Selected Solutions for Exercises in

# Numerical Methods with MATLAB: Implementations and Applications

Gerald W. Recktenwald

## Chapter 11

## Numerical Integration



The latest version of this PDF file, along with other supplemental material for the book, can be found at `www.prenhall.com/recktenwald`.

**11.2** Write a `polyInt` function that uses the built-in `polyval` function to evaluate the definite integral of a polynomial. The inputs to `polyInt` should be a vector of polynomial coefficients and the lower and upper limits of integration. Test your function by evaluating the two integrals in the preceding exercise.

**Update for MATLAB version 6:** Version 6 includes a `polyint` function that comes close to solving this Exercise. To avoid the name clash, the solution presented here is to develop a `polyIntegral` function.

**Partial Solution:** A correct implementation of `polyIntegral` gives

```
>> I1 = polyIntegral([1 1 1],-pi/2,pi)
I1 =
  20.0408

>> I1exact = (3/8)*pi^3 + (3/8)*pi^2 + (3/2)*pi;
>> err1 = I1 - I1exact
err1 =
     0

>> I2 = polyIntegral([1 0 0 -1],sqrt(3),-5)
I2 =
  160.7321

>> I2exact = ( (-5)^4/4 + 5 ) - ( 9/4 - sqrt(3) );
>> err2 = I2 - I2exact
err2 =
     0
```

How would the built-in `polyint` (MATLAB version 6) function be used to evaluate the definite integral of a polynomial?

◊

**11.3** Use the symbolic capability of the *Student Edition of* MATLAB or the *Symbolic Mathematics Toolbox*, to find the definite integral of the generalized `humps` function

$$f(x) = \frac{1}{(x-c_1)^2 + c_2} + \frac{1}{(x-c_3)^2 + c_4} + c_5$$

**Solution:** The following MATLAB session requires a version of the *Symbolic Mathematics Toolbox.*

```
>> syms a b c1 c2 c3 c4 c5 f x
>> f = 1/( (x-c1)^2 + c2 ) + 1/( (x-c3)^2 + c4 ) + c5

f =

1/((x-c1)^2+c2)+1/((x-c3)^2+c4)+c5

>> g = int(f,x,a,b)

g =

(   atan( (b-c1)/c2^(1/2) )*c4^(1/2)
  + atan( (b-c3)/c4^(1/2) )*c2^(1/2)
  + c5*b*c2^(1/2)*c4^(1/2)
)/c2^(1/2)/c4^(1/2)
- (  atan( (a-c1)/c2^(1/2) )*c4^(1/2)
   + atan( (a-c3)/c4^(1/2) )*c2^(1/2)
   + c5*a*c2^(1/2)*c4^(1/2) )/c2^(1/2)/c4^(1/2)
```

which can be rearranged as

```
g =   ( atan((c1-a)/sqrt(c2)) - atan((c1-b)/sqrt(c2)) )/sqrt(c2) ...
    + ( atan((c3-a)/sqrt(c4)) - atan((c3-b)/sqrt(c4)) )/sqrt(c4) ...
    + c5*(b-a);
```

——— ◊ ———

**11.8** F.M. White (*Fluid Mechanics*, fourth edition, 1999, McGraw-Hill, New York., problem 6.57) gives the following data for the velocity profile in a round pipe

| $r/R$ | 0.0 | 0.102 | 0.206 | 0.412 | 0.617 | 0.784 | 0.846 | 0.907 | 0.963 |
|---|---|---|---|---|---|---|---|---|---|
| $u/u_c$ | 1.0 | 0.997 | 0.988 | 0.959 | 0.908 | 0.847 | 0.818 | 0.771 | 0.690 |

$r$ is the radial position, $R = 12.35\,\text{cm}$ is the radius of the pipe, $u$ is the velocity at the position $r$, and $u_c$ is the velocity at the centerline $r = 0$. The *average* velocity in a round pipe is defined by

$$V = \frac{1}{\pi R^2}\int_0^R u\,2\pi r\,dr, \qquad \text{or} \qquad \frac{V}{u_c} = \int_0^1 2\frac{V}{u_c}\eta\,d\eta,$$

where $\eta = r/R$. What is the value of $V$ for the given data if $u_c = 30.5\,\text{m/s}$? Do not forget to include the implied data point $u/u_c = 0$ at $r/R = 1$. The data in the table is in the `vprofile.dat` file in the data directory of the NMM toolbox.

