CH[f]$ be the approximation of integral (3) by the integral of the piece-wise cubic polynomial defined by applying Hermite interpolation at each of the intervals $[x_k, x_{k+1}]$, $a = x_1 < x_2 < \ldots < x_n = b$.

Prove that

$$CH[f] = CT[f] + b_1\,\frac{h_1^2}{12} - b_n\,\frac{h_{n-1}^2}{12} + \sum_{k=2}^{n-1} b_k\,\frac{h_k^2 - h_{k-1}^2}{12} \, , \qquad (6)$$

where $h_k = x_{k+1} - x_k$, $b_k = f'(x_k)$, and $CT[f]$ is the composite trapezoidal rule.

Answer:

The cubic Hermite interpolation in the interval $[x_k, x_{k+1}]$ follows from the divided-difference table

x	$f[\,]$	$f[\,,\,]$	$f[\,,\,,\,]$	$f[\,,\,,\,]$
x_k	f_k			
x_k	f_k	b_k		
x_{k+1}	f_{k+1}	$\frac{f_{k+1}-f_k}{h_k}$	$\left(\frac{f_{k+1}-f_k}{h_k}-b_k\right)\frac{1}{h_k}$	
x_{k+1}	f_{k+1}	b_{k+1}	$\left(b_{k+1}-\frac{f_{k+1}-f_k}{h_k}\right)\frac{1}{h_k}$	$\left(b_{k+1}+b_k-2\,\frac{f_{k+1}-f_k}{h_k}\right)\frac{1}{h_k^2}$

which leads to the cubic interpolation polynomial

$$\begin{aligned} P_k(x) &= f_k + b_k\,(x-x_k) + \left(\frac{f_{k+1}-f_k}{h_k} - b_k\right)\frac{(x-x_k)^2}{h_k} \\ &\quad + \left(b_{k+1}+b_k-2\,\frac{f_{k+1}-f_k}{h_k}\right)\frac{(x-x_k)^2\,(x-x_{k+1})}{h_k^2}\,. \end{aligned}$$

Integrating on the interval $[x_k, x_{k+1}]$, we obtain

$$\begin{aligned} \int\limits_{x_k}^{x_{k+1}} P_k(x)\,dx &= \int\limits_0^{h_k} P_k(t+x_k)\,dt \\ &= f_k + b_k \int\limits_0^{h_k} t\,dt + \left(\frac{f_{k+1}-f_k}{h_k} - b_k\right)\frac{1}{h_k}\int\limits_0^{h_k} t\,dt \\ &\quad + \left(b_{k+1}+b_k-2\,\frac{f_{k+1}-f_k}{h_k}\right)\frac{1}{h_k^2}\int\limits_0^{h_k} t^2\,(t-h_k)\,dt \\ &= f_k + b_k\,\frac{h_k^2}{2} + \left(\frac{f_{k+1}-f_k}{h_k} - b_k\right)\frac{h_k^2}{3} - \left(b_{k+1}+b_k-2\,\frac{f_{k+1}-f_k}{h_k}\right)\frac{h_k^2}{12} \\ &= \frac{h_k}{2}\,(f_{k+1}+f_k) + \frac{h_k^2}{12}\,(b_k - b_{k+1})\,. \end{aligned}$$

The first term in the last expression is exactly the trapezoidal rule on the interval $[x_k, x_k+1]$. The total integral is, therefore,

$$CH[f] = \sum_{k=1}^{n-1} \int\limits_{x_k}^{x_{k+1}} P_k(x)\,dx = CT[f] + \sum_{k=1}^{n-1} \frac{h_k^2}{12}\,(b_k - b_{k+1})\,.$$

The last sum transforms as follows:

$$\begin{aligned} \sum_{k=1}^{n-1} \frac{h_k^2}{12}\,(b_k - b_{k+1}) &= \sum_{k=1}^{n-1} \frac{h_k^2}{12}\,b_k - \sum_{k=1}^{n-1} \frac{h_k^2}{12}\,b_{k+1} = \frac{h_1^2}{12}\,b_1 + \sum_{k=2}^{n-1} \frac{h_k^2}{12}\,b_k - \sum_{k=1}^{n-2} \frac{h_k^2}{12}\,b_{k+1} - \frac{h_{n-1}^2}{12}\,b_n \\ &= b_1\,\frac{h_1^2}{12} - b_n\,\frac{h_{n-1}^2}{12} + \sum_{k=2}^{n-1} b_k\,\frac{h_k^2 - h_{k-1}^2}{12}\,. \end{aligned}$$

The final expression takes the form (6).

