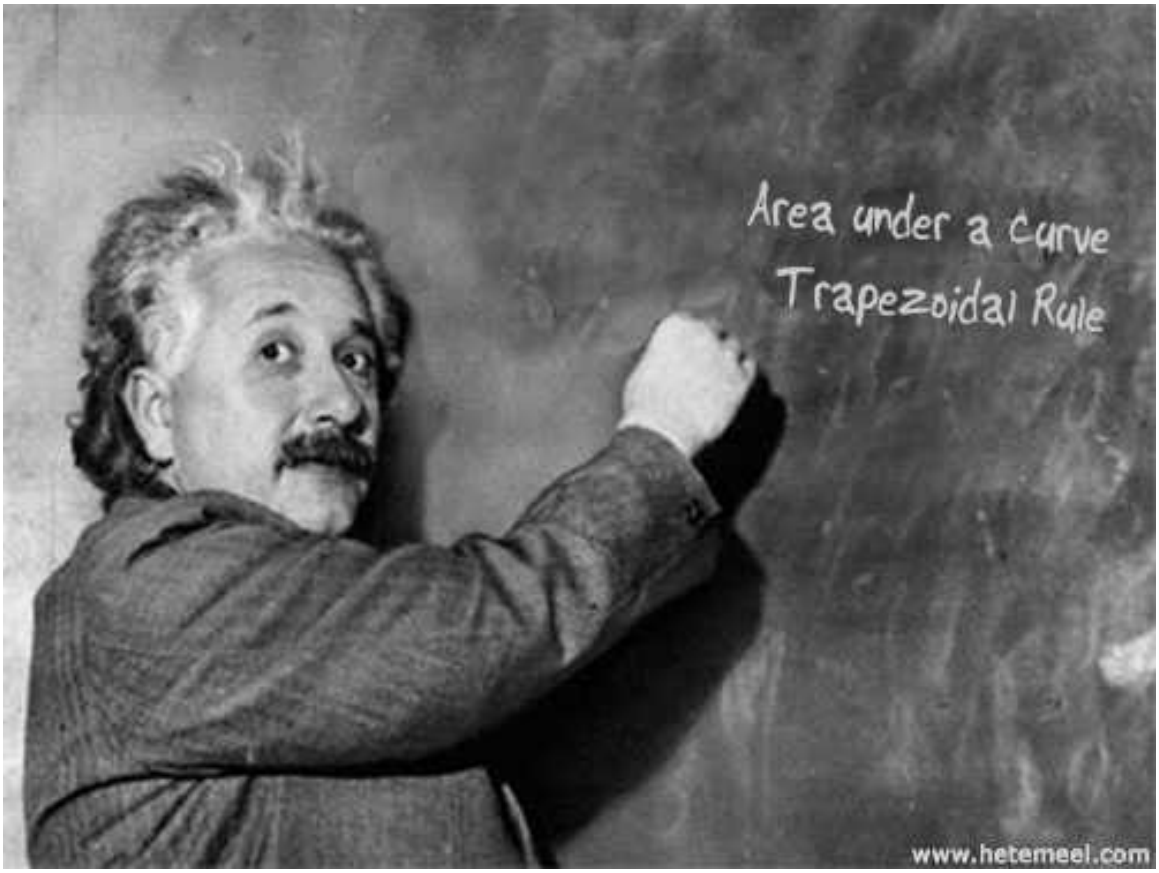
Numerical Integration with MATLAB*

Serhat Beyenir

This work is produced by OpenStax-CNX and licensed under the
Creative Commons Attribution License 3.0†

Abstract

Basic Numerical Integration with MATLAB



This chapter essentially deals with the problem of computing the area under a curve. First, we will employ a basic approach and form trapezoids under a curve. From these trapezoids, we can calculate the

*Version 1.3: Dec 2, 2011 2:14 am -0600

†http://creativecommons.org/licenses/by/3.0/

total area under a given curve. This method can be tedious and is prone to errors, so in the second half of the chapter, we will utilize a built-in MATLAB function to carry out numerical integration.

