

CHAPTER V

PROGRAMMING LANGUAGE

5.1 Microsoft Visual Basic

The "Visual" part refers to the method used to create a graphical user interface (GUI). Rather than writing numerous lines of codes to describe the appearance and locations of interface elements, you simply add prebuilt objects into a place on the screen. The "Basic" part refers to the BASIC (Beginners All-Purpose Symbolic Instruction Code) language, a language used by more programmers than any other language in the history of computing. Visual Basic has evolved from the original BASIC language and now contains several hundred statements, functions, and keywords, many of which relate directly to the Windows GUI. Beginners can create useful applications by learning just a few of the keywords, yet the power of this language allows professionals to accomplish anything that can be accomplished using any other Windows programming language. The Visual Basic programming system is included in Microsoft Excel, Microsoft Access, and many other Windows applications. The Visual Basic Scripting Edition (VBScript) is a widely used scripting language and is a subset of the Visual Basic language.

5.2 Excel Link

An Excel Link is an added in the software that integrates Microsoft Excel and MATLAB in a Microsoft Windows-based computing environment. By connecting Excel and MATLAB, the user can access the numerical, computational, and graphical power of MATLAB from an Excel worksheet and macro programming tools. Excel Link lets the user exchange and synchronize data between the two environments.

Excel Link communicates between the Excel workspace and the MATLAB workspace. It positions Excel as a front end to MATLAB. Users use Excel Link functions from an Excel worksheet or macro.

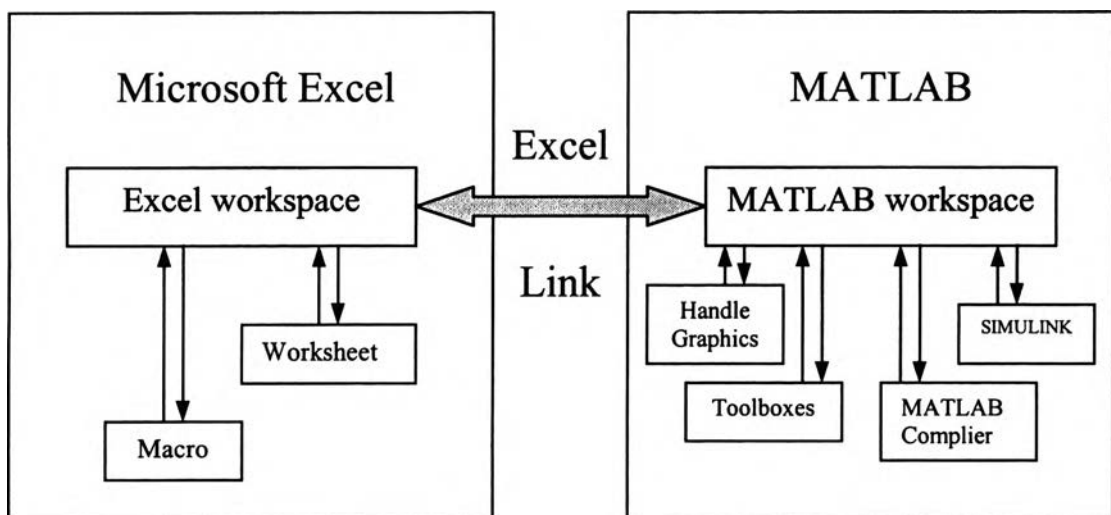


Figure 5.1 Excel Link.

Excel Link provides seven data management functions to copy data between Excel and MATLAB and to execute MATLAB commands from Excel.

MLAppendMatrix	Create or append the MATLAB matrix with data from the Excel worksheet.
MLDeleteMatrix	Delete MATLAB matrix
MLEvalstring	Evaluate commands in MATLAB
MLGetMatrix	Write contents of MATLAB matrix in the Excel worksheet
MLGetVar	Write contents of MATLAB matrix in Excel VBA (Visual Basic for Applications) variable.
MLPutMatrix	Create or overwrite MATLAB matrix with Data from the Excel worksheet.
MLPutVar	Create or overwrite MATLAB matrix with data from Excel VBA variable.

The user can command any data management function except **MLGetVar** and **MLPutvar** as a worksheet cell formula or in a macro. The user can invoke **MLGetVar** and **MLPutVar** only in a macro.

Use **MLAppendMatrix**, **MLPutMatrix**, and **MLPutVar** to copy data from Excel to MATLAB

Use **MLEvalString** to execute MATLAB commands from Excel.

Use **MLDeleteMatrix** to delete a MATLAB variable.

Use **MLGetMatrix** and **MLGetVar** to copy data from MATLAB to Excel.

5.3 MATLAB

The graphical results which are shown by MATLAB Graphics are divided depending on the type of equations.

