LabVIEW MathScript Module

Hans-Petter Halvorsen

https://www.halvorsen.blog
LabVIEW MathScript Module

• You need to install an addition module called LabVIEW MathScript Module.

• This module can be used in 2 different ways:
  – LabVIEW MathScript – A separate application similar to MATLAB
  – MathScript Node integrated in LabVIEW Code
Contents

1. LabVIEW MathScript
   – Basic Examples
   – Plotting Examples
   – Simulation Examples
   – User-defined Functions

2. MathScript Node
LabVIEW MathScript

Hans-Petter Halvorsen
LabVIEW MathScript

• Add-on Module for LabVIEW where we can do text-based programming and simulations
• GUI and syntax is identical with MATLAB
• You can run MATLAB scripts in LabVIEW MathScript with almost no changed needed (assuming you use the core functionality or the MATLAB Control Toolbox)
• LabVIEW MathScript don’t have the same speed, flexibility and toolboxes as MATLAB
• If you know MATLAB, you know LabVIEW MathScript
LabVIEW MathScript

• MathScript is a high-level, text-based programming language. MathScript includes more than 800 built-in functions and the syntax is similar to MATLAB. You may also create custom-made m-file like you do in MATLAB.

• MathScript is an add-on module to LabVIEW but you don’t need to know LabVIEW programming in order to use MathScript.

• If you want to integrate MathScript functions (built-in or custom-made m-files) as part of a LabVIEW application and combine graphical and textual programming, you can work with the MathScript Node.

• In addition to the MathScript built-in functions, different add-on modules and toolkits installs additional functions. The LabVIEW Control Design and Simulation Module and LabVIEW Digital Filter Design Toolkit install lots of additional functions.
How do you start using MathScript?

• You need to install LabVIEW and the LabVIEW MathScript RT Module.

• When necessary software is installed, start MathScript by open LabVIEW.

• In the Getting Started window, select Tools -> MathScript Window...
Output Window

Here you can see the results of the calculations.

Script Window

This is the Editor where you create your program (script). The Script can be saved as a .m file.

Command Window

You can use the Command Window to enter single commands.
Basic MathScript
Examples

Hans-Petter Halvorsen
The Command Window is the main window in MATLAB. Use the Command Window to enter variables and to run functions and M-files scripts (more about M-files later). It’s like an advanced calculator!

- Hit “ENTER” in order to execute a command.
- Use “Arrow Up” in order to browse through old Commands (“Command History”).
Case sensitive variables

MathScript/MATLAB is **case sensitive**! The variables $x$ and $X$ are not the same.

```
>> x=5;
>> x=5;
>> X=6;
>> x+X
```
```
ans =
  11
```

Unlike many other languages, where the semicolon is used to terminate commands, in MathScript/MATLAB the semicolon serves to suppress the output of the line that it concludes.

```
>> x=3
x =
  3
>> y=4;
>>
```
MathScript Basics

The "clear" command deletes all existing variables from the memory.

The "clc" command removes everything from the Command Window.

clc – clear command window

Students: Try these commands
# Built-in Constants

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>i, j</td>
<td>Used for complex numbers, e.g., z=2+4i</td>
</tr>
<tr>
<td>pi</td>
<td>π</td>
</tr>
<tr>
<td>inf</td>
<td>∞, Infinity</td>
</tr>
<tr>
<td>NaN</td>
<td>Not A Number. If you, e.g., divide by zero, you get NaN</td>
</tr>
</tbody>
</table>

```matlab
>> r=5;
>> A=pi*r^2
A =
 78.5398

>> z1=3+3i;
>> z2=3+5i;
>> z = z1+z2
z =
 6.0000 + 8.0000i
```

Students: Try these examples
Students: Try this example

$$y(x) = \frac{3x + 2}{2}$$

$$y(2) =?$$

```
>> x=2;
>> y=3*x+2/2
y =
7
>> y=(3*x+2)/2
y =
4
```

Which are correct?

