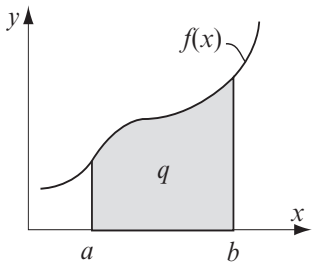
9.3 NUMERICAL INTEGRATION

Integration is a common mathematical operation in science and engineering. Calculating area and volume, velocity from acceleration, and work from force and displacement are just a few examples where integrals are used. Integration of simple functions can be done analytically, but more involved functions are frequently difficult or impossible to integrate analytically. In calculus courses the integrand (the quantity to be integrated) is usually a function. In applications of science and engineering the integrand can be a function or a set of data points. For example, data points from discrete measurements of flow velocity can be used to calculate volume.

It is assumed in the presentation below that the reader has knowledge of integrals and integration. A definite integral of a function $f(x)$ from a to b has the form:

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



The function $f(x)$ is called the integrand, and the numbers a and b are the limits of integration. Graphically, the value of the integral q is the area between the graph of the function, the x axis, and the limits a and b (the shaded area in the figure). When a definite integral is calculated analytically $f(x)$ is always a function. When the integral is calculated numerically $f(x)$ can be a function or a set of points. In numerical integration the total area is obtained by dividing the area into small sections, calculating the area of each section, and adding them up. Various numerical methods have been developed for this purpose. The difference between the methods is in the way that the area is divided into sections and the method by which the area of each section is calculated. Books on numerical analysis include details of the numerical techniques.

The following discussion describes how to use the three MATLAB built-in integration functions `quad`, `quadl`, and `trapz`. The `quad` and `quadl` commands are used for integration when $f(x)$ is a function, and `trapz` is used when $f(x)$ is given by data points.

The `quad` command:

The form of the `quad` command, which uses the adaptive Simpson method of integration, is:

```
q = quad(function,a,b)
```

The value of the integral. The function to be integrated. The integration limits.

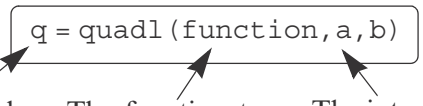
- The function can be entered as a string expression or as a function handle, in the same way as with the `fzero` command. See Section 9.1 for details. The first two methods are demonstrated in Sample Problem 9-2.
- The function $f(x)$ must be written for an argument x that is a vector (use element-by-element operations) such that it calculates the value of the function for each element of x.
- The user has to make sure that the function does not have a vertical asymptote between a and b.
- `quad` calculates the integral with an absolute error that is smaller than 1.0e–6. This number can be changed by adding an optional `tol` argument to the command:

```
q = quad('function',a,b,tol)
```

`tol` is a number that defines the maximum error. With larger `tol` the integral is calculated less accurately but faster.

The `quadl` command:

The form of the `quadl` (the last letter is a lowercase L) command is exactly the same as that of the `quad` command:



The value of the integral. The function to be integrated. The integration limits.

All of the comments that are listed for the `quad` command are valid for the `quadl` command. The difference between the two commands is the numerical method used for calculating the integration. The `quadl` command uses the adaptive Lobatto method, which can be more efficient for high accuracies and smooth integrals.

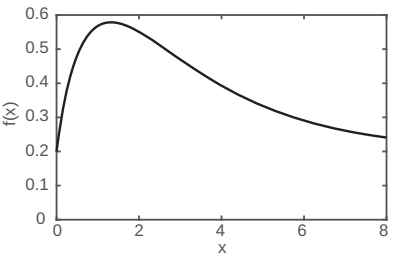
Sample Problem 9-2: Numerical integration of a function

Use numerical integration to calculate the following integral:

$$\int_0^8 (x e^{-x^{0.8}} + 0.2)\, dx$$

Solution

For illustration, a plot of the function for the interval $0 < x < 8$ is shown on the right. The solution uses the `quad` command and shows how to enter the function in the command in two ways. In the first, it is entered directly by typing the expression as an argument. In the second, an anonymous function is created and its name is subsequently entered in the command.



The use of the `quad` command in the Command Window, with the function to be integrated typed in as a string, is shown below. Note that the function is typed with element-by-element operations.

```
>> quad('x.*exp(-x.^0.8)+0.2',0,8)
ans =
    3.1604
```

The second method is to first create a user-defined function that calculates the function to be integrated. The function file (named `y=Chap9Sam2(x)`) is:

```
function y=Chap9Sam2(x)
y=x.*exp(-x.^0.8)+0.2;
```

Note again that the function is written with element-by-element operations such that the argument `x` can be a vector. The integration is then done in the Command Window by typing the handle `@Chap9Sam2` for the argument `function` in the `quad` command as shown below:

```
>> q=quad(@Chap9Sam2,0,8)
q =
    3.1604
```

The `trapz` command:

The `trapz` command can be used for integrating a function that is given as data points. It uses the numerical trapezoidal method of integration. The form of the command is

```
q = trapz(x,y)
```

where `x` and `y` are vectors with the x and y coordinates of the points, respectively. The two vectors must be of the same length.