Overview of Mathematica and Matlab suspension modeling assets

Mark Barton

SUS/Technical Meeting

6/11/15

LIGO-G1400629-v2

Mathematica vs. Matlab

- There are Mathematica and Matlab models for all the SUS and honorary-SUS suspensions.
- The Mathematica models are entirely by Mark Barton.
- The triple and quad Matlab models were originally by Calum Torrie and Ken Strain, but their innards have been replaced by matrix element generation code exported from Mathematica.
- Other Matlab models have been generated on the pattern of the triple and quad.
- A single Matlab model handles exported matrix elements from several Mathematica models with a common number of masses (quad/triple/etc).

Documentation

- The SUS Ops Manual (E1200633) links to most of the resources mentioned in this presentation:

  https://awiki.ligo-wa.caltech.edu/aLIGO/Suspensions/OpsManual/Models
- The manual for the Mathematica is T020205.
- The manual for the Matlab is T080188.
  - » Note however that Jeff K modified the calling convention for the parameter files in the version of the Matlab in the SVN relative to that described in T080188 – I'll do an update if I get the chance.

SVN Information

- All Mathematica and Matlab models for aLIGO are on the SUS SVN (LIGO.ORG credentials required).
- Web browser access root URL (read-only):
  https://redoubt.ligo-wa.caltech.edu/websvn/listing.php?repname=sus
- SVN client access root URL:
  https://redoubt.ligo-wa.caltech.edu/svn/sus/
- There are large numbers of additional Mathematica models on Mark Barton's personal laptop (mirrored on LIGO-issue iMac). These are mostly in older non-SVN-friendly format.

Model Names and SVN Locations

Suspension	Mathematica Model	Matlab
SVN Location (relative to SUS root):	^/trunk/Common/ MathematicaModels/	^/trunk/Common/MatlabTools/ (except QUAD)
QUAD	QuadLite2Lateral	^/trunk/*QUAD*/Common/MatlabTools/ QuadModel_Production
BSFM	TripleLite2	TripleModel_Production
HLTS	TripleLite2	TripleModel_Production
HSTS	TripleLite2	TripleModel_Production
OMCS	DualLite2	DualModel_Production
TMTS	DualLite2DBLateral	DualModel_Production
HAUX	TwoWireSimple	SingleModel_Production
HTTS	TwoWireSimple	SingleModel_Production
OFIS	FourWireSimple	SingleModel_Production

