LIGO Laboratory / LIGO Scientific Collaboration

LIGO- E1000488	*LIGO*	22 February 2011

aLIGO SEI

BSC-ISI, Commissioning Phase III

After insertion – Commissioning in the chamber

E1000488 – V2

Vincent Lhuillier, Fabrice Matichard, Richard Mittleman

Distribution of this document:
Advanced LIGO Project

This is an internal working note
of the LIGO Laboratory

California Institute of Technology
LIGO Project – MS 18-34
1200 E. California Blvd.
Pasadena, CA 91125
Phone (626) 395-2129
Fax (626) 304-9834
E-mail: info@ligo.caltech.edu

Massachusetts Institute of Technology
LIGO Project – NW22-295
185 Albany St
Cambridge, MA 02139
Phone (617) 253-4824
Fax (617) 253-7014
E-mail: info@ligo.mit.edu

LIGO Hanford Observatory
P.O. Box 1970
Mail Stop S9-02
Richland WA 99352
Phone 509-372-8106
Fax 509-372-8137

LIGO Livingston Observatory
P.O. Box 940
Livingston, LA 70754
Phone 225-686-3100
Fax 225-686-7189

Table of contents

1 Introduction

The testing and the commissioning of the seismic isolation platforms are decomposed into three phases:

- Phase I refers to the testing realized after assembly (before storage)
- Phase II refers to the testing realized after storage and before insertion in the chamber
- Phase III refers to the testing and the commissioning in the chamber

This document presents the tools that are used to implement the control on BSC-ISI during phase III. This introduction section gives a brief overview of the BSC-ISI control system. Section 2 deals with the calibration and the compensation filters that are introduced respectively in the input (sensors) and the ouput (actuators) filters banks. Section 3 defines the channels that will be acquired during the commissioning. Section 4 presents the tools used to identify the system. Section 5 shows in detail the steps of commissioning. LASTI is used an example to illustrate these steps. This documents shows results obtained while testing the commissioning programs tested in November 2010. Since, these programs have been retrofitted. Soon (February-March 2011), all the features will be tested and validated at LASTI and this document will be updated.

1.1 BSC – ISI Sensors and Actuators

The feedback control is implemented stage by stage independently.
Stage 1 is equipped with:

- 6 Capacitive Position Sensors (CPS)
- 3 three-axis seismometers (STS - T240) for low frequency
- 6 single axis seismometers (L4C) for high frequency
- 3 Horizontal and 3 Vertical electromagnetic actuators (Coarse actuators)

Stage 2 is instrumented with:

- 6 Capacitive Position Sensors (CPS)
- 6 single axis seismometers (GS13)
- 3 Horizontal and 3 Vertical electromagnetic actuators (Fine Actuators)

1.2 Control scheme

The following figure shows a block diagram of the BSC-ISI feedback control system. It presents:

- The different block filters (Sensors symmetrization filters, Blend filters, Damping filters, Isolation filters, Actuators compensation and symmetrizartion filters)
- The basis changes (Local to Cartesian and Cartesian to Local)
- The interactions between the DOF of both stages created by the control loops ($P_{act\ S1=>S1}$, $P_{act\ S2=>S2}$, $P_{act\ S1=>S2}$, $P_{act\ S2=>S1}$).

Although the motions of the stages are coupled to each other, the stages can be controlled independently. The control is done in the Cartesian basis using independent SISO loops. There are two types of controls filters:

- The damping filters (used to damp the first rigid body modes)
- The isolation filters (stronger isolation)

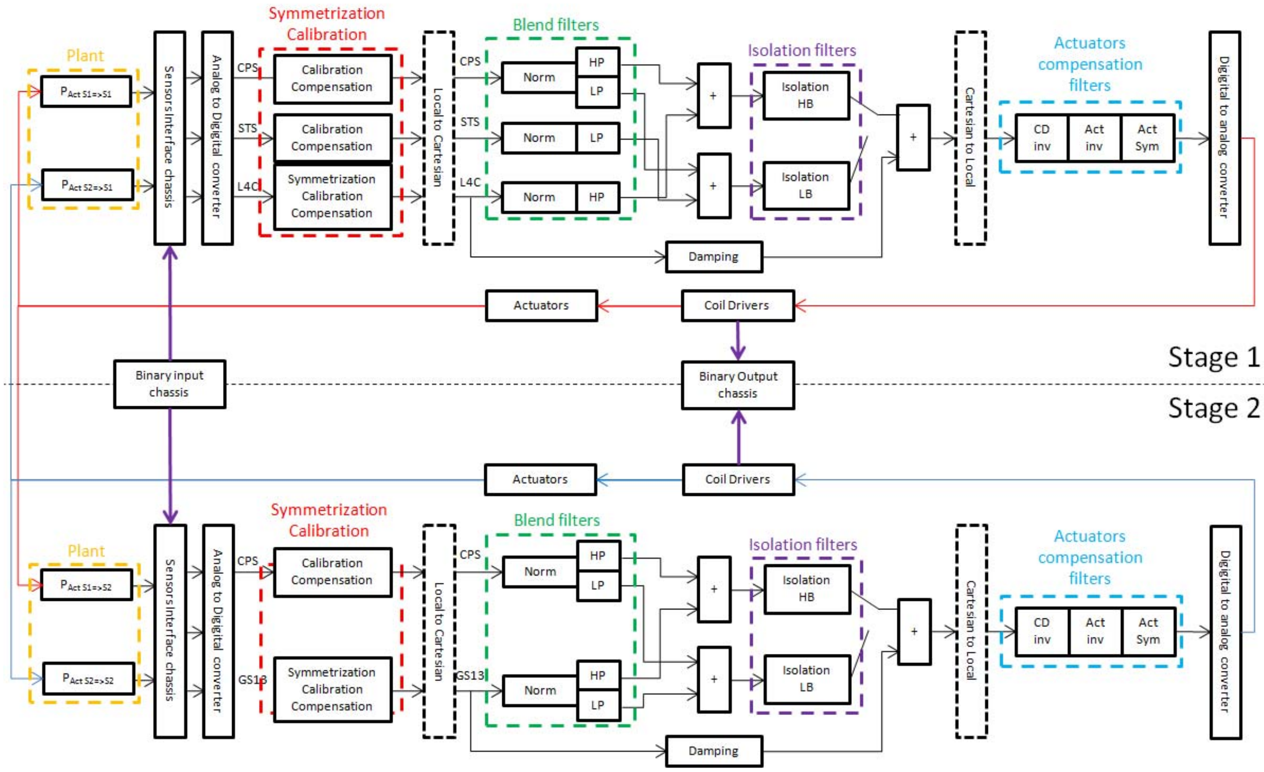

Figure 1 - Block diagram of the BSC-ISI control system

Brief description (from left to right):

1- The sensors signals pass through the interface chassis before the conversion from analog to digital. Interface chassis filters are defined in BSC-ISI_Electronics (**DCC??**)
2- The sensors signals pass through the "Sensors Symmetrization calibration filters bank". These filters calibrate the sensors, compensate the electronic of the interface chassis and symmetrize the transfer function (using a geophone model). They also correct signs.
3- Signals in the local coordinates are converted into the Cartesian Coordinates
4- Seismometer signals pass through damping filters. These filters damp the first rigid body modes resonances
5- The Blend filters bank is used to mix the relative motion sensor signals (CPS) with the inertial sensor signals (Geophones). In a first step, Geophones are normalized the CPS. Next, the two types of sensors signals pass through a low pass filter (CPS) or a high pass filter (Geophones). Finally, CPS and the geophones signals are added to obtain "super sensor signals". Note that the high pass and the pass filters are complementary. In this case, changing the blend frequency doesn't change the isolation filter.
6- The super sensor signals pass through the isolation Filters. Outputs are summated with the damping filters outputs.
7- Control drive signals in the Cartesian coordinates are converted into the Local Coordinates
8- Local Control drive signals pass through the Actuators compensation Filters bank. These filters account the actuators and the coil driver dynamics. Symmetrization filters are also added to correct gains and signs
9- The interactions between the DOF of both stages created by the control loops ($P_{act\ S1=>S1}$, $P_{act\ S2=>S2}$, $P_{act\ S1=>S2}$, $P_{act\ S2=>S1}$) are shown in yellow

1.3 Simulink Model

The “Simulink” model supported by the Real-time code generator is located in the svn at:
svn/seismic/BSC-ISI/LASTI/SimLink/M1BSCisi1.mdl

1.4 MEDM screens

Once the model has been compiled by the RCG, auto-generated and custum MEDM screens are located in the SVN at:
svn/seismic/BSC-ISI/LASTI/MedmScreens/

The figure below presents the Overview.adl

2 Input and output filters

This section presents the compensation filters that are used in the pre-filters bank and the output filters bank. These filters are the same for all BSC-ISI and will always be engaged during testing and commissioning.

2.1 Actuators and Coil drivers compensation

Filters are used to compensate the actuator roll off such that the force generated by the actuator is non-frequency dependent. Actuators have been previously identified in T0900226. Compensation filters are described below:

Stage 1:
z1=32; z2=300; p1=136; p2=900;
Compensation_Filters_Actuator_Stage_1=zpk(-2*pi*[z1 z2],-2*pi*[p1 p2],(p1*p2)/(z1*z2));

Stage 2:
z1=50; z2=290; p1=140; p2=900;
Compensation_Filters_Actuator_Stage_2=zpk(-2*pi*[z1 z2],-2*pi*[p1 p2],(p1*p2)/(z1*z2));

2.2 Coil Driver Compensation

Coil drivers are also compensated to reduce DAQ noise and introduce a non frequency dependent force. The identification of the coil driver has been realized in T0900226. The compensation filter is described below:

Coil_driver_chassis=1/zpk(-2*pi*15.9,-2*pi*0.4,-0.4/15.9);

Compensation Filters are located in the SVN at:
svncommon/seisvn/seismic/BSC-ISI/Common/Compensation_Filters_BSC_ISI

2.3 Sensors Calibration

Geophones signals and the electronic of the interface chassis are compensated such that the outputs signal of the pre filters bank give 1 count per nm (CPS) or 1 count/(nm/s) for the geophones. Thanks to the calibration, the sensors in the Cartesian basis and the super sensors are calibrated. Details concerning the super sensor calibration are given in section 5 step 8.

LASTI compensation filters for the L4C and GS13 interface chassis are defined by:

L4C Model

```
L4C_model=zpk([0 0 0],-2*pi*[pair(1,74)],1);
L4C_constant=276; % 276 V/m/s
L4C_Preamp=22; % Gain
Interface_chassis_L4C_gain=1; % Gain
AA_Interface_L4C=2.1; % Gain
```

STS

```
w_0=2*pi*0.008; Q=1/sqrt(2); STS_num=[1 0 0 0]; STS_den=[1 w_0/Q w_0^2];
STS_model=tf(STS_num,STS_den);
STS_constant=1500; % 1500V/m/s
STS_Preamp=42; % Gain
Interface_chassis_STS_gain=1; % Gain
AA_Interface_STS=2.1; % Gain
```

GS13

```
w_0=2*pi; Q=4.5; GS13_num=[1 0 0 0]; GS13_den=[1 w_0/Q w_0^2];
GS13_model=tf(GS13_num,GS13_den);
GS13_constant=2280; % 2280V/m/s
GS13_Preamp=40.2; % Gain
Interface_chassis_GS13_gain=0.5; % Gain
AA_Interface_GS13=2.1; % Gain
```

CPS

Stage 1

```
CPS_model=10000; % 10V/mm
CPS_Preamp=2; % ADE box
```

Stage 2

```
CPS_model=40000; % 40V/mm
CPS_Preamp=2; % ADE box
```

ADC

```
ADC=2^15/20; % counts/V

Gain_goal=1e9; %1count/nm or 1count/nm/s
```

Calibration gain
CPS_calibration=Gain_goal/(CPS_model*CPS_Preamp*ADC);
STS_calibration=Gain_goal/(STS_constant*STS_Preamp*Interface_chassis_STS_gain*AA_Interface_STS*ADC);
L4C_calibration=Gain_goal/(L4C_constant*L4C_Preamp*Interface_chassis_L4C_gain*AA_Interface_L4C*ADC);
GS13_calibration=Gain_goal/(GS13_constant*GS13_Preamp*Interface_chassis_GS13_gain*AA_Interface_GS13*ADC);

2.4 Interface chassis compensation

The calibration wouldn't be accurate if the electronic of the compensation chassis were not compensated. The interface chassis compensation filters are defined below.

Interface chassis compensation filters
[z,k]=conjug(500,0); [p,g]=conjug(900,45);
STS_interface_compensation=zpk([],[],1);
L4C_interface_compensation=zpk([z],[p],g/k);
GS13_interface_compensation=zpk([z],[p],g/k);

For simplicity, the interface chassis compensation and the sensor calibration are grouped into one unique filter. This filter is common to every unit. The symmetrization filters that will later introduced are specific to each unit. Consequently, they will be introduced in a second filter in the pre-filter bank.

Results presented in this report use the early scripts that don't account the sensors calibration and the electronic compensation of the interface chassis.

3 Channels and sample rate

The Front-end is running at 4096Hz. At LASTI, the channels are recorded with different acquisition rate to increase the number of saved channels. The following sample rates are defined in the table below:

Channels	Sample rate (Hz)
ST1 - CPS	1024
ST1 - STS	1024
ST1 - L4C	4096
ST2 - CPS	1024
ST2 - GS13	4096
ST1 – ACT	4096
ST2 - ACT	4096

Table 1- Channels sample rate

For aLIGO, all channels will be recorded at 4096Hz. Phase compensator is not added due to the low blend frequency. Acquired channels and sample rate are defined in the .ini file located at (LASTI):
/cvs/cds/mit/chans/daq/M1ISI.ini

4 System Identification

Transfer functions are measured via Matlab and awgstream. Measurements are launched with **Run_exc_batch** and processed with **Run_get_batch**. **Run_exc_batch** calls Run_L2L_XX or Run_C2C_XX functions to start excitations. It creates the excitation file and the batch_file. The batch file is an "exchange file" that is read by **Run_get_batch**. It carries general information such as data locations, data names, response channels, etc… The excitation file contains more specific data such as the excitation signals, GPS start time, frequency resolution, etc… The transfer functions are computed by **Run_get_batch** and results are saved in structure file (1 per section).