1 A Basic Approach

There are various methods to calculating the area under a curve, for example, Rectangle Method[1] , Trapezoidal Rule[2] and Simpson's Rule[3] . The following procedure is a simplified method.

Consider the curve below:

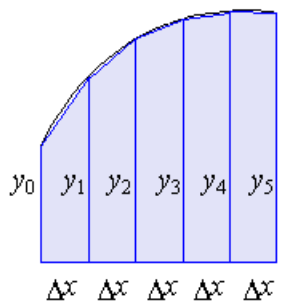


Figure 1: Numerical integration

Each segment under the curve can be calculated as follows:

$$\frac{1}{2}\left(y_0+y_1\right)\Delta \mathrm{x}+\frac{1}{2}\left(y_1+y_2\right)\Delta \mathrm{x}+\frac{1}{2}\left(y_2+y_3\right)\Delta \mathrm{x} \tag{1}$$

Therefore, if we take the sum of the area of each trapezoid, given the limits, we calculate the total area under a curve. Consider the following example.

Example 1

Given the following data, plot an x-y graph and determine the area under a curve between x=3 and x=30

Index	**x [m]**	**y [N]**
1	3	27.00
2	10	14.50
3	15	9.40
4	20	6.70
5	25	5.30
6	30	4.50

Table 1: Data Set

[1] http://en.wikipedia.org/wiki/Rectangle_method
[2] http://en.wikipedia.org/wiki/Trapezoidal_rule
[3] http://en.wikipedia.org/wiki/Simpson%27s_rule

First, let us enter the data set. For x, issue the following command `x=[3,10,15,20,25,30];`. And for y, `y=[27,14.5,9.4,6.7,5.3,4.5];`. If yu type in `[x',y']`, you will see the following tabulated result. Here we transpose row vectors with ' and displaying them as columns:

```
ans =

 3.0000   27.0000
10.0000   14.5000
15.0000    9.4000
20.0000    6.7000
25.0000    5.3000
30.0000    4.5000
```

Compare the data set above with the given information in the question (Table 1).

To plot the data type the following:

```
plot(x,y),title('Distance-Force Graph'),xlabel('Distance[m]'),ylabel('Force[N]'),grid
```

The following figure is generated:

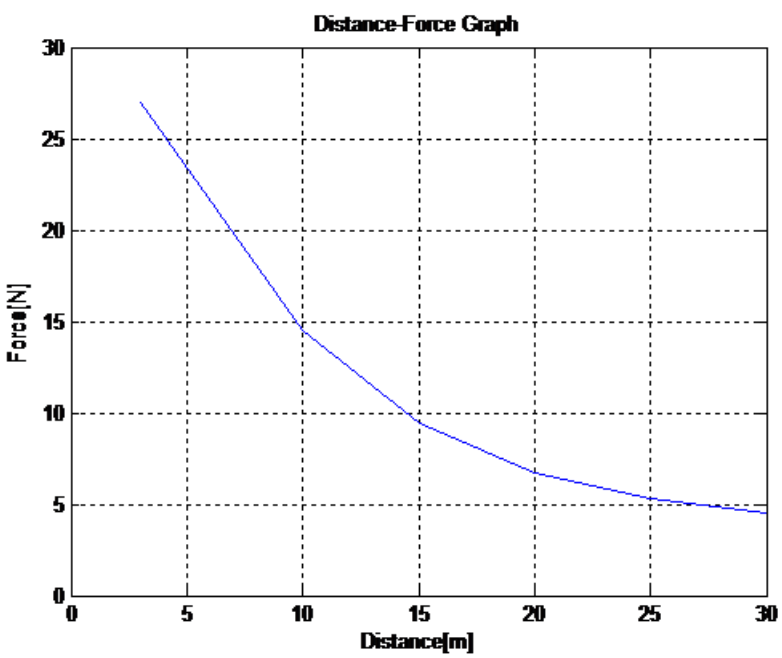


Figure 2: Distance-Force Graph

To compute dx for consecutive x values, we will use the index for each x value, see the given data in the question (Table 1).:

```
dx=[x(2)-x(1),x(3)-x(2),x(4)-x(3),x(5)-x(4),x(6)-x(5)];
```

dy is computed by the following command:

```
dy=[0.5*(y(2)+y(1)),0.5*(y(3)+y(2)),0.5*(y(4)+y(3)),0.5*(y(5)+y(4)),0.5*(y(6)+y(5))];
```

dx and dy can be displayed with the following command: `[dx',dy']`. The result will look like this:

```
[dx',dy']

ans =

    7.0000   20.7500
    5.0000   11.9500
    5.0000    8.0500
    5.0000    6.0000
    5.0000    4.9000
```

