Managing Calculation Scenarios in SIMPACK using MATLAB

Marc Stiepel

6\textsuperscript{th} December 2018
1. Examples of Railway Vehicle Platforms

2. Dynamic Assessment of Railway Vehicles

3. SIMPACK – MATLAB Scenario-Manager

4. Summary
Examples of Railway Vehicle Platforms

Coradia Lint – Coradia iLint

- Regional Trains with \( v_{\text{max}} = 140\text{km/h} \)
- Single or multiple Carbodies
  - Non-articulated train, up to 3 carbodies \( \Rightarrow \) Lint 27 / 54 / 81
  - Articulated train, 2 carbodies \( \Rightarrow \) Lint 41
- Power Source
  - Diesel powered train \( \Rightarrow \) standard Coradia Lint
  - Hydrogen powered train \( \Rightarrow \) Coradia iLint
- Bogie Types
  - Classical motor and trailer bogies
  - Jakobs bogie for Lint 41 (articulated)
- Well accepted Platform
  - Appr. 900 trains in service
  - more than 80 Mio. km/year
  - at 20 customers

Lint 27
Lint 41 (articulated)
Lint 54
iLint – Lint 54 hydrogen power
Lint 81
Examples of Railway Vehicle Platforms

**Coradia Continental**

- Regional Trains with $v_{\text{max}} = 160\text{km/h}$
- Electrical multiple Units, articulated
  - 3-car / 4-car / 5-car / 6-car units
  - Flexible length of end car, standard + XL
  - 2 floor-levels available
- Bogie Types
  - Classical end-bogies incl. traction
  - Jakobs bogie as
    - Motor bogie
    - Trailer bogie
- Well accepted Platform
  - 222 trains in service
  - more than 35 Mio. km/year
  - in 13 different versions
  - at 6 customers
Examples of Railway Vehicle Platforms

Coradia Stream

- Regional and Intercity Trains with $v_{\text{max}} = 200\text{km/h}$

- Electrical multiple Units
  - 3-car, 4-car, ..., up to 8-car units
  - Articulated
  - Articulated with short cuppling (*mixed architecture*)

- Bogie Types
  - Classical end-bogies as
    - Motor bogie
    - Trailer bogie
  - Jakobs bogie as trailer bogie

- New Development, currently
  - 7 different units
  - 2 contracts, 180 vehicles

8-car unit

3-car unit

4-car unit

5-car unit
Examples of Railway Vehicle Platforms

Railway Vehicles - Vehicle Platforms

- **Vehicle Platform Definition**
  - Typical for trams / metros, regional trains, High Speed (up to 360km/h), locomotives
  - Predefined set of carbodies / carbody types
  - Standard types of bogies
    - End-/ mid-bogie
    - Motor-/ trailer-bogie

- **Modularisation**
  - Flexible number of carbodies
  - Varying internal layout of car bodies, => depending on customer requirements
  - Mixed architecture, non-articulated / articulated
  - Traction level

Railway Vehicle Platforms offer a high modularity while number of unique vehicles is low!
Dynamic Assessment of Railway Vehicles

Railway Vehicle Dynamics – Calculations and Tests

Vehicle Dynamic design – Determination of Suspension Characteristics

• Safety against derailment
• Vehicle sway, gauging parameters
• Dynamic assessment, curving performance, running safety, track loading
• Running stability
• Safety against crosswind
• Ride characteristics, passenger comfort
• ...

Homologation of Railway Vehicles / Vehicle Platforms

• Single vehicle / reference vehicle
  - Full range of on-track tests is required for approval (EN 14363)
  - Additional calculational proofs
• Homologation of platform vehicles – extension based on reference solution
  - Additional tests, simplified range or
  - Calculational proof instead of on-track test
• ...