In **Run_exc_batch** and **Run_get_batch** , locations of the data and the batch file (current + archive) have to be specified. Site, Unit, State (Undamped, Damped, Isolated …) must be edited. Date is automatically added.

Note 1: The batch_file is erased at every new set of measurements. If the batch file has been overwritten, Run_get_batch_retrieve can be used to compute transfer functions. In that case, the batch_file_archive name must be edited in Run_get_batch_retrieve.

Note 2: The identification scripts calls the Schroeder phase scripts that are common to every systems (BSC-ISI, HAM-ISI, HEPI-BSC, HEPI-HAM). These scripts are located at the top of the SVN arborescence (svncommon/seisvn/seismic/Common/MatlabTools/Schroeder_Phase_Scripts)

Scripts used at LASTI to measure transfer functions are located in the SVN at:
/svncommon/seisvn/seismic/BSC-ISI/LASTI/Scripts/Data_Collection

- Run_exc_batch
- Run_get_batch
- Run_get_batch_retrieve
- Run_L2L_10mHz_100mHz.m
- Run_L2L_100mHz_1Hz.m
- Run_L2L_1Hz_10Hz.m
- Run_L2L_10Hz_100Hz.m
- Run_L2L_100Hz_1000Hz.m
- BSC_ISI_L2L_exc_chan_list.m
- BSC_ISI_L2L_resp_chan_list.m

Batch_file location is specific for each Unit. For instance, **the LASTI/Prototype/Phase_3/ batch file is located in the SVN at:** svncommon/seisvn/seismic/BSC-ISI/LASTI/Prototype/Phase_3/Data/ Transfer_Functions/Measurements
and the archived batch file at Phase_3/Data/Transfer_Functions/Measurements/Batch_file-Archive

4.1 System Identification

Two sets of measurements are used to identify the system:

- Quick measurements (14h) that are launched during a night. These quick measurements enable to design the symmetrization filters (sensors - actuators), the damping loops, the blend filters (CPS – L4C – GS13) and the isolation loops (High-blend control). Due to T240 saturations and a low coherence at low frequency, these measurements don't allow to design the low blend control.
- Long measurements that are launched during the week end (with or without the damping loop) to measure the T240 transfer function without saturation

Parameters used for a quick identification (High Blend control) are reported below:

	Section	Freq min	Freq max	Fres	Amplitude (H-V)	Nrep	Time Stage 1 (min)
Stage 1	1	0.01	0.1	0.01	2000 - 2000	3	32
	2	0.1	1	0.02	140 - 200	25	127
	3	1	10	0.05	80 - 110	75	154
	4	10	100	0.1	120 - 120	50	53
	5	100	1000	0.1	33 - 50	50	53
							419

	Section	Freq min	Freq max	Fres	Amplitude (H-V)	Nrep	Time Stage 2 (min)
Stage 2	1	0.01	0.1	0.01	2000 - 2000	3	32
	2	0.1	1	0.02	500 - 550	25	127
	3	1	10	0.05	320 - 300	75	154
	4	10	100	0.1	180 -180	50	53
	5	100	1000	0.5	50 - 50	50	53
							419

Overall ID	**838**

Table 2 - System identification parameters – quick measurements

Parameters used for a full identification (low Blend control) are reported below:

	Section	Freq min	Freq max	Fres	Amplitude (H-V)	Nrep	Time Stage 1 (min)
Stage 1	1	0.01	0.1	0.01			
	2	0.1	1	0.02			
	3	1	10	0.05			
	4	10	100	0.1			
	5	100	1000	0.1			

	Section	Freq min	Freq max	Fres	Amplitude (H-V)	Nrep	Time Stage 2 (min)
Stage 2	1	0.01	0.1	0.01			
	2	0.1	1	0.02			
	3	1	10	0.05			
	4	10	100	0.1			
	5	100	1000	0.5			
							0

Overall ID	**0**

Table 3 - System identification parameters – long measurements

4.2 Filters to engage before measuring the transfer functions

The transfer functions are measured with:

- The actuators compensation filters ON (Output filters bank)
- The filters X10, Inv 10, Whiten, DeWhiten ON (Input Filters bank)
- The calibration filters (Input Filters bank)

4.3 Measuring the transfer functions

The transfer functions are measured from the excitation channels of the output filters banks to the input in1 of the pre-filters bank. Excitation and response channels are defined in BSC_ISI_L2L_exc_chan_list.m and BSC_ISI_L2L_resp_chan_list.m.

Work to do

- Open Matlab and run Run_exc_batch
- Open a second Matlab and run Run_get_batch

4.4 Plotting the data once processed

The transfer functions computed by Run_get_batch are stored in .mat files (one .mat file by section). Transfer functions will be later concatenated into a unique structure file (see Control_Scripts/ Step_1_TF_L2L_ST1_ST2_LASTI_Prototype_BSC_ISI).

4.5 Comparison with the reference transfer functions

Transfer functions can be read and compared to reference transfer function (December 8, 2010) using the scripts located in the SVN at:
/svncommon/seisvn/seismic/BSC-ISI/LASTI/Prototype/Phase_3/Test_Scripts/
TF_L2L_ST1_ST2_Comparison_LASTI_Prototype_BSC_ISI

5 Design of the control loops

5.1 Presentation

The control scripts have been written to reduce the number of tasks the commissioner will have to carry out. The control design is divided into 16 steps. The design scripts are specific for each unit. These scripts calls several functions and filters that are common to all BSC-ISIs such as the plot functions, the complementary filters, the basis change matrices, etc…

For instance, LASTI prototype control scripts are located in the svn at:
/svncommon/seisvn/seismic/BSC-ISI/LASTI/Prototype/Phase_3/Control_Scripts

- Step_0_Run_all_LASTI_Prototype_BSC_ISI
- Step_1_TF_L2L_ST1_ST2_LASTI_Prototype_BSC_ISI
- Step_2_Symmetrization_CalibrationLASTI_Prototype_BSC_ISI
- Step_3_TF_Base_Change_L2C_LASTI_Prototype_BSC_ISI
- Step_4_C2D_Sym_Filters_LASTI_Prototype_BSC_ISI
- Step_5_Damping_Loops_ST2_LASTI_Prototype_BSC_ISI
- Step_6_Damping_Loops_ST1_LASTI_Prototype_BSC_ISI
- Step_7_C2D_Damping_Filters_LASTI_Prototype_BSC_ISI
- Step_8_Normalization_LASTI_Prototype_BSC_ISI
- Step_9_C2D_Blend_Filters_LASTI_Prototype_BSC_ISI
- Step_10_Isolation_Loops_ST2_Z_RX_RY_LASTI_Prototype_BSC_ISI
- Step_11_Isolation_Loops_ST1_Z_RX_RY_LASTI_Prototype_BSC_ISI
- Step_12_Isolation_Loops_ST2_X_Y_RZ_LASTI_Prototype_BSC_ISI
- Step_13_Isolation_Loops_ST1_X_Y_RZ_LASTI_Prototype_BSC_ISI
- Step_14_Open_Loop_check_LASTI_Prototype_BSC_ISI
- Step_15_C2D_Isolation_Filters_LASTI_Prototype_BSC_ISI
- Step_16_Tilt_decoupling_LASTI_Prototype_BSC_ISI
- Matlab_Path_Definition_LASTI_Prototype_BSC_ISI

Scripts common to Seismic are located in the svn at:
/svncommon/seisvn/seismic/Common/

Scripts that will be common to BSC's are located in the svn at:
/svncommon/seisvn/seismic/BSC-ISI/Common/

It contains subfolders relative to the control design:

- Base_Change_BSC_ISI
- Complementary_Filters_BSC_ISI
- Compensation_Filters_BSC_ISI
- Control_Scripts_Functions_BSC_ISI

Base_Change_BSC_ISI contains that basis change matrices that will be used by Step_3_TF_Base_Change_L2C_LASTI_Prototype_BSC_ISI
Complementary_Filters_BSC_ISI contains a bank of complementary filters (corner/blend frequency from 100mHz to 1Hz with a 50mHz increment) that will be used to build super sensors.
Compensation_Filters_BSC_ISI contains filters that compensate the actuators dynamic and the coil drivers response. They also contains the calibration filters of the pre-filter bank.

Control_Scripts_Functions_BSC_ISI contains all the functions that create plots (Transfer functions, Open Loops, closed loops, etc…)

5.2 Scripts step by step

This section presents how the control design scripts are working. Each program is briefly presented and indications are provided to adapt the code to a new set of data.
At each step, three different types of results can be saved:

- Parameters file (.mat file)
- Data (.mat file)
- Figures (.fig and .pdf)

These results file are automatically saved at a specific location (different from the working folder). Consequently, the folders arborescence mustn't be modified. The folder that contains the Control scripts is defined as the reference folder.

5.2.1 Matlab_Path_Definition_LASTI_Prototype_BSC_ISI

This script is used to define the Matlab path. Due to the numerous systems, it is preferable to redefine the matlab path without saving it every time we work on a specific unit. It reduces the risk of destructive interactions when working on different units.
For instance, at LASTI Matlab_Path_Definition_LASTI_Prototype_BSC_ISI is run at the beginning of every design scripts and the path is never saved.

5.2.2 Step_0_Run_all_LASTI_Prototype_BSC_ISI

Step_0_Run_all_ LASTI_Prototype_BSC_ISI is the first program to run. It opens every program that will be called during the control design. It also checks the path definition.

5.2.3 Step_1_TF_L2L_ST1_ST2_LASTI_Prototype_BSC_ISI

This step must be executed from the control scripts directory (Reference point in the SVN arborescence to save figures and data). In details, this script:

- Loads the data (5 sections) located at /svncommon/seisvn/seismic/BSC-ISI/LASTI/Prototype/Phase_3/Data/Transfer_Functions/Measurements/Undamped
- Concatenates the transfer functions and coherence functions (5 sections into 1). Next, these data are stored into 3D matrices. With the resolution mentioned earlier, 3D matrix dimensions are [33 (sensors), 12 (actuators), 10135 (frequency)]. Finally, 3D matrices are transformed in frd structure to ease data "readability".
- *At LASTI, different sampling rates are used to increase the number of acquired channels. Transfer function phases are corrected such that the phase lag is equivalent to a 4096Hz sample frequency (Front-end phase). Not done (Blend frequency is low and not usefull for aLIGO)*
- Plots transfer functions and coherence functions of the main couplings in the local base
- **Saves the results in the svn at**
    - Figures at /svncommon/seisvn/seismic/BSC-ISI/LASTI/Prototype/Phase_3 /Data/Figures/Transfer_Functions/Measurements/Undamped. A date stamp is automatically added on figures files

- Saves transfer functions, coherences functions (3D matrices then frd) and a frequency vector at /svncommon/seisvn/seismic/BSC-ISI/LASTI/Prototype/Phase_3/Data /Transfer_Functions/Simulations/Undamped **TF_L2L_Sym_LASTI_BSC-ISI-Prototype_10mHz_1000Hz_2010_12_08**
- Save Parameters file at /svncommon/seisvn/seismic/BSC-ISI/LASTI/Prototype/ Phase_3/Data/Transfer_Functions/Simulations/ **Parameters_LASTI_Prototype_BSC_ISI**

The following figure shows the way the transfer functions and the coherence functions are stored.

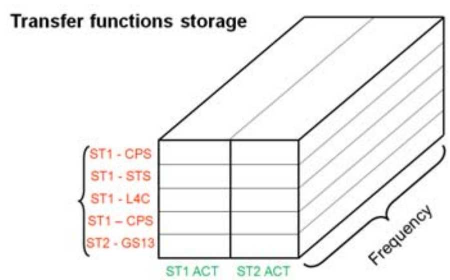

Figure 2 - Transfer functions and coherence functions storage [33, 12, 10135]

Step_1_TF_L2L_ST1_ST2_LASTI_Prototype_BSC_ISI calls:

- **Plot_TF_L2L_BSC_ISI**: This function plots Local to Local transfer functions and automatically save figure in the folder /Data/Figures/Transfer_Functions/Simulations/Undamped

The figure below shows the transfer functions (main couplings) from local actuators of stage 1 to the capacitive position sensors of stage 1.

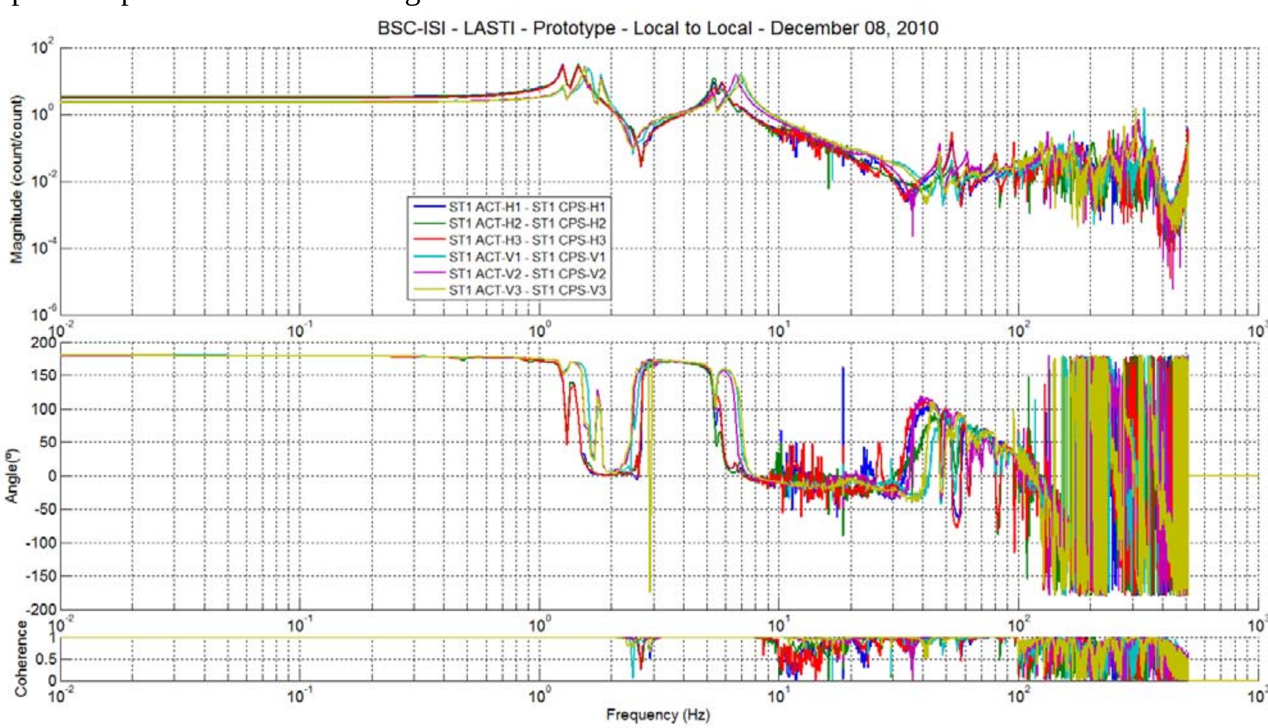

Figure 3 - Local to Local transfer functions - Main couplings - ST1 CPS

The figure below shows the transfer functions (main couplings) from local actuators of stage 1 to the L4C.