3. (a) The *Gauss-Lobatto* quadrature rule has the form

$$\int_{-1}^{1} f(x)\,dx \approx w_1\, f(-1) + w_n\, f(1) + \sum_{k=2}^{n-1} w_k\, f(x_k) = Lo[\,f\,]\,, \tag{7}$$

where the abscissas x_k and weights w_k are chosen so that $f(x)$ is integrated exactly if it is a polynomial of order $2n-3$ or less.

Derive the abscissas and weights for $n = 3$ and $n = 4$ and apply the Gauss-Lobatto rule to integral (2). Are the results more accurate than the result of Simpson's rule?

Hint: Use the symmetry of the interval to constrain the abscissas and weights.

Answer:

First, let us consider the case $n = 3$. By the symmetry of the interval, the point x_2 has to be in the center, and we can immediately set $x_2 = 0$. Integrating polynomials of the first four powers, we obtain

$$\begin{aligned} I[1] &= 2 = w_1 + w_2 + w_3 \\ I[x] &= 0 = -w_1 + w_3 \\ I[x^2] &= \frac{2}{3} = w_1 + w_3 \\ I[x^3] &= 0 = -w_1 + w_3 \end{aligned}$$

The last two equations define $w_1 = w_3 = 1/3$. From the first equation, we get

$$w2 = 2 - w_1 - w_3 = 4/3$$

The rule is then

$$Lo_3[\,f\,] = \frac{1}{3}\left[\,f(-1) + 4\,f(0) + f(1)\right]\,,$$

which is equivalent to Simpson's rule applied on the interval $[-1,1]$. In the example problem, the result is equivalent to that of Simpson's rule: $Lo_3[\,f\,] = \frac{10}{3} = 3.3333\ldots$

In the case of $n = 4$, the symmetry suggests $x_2 = -x_3 = x$ and $w_2 = w_3 = w$. Integrating polynomials of the first six powers, we obtain

$$\begin{aligned} I[1] &= 2 = w_1 + 2\,w + w_4 \\ I[x] &= 0 = -w_1 + w_4 \\ I[x^2] &= \frac{2}{3} = w_1 + 2\,w\,x^2 + w_4 \\ I[x^3] &= 0 = -w_1 + w_4 \\ I[x^4] &= \frac{2}{5} = w_1 + 2\,w\,x^4 + w_4 \\ I[x^5] &= 0 = -w_1 + w_4 \end{aligned}$$

Subtracting the third equation from the first one yields

$$2\,w\,(1 - x^2) = 2 - \frac{2}{3} = \frac{4}{3}$$

Subtracting the fifth equation from the third one yields

$$2\,w\,x^2\,(1-x^2) = \frac{2}{3} - \frac{2}{5} = \frac{4}{15}$$

Dividing by the previously equation, we get

$$x^2 = \frac{1}{5}$$

Therefore, $x_2 = -1/\sqrt{5}$ and $x_3 = 1/\sqrt{5}$. Next,

$$2\,w\,(1-x^2) = \frac{2}{5}\,w = \frac{4}{3}$$

and $w = 5/6$. Solving the remaining equations for w_1 and w_4, we get $w_1 = w_4 = 1/6$. The rule is then

$$Lo_4[f] = \frac{1}{6}\left[f(-1) + 5\,f\left(-\frac{1}{\sqrt{5}}\right) + 5\,f\left(\frac{1}{\sqrt{5}}\right) + f(1)\right] .$$

Applying it to the example problem yields

$$Lo_4[f] = \frac{1}{3}\left[1 + 5\,\frac{2}{1+\frac{1}{5}}\right] = \frac{28}{9} = 3.1111\ldots$$

This is more accurate than Simpson's result.