ALSTOM - 30/11/2018 – P 7

© ALSTOM SA. 2015. All rights reserved. Information contained in this document is indicative only. No representation or warranty is given or should be relied on that it is complete or correct or will apply to any particular project. This will depend on the technical and commercial circumstances. It is provided without liability and is subject to change without notice. Reproduction, use or disclosure to third parties, without express written authorization, is strictly prohibited.
Dynamic Assessment of Railway Vehicles

Assessment of On-Track measurements
Evaluation of the Test-Results

- Using in-house development – no commercial solution available
- Based on MATLAB
- Full range of assessment according to EN14363 is implemented

Pros
+ Powerful tools for signal processing
+ Code generation and debugging is comparatively easy
+ Availability of powerful plot tools => suitable for documentation

Cons
- Maintenance of the software has to be managed
Dynamic Assessment of Railway Vehicles

Theoretical studies using MBS software packages

ALSTOM has chosen SIMPACK as standard MBS software

Pros
+ Generally accepted Wheel-Rail-Contact
+ Excellent solver technology
+ Multi-level sub-structuring of models

Cons
- Post processing of results - documentation
  - No suitable standard channel names are available
    => renaming is always necessary
- Signal processing is restricted and not powerful
- Scripting language
  - Implementation and debugging is uncomfortable
  - No extended Mathematics / Numerics or signal processing included

=> Check availability of options to combine SIMPACK and MATLAB
### Extended Options available with – SIMPACK 9.x / 10.x / 20xx

#### Solver Part
- Solver call from the command line
  - => in combination with SIMPACK-Scripting
- Use of Component Object Model
  - => COM-interface
- defined interface SIMPACK ↔ MATLAB
- Modification of SIMPACK model possible
  - => Parameter variation within MATLAB
- Application of the interface had to be developed
  - => both solutions have been implemented!

#### Result Export
- Use of MATLAB result export
  - Large extra files if combined with SBR
  - Extending duration for measurements
- Use of Component Object Model
  - => COM-interface
- Selection and renaming of channels had to be implemented
  - => solved
- Allows direct reading of SIMPACK SBR files within MATLAB!
  - => No MATLAB result export necessary

=> Start of development of “SIMPACK / MATLAB Scenario Manager”
The Idea - Requirements

Effective calculation of standard tasks

- Static calculations
  - Safety against derailment => twist and Y-force calculation
  - Static sway
  - ...
- Dynamic calculations
  - Curving analysis, entry, passing, s-shaped, …
  - Nonlinear stability
  - Crosswind calculation
  - …
- Linear analysis

Standardized evaluation of results

- Include renaming of channels
- Automated generation of plots / tables
  - “Documentation ready”
  - Supporting several languages
- Include required filtering
- Full data access in MATLAB
  - Time series data from SBR incl. meta data
  - Evaluated results - tables
  => for specific and further evaluations

For all required vehicle configurations

- Load-cases, empty, normal payload, …
- Airspring / auxiliary spring operation
- Failure modes, e.g. Anti-Yaw-Damper

Solution should be independent from: model structure => applicable to platforms
model naming => use of external models
Scenario Manager – The Solution

On SIMPACK side – Recommended Setup …

Split model into **Scenario** defining

- Operational modes
  - Airspring
  - Aux. spring (with/without levelling)
  - …

- Load cases
  - Operational load
  - Normal load
  - …

- Calculational modes
  - Static calc. (twist test, …)
  - Dynamic calc., curving, ….

- Tracks

and **Vehicle**

- Loaded as substructure into the scenario => can be exchanged easily

- Supporting basic settings from scenario
  - Load cases
  - Operational modes
  - Calculational modes

- Does not contain modelling elements

- Assembly consisting of
  - Carbodies
  - Bogies
  - Articulations
  - …

**Same scenario can / will be used for numerous vehicles / vehicle platforms**
On MATLAB side …

Object oriented solution is chosen consisting of a

- Project => container
- Scenario => representing the SIMPACK Scenario
- Vehicle => representing the SIMPACK Vehicle
- Calculation-objects => defining solver-runs
  - Predefined standard tasks, curving, crosswind, nonl. stability, ...
  - List of predefined SubVars to vary => SubVar-names can be defined
  - List of additional SubVars to vary => allow individual variations
  - Access to tracks, solver-settings, ...
- Evaluation-objects => perform the evaluation of any SBR
  - Reads the SIMPACK-SBR via the COM-interface
  - Calculates eval.-values, Max. or End value of force, RMS of acc., ...
  - Ability to plot results, time-series of calculation incl. evaluation-values
  - Bar-plots of set of calculations for comparison
Scenario Manager – Defining the Scenario