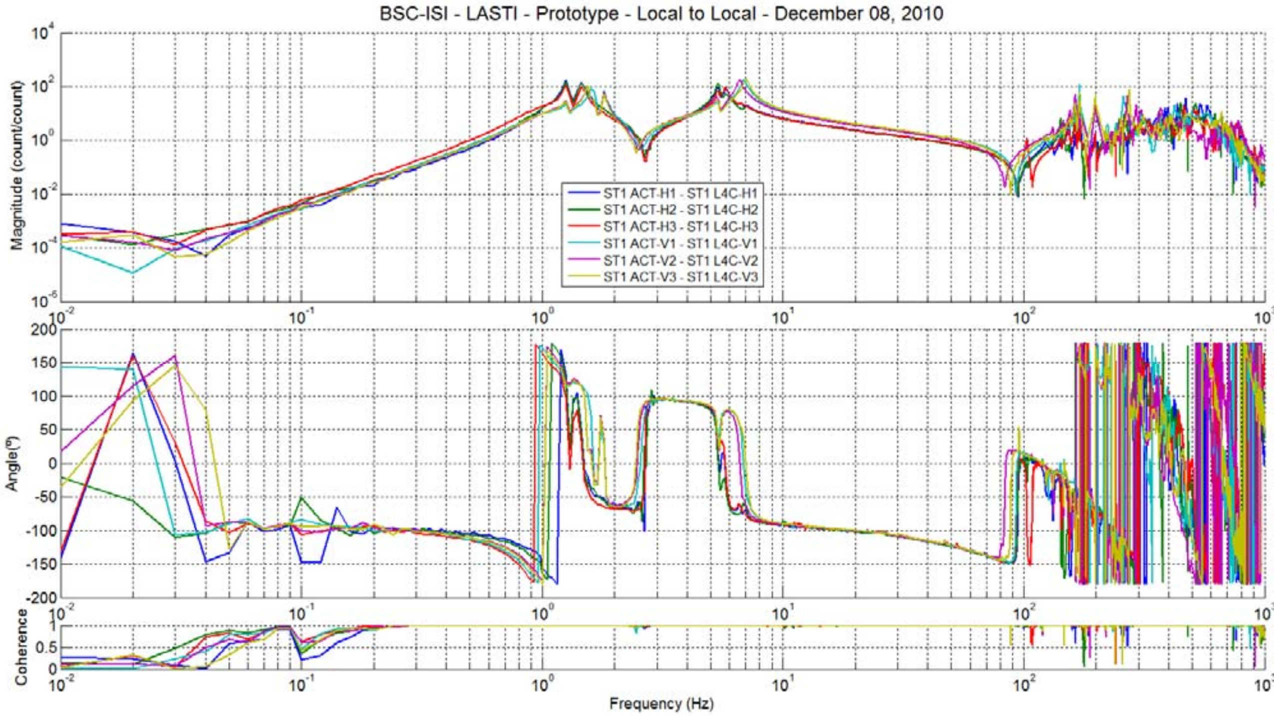

Figure 4 - Local to Local transfer functions - Main couplings - ST1 L4C

The figure below shows the transfer functions (main couplings) from local actuators of stage 2 to the capacitive position sensors of stage 2.

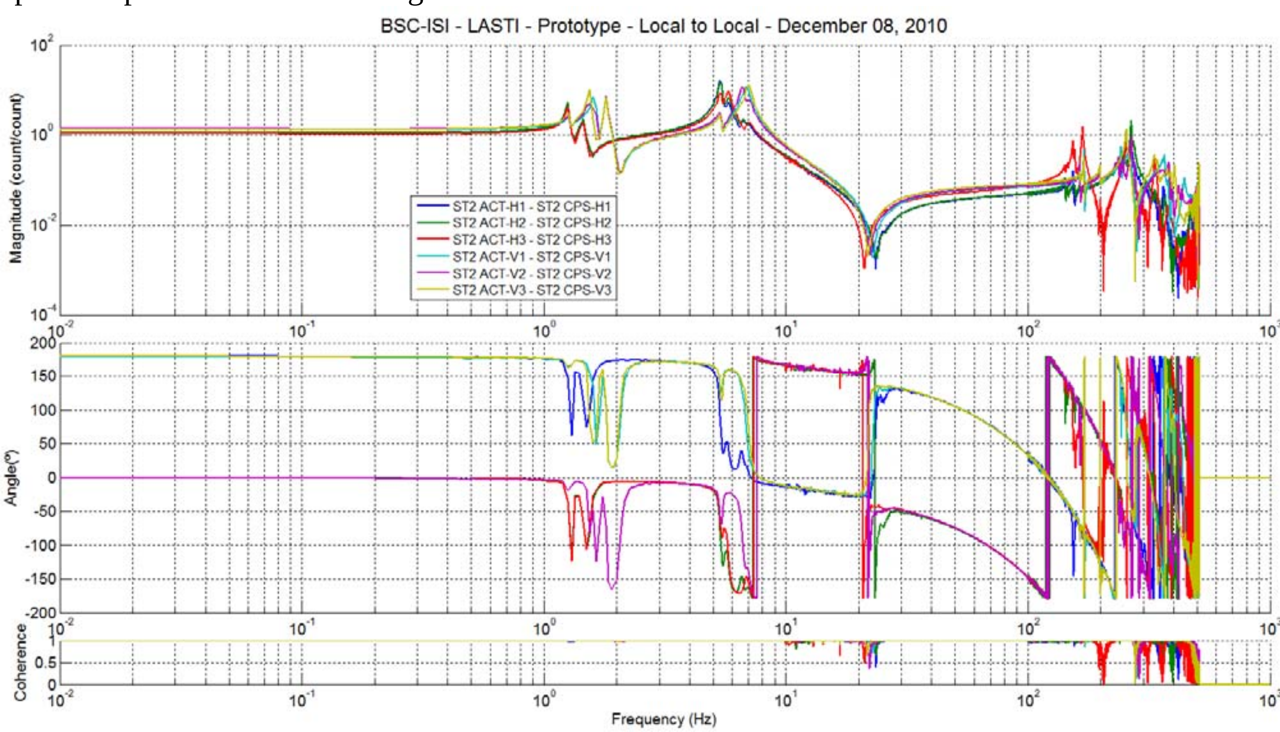

Figure 5 - Local to Local transfer functions - Main couplings - ST2 CPS

The figure below shows the transfer functions (main couplings) from local actuators of stage 2 to the GS13.

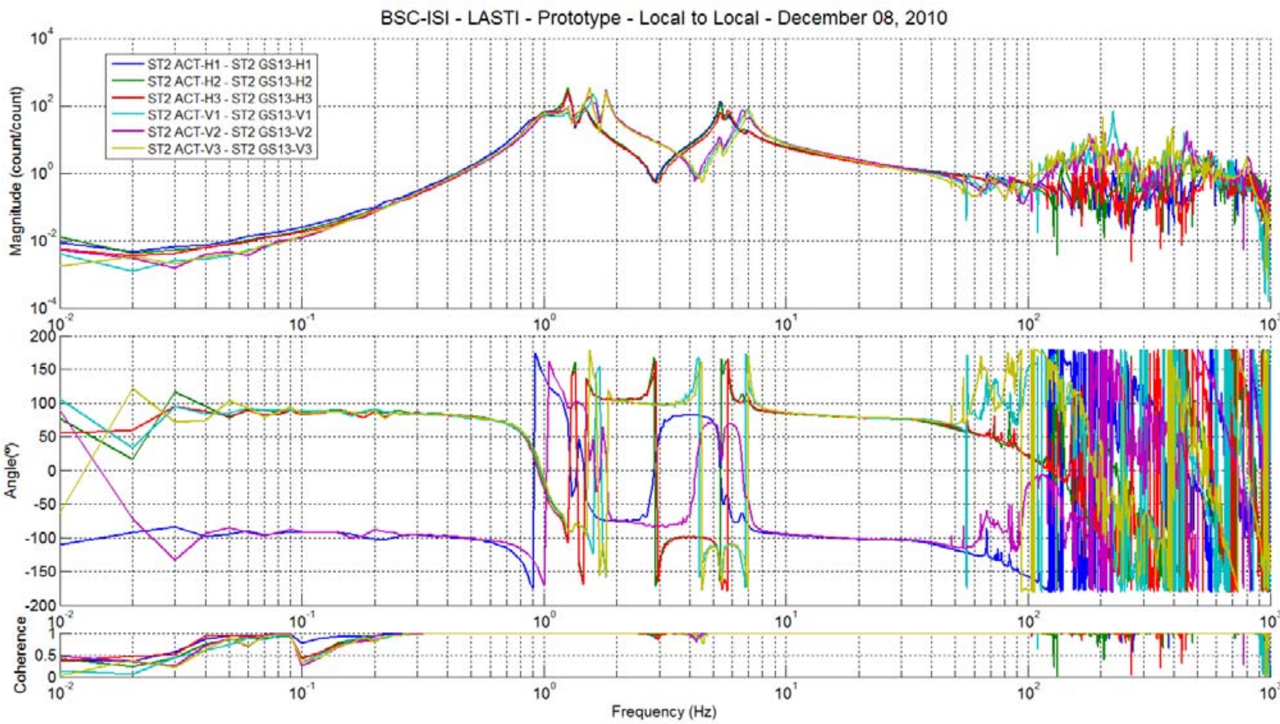

Figure 6 - Local to Local transfer functions - Main couplings - ST2 GS13

Due to saturation, STS transfer function are not presented.

Work to do:

- **Edit save_parameters file**
- **Edit the data files**
- **Edit title_L2L_str**
- **Edit save_data_TF_L2L**
- **Edit date_str**

Every data name must contain information relative to the site (X1, X2, LHO, LLO), Unit (BSC X) and date (YYYY_MM_DD)

Step_2_Symmetrization_Calibration_LASTI_Prototype_BSC_ISI

During this step, symmetrization filters (input and output) are designed. Then, the symmetrized and calibrated transfer functions (from Output EXC to Input In2) are calculated. In details, this script:

- Reads the undamped transfer functions (created by Step 1)
- Account for the sensors calibration and the geophone interface chassis compensation. All transfer functions plotted are presented in nm/count (exc channel) or in (nm/s)/count (exc channel).
- Computes the Actuators corrections

  1 - **Actuators symmetrization gains**

  CPS are used as reference. Symmetrization gains are calculated after averaging the CPS transfer functions in the [100mHz; 500mHz] frequency band.

  2- **Signs corrections** (the phase must be 0º at DC)

  3- **Stored transfer functions** in a 3D matrix (frd) after the actuators gains correction
- Computes Geophones symmetrization (due to a bad coherence at low frequencies, the symetrization gains are computed in the [0.5; 0.7]Hz frequency band.

  1- L4C

  1.1- **Signs corrections:** At low frequency (0.2Hz is the reference frequency), L4C transfer functions have a third order slope. Consequently, the phase must be -90º. If geophone phase is positive, Geophones are out-of-phase. Then, -1 gain is added.

  1.2- **Fitting Filters:** The L4C has a low Q and its internal resonance is in the vicinity of the BSC-ISI first suspension mode. Consequently, identifying the L4C dynamic is tricky. But, the BSC-ISI motions are also sensed by CPS. At low frequency, CPS of stage 1 and L4C observe approximately the same motion (one is relative, one is absolute). So, the L4C transfer functions (the instrument itself) can be calculated by removing the BSC-ISI response to the measured L4C transfer functions using the CPS transfer functions. The Cartesian to local transfer functions are preferred to local to local because the CPS and the L4C are not collocated (on aLIGO BSC-ISI). Next, the L4C transfer function (the instrument itself) can be fitted.