Students: Calculate this expression, try with different values for $x$ and $y$

$$z = 3x^2 + \sqrt{x^2 + y^2} + e^{\ln(x)}$$
Mathematical Expressions

Given the following Mathematical Expression:

\[ z = 3x^2 + \sqrt{x^2 + y^2} + e^{\ln(x)} \]

Students: Calculate this expression, try with different values for \( x \) and \( y \)
Students: Calculate this expression, try with different values for $x$ and $y$

$$z = 3x^2 + \sqrt{x^2 + y^2} + e^{\ln(x)}$$

Solutions:

```matlab
>> x=2;, y=2
>> z = 3*x^2 + sqrt(x^2 + y^2) + exp(log(x))
ans =
  16.8284
...
Students: Use MATLAB in order to find the surface area of a cylinder based on the height \( h \) and the radius \( r \) of the cylinder.

\[ r = 3 \]

\[ h = 8 \]

\[ A = ? \]
Solving Mathematical Problems - Solutions

Solutions:

```
>> h=8
>> r=3
>> A = 2*pi*r^2 + 2*pi*r*h;
A =
  207.3451
```

MathScript Code:
Plotting in MathScript
Plotting in MathScript

Example:

\[ y(t) = 2x + 4 \]

Useful MathScript functions for plotting:

- xlabel
- ylabel
- axis
- title
- grid
- text

Students: Try these functions

How does they work?
Type `help <function name>` in Command window

How do you get another color? or line type?

```
x = 0:5;
y = 2*x + 4;
plot(x, y)
```
Basic Plotting Examples

$$\begin{align*}
\text{Students: Try this example} \\
\gg x &= 0:0.1:2*\text{pi}; \\
\gg y &= \sin(x); \\
\gg \text{plot}(x,y)
\end{align*}$$

$$\begin{align*}
\text{Students: Try also these examples:} \\
\gg x &= 0:0.1:2*\text{pi}; \\
\gg y &= \sin(x); \\
\gg y2 &= \cos(x); \\
\gg \text{plot}(x,y, x, y2)
\end{align*}$$

$$\begin{align*}
\gg \text{plot}(x,y, 'r*', x, y2, 'g+')
\end{align*}$$
## Plotting Functions

### Plotting functions:

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>plot</td>
<td>Create a Plot</td>
</tr>
<tr>
<td>figure</td>
<td>Define a new Figure/Plot window</td>
</tr>
<tr>
<td>grid on/off</td>
<td>Create Grid lines in a plot</td>
</tr>
<tr>
<td>title</td>
<td>Add Title to current plot</td>
</tr>
<tr>
<td>xlabel</td>
<td>Add a Label on the x-axis</td>
</tr>
<tr>
<td>ylabel</td>
<td>Add a Label on the x-axis</td>
</tr>
<tr>
<td>axis</td>
<td>Set xmin, xmax, ymin, ymax</td>
</tr>
<tr>
<td>hold on/off</td>
<td>Add several plots in the same Figure</td>
</tr>
<tr>
<td>legend</td>
<td>Create a legend in the corner (or at a specified position) of the plot</td>
</tr>
<tr>
<td>subplot</td>
<td>Divide a Figure into several Subplots</td>
</tr>
</tbody>
</table>

---

**Students: Try this example**

```matlab
>> x=0:0.1:2*pi;
>> y=sin(x);
>> plot(x,y)
>> title('Plot Example')
>> xlabel('x')
>> ylabel('y=sin(x)')
>> grid on
>> axis([0,2*pi,-1,1])
>> legend('Temperature')
```

**Students: Try also to change some of the commands and see what happens with the plot**
Subplots

Students: Try these examples

```matlab
>> x=0:0.1:2*pi;
>> y=sin(x);
>> y2=cos(x);

>> subplot(2,1,1)
>> plot(x,y)