The MATLAB Scenario

- MATLAB object of type `ScenarioM.Scenario`
- represents the SIMPACK Scenario
- is defined as
  - MATLAB script .m-file
  - MATLAB binary file .mat-file
- defines
  - SIMPACK-model to be used
  - Substructure name of vehicle within the model
  - Operational modes, load cases and calculational modes
- mainly on the basis of SubVars

Independence from model structure & model naming

% --- basic configuration ---------------------------------------------
% filename of Simpack-Model containing the calc. Scenario
Sc       = ScenarioM.Scenario;
Sc.SPCK.fFile = 'D:\SIMPACK\Modelle\Scenario.spck';
% vehicle Substructure within Main-Model
Sc.Veh.SimpackName = '$S_Veh$';

% --- opModes --------------------------------------------------------
Sc.opMode.SubVar               = {'$G_Flag.$_Deflated'...
     '$G_Flag.$_Levelling'};
Sc.opMode.State(1).DisplayName = 'Airspring w. Lev.';
Sc.opMode.State(1).Value       = [0 1];
Sc.opMode.State(2).DisplayName = 'Aux. Spring';
Sc.opMode.State(2).Value       = [1 0];

% --- loadCases ------------------------------------------------------
Sc.loadCase.SubVar               = '$_LoadSetIndex';
Sc.loadCase.State(1).DisplayName = 'op. Load';
Sc.loadCase.State(1).Value       = 1;
Sc.loadCase.State(2).DisplayName = 'norm. Load';
Sc.loadCase.State(2).Value       = 2;

% --- calcModes ------------------------------------------------------
Sc.calcMode(1)             = ScenarioM.cState();
Sc.calcMode(1).DisplayName = 'dynamic Calc.';
Sc.calcMode(1).Track       = '$Trk_Track$';
Sc.calcMode(1).SubVar(1)   = ScenarioM.SubVar('Name', '$G_Flag.$_B55', 
     'Value', 0);
Sc.calcMode(1).SubVar(2)   = ScenarioM.SubVar('Name', '$_vVeh$', 
     'Value', 2, 'Unit', 'm/s');
Scenario Manager – Defining the Vehicle

The MATLAB Vehicle

- MATLAB object of type *ScenarioM.Veh.Vehicle*
- represents the SIMPACK Vehicle
- is defined as
  - MATLAB script .m-file
  - MATLAB binary file .mat-file
- defines
  - Carbodies, bogies, wheelsets (implemented as objects)
  - Standard names – *DisplayNames* – support of TeX-style
    => to be used for plots
  - Element names – Substructure, Body, FEs, ...
    => access to SBR-channels

Independence from model structure & model naming
The MATLAB Calculation-Object

split in Set & Case

- Set defining => setup, can be referenced by multiple cases
  - SubVar names for calculation input
  - Solver-setting, track to be used
  - Possibility to set additional SubVars

- Case setting => calculation task defining the solver-run
  - Values for the input parameters
  - Vehicle configuration, load case, operational mode
  - Solver-setting, track can be overwritten in the case
  - Possibility to set additional SubVars

- Several standard tasks implemented => curving, crosswind, ...