  At low frequency, we can estimate the transfer function of the instrument ($L4C_{Sensor}$) using:

$$L4C_{Sensor} = \frac{TFC2L\ L4C_{Measured}}{TFC2L\ CPS_{Measured}}$$

  1.3- **Symmetrization filters gains:** *The synmmetrization filters gains are computed such that the symmetrized transfer functions (H1-H2-H3 & V1-V2-V3) are superposed to the "averaged initial transfer function" of H1-H2-H3 & V1-V2-V3*

The figure below shows how the L4c transfer function is identified (instrument itself). It shows:

- In black, the measured transfer function
- In blue, the instrument transfer function obtained after dividing the geophone cartesian to local transfer function by the CPS Cartesian to local transfer function
- In green, the model with a random gain (to be close to the identified transfer function
- In red, the fitting filter

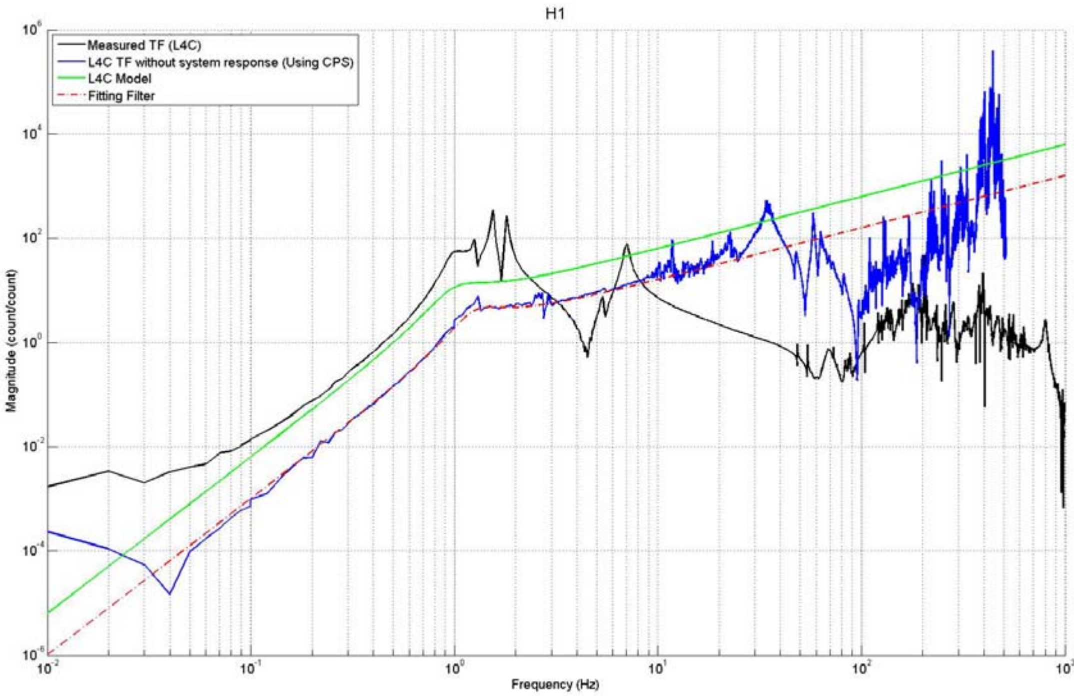

Figure 7 - L4C Identification

2- **GS13**

2.1- **Phase corrections**

2.2-**Fitting Filters** GS13 has a higher Q than L4C. Consequently, fitting the GS13 transfer functions can be done withour removing the system response. Moreover, we won't be able to do it when the BSC-ISI will be in the chamber

2.3- **Symmetrization filters gains**

- **Saves the results in the svn at**
  - Figures at /svncommon/seisvn/seismic/BSC-ISI/LASTI/Prototype/Phase_3/Data/Figures/Transfer_Functions/Simulations/Undamped
  - Data at /svncommon/seisvn/seismic/BSC-ISI/LASTI/Prototype/Phase_3/Data/Transfer_Functions/Simulations/Undamped
  - Symmetrization Filters at /svncommon/seisvn/seismic/BSC-ISI/LASTI/Prototype/Phase_3/Continuous_Filters

The figure below shows the transfer functions (main couplings) from local actuators of stage 1 to the capacitive position sensors of stage 1 after symmetrization and calibration.

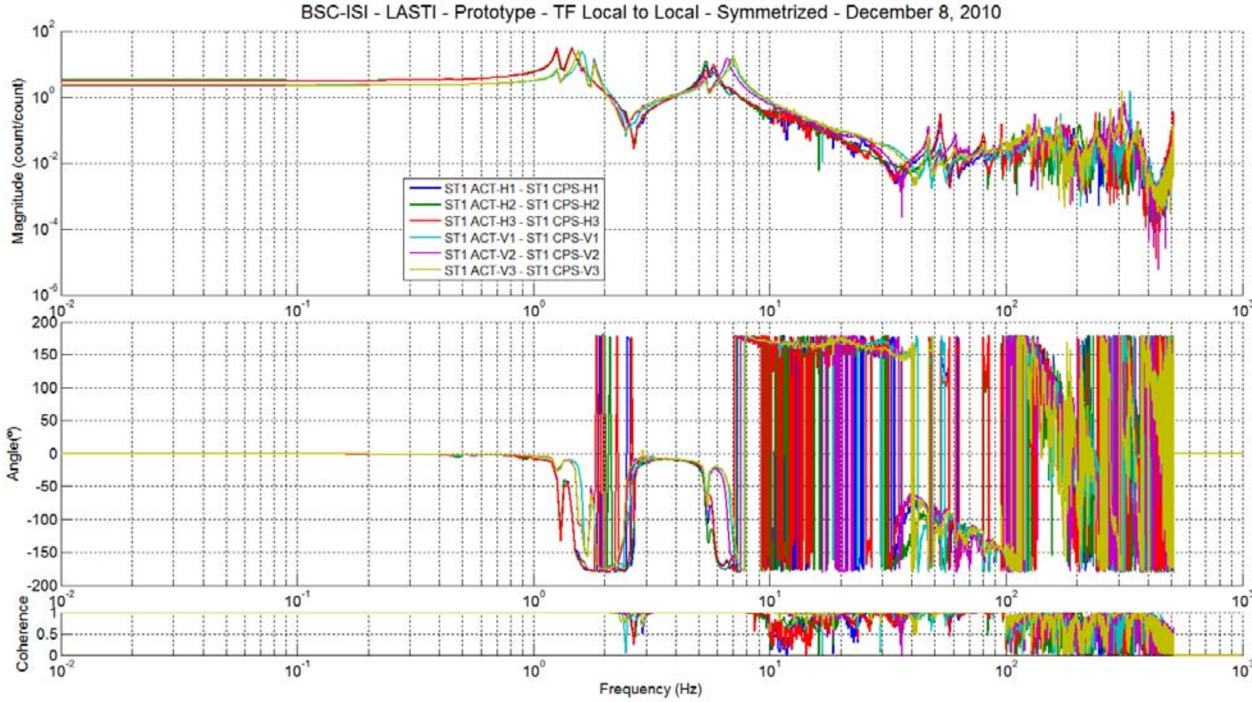

Figure 8 - Local to Local transfer functions - Main couplings - ST1 CPS
Actuators & sensors symmetrization + Sensors calibration (not on this figure)

The figure below shows the transfer functions (main couplings) from local actuators of stage 1 to the L4C after symmetrization and calibration.

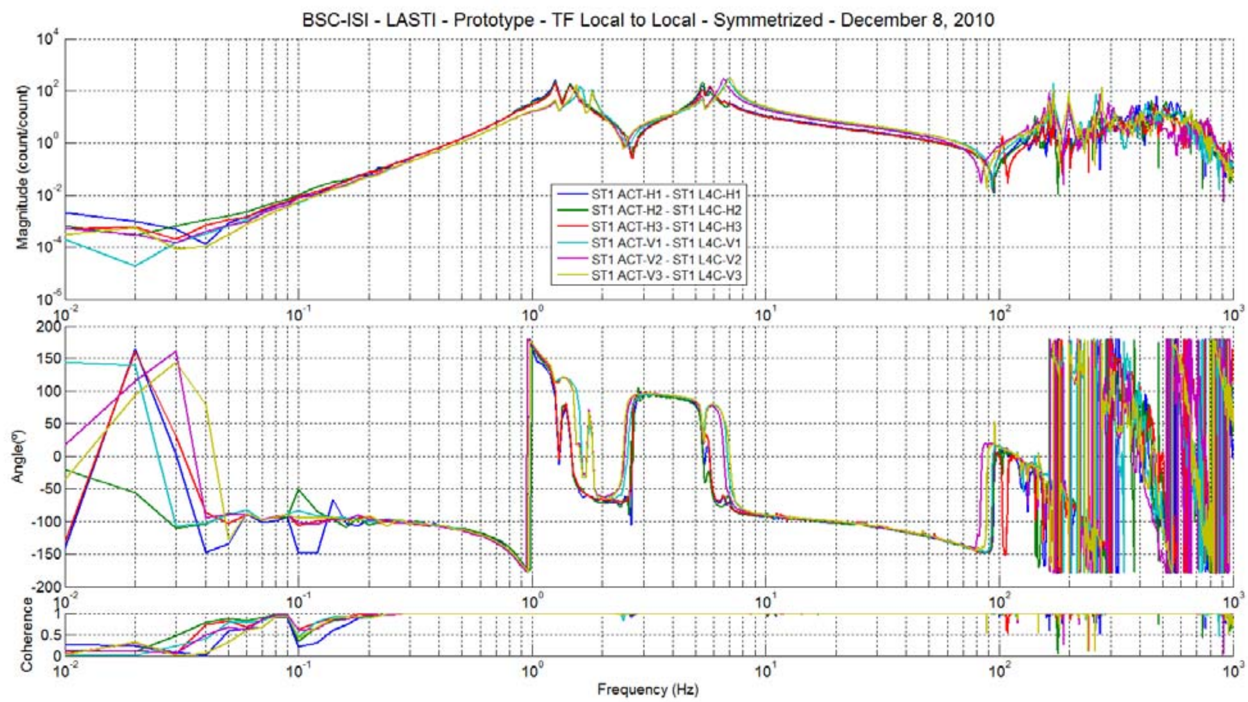

Figure 9 - Local to Local transfer functions - Main couplings - ST1 L4C
Actuators & sensors symmetrization + Sensors calibration (not on this figure)

The figure below shows the transfer functions (main couplings) from local actuators of stage 2to the capacitive position sensors of stage 2 after symmetrization and calibration.

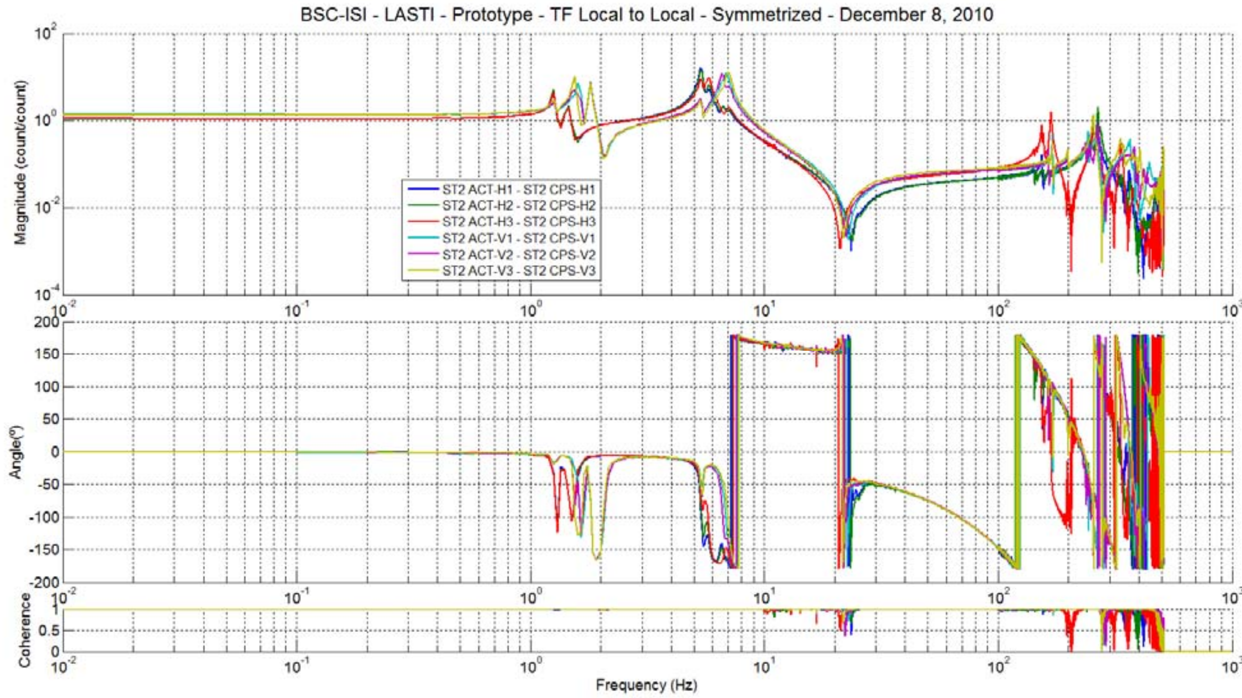

Figure 10 - Local to Local transfer functions - Main couplings - ST2 CPS
Actuators & sensors symmetrization + Sensors calibration (not on this figure)

The figure below shows the transfer functions (main couplings) from local actuators of stage 1 to the GS13 after symmetrization and calibration.

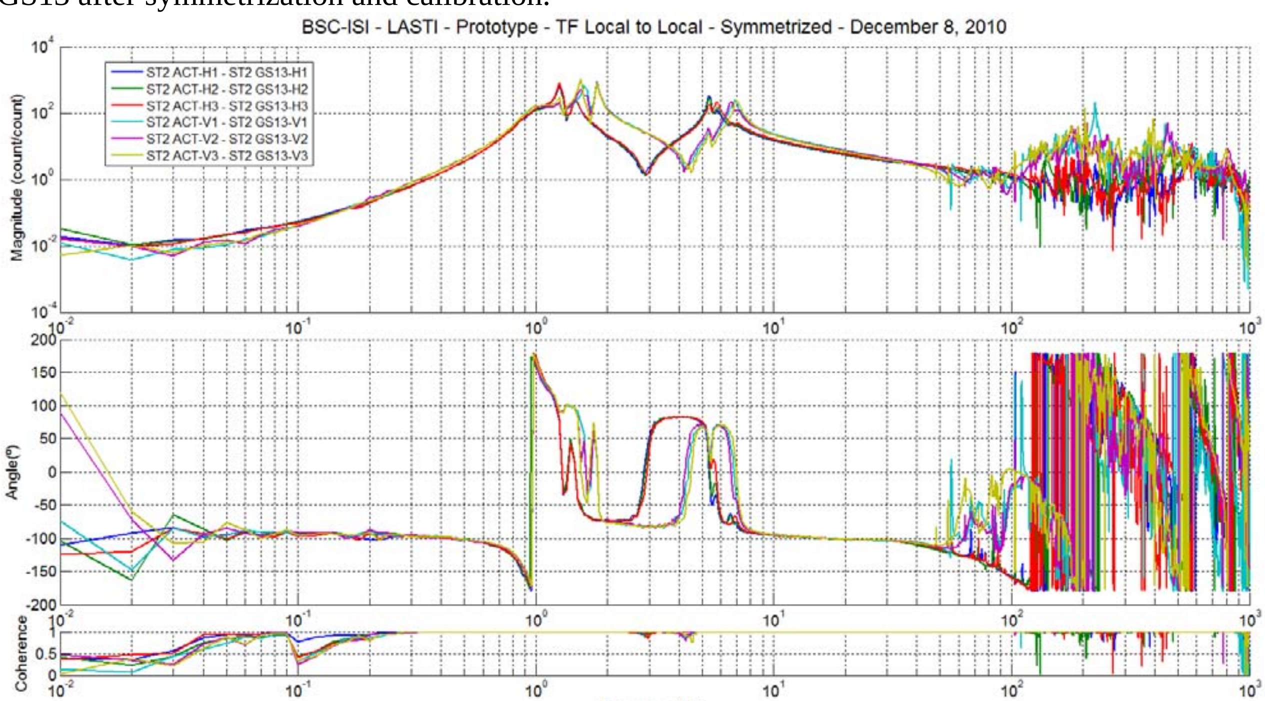

Figure 11 - Local to Local transfer functions - Main couplings - ST2 GS13
Actuators & sensors symmetrization + Sensors calibration (not on this figure)