5.3.1 Plot

For the one-dimensional elliptic equation and one-dimensional parabolic equation, the graphic results are shown by using Plot which purposes a linear 2-D plot. The syntax of Plot is :

`plot(Y)` plots the columns of Y versus their index

`plot(...,'PropertyName', 'PropertyValue',...)` sets properties to the specified property values for all line graphic objects created by plot

`h = plot(...)` returns a column vector of handles to line graphics objects, one handle per line.

Other line characteristics can be specified by using graphics properties:

- `LineWidth` – specifies the width (in points) of the line.
- `MarkerEdgeColor` – specifies the color of the marker or the edge color for filled markers (circle, square, diamond, pentagram, hexagram, and the four triangles).
- `MarkerFaceColor` – specifies the color of the face of the filled markers.
- `MarkerSize` – specifies the size of the marker in units of points.

The user can adjust the axis tick-mark locations and the labels appearing at each tick. MATLAB also enables the user to add axis labels and titles.

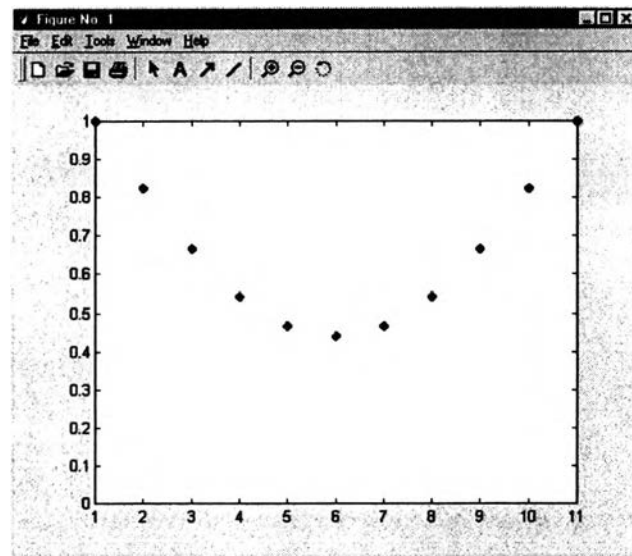


Figure 5.2 Example of plot function.

5.3.2 Surf Plots

Surf plots are used for showing the graphical results of two- and three-dimensional elliptic equations and for those of two-dimensional parabolic equations.

The syntax of mesh is `Surf(Z)` which displays mathematical functions over a rectangular region. Surf creates colored parametric surfaces specified by X, Y, and Z with color specified by Z. `Surf(Z)` creates a three-dimensional shaded surface from the z components in matrix Z, using $x = 1:n$ and $y = 1:m$. The height, Z, is a single-valued function defined over a geometrically rectangular grid. Z specifies the color data as well as surface height, so color is proportional to surface height.

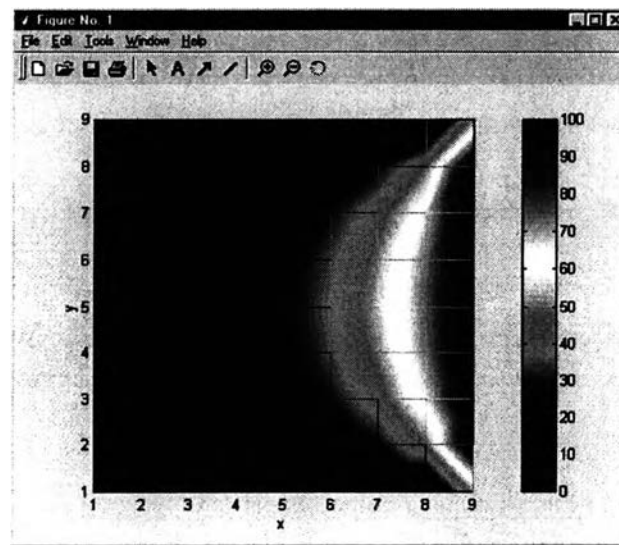


Figure 5.2 The graphical results using surf plot.

5.3.3 Movie

Movie in MATLAB Graphic purpose is to play recorded movie frames. The movie function displays each frame as it loads the data into the memory, and then it plays the movies. This eliminates long delays with a blank screen when you load a memory-intensive movie. For a parabolic equation, a movie can be shown with successive graphical results because the numerical results of parabolic equation depends on the time step. Use MATLAB movie to capture each frame in each time step and play it like an animation in which the graphs change as the time increases.

$F = \text{getframe}$ returns a movie frame. The frame is a snapshot of the current axes or figures.

$M = \text{moviein}(n)$ creates matrix M having n column to store n frames of a movie based on the size of the current axes.

$\text{Movie}(M, n, \text{fps})$ plays the movie n times at fps frames per second. The default value is 12 frames per second. Computers that cannot achieve the specified speed play as fast as possible.