**Typographical Error:** A factor of 2 is missing from the second integral expression. The correct formula for $V/u_c$ is

$$\frac{V}{u_c} = \int_0^1 2\frac{V}{u_c}\eta\,d\eta$$

**Numerical Answer:** Using `trapzDat` function with the correct form of the integral for $V/u_c$, gives $V = 25.4870$ m/s.

◊

**11.12** Use the Trapezoid rule to evaluate

$$\beta(m,n) = \int_0^1 x^{m-1}(1-x)^{n-1}\,dx$$

for any $m$ and $n$ and for a sequence of decreasing panel sizes $h$. Print the value of $\beta(m,n)$, and the error relative to the value returned by the built-in `beta` function. Use your function to evaluate $\beta(1,2)$, $\beta(1.5,2.5)$, $\beta(2,3)$, and $\beta(2,5)$. Comment on the convergence rate. (*Hint*: The values of $m$ and $n$ can be passed around (not through) `trapezoid` with global variables.)

**Partial Solution:** The solution is obtained by writing two m-file functions. One function evaluates the integrand, and the other calls `trapezoid` with a sequence of decreasing panel sizes. This second m-file is obtained with minor modifications to the `demoTrap` function. I've called it `betaTrap`. The prologue of the `betaTrap` function is

```
function betaTrap(m,n)
% betaTrap  Evaluate beta function with trapezoid rule
%
% Synopsis:  betaTrap
%            betaTrap(m,n)
%
% Input:  m,n = (optional) parameters of the beta function
%               Default: m = 1, n = 2
%
% Output: Table of integral values as a function of decreasing panel size
```

Running `betaTrap` for $m = 1$ and $n = 2$ gives

```
>> betaTrap

  Iexact =    0.500000000

   n      h          I              error          alpha
   3   0.50000   0.500000000    0.000000000
   5   0.25000   0.500000000    0.000000000   -0.00000
   9   0.12500   0.500000000    0.000000000   -0.00000
  17   0.06250   0.500000000    0.000000000   -0.00000
  33   0.03125   0.500000000    0.000000000   -0.00000
  65   0.01562   0.500000000    0.000000000   -0.00000
 129   0.00781   0.500000000    0.000000000   -0.00000
 257   0.00391   0.500000000    0.000000000   -0.00000
```

The numerical integral is exact in this case becuase the integrand reduces to $(1-x)$. The trapezoid rule integrates a linear function with no truncation error.

Running `betaTrap` for $m = 1.5$ and $n = 2.5$ gives

```
>> betaTrap(1.5,2.5)

  Iexact =    0.196349541
```

| n | h | I | error | alpha |
|---|---|---|---|---|
| 3 | 0.50000 | 0.125000000 | -0.071349541 | |
| 5 | 0.25000 | 0.170753175 | -0.025596365 | 1.47897 |
| 9 | 0.12500 | 0.187231817 | -0.009117724 | 1.48919 |
| 17 | 0.06250 | 0.193113697 | -0.003235844 | 1.49453 |
| 33 | 0.03125 | 0.195203315 | -0.001146226 | 1.49725 |
| 65 | 0.01562 | 0.195943901 | -0.000405639 | 1.49862 |
| 129 | 0.00781 | 0.196206057 | -0.000143484 | 1.49931 |
| 257 | 0.00391 | 0.196298800 | -0.000050741 | 1.49965 |

The theoretical value of $\alpha = 2$ is not obtained becuase the derivative of the integrand is not defined at the lower limit of integration. The truncation error for the composite trapezoid rule is bounded by $Ch^2 f''(\xi)$ where $C$ is a constant and $f''(\xi)$ is the second derivative of the integrand evaluated at some point $\xi$ in the limits of integration. For $m = 1.5$ and $n = 2.5$, $f = \sqrt{x}(1-x)^{3/2}$ and $f''(x) \to \infty$ as $x$ approaches zero. Although the formulas in the trapezoid rule do not encounter a division by zero, the truncation error reduces more slowly than $\mathcal{O}(h^2)$ because of the contribution of $f''(\xi)$.

