

Term Project CS177 : Symbolic Integration using MATLAB Symbolic Math Toolbox

Mehreen Asad

2011-05-04

Abstract

This tutorial incorporates the basic concepts used in MATLAB Symbolic Math Toolbox with specific emphasis on Symbolic Integration. It starts by giving the basic concepts of Symbolic Operations and MATLAB Symbolic Commands. The tutorial then introduces Symbolic Integration Concepts for both Indefinite and Definite Integration. To further illustrate the concepts, four examples are provided with solutions to give the reader a full insight into the concept of Symbolic Integration in MATLAB. The tutorial is ideal for people who have never used MATLAB Symbolic Math Toolbox before.

1 Symbolic Toolbox Basics

1.1 Concepts and Symbolic Matlab Commands

Symbolic operations, like numerical operations can be performed by MATLAB when the Symbolic Math Toolbox is installed. Symbolic operations are carried out in the similar manner as numeric operations except for the fact that the former are performed on symbolic variables rather than numeric values. The Symbolic Math Toolbox Software can be used to perform mathematical operations and incorporates: *Calculus, Simplifications and Substitutions, Variable-Precision Arithmetic, Linear Algebra, Solving Equations and Integral Transforms and Z-Transforms*. The focus of this tutorial is on Symbolic Integration which is a part of Symbolic Calculus. First basic Symbolic concepts are introduced in the tutorial and then *Symbolic Integration* is discussed in detail illustrated with sample problems and their solutions and graphs, starting with relatively simple examples and then moving on to a bit complex ones involving some of the MATLAB commands mentioned here in the tutorial.

1.1.1 Creating a Symbolic Object

Symbolic objects can be created using the *sym* or *syms* command. The command is of the form:
`a=sym('a')` or `sym a`.

The only difference between these two commands is that the object name is displayed in the first command. *syms* is used to declare multiple symbolic objects at the same time. Symbolic objects can be numbers as well as variables. If a number is declared as a symbolic object, then all calculations related to that symbolic number are carried out exactly with no numeric approximation. Consider the following example:

`a=sym(3); b=sym(8); f=b/a+sqrt(2)` gives the result:

$$f = 8/3 + \sqrt{2}$$

instead of any numerical value.

1.1.2 Creating Symbolic Expressions

A Symbolic Expression can be created by first defining symbolic objects and then writing an expression in terms of those symbolic objects.

Using, the *syms* command: `syms a b c x y`

$$f = ax^2 + bxy + cy$$

1.1.3 Matlab Commands:

The *double(f)* command is used to transform a symbolic expression *f* written in exact form to numerical form.

The *collect(f)* Command groups the terms in the expression *f* that have the variable with the same power.

The *expand(f)* command expands expressions by carrying out products of terms that include summation and uses trigonometric identities and exponential and logarithmic laws to expand corresponding terms that include summation.

The *factor(f)* command is the opposite of expand command and reduces the polynomial expression to be a product of polynomials of a lower degree.

The *simplify(f)* and *simple(f)* are used for simplifying the form of an expression. The command `[F how] = simple(f)` gives the simplified expression as well as the method used for simplifying the expression by assigning it to variable *how*.

The *pretty(f)* command displays a symbolic expression in a format resembling the mathematical format.

The *findsym(f)* command displays the names of all symbolic variables in *f* in alphabetical order.

The *solve(eq)* and *solve(eq,var)* commands are used for solving the equation. The *solve(eq)* solves the equation for default variable *x* and the *solve(eq,var)* command can be used to solve the equation in terms of the variable *var*. If there is only one variable, then the solution is numerical, otherwise the solution is in terms of other variables.

2 Symbolic Integration Concepts

2.1 Concepts

Symbolic Integration can be carried out by using the *int* command. Both indefinite and definite Integrals are determined by the command *int*. *f* can be a pre-defined symbolic expression or an an expression can be typed in for *f*

2.2 Indefinite Integration

If *f* is a symbolic expression, the *int(f)* attempts to find another symbolic expression **F** such that $\text{diff}(F)=f$.

If the expression contains one variable, then integration is carried out with respect to that variable using the *int(f)* command.

If the expression contains more than one variable, then integration is carried out with respect to the desired variable using the `int(f,var)` command.

2.3 Definite Integration

For definite Integration, the commands `int(f,a,b)` and `int(f,var,a,b)` are used for integration *w.r.t* a particular variable. **a** and **b** are the limits of Integration. The limits can be *inf* or *-inf*.

In contrast to differentiation, symbolic Integration is a more complicated task. The antiderivative **F** may not exist in closed form, or even if exists, MATLAB might not be able to find it. In case of such an exception, MATLAB returns `int(f)` and a message *Explicit Integral could not be found*.