Work to do:

- **Edit initial data**
- **Edit save_parameters file**
- **Edit save_data_TF_L2L_Symmetrized**
- **Edit save_Symmetrisation_Filters**
- **Edit title_L2L_Symmetrized_str**
- **Edit Fitting Filters**

Step_2_Symmetrization_LASTI_Prototype_BSC_ISI calls:

- **Plot_TF_L2L_BSC_ISI**: This function plots Local to Local transfer functions and automatically save figure in the specified folder
- **Plot_L4C_Fitt_BSC_ISI**: This function is used to plot L4C transfer functions (example: Act CS H1 to L4C H1) without the system response and the L4C fitting filters.
- **Plot_TF_L4C_GS13_Symmetrization_BSC_ISI**: This function is used to plot L4C and GS13 transfer functions (example: Act FN H1 to GS13 H1) and the L4C and GS13 fitting filters.
- **Plot_L4C_Fitt_all_BSC_ISI**: This functions is used to plot all L4C transfer functions (example: Act CS X to L4C X) without the system response and the L4C fitting filters.
- **Plot_TF_L4C_GS13_Symmetrization_all_BSC_ISI**: This function is used to plot L4C and GS13 transfer functions before and after symmetrization

Step_3_TF_Base_Change_L2C_LASTI_Prototype_BSC_ISI

Step 3 calculated the transfer functions in the Cartesian basis. In details, this script:

- Loads symmetrized transfer functions and basis change matrices
- Computes the transfer functions in the Cartesian basis. The base change matrices don't change the unit of the sensors readout. The Cartesian displacements are in nm/count and the Cartesian velocity is in (nm/s)/count.
- **Saves the results in the svn at**
  - o Symmetrized transfer functions in the Cartesian basis at /svncommon/seisvn/seismic/BSC-ISI/LASTI/Prototype/Phase_3/Data/ Transfer_Functions/Simulations/Undamped
  - o Figures at /svncommon/seisvn/seismic/BSC-ISI/LASTI/Prototype/Phase_3/ Data/ Figures/Transfer_Functions/Simulations/Undamped

The Cartesian to Cartesian transfer functions of the main couplings are presented in the figures below. They include the symmetrization (actuators + sensors) and the sensors calibration realized during step 2.

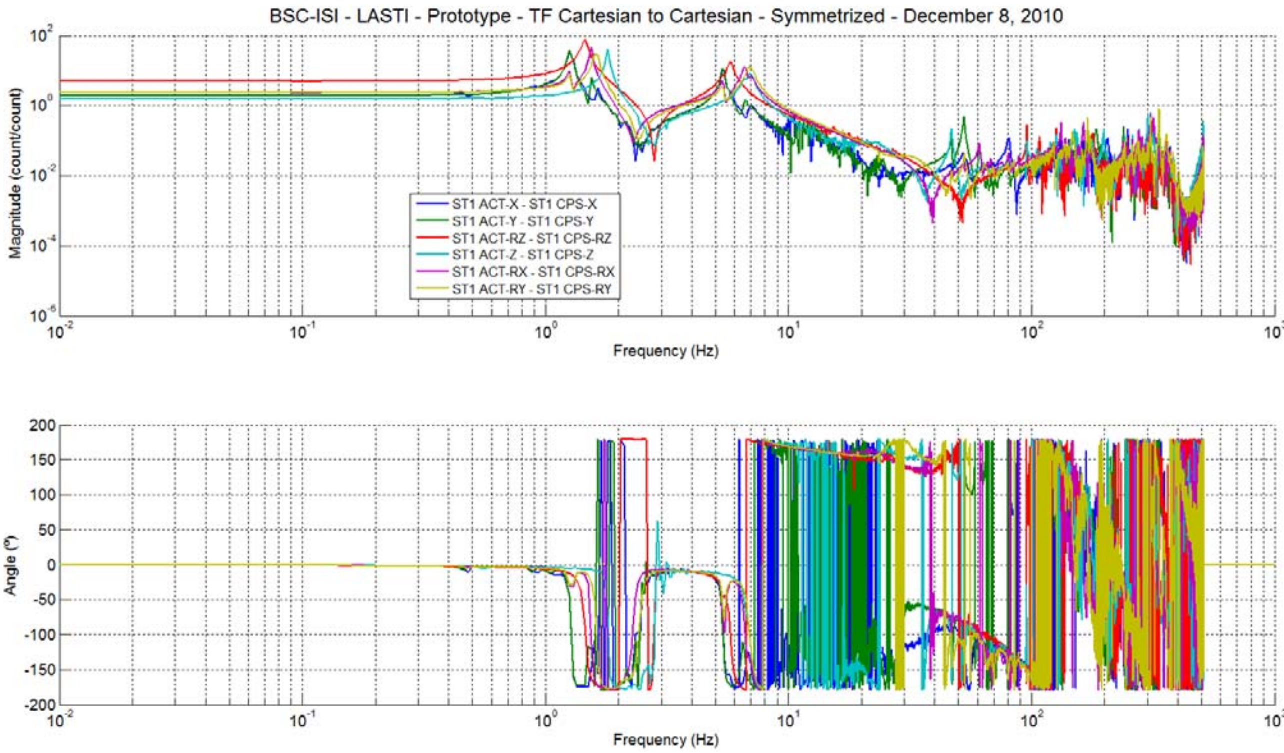

Figure 12 - Cartesian to Cartesian transfer functions - Main couplings - ST1 CPS

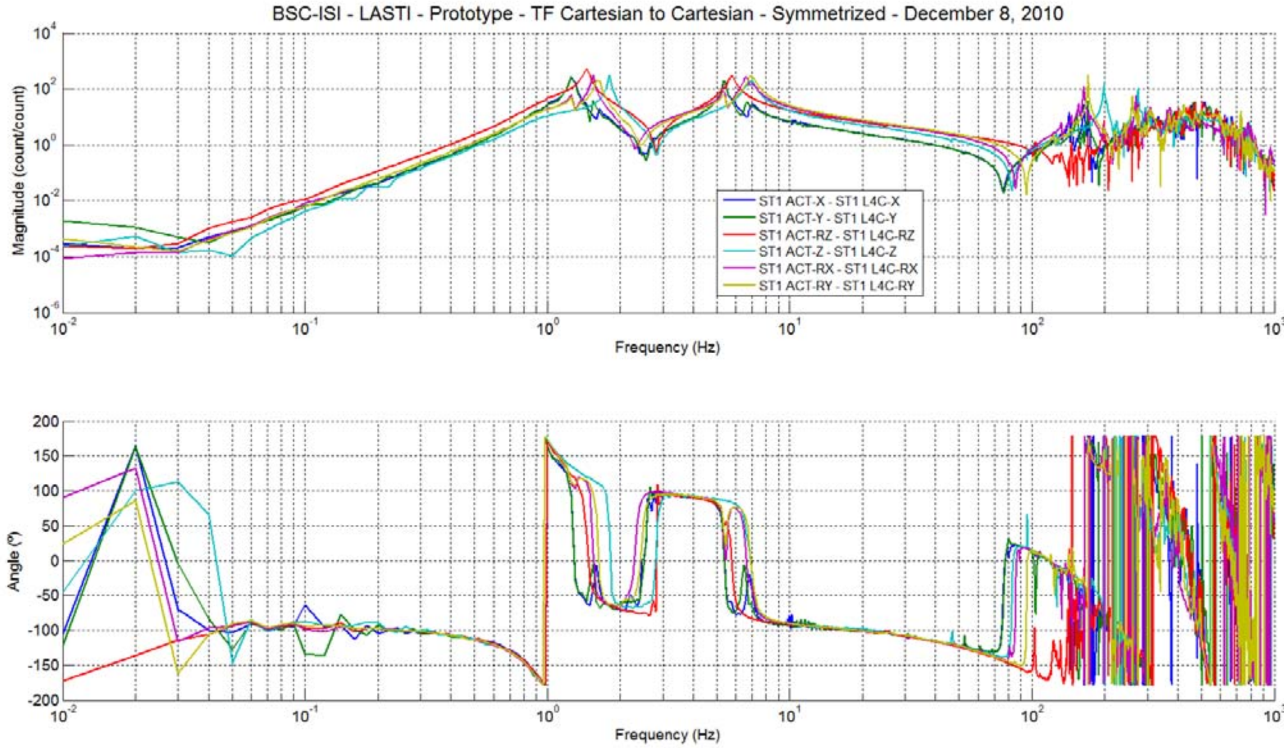

Figure 13 - Cartesian to Cartesian transfer functions - Main couplings - ST1 L4C

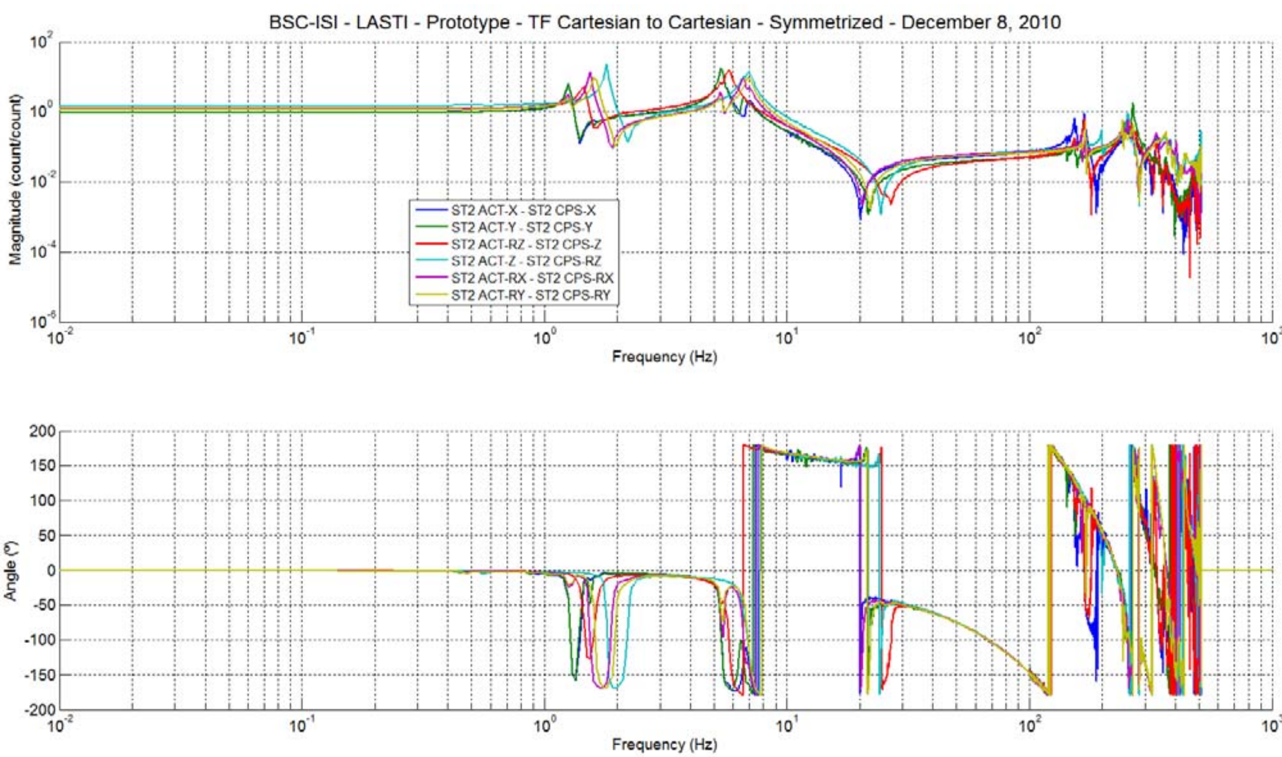

Figure 14 - Cartesian to Cartesian transfer functions - Main couplings - ST2 CPS

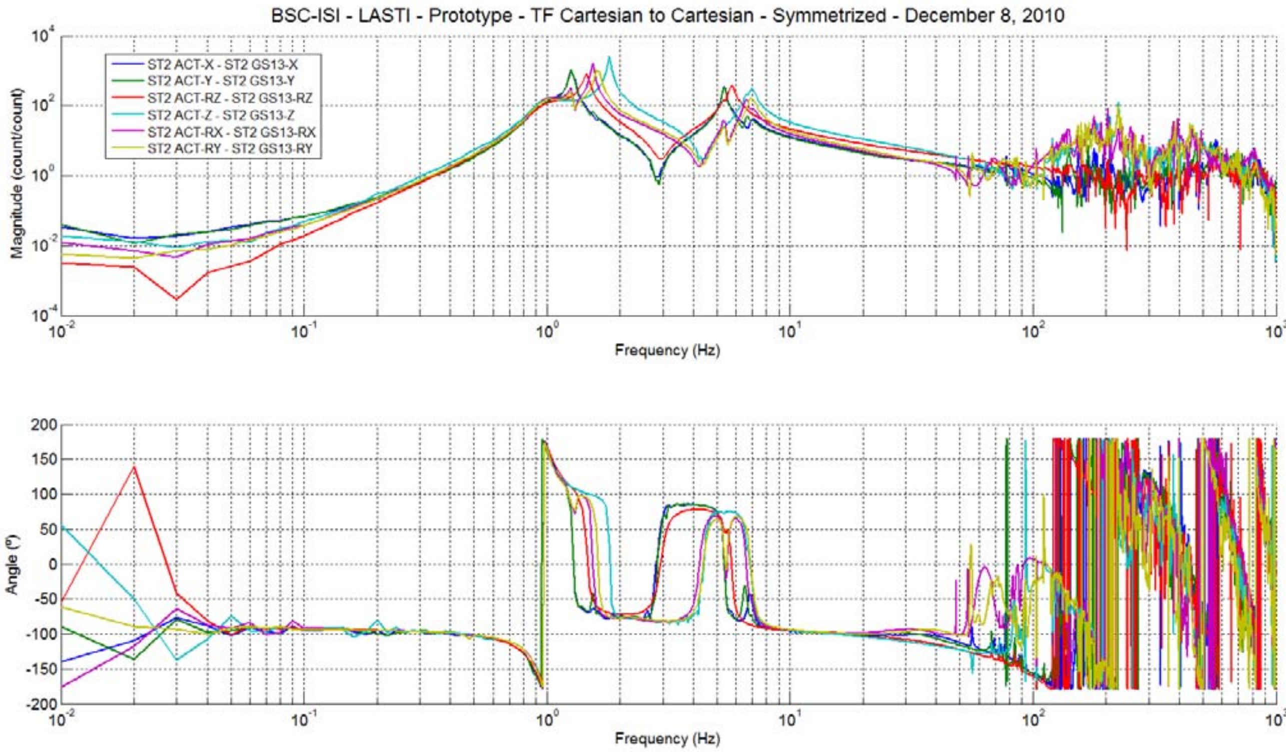

Figure 15 - Cartesian to Cartesian transfer function – Main couplings - ST2 GS13

Work to do:

- **Edit initial data**
- **Edit save_parameters file**
- **Edit save_data_TF_C2C**
- **Edit title_C2C_Symmetrized_str**

Step_4_C2D_Sym_Filters_LASTI_Prototype_BSC_ISI

Step 4 digitizes the symmetrization filters computed during step 2. Once digitized, measured Cartesian to Cartesian transfer functions can be compared with the simulated transfer functions. In details, this scripts:

- Loads Symmetrization Filters
- Digitizes Symmetrization filters
- **Save digitized Symmetrization filters in a txt file in the svn at:** svncommon/seisvn/seismic/BSC-ISI/LASTI/Prototype/Phase_3/Digitized_Filters

Work to do:

- **Edit initial data**
- **Edit save_parameters file**
- **Edit the continuous symmetrization filters file**
- **Edit the digitized symmetrization filters file (Site, Unit). A date stamp is automatically added**

Step_5_Damping_Loops_ST2_LASTI_Prototype_BSC_ISI

During this step, damping filters of stage 2 are designed. In details, this script:

- Loads the Cartesian to Cartesian transfer functions of the undamped structure (measured or simulated).
- Computes the gain of the damping filters. Filters are designed in a SISO configuration. The filters shape (poles and zeros) is the same for every DOF and for every unit.
- Computes the SISO and the MIMO transfer functions when the Damping Filters of stage 1 are engaged

If isolation and damping filter are engaged, the control drive transfer functions are given by:

$$U_1/Exc = [TF_{C2C}CPS_{CS} * CPS_{Blend\ CS} + TF_{C2C}GEO_{CS} * GEO_{Blend\ CS}] * Controller_{S1} + TF_{C2C}GEO_{CS} * Damping_{S1}$$

$$U_2/Exc = [TF_{C2C}CPS_{FN} * CPS_{Blend\ FN} + TF_{C2C}GEO_{FN} * GEO_{Blend\ FN}] * Controller_{S2} + TF_{C2C}GEO_{FN} * Damping_{S2}$$

In case of exclusive damping, the MIMO open loop is given by:

$$Open\ Loop_{[30,30]} = TF_{C2C\ [30,12]} * Damping_{[12,30]}$$

with

$$Damping = \begin{bmatrix} 0 & 0 & Damp\ L4C & 0 & 0 \\ 0 & 0 & 0 & 0 & Damp\ GS13 \end{bmatrix}$$

Where $Damp\ L4C$ and $Damp\ GS13$ are diagonal matrices. Diagonal terms are the damping filters.

$$DOF\ interraction_{[30,30]} = [I - Open\ Loop]^{-1}$$

$$Closed\ Loop_{[30,12]} = DOF\ interraction_{[30,30]} * TF_{C2C\ [30,12]}$$

Note: In Matlab, the MIMO closed loop transfer function is computed using:

$$Closed\ Loop_{[30,12]} = [I - Open\ Loop]\backslash TF_{C2C\ [30,12]}$$

Where \ performs the matrix left division.

The MIMO suppression is given by:

$$Suppression_{MIMO[30,12]} = Closed\ Loop_{[30,12]}./Open\ Loop_{[30,12]}$$

with ./ the Hadamard division.

- **Saves the results in the svn at**
  - SISO and MIMO transfer functions in the SVN at /svncommon/seisvn/seismic/BSC-ISI/LASTI/Prototype/Phase_3/Data/ Transfer_Functions/Simulations/Damping
  - Save figures of the SISO and the MIMO transfer functions in the SVN at /svncommon/seisvn/seismic/BSC-ISI/LASTI/Prototype/Phase_3/Data/ Figures/Transfer_Functions/Simulations/Damping

Step_5_Damping_Loops_ST2_LASTI_Prototype_BSC_ISI calls:

- Plot_Damping_Filters_ST2_SISO_BSC_ISI: This function is used when tuning the damping filters of Stage 2. It plots the plant, the Damping Filters, the SISO Open Loop, the SISO Suppression, the SISO Closed Loop for each DOF. The figure is automatically saved.
- Plot_Damping_Filters_MIMO_BSC_ISI: This function plots the plant, the Damping Filters, the MIMO Open Loop, the MIMO Suppression, the MIMO Closed Loop for each DOF. The figure is automatically saved.
- **Saves the results in the svn:**
    - o SISO and MIMO transfer functions at /svncommon/seisvn/seismic/BSC-ISI/LASTI/Prototype/Phase_3/Data/Transfer_Functions/Simulations/Damping
    - o Figures of the SISO and the MIMO transfer functions at /svncommon/seisvn/seismic/BSC-ISI/LASTI/Prototype/Phase_3/Data/Figures/Transfer_Functions/Simulations/Damping

The figure below presents how the damping filter (stage 2 direction X):

- In blue, the TF of the undamped structure
- In green, the Damping filter
- In black, the open loop
- In red, the sensitivity
- In magenta, the closed loop

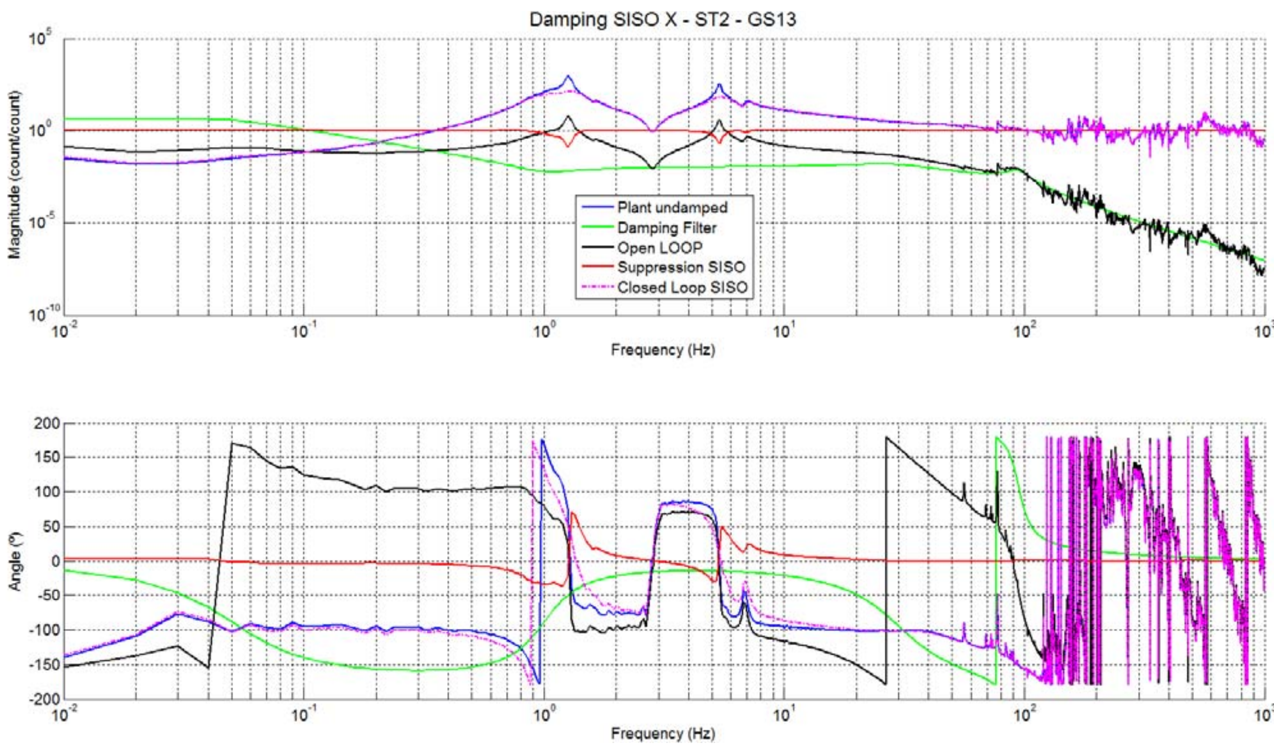

Figure 16 – SISO Cartesian to Cartesian transfer functions - ST2-Act X to ST2-GS13 X - Damping Stage 2

The figures 15 and 16 compare the simulated SISO and the simulated MIMO responses. They present:

- In blue, the TF of the undamped structure
- In green, the Damping filter
- In black, the open loop SISO – Based on the undamped structure
- In red, the suppression SISO – Based on the undamped structure
- In magenta, the closed loop SISO – Based on the undamped structure
- In dashed blue, the MIMO sensitivity
- In dashed green, the MIMO closed loop

Note: In the legend, "Stage by Stage independently" means the open loop is calculated using the undamped structure. This legend has a real meaning when tuning stage 1 damping filters; where the open loops can also be computed using the transfer function once the structure is damped by stage 2.

Figure 17 shows the weak interactions created by the damping filters between the different DOF.

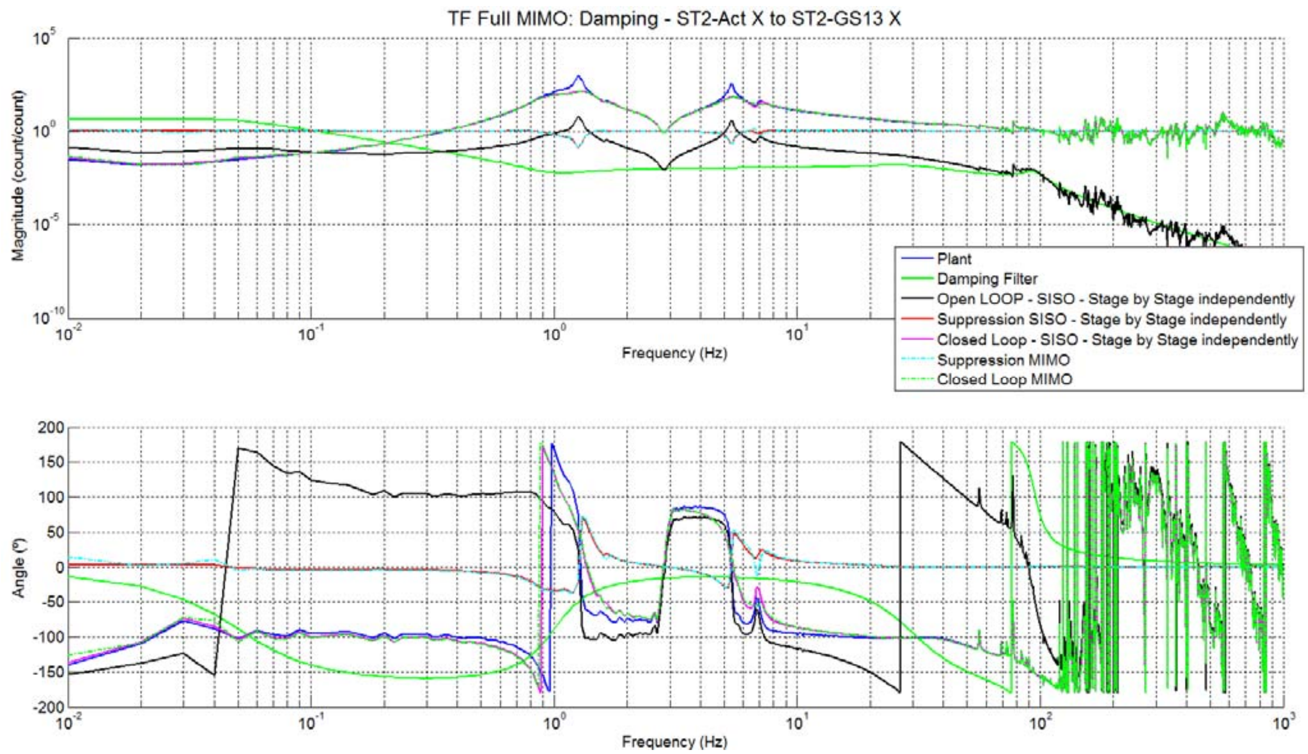

Figure 17 – MIMO Cartesian to Cartesian transfer functions - ST2-Act X to ST2-GS13 X - Damping Stage 2

Figure 18 shows the strong influence of the stage 2 on stage 1 once the stage 2 damping filters are engaged. This figure shows the transfer function from “actuator X of stage 1” to the “L4C in the X direction” when the damping filters of stage 2 are engaged.