>> subplot(2,1,2)
>> plot(x,y2)
```

```matlab
>> x=0:0.1:2*pi;
>> y=sin(x);
>> y2=cos(x);
>> y3=tan(x);

>> subplot(3,1,1)
>> plot(x,y)

>> subplot(3,1,2)
>> plot(x,y2)

>> subplot(3,1,3)
>> plot(x,y3)
```

```matlab
>> x=0:0.1:2*pi;
>> y=sin(x);
>> y2=cos(x);
>> y3=tan(x);
>> y4=atan(x);

>> subplot(2,2,1)
>> plot(x,y)

>> subplot(2,2,2)
>> plot(x,y2)

>> subplot(2,2,3)
>> plot(x,y3)

>> subplot(2,2,4)
>> plot(x,y4)
```
MathScript Simulation Example

Hans-Petter Halvorsen

https://www.halvorsen.blog
Simulation Example

Assume the following model:

\[ \dot{x} = -ax + bu \]

We start by setting \( a = 0.25 \) and \( b = 2 \).

We can use e.g., the Euler Approximation:

\[ \dot{x} \approx \frac{x(k+1) - x(k)}{T_s} \]

Where \( T_s \) is the Sampling Time.

Then we get:

\[ \frac{x(k + 1) - x(k)}{T_s} = -ax(k) + bu(k) \]

Finally we get:

\[ x(k + 1) = (1 - T_s a)x(k) + T_s bu(k) \]
% Simulation of discrete model
clear, clc

% Model Parameters
a = 0.25; b = 2;

% Simulation Parameters
Ts = 0.1; %s
Tstop = 20; %s
uk = 1; % Step Response
x(1) = 0;

% Simulation
for k=1:(Tstop/Ts)
    x(k+1) = (1-a*Ts).*x(k) + Ts*b*uk;
end

% Plot the Simulation Results
k=0:Ts:Tstop;
plot(k,x)
grid on
% Simulation of discrete model
clear, clc

% Model Parameters
a = 0.25; b = 2;

% Simulation Parameters
Ts = 0.1; %s
Tstop = 20; %s
uk = 1; % Step Response
x(1) = 0;

% Simulation
for k=1:(Tstop/Ts)
    x(k+1) = (1-a*Ts).*x(k) + Ts*b*uk;
end

% Plot the Simulation Results
k=0:Tstop;
plot(k,x)
grid on
Creating Scripts and User-defined Functions in MathScript

https://www.halvorsen.blog

Hans-Petter Halvorsen
When using the Script Editor, you may create several lines of code and execute all in one batch. You can easily do changes in your code, create comments, etc.

MathScript/MATLAB Scripts are saved as so-called .m files (file extension is .m)

Clear
clc

x=0:0.1:2*pi;
y=sin(x);
y2=cos(x);
y3=tan(x);
y4=atan(x);

%plotting sin(x)
subplot(2,2,1)
plot(x,y)

%plotting cos(x)
subplot(2,2,2)
plot(x,y2)

%plotting tan(x)
subplot(2,2,3)
plot(x,y3)

%plotting atan(x)
subplot(2,2,4)
plot(x,y4)

Students: Try this example
User-defined Functions in MathScript

1. Create your function in the Script window.
2. Save your function as a .m file.
4. Add Search Folder for your Code.
5. Test your function in the Command window.
Function Example

MathScript/MATLAB contains hundreds of built-in functions, but very often you need to create your own functions.

You Create the Function in the Editor

You Use the Function in the Command Window or in a Script

Students: Try this example
Function Example

1. Create the Function

```
function Tf = fahrenheit(Tc)
Tf = (9/5)*Tc + 32;
```

The function needs to be saved as "fahrenheit.m" on your harddrive

2. Execute the Function

\[ T_F = \frac{9}{5} T_C + 32 \]

```
Tc = 23;
Tf = fahrenheit(Tc)
```

This can be done from Command window or Script window

Students: Try this example
Function Example

Example: Convert from Celsius to Fahrenheit

\[ T_F = \frac{9}{5} T_C + 32 \]

Students: Create a User-defined Function that converts from Temperature in Celsius to Temperature in Fahrenheit.

Try the function in the Command window like this:

```matlab
>> Tc = 20;
>> Tf = fahrenheit(Tc)
Tf =
68
```

You need to create this function.
Function Example

Solutions: Convert from Celsius to Fahrenheit

\[ T_F = \frac{9}{5} T_C + 32 \]

function Tf = fahrenheit(Tc)
% This function converts a temperature from celsius to fahrenheit
Tf = (9/5)*Tc + 32;

clear
clc
t = 0:0.1:24;
Tc = (sin(t)+1)*20;
Tf = fahrenheit(Tc);
plot(t,Tc, t,Tf)
title('Temperature Simulation')
xlabel('t')
ylabel('Temperature')
grid on
axis([0,24, 0, 120]);
legend('Celsius', 'Fahrenheit')
Tips & Tricks

Use Comments (%)

% This is a comment
x=2; % Comment2
y=3*x % Comment3

- but that have to make sense!

Decimal sign: Use "." – NOT ","!
i.e. y=3.2 – not y=3,2

Use english names on variables, functions, files, etc. This is common practice in programming!
Use always variables – Do not use numbers directly in the expressions!

Functions:
• Only ONE function in each File!
• The Filename (.m) AND the Name of the Function MUST be the same!

DO NOT use "spaces" in Filename or names that are similar to built-in functions in MathScript/MATLAB!

Yes:

a=2;
b=4;
y=a+b

No:

y=2+4

Always include these lines in your Script

clear
clc
close all...

...
Tips & Tricks

Greek letters: In math and control theory it is common to use Greek letters in formulas, etc. These cannot be used directly in MathScript/MATLAB, so you need to find other good alternatives. Examples:

- \( \omega_0 \) – \( w_0 \)
- \( \zeta \) – zeta or just \( z \)
- etc.

A Golden Rule: One Task – one file, i.e. **DON’T** put all the Tasks in one single file!!

Mathematical expressions: The following applies in MathScript/MATLAB

\[
z = 3x^2 + \sqrt{x^2 + y^2} + e^{\ln(x)}
\]

\[
z(2,2) = ?
\]

```matlab
x = 2;
y = 2;
z = 3*x^2 + sqrt(x^2 + y^2) + exp(log(x))
```

Use `help` in order to find out how to use a function in MathScript/MATLAB. In order to get help for the `tf` function, type the following in the Command window:

```
help tf
```

Greek letters:

- \( \omega_0 \) – \( w_0 \)
- \( \zeta \) – zeta or just \( z \)
- etc.

<table>
<thead>
<tr>
<th>Function</th>
<th>Equivalent in MathScript/MATLAB</th>
</tr>
</thead>
<tbody>
<tr>
<td>( x^2 )</td>
<td>( x^2 )</td>
</tr>
<tr>
<td>( \sqrt{x} )</td>
<td><code>sqrt(x)</code></td>
</tr>
<tr>
<td>( \ln(x) )</td>
<td><code>log(x)</code></td>
</tr>
<tr>
<td><code>log10(x)</code></td>
<td><code>log10(x)</code></td>
</tr>
<tr>
<td><code>exp(x)</code></td>
<td><code>exp(x)</code></td>
</tr>
<tr>
<td>( \pi )</td>
<td><code>pi</code></td>
</tr>
</tbody>
</table>
MathScript Node

Hans-Petter Halvorsen

https://www.halvorsen.blog
MathScript Node

• With MathScript Node you can create and use MathScript/MATLAB code within LabVIEW
Simulation using MathScript Node

Students: Try the same example inside LabVIEW using the MathScript Node

Just copy and paste the code from the previous example.
Hans-Petter Halvorsen

University of South-Eastern Norway

www.usn.no

E-mail: hans.p.halvorsen@usn.no
Web: https://www.halvorsen.blog