**11.16** Evaluate

$$I = \int_0^1 \sqrt{x}\, dx$$

using the NMM routines `trapezoid`, `simpson`, and `gaussQuad`. For each routine, evaluate the integral for at least three different panel sizes. Present a table comparing the measured truncation error as a function of panel size. Report any problems in obtaining values of $I$. Which routine works best for this problem?

**Partial Solution:** The solution is obtained by modifying the code in the `demoTrap`, `demoSimp`, and `demoGauss` functions. For convenience the modified code from these function is combined into a single m-file called `intSqrt`. Running `intSqrt` gives:

```
>> intSqrt

Evaluate Integral with Trapezoid Rule:  Iexact =  0.6666667

    n      h          I              error         alpha
    3   0.50000    0.603553391   -0.063113276
    5   0.25000    0.643283046   -0.023383620    1.43245
    9   0.12500    0.658130222   -0.008536445    1.45379
   17   0.06250    0.663581197   -0.003085470    1.46815
   33   0.03125    0.665558936   -0.001107730    1.47788
   65   0.01562    0.666270811   -0.000395855    1.48456
  129   0.00781    0.666525657   -0.000141009    1.48918
  257   0.00391    0.666616549   -0.000050118    1.49240


Evaluate Integral with Simpson's Rule:  Iexact =  0.6666667

    n      h          I              error         alpha
    3   0.50000    0.656526265   -0.010140402
    5   0.25000    0.663079280   -0.003587387    1.49911
    9   0.12500    0.665398189   -0.001268478    1.49983
   17   0.06250    0.666218183   -0.000448484    1.49997
   33   0.03125    0.666508103   -0.000158564    1.49999
   65   0.01562    0.666610606   -0.000056061    1.50000
  129   0.00781    0.666646846   -0.000019820    1.50000
  257   0.00391    0.666659659   -0.000007008    1.50000


Evaluate Integral with Gauss-Legendre Rule:  Iexact =  0.6666667

Gauss-Legendre quadrature with 4 panels, H = 0.250000

 nodes       I           error
   1    0.6729773970    6.31e-03
   2    0.6675777702    9.11e-04
   3    0.6669809064    3.14e-04
   4    0.6668117912    1.45e-04
   5    0.6667454321    7.88e-05
   6    0.6667141381    4.75e-05
   7    0.6666974690    3.08e-05
   8    0.6666877808    2.11e-05
```

```
Gauss-Legendre quadrature with 8 nodes

 panels    H           I            error       alpha
    2    0.50000   0.6667263866     5.97e-05
    4    0.25000   0.6666877808     2.11e-05      1.50
    8    0.12500   0.6666741317     7.46e-06      1.50
   16    0.06250   0.6666693059     2.64e-06      1.50
   32    0.03125   0.6666675998     9.33e-07      1.50
   64    0.01562   0.6666669966     3.30e-07      1.50
  128    0.00781   0.6666667833     1.17e-07      1.50
  256    0.00391   0.6666667079     4.12e-08      1.50
```

None of the integration rules achieves its theoretical truncation error. The reason is that the integrand is not sufficiently differentiable at the lower limit of integration. (See also the solution to Exercise **11-12**.) The Gauss–Legendre rule with eight nodes per panel obtains the result with the smallest error, though its performance is considerable worse on this integrand than on the integrands demonstrated in the Examples in the text.

◊

**11.22** Write an m-file function that evaluates $\int_0^{2\pi} \sin^2(x)\,dx$ using the composite trapezoid rule, composite Simpson's rule, and composite Gauss–Legendre quadrature with four nodes per panel. Place the calls to `trapezoid`, `simpson`, and `gaussQuad` inside a loop and repeat the calculations for `np = [2 4 8 16 32 64]`, where `np` is the number of panels. Record the number of function evaluations, `n`, for each method. Print the absolute error $|I - I_{\text{exact}}|$ for the three methods versus $n$. (See, for example, [13,§ 2.9] for help in explaining the results.)