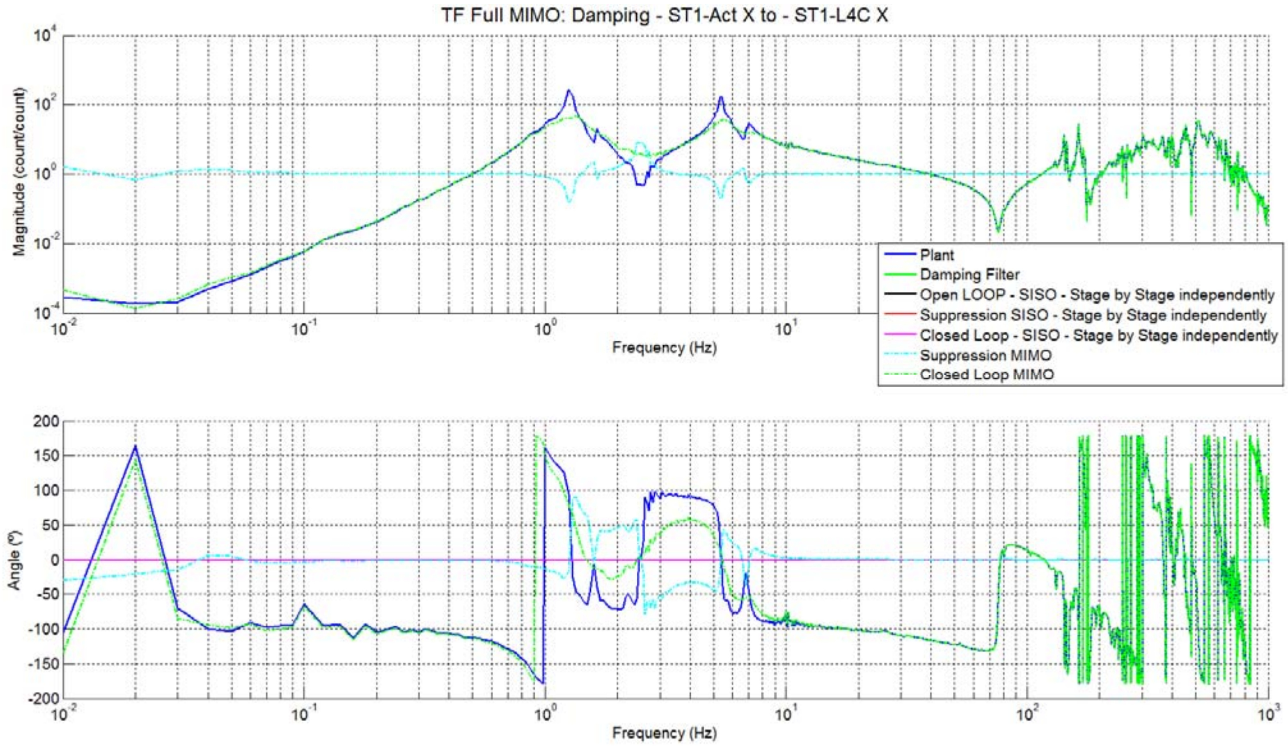

Figure 18 – MIMO Cartesian to Cartesian transfer functions – ST1-Act X to ST1-L4C X - Damping Stage 2

Work to do:

- **Edit initial data**
- **Edit save_parameters file**
- **Edit save_Damping_Filters (date)**
- **Edit Damping_Filters and UG_Damping**
- **Edit save_data_TF_C2C_Damped_ST2_MIMO and save_Damping_Filters**

Step_6_Damping_Loops_ST1_LASTI_Prototype_BSC_ISI

- Loads the Cartesian to Cartesian transfer functions of the structure when the damping filters of Stage 1 are engaged
- Computes the gain of the damping filters once the operator placed poles and zeros. Filters are designed in a SISO configuration.
- Computes the SISO and the MIMO transfer functions when the Damping Filters of stage 1 and stage 2 are engaged
- **Saves the results in the svn at**
  - o SISO and MIMO transfer functions in the SVN at /svncommon/seisvn/seismic/BSC-ISI/LASTI/Prototype/Phase_3/Data/ Transfer_Functions/Simulations/Damping
  - o Figures of the SISO and the MIMO transfer functions in the SVN at /svncommon/seisvn/seismic/BSC-ISI/LASTI/Prototype/Phase_3/Data/ Figures/Transfer_Functions/Simulations/Damping

The damping filters of Stage 1 are tuned using the MIMO response of the system when stage 2 damping filters are engaged.

The following figure presents:

- In blue, the TF of the undamped structure
- In dashed blue, the plant when damping filters of stage 2 are engaged and damping filters of stage 1 are disengaged
- In green, the Damping filter
- In black, the open loop computed using the MIMO response of the system when the damping filters of stage 2 are engaged
- In red, the sensitivity SISO relative to the MIMO response of the system when the damping filters of stage 2 are engaged
- In magenta, the closed loop SISO using the MIMO response of the system when the damping filters of stage 2 are engaged

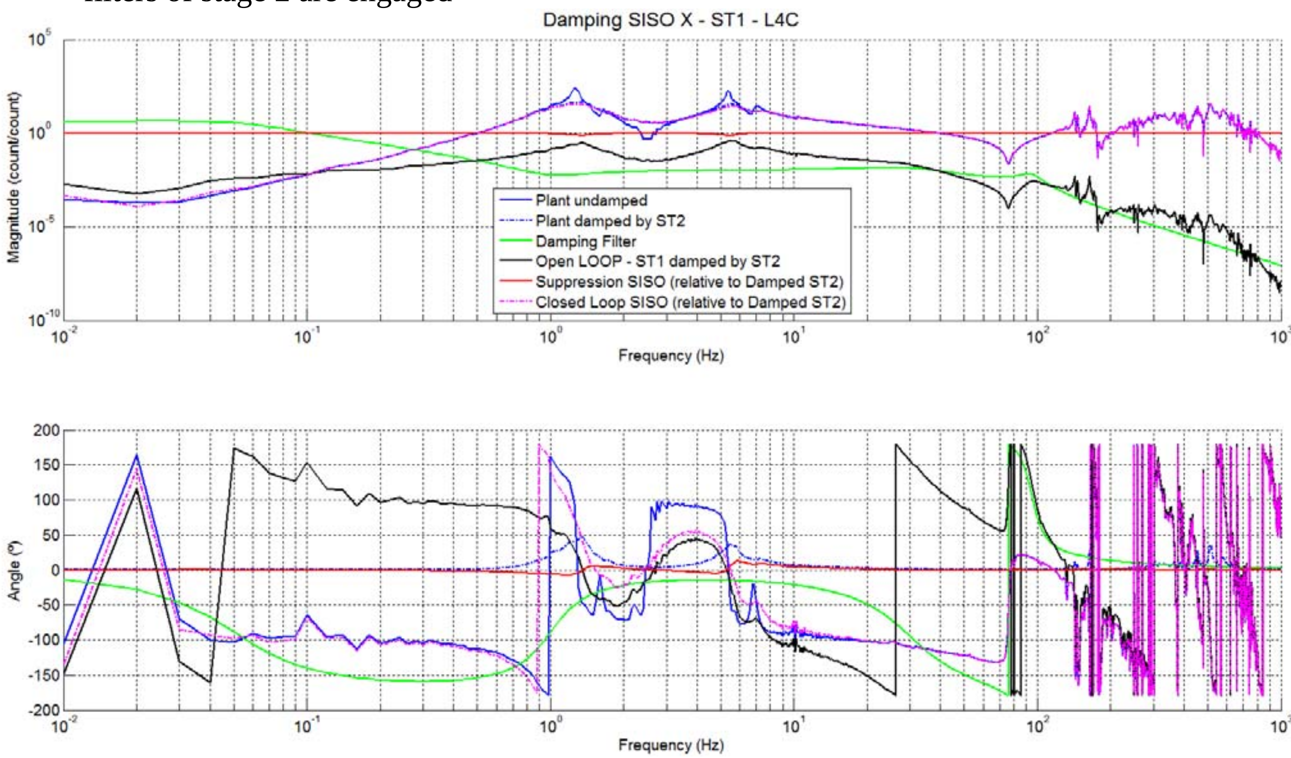

Figure 19 – SISO Cartesian to Cartesian transfer functions - ST1-Act X to ST1-L4C X - Damping Stages 1 & 2

Note the quasi-null effect of the damping filters of stage 1. All the damping is provided by the stage 2 damping loops. Open loops of stage 1 damping filters are below unity. Do we need to implement damping filters on Stage 1?

Figure 20 and 21 present the MIMO response of the structure when the damping filters are engaged on stages 1 and 2. They also show the low influence of the stage 1 damping filters.

- In blue, the TF of the undamped structure
- In green, the Damping filter
- In black, the open loop SISO – Based on the undamped structure
- In red, the sensitivity SISO – Based on the undamped structure
- In magenta, the closed loop SISO – Based on the undamped structure
- In dashed blue, the MIMO sensitivity
- In dashed green, the MIMO closed loop

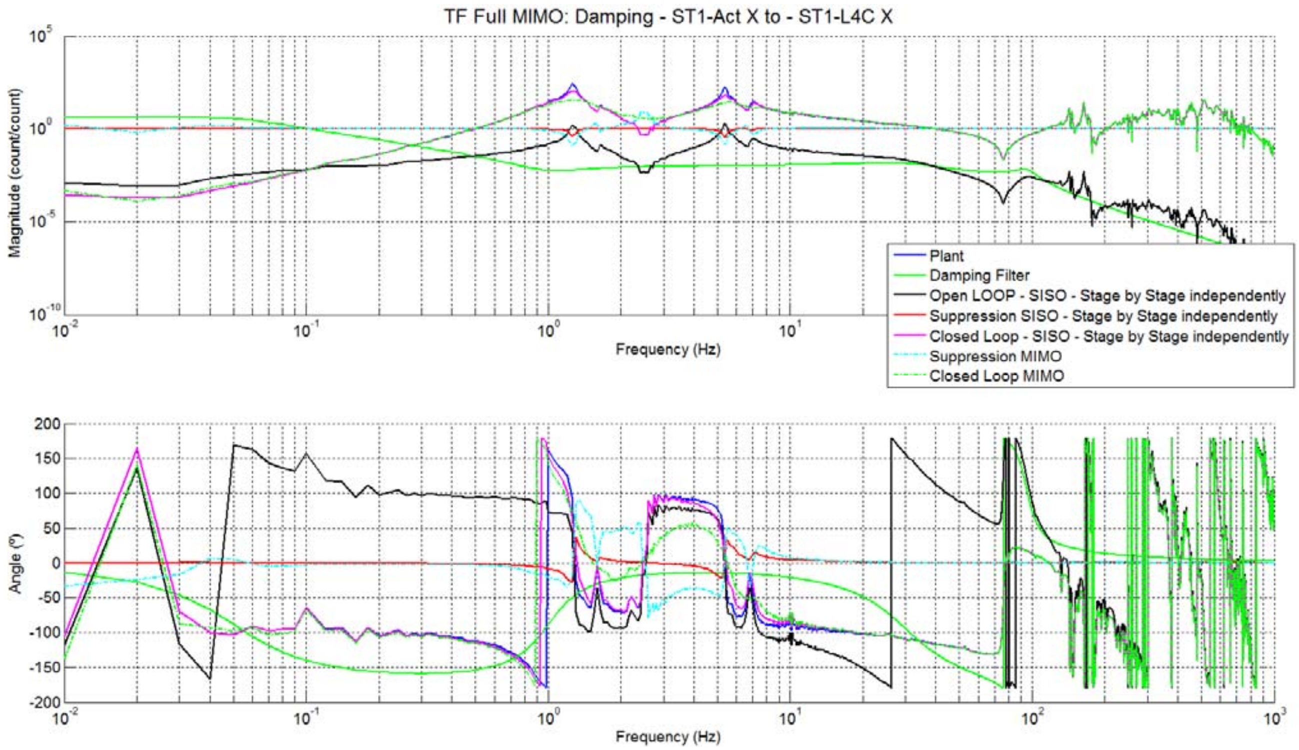

Figure 20 – MIMO Cartesian to Cartesian transfer functions - ST1-Act X to ST1-L4C X - Damping Stages 1 & 2

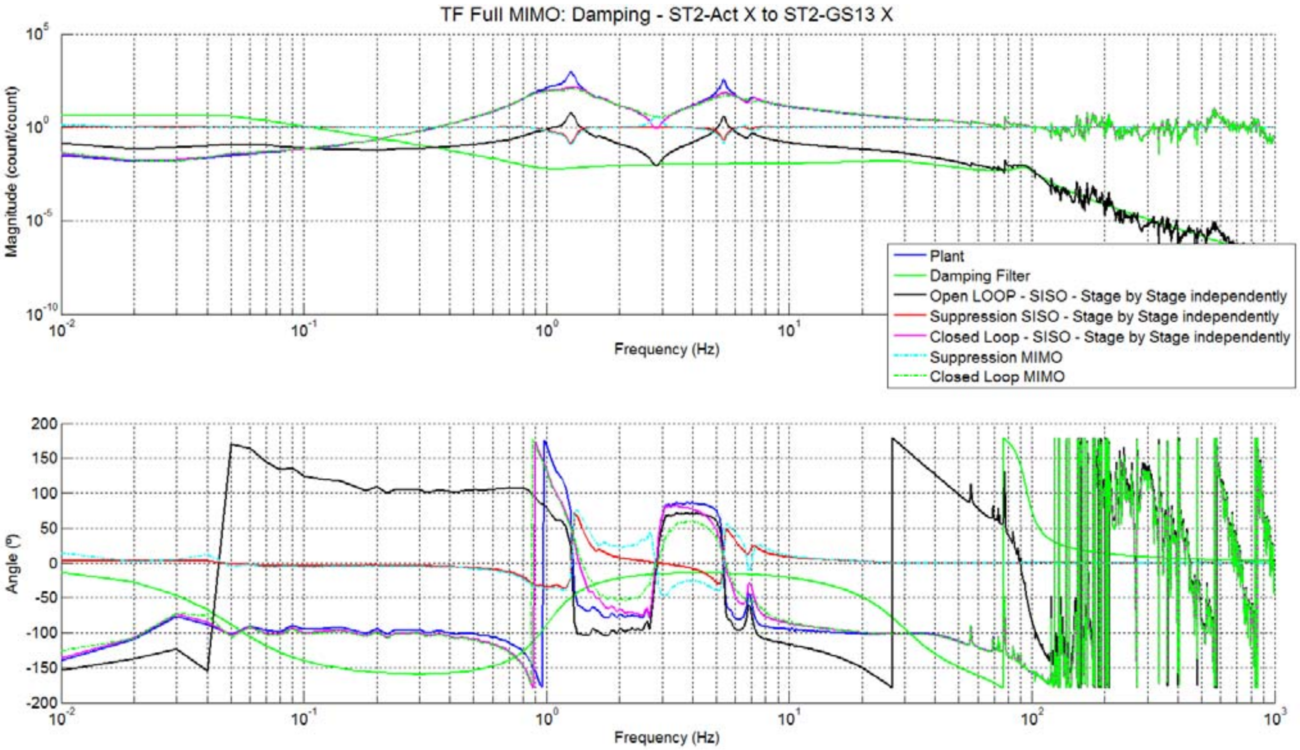

Figure 21 - Cartesian to Cartesian transfer functions – ST2-Act X to ST2-GS13 X - Damping Stages 1 & 2