(b) The *Gauss-Laguerre* quadrature rule has the form

$$\int_0^\infty e^{-x}\,f(x)\,dx \approx \sum_{k=1}^{n} w_k\,f(x_k) = La[f]\,, \tag{8}$$

where the abscissas x_k and weights w_k are chosen so that $f(x)$ is integrated exactly if it is a polynomial of order $2\,n-1$ or less.

Derive the abscissas and weights for $n=1$ and $n=2$. Test your formulas by approximating π with

$$\pi = \left[2\,\Gamma(3/2)\right]^2 \approx \left(2\,La[\sqrt{x}]\right)^2\,, \tag{9}$$

where

$$\Gamma(x) = \int_0^\infty t^{x-1}\,e^{-t}\,dt \tag{10}$$

Answer:

First, let us consider the case $n=1$. Integrating polynomials of the first two powers and applying the general formula

$$I[x^k] = \int_0^\infty e^{-x}\,x^k\,dx = k!$$

we obtain

$$
\begin{aligned}
I[1] &= 1 = w_1 \\
I[x] &= 1 = w_1\, x_1
\end{aligned}
$$

The rule is then simply

$$La_1[f] = f(1)$$

In the example problem, we get

$$\pi \approx \left(2\, La_1[\sqrt{x}]\right)^2 = 4\,.$$

In the case $n = 2$, we need to integrate the first four powers, obtaining

$$
\begin{aligned}
I[1] &= 1 = w_1 + w_2 \\
I[x] &= 1 = w_1\, x_1 + w_2\, x_2 \\
I[x^2] &= 2 = w_1\, x_1^2 + w_2\, x_2^2 \\
I[x^3] &= 6 = w_1\, x_1^3 + w_2\, x_2^3
\end{aligned}
$$

Manipulating these equations by multiplications and subtractions, we can simplify them as follows:

$$
\begin{aligned}
(w_1 + w_2)(w_1\, x_1^3 + w_2\, x_2^3) - (w_1\, x_1 + w_2\, x_2)(w_1\, x_1^2 + w_2\, x_2^2) &= w_1\, w_2\, (x_1 - x_2)^2\, (x_1 + x_2) = 4 \\
(w_1\, x_1 + w_2\, x_2)(w_1\, x_1^3 + w_2\, x_2^3) - (w_1\, x_1^2 + w_2\, x_2^2)^2 &= w_1\, w_2\, x_1\, x_2\, (x_1 - x_2)^2 = 2 \\
(w_1 + w_2)(w_1\, x_1^2 + w_2\, x_2^2) - (w_1\, x_1 + w_2\, x_2)^2 &= w_1\, w_2\, (x_1 - x_2)^2 = 1
\end{aligned}
$$

Dividing the first and second equations by the last one yields the system of equations

$$
\begin{aligned}
x_1 + x_2 &= 4 \\
x_1\, x_2 &= 2
\end{aligned}
$$

with the solution $x_1 = 2 - \sqrt{2}, x_2 = 2 + \sqrt{2}$.

An easier way to get this solution is by considering polynomials orthogonal on the interval $[0, \infty]$ with the weight $w(x) = e^{-x}$ (*Laguerre* polynomials). We constructed a more general case of these polynomials in Homework 7, Problem 1(b). The Laguerre polynomial of the second order is (up to scaling by a constant)

$$P(x) = x^2 - 4x + 2$$

and its zeroes are $x_1 = 2 - \sqrt{2}$ and $x_2 = 2 + \sqrt{2}$.