MathematicaModels Directory

^/trunk/Common/MathematicaModels

DualLite, for OMCS

DualLite2DBLateral, for TMTS

Cases with various parameter sets

Case definition notebook

User calculation directory

Calculation directory for Ops Manual stuff

Pre-computed results

Standard calculation notebooks

Model definition notebook

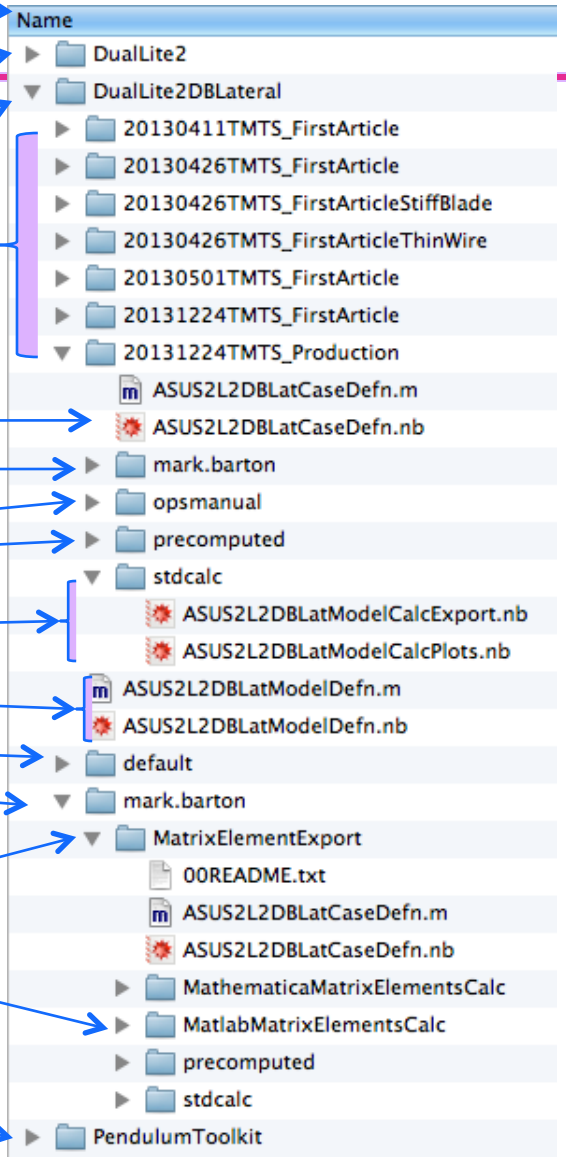
Default case directory

User case directory for Mark Barton

Dummy case for calculation of symbolic matrix elements

Calculation of Matlab matrix elements

Pendulum Toolkit



PendulumToolkit

- ^/trunk/Common/MathematicaModels/PendulumToolkit
- aWiki: Suspensions/MathematicaModels/Toolkit
- Key files:
  - PendUtil.nb – main toolkit
  - IFOModel.nb – adapted from GWINC, somewhat out of date
  - MatlabExport.nb – export variables/vectors/arrays in Matlab format
  - E2EExport.nb – export variables/vectors/arrays in E2E format
  - MyShapes.nb – shapes for drawing eigenmodes
  - StatusWindow.nb – draws the window reporting on progress of calculations
  - RotationsXYZ.nb – yaw/pitch/roll geometry
  - WedgeMOI.nb – calculates the mass and MOI of optics with wedge and/or flats
  - WikiExport.nb – exports wiki text and mode plots for use in Operations Manual
  - DualDBLatMatlabParamExport.nb etc – export parameter files in Matlab format
  - QuickDual.nb etc – use fast Matlab-style symbolic calculation in Mathematica

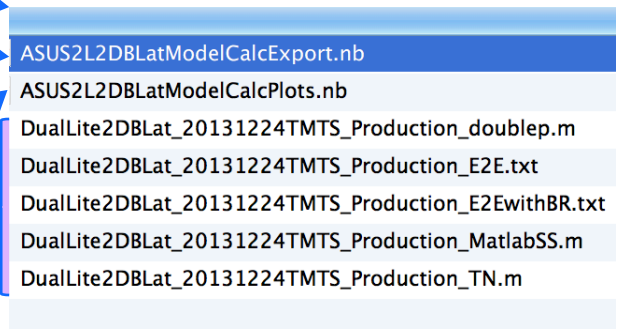
Standard Calculation

^/trunk/Common/MathematicaModels/DualLite2DBLateral/20131224TMTS_Production/stdcalc

Standard set of exported data (see below)

Standard calculation and set of example plots (mode shapes, transfer functions and thermal noise)

Standard set of exported data:
Matlab-style parameter file
SS matrices in E2E format
SS matrices with back reaction
SS matrices in Matlab format
Thermal noise data

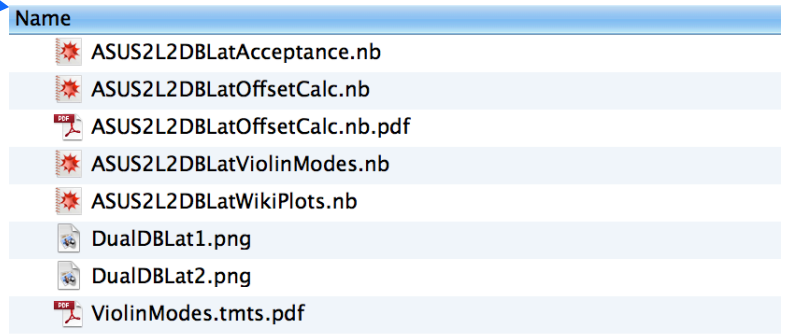


Note: the files in this directory should not be modified unless an outright error is discovered. Rather, create your own albert.einstein calculation directory sibling to stdcalc and copy over either/ both of the .nb files.

Personal Calc Directory mark.barton

^/trunk/Common/MathematicaModels/DualLite2DBLateral/20131224TMTS_Production/mark.barton

Every user is encouraged to create a personal calculation directory named in albert.einstein format for any cases they want to work with. It should be sibling to stdcalc.



My calc directories contain random useful stuff that was generated in response to queries by Norna et al., including (typically) notebooks on violin modes and buoyancy correction – feel free to rummage through, copy anything and adapt it.

Before trusting the contents of a notebook, check that the modelcase value at the top matches the case directory the notebook is in. The notebook may have been copied from another case without being re-run.

