Integrating .m, .mdl and Real-Time Hardware for Math, Signal Processing, & Controls

Jeannie Falcon, Ph.D.
Chief Engineer, Control & Simulation

Lecturer, Department of Aerospace Engineering & Engineering Mechanics
Lecturer, Department of Mechanical Engineering
University of Texas at Austin
High-Level Design Models

Data Flow

C Code

Textual Math

1. \( c = 0.285 + 0.013i; \)
2. \( [X,Y] = \text{meshgrid}(x, y); \)
3. \( z = X + iY; \)
4. for \( k=1:30 \)
5. \( z = z^2 + c; \)
6. end

Simulation

Statechart

NATIONAL INSTRUMENTS

LabVIEW™

Desktop

Real-Time

FPGA

Microprocessors

ni.com
Embedded Software Development

- Design
- Prototyping
- Deployment
- System Test
- HIL Validation

Kc, Kp

ni.com

National Instruments
Deploy to Hardware Through LabVIEW

MathScript RT Module

Control Design & Simulation Module

The MathWorks Inc. software development environment

Your .m code

MATLAB®

Your .mdl code

Simulink®

Simulink Coder™

LabVIEW Real-Time

NI VeriStand

CompactRIO, Single-Board RIO, PXI, or desktop

MATLAB® and Simulink® are registered trademarks of The MathWorks, Inc.
MIT Feedback Control Systems Laboratory

- MIT Aerospace Engineering
- Prof. Jonathan How
- Students design and implement dynamic output feedback controller using CompactRIO, LV CD&Sim & MathScript
- 6 stations with 44 students (grad & undergrad)
- Ability to run simulation and control the physical plant all within the same VI
SW Product Overview

- Math & Optimization
- Statistics & Data Analysis
- Database Connectivity
- Application Deployment
- Computational Biology
- Financial Analysis

MATLAB®

- Control Design
- Signal Processing
- Communications
- Test and Measurement
- Image Processing

SIMULINK®

- Physical Modeling
- Fixed-Point Modeling
- Event-Based Modeling
- Simulation Graphics

- Code Generation
- RCP & HIL
- Embedded Targets
- Verification & Validation

MATLAB® and Simulink® are registered trademarks of The MathWorks, Inc.
SW Product Overview

- Math & Optimization
- Statistics & Data Analysis
- Database Connectivity
- Application Deployment
- Computational Biology
- Financial Analysis

- Physical Modeling
- Fixed-Point Modeling
- Event-Based Modeling
- Simulation Graphics

- Code Generation
- RCP & HIL
- Embedded Targets
- Verification & Validation

- Control Design
- Signal Processing
- Communications
- Test and Measurement
- Image Processing

MATLAB® and Simulink® are registered trademarks of The MathWorks, Inc.
SW Product Overview

- Math & Optimization
- Statistics & Data Analysis
- Database Connectivity
- Application Deployment
- Computational Biology
- Financial Analysis

- Motion Control
- Sound & Vibration

- Control Design
- Signal Processing
- Communications
- Test and Measurement
- Image Processing

- Physical Modeling
- Fixed-Point Modeling
- Event-Based Modeling
- Simulation Graphics

- Code Generation
- RCP & HIL
- Embedded Targets
- Verification & Validation

PLUS HARDWARE
LabVIEW MathScript RT Module

- Text-based controls, signal processing, analysis, and math
  - 900 built-in functions / user-defined functions
  - Reuse many of your .m file scripts created with The MathWorks, Inc. MATLAB® software and others
  - Based on original math from NI MATRIXx software
- A native LabVIEW solution
  - Interactive and programmatic interfaces
  - Does not require 3rd-party software
  - Enables hybrid programming

MATLAB® is a registered trademark of The MathWorks, Inc.
Octave vs MathScript

Octave:

octave:1> A=[0,2,0,1;2,2,3,2;4,-3,0,1;6,1,-6,-5]
A =
     0   2   0   1
     2   2   3   2
     4  -3   0   1
     6   1  -6  -5

octave:14> det(A)
ans = -234

octave:11> cond(A)
ans = 9.7626

MathScript:

>> A=[0,2,0,1;2,2,3,2;4,-3,0,1;6,1,-6,-5]
A =
     0   2   0   1
     2   2   3   2
     4  -3   0   1
     6   1  -6  -5