Equations for the unknowns weights w_1 and w_2 are

$$
\begin{aligned}
w_1 + w_2 &= 1 \\
w_1 \left(2 - \sqrt{2}\right) + w_2 \left(2 + \sqrt{2}\right) &= 1
\end{aligned}
$$

with the solution

$$
\begin{aligned}
w_1 &= \frac{2 + \sqrt{2}}{4} \\
w_2 &= \frac{2 - \sqrt{2}}{4}
\end{aligned}
$$

The rule is then

$$La_2[f] = \frac{2+\sqrt{2}}{4}\, f\left(2-\sqrt{2}\right) + \frac{2-\sqrt{2}}{4}\, f\left(2+\sqrt{2}\right) .$$

In the example problem, we get

$$\begin{aligned} La_2[\sqrt{x}] &= \frac{2+\sqrt{2}}{4}\sqrt{2-\sqrt{2}} + \frac{2-\sqrt{2}}{4}\sqrt{2+\sqrt{2}} \\ &= \frac{\sqrt{2+\sqrt{2}}}{4}\sqrt{2+\sqrt{2}}\sqrt{2-\sqrt{2}} + \frac{2-\sqrt{2}}{4}\sqrt{2+\sqrt{2}} = \frac{\sqrt{2+\sqrt{2}}}{2} \end{aligned}$$

and

$$\pi \approx \left(2\, La_2[\sqrt{x}]\right)^2 = 2+\sqrt{2} \approx 3.41421$$

4. (Programming) Implement the adaptive quadrature method. Test your program by computing integral (2) with the precision of six significant decimal digits. Plot or tabulate the values of x and $f(x)$ that were involved in the computation.

   The algorithm of adaptive integration can be defined either recursively

```
ADAPTIVE RECURSIVE(f(x), a, b, h, I)
if h < xtol
  then
        WARNING( 'did not converge' )
        return I
c ← a + h/2
I1 ← R(f, a, c)
I2 ← R(f, c, b)
D ← I2 + I1 − I
if |D| < itol
  then J ← I + α D
  else J ← ADAPTIVE RECURSIVE(f, a, c, h/2, I1) + ADAPTIVE RECURSIVE(f, c, b, h/2, I2)
return J
```

or sequentially

```
ADAPTIVE NONRECURSIVE(f(x), a, b, h, I)
J ← 0
PUSH(f, a, b, h, I)
while POP(f, a, b, h, I)
do
    if h < xtol
      then
            WARNING( 'did not converge' )
            return I
    c ← a + h/2
    I1 ← R(f, a, c)
    I2 ← R(f, c, b)
    D ← I2 + I1 − I
    if |D| < itol
      then J ← J + I + α D
      else
            PUSH(f, a, c, h/2, I1)
            PUSH(f, c, b, h/2, I2)
return J
```

Both algorithms involve the rule $R(f,a,b)$ for computing integral (3), the minimum allowed interval $xtol$ and the requested precision $itol$. They are initialized with $I = R(f,a,b)$ and $h = b-a$. The sequential algorithm operates a queue of intervals using a pair of functions PUSH and POP.

Run your program taking R to be the trapezoidal rule. What is the appropriate value of constant α?

Answer:

An appropriate value for α follows from Richardson's extrapolation. The error of the trapezoidal rule is

$$I[f] - T[f] = A h^3 + O(h^5)$$

where A does not depend on h. The composite trapezoidal rule applied on two intervals has the error

$$I[f] - CT[f] = 2\,A\,(h/2)^3 + O(h^5) = A\,\frac{h^3}{4} + O(h^5)$$

Multiplying the second equation by α and the first equation by $1-\alpha$ and adding them together, we get