- Generalized task available

% Set
XWindS = ScenarioM.XWind.XWindS;
XWindS.calcMode = Proj.calcMode(1); % calcMode to be used
XWindS.Track = 'Trk_CrossWind'; % Track for calc.
XWindS.SlvSet = 'SLV_XWind'; % Solver-Setting
XWindS.ParN4vVeh = '$vVeh$'; % SubVar def. veh. velocity
XWindS.ParN4vWind = '$vWind$'; % SubVar def. wind velocity
XWindS.ParN4betaW = '$betaW$'; % SubVar def. wind angle
XWindS.ParN4aq = '$aq$'; % SubVar def. unb. lat. acc.
XWindS.SubVar(1) = ScenarioM.SubVar('Name', 'Flag.XWind', ...
  'Value', 1)); % => free parameter

% Case
XWindC = ScenarioM.XWind.XWindC();
XWindC.Set = XWindS;
XWindC(1).loadCase = 1; % load Case
XWindC(1).opMode = 1; % op. Mode
XWindC.vVeh = '120km/h'; % veh. velocity
XWindC.aq = '0m/s^2'; % unb. lat. acc.
XWindC.vWind = '20m/s'; % wind velocity
XWindC.betaW = 90; % wind angle
XWindS.SubVar = []; % add. SubVars to set
XWindC.calc(); % run the calc.

=> Allows repetition of complete sets of different calculations
The MATLAB Evaluation-Objects

- Evaluation-Objects are based on the Scenario and the Vehicle
- SBR is processed automatically
- Evaluation is independent from the calculation-objects
- Several standard tasks implemented => curving, crosswind, …
- General access of SIMPACK-results is possible
- Evaluations include
  - Signal-processing
  - Determination of characteristic values, Min / Max, RMS, …
- Results are accessible
  - Directly in MATLAB
  - As formatted tables in Excel – including export as graphics
  - Plot of time history including presentation of characteristic values
  - Bar-plots of characteristic values used for comparing sets of calculations

MATLAB code example

```matlab
XWind = ScenarioM.Eval.EvXWind;
% set Scenario
XWind.Scenario = "...\ExampleSc.m";
% set Vehicle
XWind.Veh = "...\ExampleVeh.m";
% Define static Wheelloads
XWind.Q0 = Q0;
% set the Result-file
XWind.Result = "...\Example.sbr";
% plot results
XWind.plot();
```
Scenario Manager – The Evaluation-Object

Evaluation-Object Example

• Using in-house Plot-environment in MATLAB
• General information is always included
  - Vehicle name
  - Load case, operational mode
  - Result information, filename, date
• Example for crosswind calculation acc. to EN14067-6
  - Wheelloads of all wheels \( RS_i \cdot Q_{re/l} \)
    => raw data from SBR
  - Sum of wheelloads per bogie side \( \Sigma Q_{re/l} \)
    => calculated and filtered
  - Residual wheelload
    => criteria for calculation
  - Wind velocity acting on carbodies \( WK_{CB} \cdot v_{Wind} \)
    => standardized gust acc. to EN 14067-6
Scenario Manager – The Evaluation-Object

**Evaluation-Object – Example**

- Comparison of multiple tasks / runs
- Curving performance – dyn. derailment coefficient Y/Q
  - Max. Y/Q within bogie while curving
  - 4 bogies
  - 4 curves differing in radius – 300m / 500m / 800m / 1000m
- 4 Vehicle configurations
  - Operation on airspring / auxiliary spring
  - For operational load and exceptional load of vehicle

MATLAB code example

```matlab
% bar-plot of selected results
Curve(1:16).bar();
```

Overview via
- A single characteristic value
- 16 calculations in total
Managing Calculation Scenarios in SIMPACK using MATLAB

Summary

- ALSTOM CoE Regional Bogies is assessing Railway Vehicle Dynamics using following software packages:
  - SIMPACK for Multi-Body-System calculations
  - MATLAB for numerical calculations and the evaluation of on-track measurements

- A tool for managing SIMPACK calculations and evaluations within MATLAB has been developed
  - Based on a simple description of the SIMPACK Scenario / Vehicle in MATLAB
  - Using the Component Object Model of SIMPACK => access to SIMPACK model & SBR
  - Independent from model structure and naming => generalized application
  - Serves as replacement for SIMPACK DoE => higher flexibility
  - Automatized output of plot pages, tables, … => „Documentation ready“

The Scenario Manager is increasing the efficiency in solving standard tasks during vehicle design