3 Symbolic Integration Examples

3.1 Example 1:

3.1.1 Problem:

Evaluate the indefinite Integral

$$I = \int \frac{\sin^2(x)\cos(x)}{(2+3\sin(x))^2} dx.$$

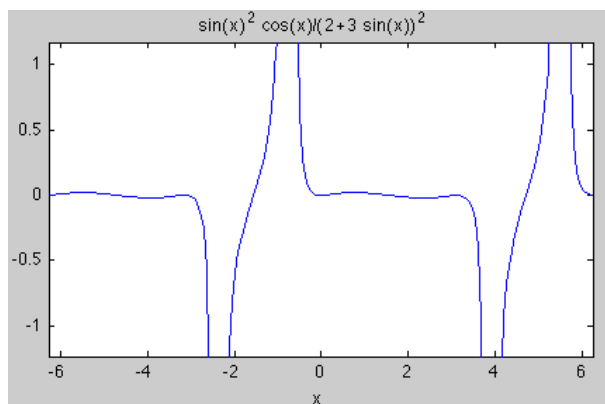
Display the result in mathematical format using `pretty` command. Plot both the function and the Integral using subplot.

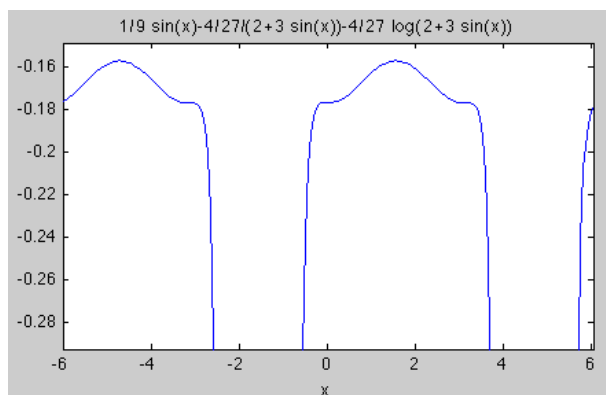
3.1.2 Solution:

The problem is solved by first declaring a symbolic variable **x** and then defining the function *f* in terms of the symbolic variable. Then the Integral is computed using the `int` command in Matlab.

The matlab commands are given in the project.m script file which contains solutions to all the problems mentioned in the tutorial.

Following are the plots for function and the Integral of Problem 1.





3.2 Example 2:

3.2.1 Problem:

The differential volume element of the cone is given by:

$$dV = \pi(R^2)\left(1 - \frac{y}{H}\right)^2 dy$$

Use MATLAB to evaluate the integral of dV from 0 to H symbolically and show that the volume of the cone is

$$V = \frac{1}{3}\pi(R^3)H$$

3.2.2 Solution:

The *int* command is used to calculate the Integral for the given equation which gives the Volume V of the cone.

3.3 Example 3:

3.3.1 Problem:

The rms value of an AC Voltage is defined by:

$$v_{rms} = \sqrt{\frac{1}{T} \int_0^T v^2(t) dt} \quad (1)$$

where T is the period of the waveform. A voltage is given by

$$v(t) = V \cos(\omega t)$$

Show that

$$v_{rms} = \frac{V}{\sqrt{2}}$$

and is independent of ω . Note: The relationship between the period T and the radian frequency ω is:

$$T = \frac{2\pi}{\omega}$$

3.3.2 Solution:

First, the values of $v(t)$ and T as given in the problem are substituted in Eq (1) and then the equation is being solved for

$$v_{rms} = \frac{V}{\sqrt{2}}$$

First symbolic variables are declared using *syms* command: `syms T v V w t`

Then, $v(t)$ and T are defined in terms of symbolic variables. Next, eq(1) is solved in terms of V using the *int* command for symbolic integration. The limits of integration are set from 0 to T . The solution is then further simplified by *simple* command and [how] specifies the name of the simplification method.

3.4 Example 4:

3.4.1 Problem:

A horizontal cylindrical tank is used for storing fuel. The tank has a diameter of 6 m and is 8 m long. The amount of fuel in the tank can be estimated by looking at the level of the fuel through a narrow vertical glass window at the front of the tank. A scale that is marked next to the window shows the level of the fuel corresponding to 40, 60, 80, 120, 160 thousand litres. Determine the vertical position (measured from the ground) of the lines of the scale.

The volume V of the fuel is given by:

$$V = AL = L \int_0^h w dy$$

where A is the cross-sectional area of the fuel and L is the length of the tank. The width w of the top surface of the fuel is written as a function of y and is given by:

$$w = 2\sqrt{R^2 - (R - y)^2}$$

where R is the radius of the tank.

3.4.2 Solution:

Here we have $R = 3$ and $L = 8$. Then we declare w , y , h as symbolic variables. Next, w is written as a symbolic expression in terms of R and y . The Volume of the fuel at height h can be calculated by substituting w in the integral in the equation for the volume and carrying out the Integration using the *int(S,y,0,h)* command where S is the symbolic expression and is obtained by multiplying L with w . The result is an equation that gives the volume v as a function of h . The value of h for a given V is obtained by solving the equation for h using the MATLAB command *solve*. In the current scenario, the values of h have to be determined for volumes of 40, 60, 80, 120 and 160 thousand litres. The solution is given in project.m script file.

3.5 References:

Web Reference for Problem 1

Web Reference for Matlab Symbolic Math Toolbox Demos and Documentation

MATLAB: An Introduction With Applications (Third Edition) by Amos Gilat