$$I[f] - R[f] = (1-\alpha)\,A\,h^3 + \alpha\,A\,\frac{h^3}{4} + O(h^5)\,,$$

where

$$R[f] = (1-\alpha)\,T[f] + \alpha\,CT[f] = T[f] + \alpha\,(CT[f] - T[f])\;.$$

The error is minimized if

$$1 - \alpha + \frac{\alpha}{4} = 0$$

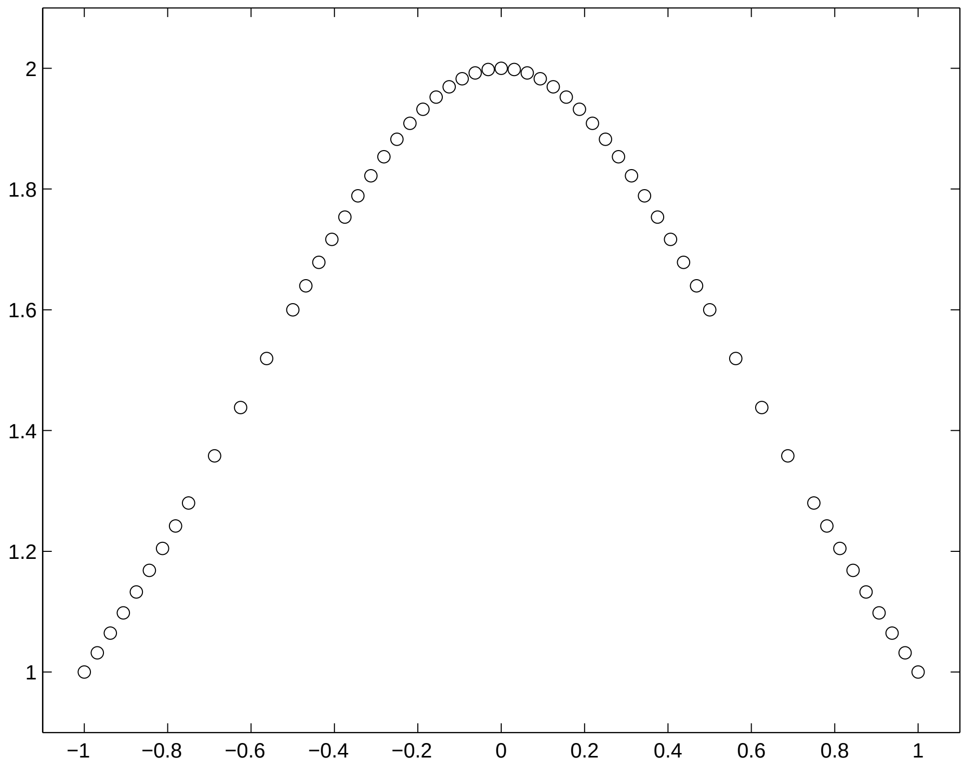
or

$$\alpha = \frac{4}{3}$$

The combination

$$R[f] = T[f] + \frac{4}{3}\left(CT[f] - T[f]\right)$$

is equivalent to Simpson's rule.



5. (Programming) The length of a parametric curve $\{x(t), y(t)\}$ is given by the integral

$$\int_a^b \sqrt{[x'(t)]^2 + \left[y'(t)\right]^2}\, dt \tag{11}$$

The curve that you interpolated in Homework 6 is close to the *hypotrochoid*

$$x(t) = \frac{3}{2}\cos t + \cos 3t\ ; \tag{12}$$

$$y(t) = \frac{3}{2}\sin t - \sin 3t\ , \tag{13}$$

defined on the interval $0 \le t \le 2\pi$.

Estimate the length of this curve to six significant decimal digits applying a numerical method of your choice.

Identify the method and plot the $\{x, y\}$ points involved in the computation.

Answer:

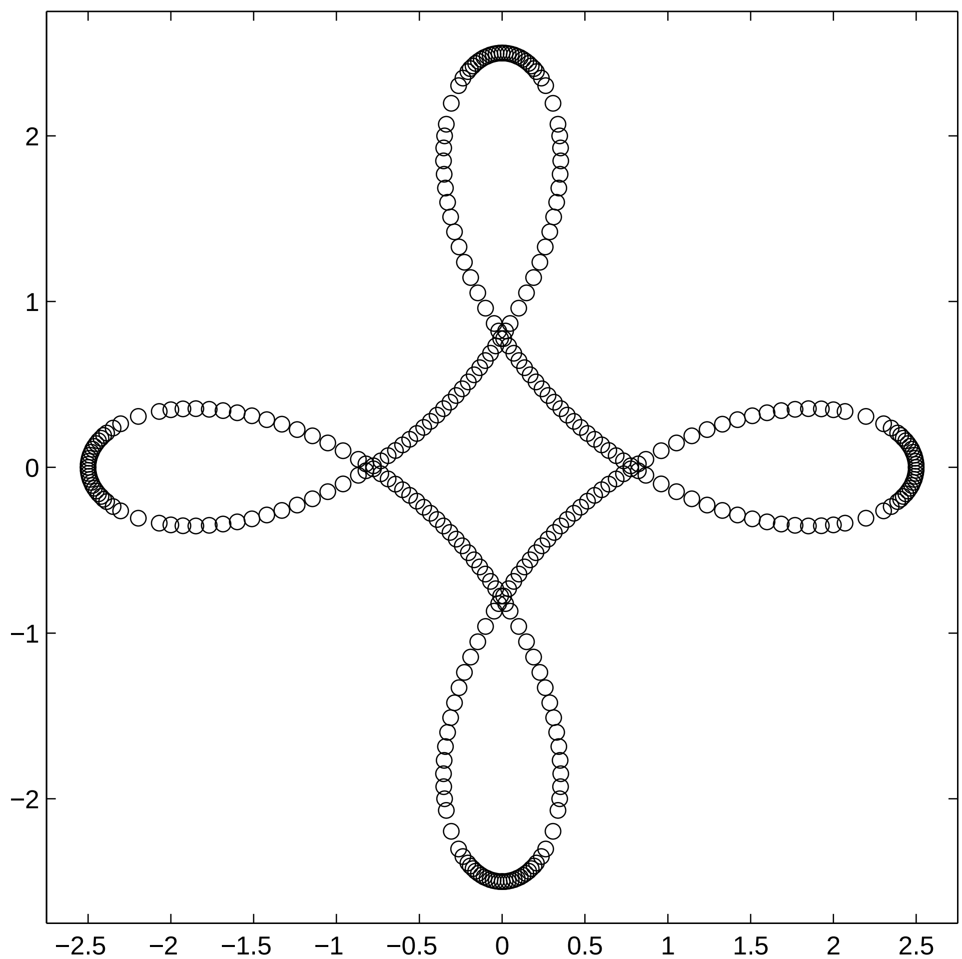
The integrand simplifies as follows:

$$f(t)=\sqrt{[x'(t)]^2+\left[y'(t)\right]^2}=\sqrt{\frac{3}{2}^2+1+3\,(\cos t\,\cos 3t-\sin t\,\sin 3t)}=\frac{3}{2}\sqrt{5-4\cos 4t}$$

This function is periodic with the period of $\pi/2$. Therefore, we can integrate it on the interval $[0,\pi/2]$ and multiply the result by four:

$$\int_0^{2\pi} f(t)\,dt = 4\int_0^{\pi/2} f(t)\,dt$$

Applying the adaptive integration method of the previous problem leads to the following distribution of points:



The estimated length of the curve is 20.0473

Solution:

C program:

```
#include <math.h>   /* for math functions */
#include <stdio.h>  /* for output */
#include <stdlib.h> /* for malloc */
#include <assert.h> /* for assertion */

static const double xtol=1.e-8; /* x tolerance */
static const double itol=1.e-4; /* integral computation tolerance */
static const double pi=3.14159265358979323846264338327950;

/* Function: adaptive_recursive
   ----------------------------
   Adaptive integration by the recursive algorithm.
   Uses the trapezoidal rule, upgraded to Simpson's
   by one step of Richardson's extrapolation.
   f(x) - integrand function
   a,b  - integration limits
   fa = f(a)
   fb = f(b)
   h = b-a
   I - the trapezoidal estimate of the integral
*/
double adaptive_recursive(double (*f)(double x),
                          double a, double b,
                          double fa, double fb,
                          double h, double I)
{
    double c, fc, I1, I2, D;

    h /= 2;
    if (h < xtol) {
        fprintf(stderr,"Adaptive integration did not converge.\n");
        return I;
    }
    c = a + h; /* midpoint */
    fc = f(c);

    I1 = 0.5*h*(fa+fc); /* trapezoidal from a to c */
    I2 = 0.5*h*(fc+fb); /* trapezoidal from c to b */
    D = 4.*(I1 + I2 - I)/3.;
    if (fabs(D) < itol) {
        I += D;
    } else {
        I = adaptive_recursive(f,a,c,fa,fc,h,I1) +
            adaptive_recursive(f,c,b,fc,fb,h,I2);
    }
    return I;
}

/* Function: adaptive_nonrecursive
   -------------------------------
   Adaptive integration by the non-recursive algorithm.
   Uses the trapezoidal rule, upgraded to Simpson's
   by one step of Richardson's extrapolation.
   f(x) - integrand function
   a,b  - integration limits
```

```
   fa = f(a)
   fb = f(b)
   h = b-a
   I - the trapezoidal estimate of the integral
*/
double adaptive_nonrecursive(double (*f)(double x),
                             double a, double b,
                             double fa, double fb,
                             double h, double I)
{
    double c, fc, I1, I2, D, J;
    /* Stack is implemented with a linked list */
    struct Stack {
        double a,b,fa,fb,h,I;
        struct Stack* next;
    } *p, *first;

    J = 0.;

    /* allocate and PUSH the first entry to the stack */
    assert(first = (struct Stack*) malloc(sizeof(*first)));
    first->a = a; first->fa = fa;
    first->b = b; first->fb = fb;
    first->h = h; first->I = I;
    first->next = NULL;

    while ((p = first) != NULL) { /* POP from the stack */
        first = p->next;

        a = p->a; fa = p->fa;
        b = p->b; fb = p->fb;
        h = p->h/2; I = p->I;

        if (h < xtol) {
            fprintf(stderr,"Adaptive integration did not converge.\n");
            return I;
        }
        c = a+h;
        fc = f(c);

        I1 = 0.5*h*(fa+fc); /* trapezoidal from a to c */
        I2 = 0.5*h*(fc+fb); /* trapezoidal from c to b */
        D = 4.*(I1 + I2 - I)/3.;
        if (fabs(D) < itol) {
            J += (I+D);
            free (p);
        } else {
            p->b = c; p->fb = fc;
            p->h = h; p->I = I1;

            /* allocate and PUSH the first entry to the stack */
            assert(first = (struct Stack*) malloc(sizeof(*p)));
            first->a = c; first->fa = fc;
            first->b = b; first->fb = fb;
            first->h = h; first->I = I2;
            first->next = p;
        }
    }
```

```
    return J;
}

/* example function: problem 4 */
static double func4(double x) {
    double y;

    y = 2./(1.+x*x);

    printf("%f %f\n",x,y);
    return y;
}

/* example function: problem 5 */
static double func5(double t) {
    double x, y, f;
    int k;

    f = sqrt(5.-4.*cos(4.*t));

    for (k=0; k < 4; k++) {
        /* account for periodicity */
        x = 1.5*cos(t) + cos(3.*t);
        y = 1.5*sin(t) - sin(3.*t);
        printf("%f %f\n",x,y);
        t += 0.5*pi;
    }

    return f;
}

/* main program */
int main(void) {
    double a, b, fa, fb, h;
    double I;

    a = -1.; fa=func4(a);
    b =  1.; fb=func4(b);
    h = b-a; I = 0.5*h*(fa+fb);

    I = adaptive_nonrecursive(func4, a, b, fa, fb, h, I);
    fprintf(stderr,"Integral=%f\n",I);

    a = 0.;     fa=func5(a);
    b = 0.5*pi; fb=func5(b); /* integrate to p/2 due to periodicity */
    h = b-a; I = 0.5*h*(fa+fb);

    I = 6.*adaptive_recursive(func5, a, b, fa, fb, h, I);
    fprintf(stderr,"Integral=%f\n",I);

    return 0;
}
```