**Partial Solution:** The computations are carried out with the `compint_sinx2` function. The prologue for `compint_sinx2` is

```
function compint_sinx2(a,b)
% compint_sinx2  Compare quadrature rules for integral of (sin(x))^2
%
% Synopsis:  compint_sinx2
%            compint_sinx2(a,b)
%
% Input:     a,b = (optional) limits of integration.  Default: a=0; b=pi
%
% Output:    Values of integral obtained by trapezoid and simpsons rules
%            for increasing number of panels
```

Running `compint_sinx2` with the default inputs gives:

```
>> compint_sinx2

        Integral of (sin(x))^2 from 0*pi to 1*pi
        Iexact =    1.570796327

      Trapezoid              Simpson             Gauss-Legendre
   n       error          n       error          n       error
    3    0.00000e+00      5    0.00000e+00      8    0.00000e+00
    5    0.00000e+00      9    0.00000e+00     16    0.00000e+00
    9    0.00000e+00     17    0.00000e+00     32    0.00000e+00
   17    0.00000e+00     33    0.00000e+00     64   -2.22045e-16
   33    0.00000e+00     65    0.00000e+00    128    0.00000e+00
   65    0.00000e+00    129    0.00000e+00    256    0.00000e+00
```

Note that `n` is the number of nodes at which the integrand is evaluated, not the number of panels used by the various composite rules. All methods give neglible errors regardless of the number of panels. The trapezoid rule is known to rapidly converge for a periodic integrand when the limits of the integral are points at which the integrand and its derivative assume periodic values. See Davis and Rabinowitz [13,§ 2.9] for details. $\sin^2 x$ is an extreme example of the special behavior for periodic integrands.

A slightly more interesting results is obtained if `np = [ 1 2 4 8 16 32]` and the Gauss–Legendre rule with two (instead of four) points per panel is used. Making these changes and running the modified `compint_sinx2` gives

```
>> compint_sinx2

    Integral of (sin(x))^2 from 0*pi to 1*pi
    Iexact =    1.570796327
```

```
   Trapezoid              Simpson             Gauss-Legendre
n         error        n         error        n         error
 2   -1.57080e+00       3    5.23599e-01       2   -3.77963e-01
 3    0.00000e+00       5    0.00000e+00       4    0.00000e+00
 5    0.00000e+00       9    0.00000e+00       8    0.00000e+00
 9    0.00000e+00      17    0.00000e+00      16    0.00000e+00
17    0.00000e+00      33    0.00000e+00      32    0.00000e+00
33    0.00000e+00      65    0.00000e+00      64    0.00000e+00
```

This result allows direct comparison with somewhat more comparable number of nodes in each row. The first row shows the error for applying the basic rule for each method.

If, for the integral in this Exercise, the limits of the integral are shifted to 0 and some non-rational multiple of $\pi$, the integration schemes behave as usual. Rerunning the modified version of `compint_sinx2` gives

```
>> compint_sinx2(0,5.12*pi)

    Integral of (sin(x))^2 from 0*pi to 5.12*pi
    Iexact =    7.871340417

   Trapezoid              Simpson             Gauss-Legendre
n         error        n         error        n         error
 2   -6.78146e+00       3    2.83874e+00       2   -7.23535e+00
 3    4.33692e-01       5   -6.75466e-01       4    2.67829e-01
 5   -3.98177e-01       9    5.76797e-01       8   -4.38233e-01
 9    3.33054e-01      17   -2.84133e-02      16    1.95645e-02
17    6.19534e-02      33   -1.10178e-03      32    7.40447e-04
33    1.46620e-02      65   -6.25700e-05      64    4.17971e-05
```

Now the truncation errors do not decrease so dramatically because the limits of the integral do not produce periodic values of the integrand and its derivatives.

◊