Ops Manual Export (i)

^/trunk/Common/MathematicaModels/DualLite2DBLateral/20131224TMTS_Production/opsmanual

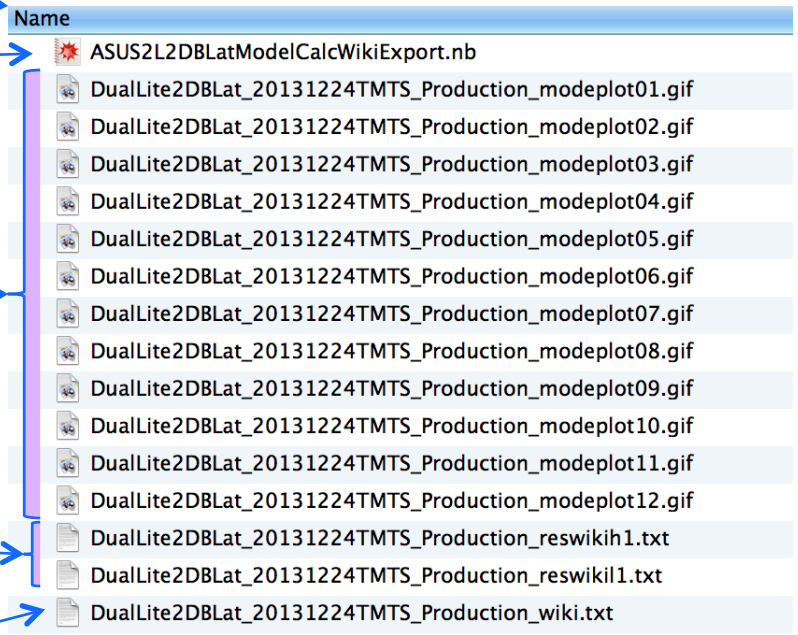
Notebook that does export

Automatically generated mode plot GIFs, to be uploaded as attachments to Ops Manual model page for 20131224TMTS_Production,

Suspensions/OpsManual/TMTS/Models/20131224TMTS_Production

Automatically generated wiki source-code fragments, to be added to the Resonance Wiki

Automatically generated wiki source-code, to be used as the Ops Manual model page



Ops Manual Export (ii)

Export to wiki

- **Setup**

Switches to enable loading of previously saved results instead of recalculating from scratch

```
useprecomputed = True; (* set to True to use saved results from precomputed subdirectory *)
If[useprecomputed,
    exceptdamping = False, (* False by default, True to recalculate just damping-dependent stuff*)
    exceptdamping = True (* DON"T CHANGE *)
];

loadcasefromuser["ASUS2L2DBLatCaseDefn.m"];

modelcase

20131224TMTS_Production

modelcasecomment

Production case of TMS with data from D0902773-v8
  of 12/24/13, equivalent to tmtsopt_production r6273 of 2/
```

- **Export**

```
modenames = {
    modeY1, modeT1, modeL1, modeR1, modeP1, modeV1,
    modeT2, modeL2, modeV2, modeP2, modeY2, modeR2
  };
```

This code assigns names to modes. It will need editing depending on the exact order (list is ascending order by frequency). Easiest is to run this notebook with any old names, upload the generated wiki source and GIF attachments to the wiki, review it there, adjust the list as needed, then re-export. The GIFs will not change and do not need to be re-uploaded.

Matrix Element Export

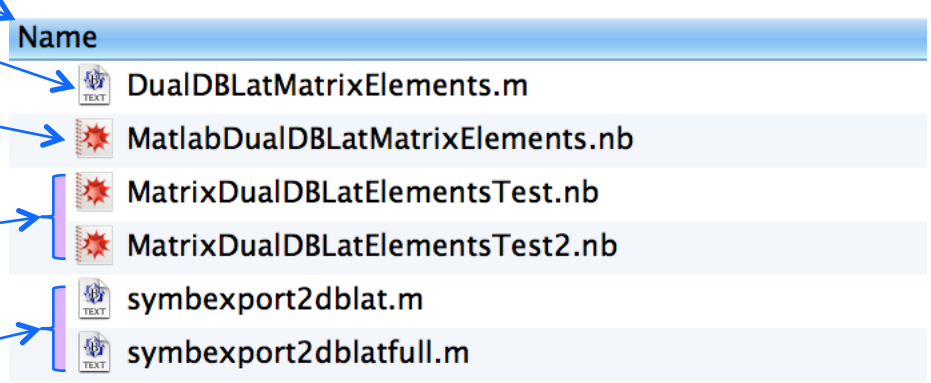
^/trunk/Common/MathematicaModels/DualLite2DBLateral/mark.barton/MatrixElementExport/MatlabMatrixElementsCalc

Symbolic Mathematica matrix elements, for testing

Exports Matlab matrix elements, plus equivalent Mathematica set for testing

Tests against Stage0 and Stage2 results (without and with wire bending)

Symbolic Matlab matrix elements, for copying to DoubleModel_Production



Matlab Models

^/trunk/Common/MatlabTools/DoubleModel_Production
(handles OMCS and TMTS; representative of others)

Does comparison plots of selected parameter sets

Defines labels for inputs and outputs

Does quick summary of parameters, modes

Maps short names like tmtsopt_metal
used in captions to parameter file names

Assembles exported matrix elements
into State-Space A/B/C/D

Exported matrix elements from Mathematica

Parameter sets for OMCS and TMTS