THE FLIGHT TEST ENGINEER’S USE OF MATLAB

Thomas Lampe
MATLAB NORDIC EXPO 2014-05-22
DISPOSITION

- Company presentation
- Flight Test Instrumentation overview
- Analysis System overview
- Stella – in-house developed toolbox in MATLAB
- Real-time data processing
- Conclusions
SAAB GROUP

*Independent subsidiary reported as a business area
THE INSTRUMENTED TEST AIRCRAFT

Telemetry

Instrumentation system

Database:
- Test card
- Instrumentation setup
- Weight data
- Observations
- ...

Data files
- raw
- processed (HDF5)

Analysis
ANALYSIS SYSTEM OVERVIEW

Purpose:
- Applied data processing
- Analysis
- Visualization
- Data export
- Report generation

Key requirements:
- Stability and flexibility
STELLA TOOLBOX

Stella HDF5 datafiles
hubble

Matlab + Stella
imago

stellaRT

Telemetry

jbrowser
grape

arena
TIMELINE

First flight with Gripen

2000

First production release of Stella in Matlab 5.3.1

2010

First release of stellaRT
MAINTENANCE

- Design solutions from year 2000 (Matlab 5.3.1) and onwards
- Platforms: Solaris+Windows ► Linux ► Linux x64
- Minimal interference between Matlab installations and Stella installations
- Name conflicts …
- Testing, testing, testing …
DATA FILES

Stella HDF5
- Variable/time segment block structure
- Archive format
- Associated with one test
- Main source for post-flight analysis.

"sim" files
- "Data stream" structure
- Chronological
- Adapted for real-time analysis
STELLA DATA OBJECTS

- param (float type) / pardi (discrete/integer type)

```matlab
>> s2737

s2737 =

status: []
  date: 2012-02-20
  start: 50435.4953
  end: 50439.5461
  note: []
    id: '7564'
    unit: 'm/m'
  system: ...
  sync: 2
  freq: NaN
  freq.orig: 256
  vrtime: [1x1038 double]
  vrn: [1x1038 double]
```

- Date - separate variable!
- Seconds since 00:00 UTC
- Test identity
- Time reference
- Data vector, always "double"
## STELLA FUNCTION REFERENCE

**Function:** spike

Remove spikes (outliers) from 1-D signals.

**Syntax**

```
[par1, par2, ..., distMin] = spike(par1, par2, ..., aaplMin, widthMax)
```

**Description**

The spike command finds and replaces spikes in the parameters `par1`, `par2`, ..., `parN`, according to input arguments. A spike has a minimum difference in amplitude, `aaplMin`, towards the surrounding samples. To be detected it also must contain no more than `widthMax` samples. `distMin` is the minimum distance between two spikes. Spikes will be replaced with NaN. The default value 2 will be used for `widthMax` and `distMin`, if they are not specified.

**Example**

![Example](image.png)

Figure 1: Definitions in spike.
CODE SAMPLES

Examples

Remove spikes from the parameter p3 and plot the result:

```matlab
>> p3s = spike(p3,1,1,2);
>> plot(p3s,vrtime,p3s,vrtime,p3s,vrtime,p3s,vrtime,'o-','r')
```

Figure 3: The samples shown in dashed red are those which have been found and replaced with NaN.
STELLA GRAPE
CASE 1: TAKE-OFF DISTANCE

- Rough estimation of take-off distances (lift-off and 15 m obstacle clearance)
- Variables:
  1. weight-on-wheel logic
  2. radio altitude [m]
  3. wheel speed
  4. latitude
  5. longitude
CASE 1: CALCULATION OF DISTANCE

% --- Sample code from lolatoTM reference manual ----
% vrlola is a 2x1 double containing longitude and latitude (in radians)
% at brake release

mp.zone = 'local';
[vrxy,mpout]=lolatoTM(vrlola,mp); % vrxy will contain zeros. The important
% output is mpout.

mp2 = mpout;
parlola2 = join (parlo; parla); % parlo and parla are synchronized
% single-row param objects
vrxy2 = lolatoTM(parlola2,mp2); % The system defined above is used

% ---

[x, y] = split(vrxy2);

dist = sqrt(x^2 + y^2);
CASE 1: DISTANCE TO LIFT-OFF

Stella plot tool "imago"
CASE 1: VISUALIZATION AND ANALYSIS
CASE 1: DISTANCE TO 15 M OBSTACLE CLEARANCE

"Seeing is believing"
Real-time applications – stellaRT

- Real-time evaluation
  - Connection to telemetry data server
- “Simulated evaluation”
  - Real-time simulation with "sim" file (can be created from Stella HDF5 file)
stellaRT script execution

 Scripts `stellaRT_start.m`
  `stellaRT_run.m`
  `stellaRT_evaluate.m`
  `stellaRT_action.m`
  `stellaRT_pause.m`
  `stellaRT_stop.m`

are user-defined for each project.

 Scripts can be renamed in a stellaRT project
USER CASE 2: GROUND COLLISION AVOIDANCE SYSTEM (GCAS)
USER CASE 2: GROUND COLLISION AVOIDANCE SYSTEM (GCAS)
CONCLUSIONS

MATLAB with the Stella toolbox is a reliable and flexible platform for flight test data analysis.

Real-time and early analysis based on telemetered data are prioritized development areas.