Our results so far are shown below

x [m]	y [N]	dx [m]	dy [N]
3	27.00		
10	14.50	7.00	20.75
15	9.40	5.00	11.95
20	6.70	5.00	8.05
25	5.30	5.00	6.00
30	4.50	5.00	4.90

Table 2: x, y and corresponding differential elements

If we multiply dx by dy, we find da for each element under the curve. The differential area da=dx*dy, can be computed using the 'term by term multiplication' technique in MATLAB as follows:

```
da=dx.*dy

da =

  145.2500   59.7500   40.2500   30.0000   24.5000
```

Each value above represents an element under the curve or the area of trapezoid. By taking the sum of array elements, we find the total area under the curve.

```
sum(da)

ans =

  299.7500
```

The following (Table 3) illustrates all the steps and results of our MATLAB computation.

x [m]	y [N]	dx [m]	dy [N]	dA [Nm]
3	27.00			
10	14.50	7.00	20.75	145.25
15	9.40	5.00	11.95	59.75
20	6.70	5.00	8.05	40.25
25	5.30	5.00	6.00	30.00
30	4.50	5.00	4.90	24.50
				299.75

Table 3: Computation of the approximate area under a curve

2 The Trapezoidal Rule

Sometimes it is rather convenient to use a numerical approach to solve a definite integral. The trapezoid rule allows us to approximate a definite integral using trapezoids.

2.1 The `trapz` Command

Z = trapz(Y) computes an approximation of the integral of Y using the trapezoidal method.

Now, let us see a typical problem.

Example 2

Given Area = $\int_2^5 x^2 dx$, an analytical solution would produce 39. Use trapz command and solve it

1. Initialize variable x as a row vector, from 2 with increments of 0.1 to 5: `x=2:.1:5;`
2. Declare variable y as `y=x^2;`. Note the following error prompt: `??? Error using ==> mpower Inputs must be a scalar and a square matrix.` This is because x is a vector quantity and MATLAB is expecting a scalar input for y. Because of that, we need to compute y as a vector and to do that we will use the dot operator as follows: `y=x.^2;`. This tells MATLAB to create vector y by taking each x value and raising its power to 2.
3. Now we can issue the following command to calculate the first area, the output will be as follows:

```
area1=trapz(x,y)

area1 =

   39.0050
```

Notice that this numerical value is slightly off. So let us increase the number of increments and calculate the area again:

```
x=2:.01:5;
y=x.^2;
area2=trapz(x,y)

area2 =

   39.0001
```

Yet another increase in the number of increments:

```
x=2:.001:5;
y=x.^2;
area3=trapz(x,y)

area3 =

   39.0000
```

Example 3

Determine the value of the following integral:
$\int_0^{\pi} \sin(x)\, dx$

1. Initialize variable x as a row vector, from 0 with increments of pi/100 to pi: `x=0:pi/100:pi;`
2. Declare variable y as `y=sin(x);`
3. Issue the following command to calculate the first area, the output will be as follows:

```
area1=trapz(x,y)

area1 =

1.9998
```

let us increase the increments as above:

```
x=0:pi/1000:pi;
y=sin(x);
area2=trapz(x,y)

area2 =

2.0000
```

3 Summary of Key Points

1. In its simplest form, numerical integration involves calculating the areas of segments that make up the area under a curve,
2. MATLAB has built-in functions to perform numerical integration,
3. `Z = trapz(Y)` computes an approximation of the integral of Y using the trapezoidal method.