>> det(A)
ans =
    -234

>> cond(A)
ans =
     9.7626
LabVIEW MathScript RT Background

- Textual node & interactive window
- Added plug-ins:
  - LabVIEW Control Design & Simulation Module
  - LabVIEW Digital Filter Design Toolkit
- Compatible with The MathWorks, Inc.:
  - MATLAB®
  - Signal Processing Toolbox™
  - Control System Toolbox™
  - DSP System Toolbox™

*Useful for desktop or real-time applications with hardware*

MATLAB® and Simulink® are registered trademarks of The MathWorks, Inc.
DEMO: MathScript Interactive Environment

![MathScript Interactive Environment](image)

- `grid on` and `grid off` commands to toggle the grid.
- `a=[1 2; 3 4]` creates a 2x2 matrix.
- `b=imw(a)` modifies a variable.
- `eig(a)` calculates eigenvalues of matrix `a`.

Output includes:
- Matrix `a`:
  
<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
</tr>
</tbody>
</table>

- Matrix `b`:
  
<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>-2</td>
<td>1</td>
</tr>
<tr>
<td>1.5</td>
<td>-0.5</td>
</tr>
</tbody>
</table>

- Eigenvalues of `a`:
  
  - 0.37228
  - 5.3723

- Graphical interface with data tables and variables.
LabVIEW MathScript DEMO

- Graphical and textual programming
- Interactive user-interface
LabVIEW MathScript – Heat Equation

C:\Program Files (x86)\National Instruments\LabVIEW 2012\examples\MathScript\MathScript - Heat Equation
Debugging A MathScript Node (DEMO)

- Execution highlighting and single-stepping
- Probe tool for:
  - Variables in each node
  - Output defined within node
- Syntax error indication for each line
- Error indicator for node
Importing your .m file and adding interactivity

Fs = 150; % Sampling frequency
t = 0:1/Fs:1; % Time vector of 1 second
f = 5; % Create a sine wave of f Hz.
x = sin(2*pi*t/f);
nfft = 1024; % Length of FFT
% Take fft, padding with zeros so that length(X) is equal to nfft
X = fft(x,nfft);
% FFT is symmetric, throw away second half
X = X(1:nfft/2);
% Take the magnitude of fft of x
mx = abs(X);

% Frequency vector
f = (0:nfft/2-1)*Fs/nfft;
% Generate the plot, title and labels.
figure(1);
plot(t,x);
title('Sine Wave Signal');
xlabel('Time (s)');
ylabel('Amplitude');
figure(2);
plot(f,mx);
title('Power Spectrum of a Sine Wave');
xlabel('Frequency (Hz)');
ylabel('Power');

http://www.utdallas.edu/~dlm/3350%20comm%20sys/FFTandMatLab-wanjun%20huang.pdf
Dr. Andy Clegg, Lead Engineer & ISC Managing Director:
“We used the LabVIEW MathScript RT Module to run a textual node containing our m-file code, developed with MATLAB on a desktop, for kinematics on the real-time CompactRIO controller. MathScript has significant benefits for real-time deployment on the CompactRIO controller including determinism, easy debugging, and no extra compilation steps.”
Video “TAS (Turbine Access System)”

- http://www.youtube.com/watch?v=N21en93lrqg
NI USRP | RF Direction Finding & Beamforming

- Testing MUSIC direction finding algorithm
- Rapid prototyping in LabVIEW with MathScript RT
- Synchronized up to 12 USRP devices
- Reference provides continuous phase alignment compensation

Prof. Athanassios Manikas
Comm & Array Processing Chair
Imperial College, London
USRP Demo
Don’t Get Confused About the Nodes

- Native LabVIEW code
- Works with:
  - LabVIEW on Windows, Mac or Linux
  - LabVIEW Real-Time
- Requires MATLAB
- Only works with LabVIEW on Windows
- Does not work with LabVIEW Real-Time
Recommended options for analyzing data with MathScript (DEMO)

- DAQ Assistant plus MathScript node
- Instrument Drivers plus MathScript node
- I/O Asst plus MathScript node
Working with LabVIEW MathScript

- Develop scripts interactively with the MathScript Window
- Move to the MathScript Node to “Instrument your Algorithms”
- Move back and forth as necessary to complete your work