Work to do:

- **Edit initial data**
- **Edit save_parameters file**
- **Edit save_Damping_Filters (date)**
- **Edit Damping_Filters and UG_Damping**
- **Edit save_data_TF_C2C_Damped_ST2_MIMO and save_Damping_Filters**

Damping filters can be engaged using the function "Switch_Damping_Loops_BSC_ISI.m". This function is located in the sVN at: /seismic/BSC-ISI/LASTI/Scripts/

Comparisons between the simulations and experiments with damping filters engaged

Figures 22 and 23 compare the system damped in simulations with the measured damped system. Simulations use the undamped, non symmetrized and non-calibrated transfer functions. The two following figures shows a good agreement between simulations and experiments (Main couplings; cross coupling from stage 1 to stage 2)

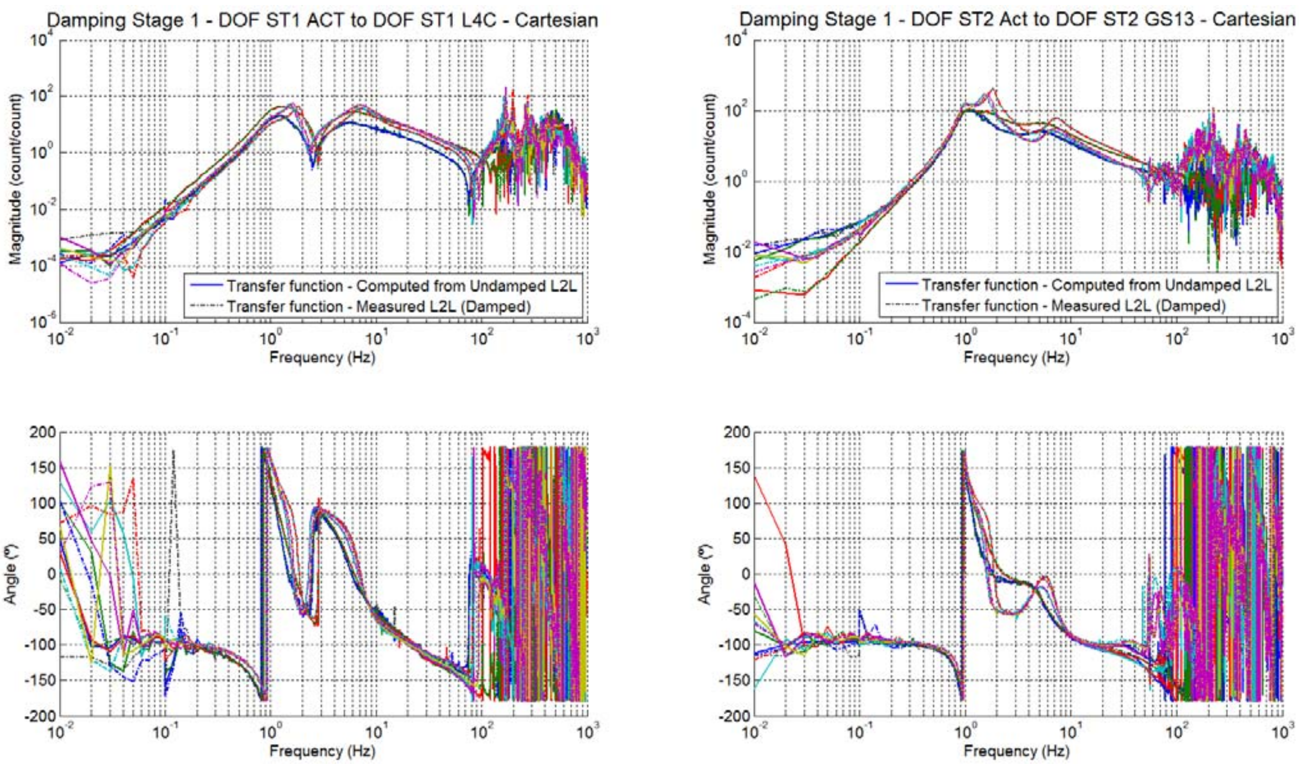

Figure 22 - Main couplings - Simulations (solid) - Measurements (dashed)

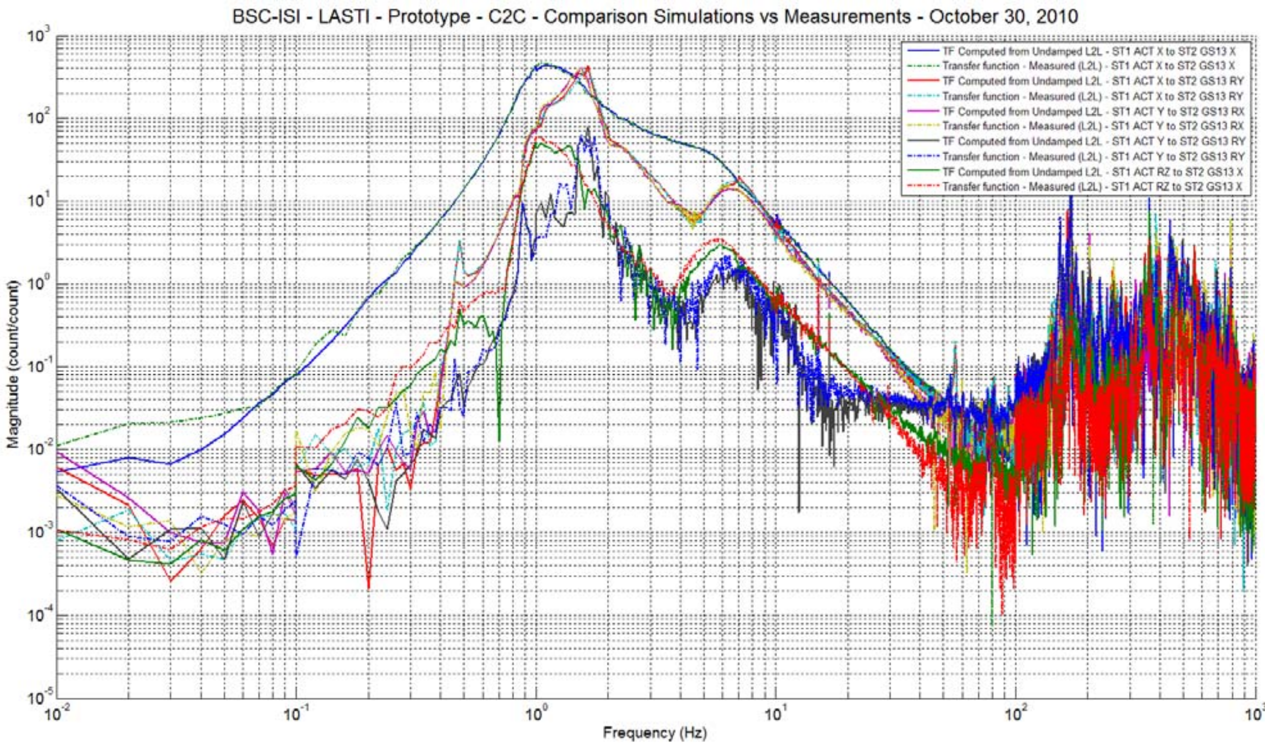

Figure 23 - Cross couplings - Simulations (solid) - Measurements (dashed)

Step_7_C2D_Damping_Filters_LASTI_Prototype_BSC_ISI

Step 7 digitizes the damping filters computed during steps 5 and 6. In details, this scripts:

- Loads damping filters
- Digitizes damping filters
- **Save digitized damping filters in a txt filen in the svn at** /svncommon/seisvn/seismic/BSC-ISI/LASTI/Prototype/Phase_3/Digitized_Filters

Work to do:

- **Edit save_parameters file**
- **Edit continuous damping filters file**

Step_8_Normalization_LASTI_Prototype_BSC_ISI

During this step, the relative motion sensor signals (CPS) are mixed with the inertial sensor signals (Geophones) to create the super sensors. In a first step, Geophones are normalized to the CPS. Next, the two types of sensors signals pass through a low pass filter (CPS) or a high pass filter (Geophones). Finally, CPS and the geophones signals are added to obtain "super sensor signals". In details, this script:

- Loads the transfer functions when the BSC-ISI is damped on both stages
- Converts the geophones in nm/count using the geophones models
- Normalizes the Geophones transfer functions to the CPS transfer functions (This normalization is a simple gain (close to 1for Stage 1 geophones)
- Computes the geophones normalization filters
- Loads the low and the high pass filters (complementary filters) located in the svn at /svncommon/seisvn/seismic/BSC-ISI/Common/Complementary_Filters_BSC_ISI
- Multiplies the CPS transfer functions by low pass filters (the corner frequency is the blend frequency)
- Multiplies the geophones transfer functions by high pass filters (the corner frequency is the blend frequency)
- Creates the Blend Filters

$$CPS_{Blend} = LP$$
$$GEO_{Blend} = HP * Normalization_{GEO}$$

- Computes the super sensors transfer functions

$$Super\ Sensor_{C2C} = TF_{C2C\ CPS} * CPS_{Blend} + TF_{C2C\ GEO} * GEO_{Blend}$$

- **Saves the results in the svn at**
  - Super sensors transfer function at svncommon/seisvn/seismic/BSC-ISI/LASTI/ Prototype/Phase_3/Data/Transfer_Functions/Simulations/Super_Sensors
  - Figures in the svn at svncommon/seisvn/seismic/BSC-ISI/LASTI/Prototype/Phase_3/ Data/Transfer_Functions/Simulations/Super_Sensors
  - Blend Filters in the svn at /svncommon/seisvn/seismic/BSC-ISI/LASTI/Prototype/ Phase_3/Continuous_Filters

The Super Sensor Calibration is based on the CPS Cartesian to Cartesian transfer functions and the sensitivity is 1nm/count.

Note 1: We are summing two signals coming from sensors that measure 2 different types of motions. For example, on stage 2, GS13s measure inertial motion and CPSs measure relative motion between stage 1 and stage 2. If Stage 1 jiggles while stage 2 is still, the CPS readout is large and the GS13 readout is low. The super sensor signal is high but what does it says? What does the calibration means?

Note 2: If the geophones gain correction (calibration correction) is not added, it means that the contribution of the geophones at the blend frequency won't be equal to the contribution of the CPS. Not normalizing the geophones to the CPS means that the blend frequency is not the actual blend frequency.

Since the local to local transfer functions have been symmetrized using the geophones model (step 2), the Cartesian to Cartesian transfer functions can be fitted using the geophones models.

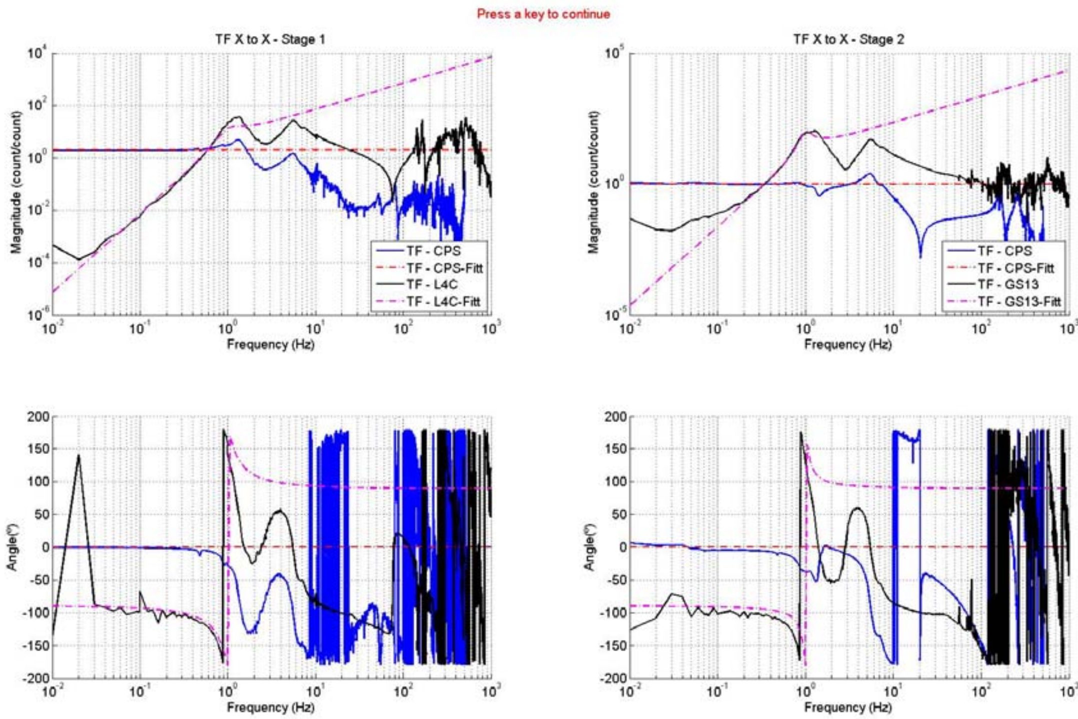

Figure 24 - Normalization filters

Figure 25 presents:

- The complementary filters (Low pass and high pass)
- The capacitive position sensor blend filter
- The L4C blend filter
- The contribution of the capacitive position sensor in the super sensor
- The contribution of the geophone on the super sensor
- The super sensor transfer function

Note that the super sensor is normalized to unity. In the latest control scripts, the super sensors is calibrated with a 1nm/count sensitivity.

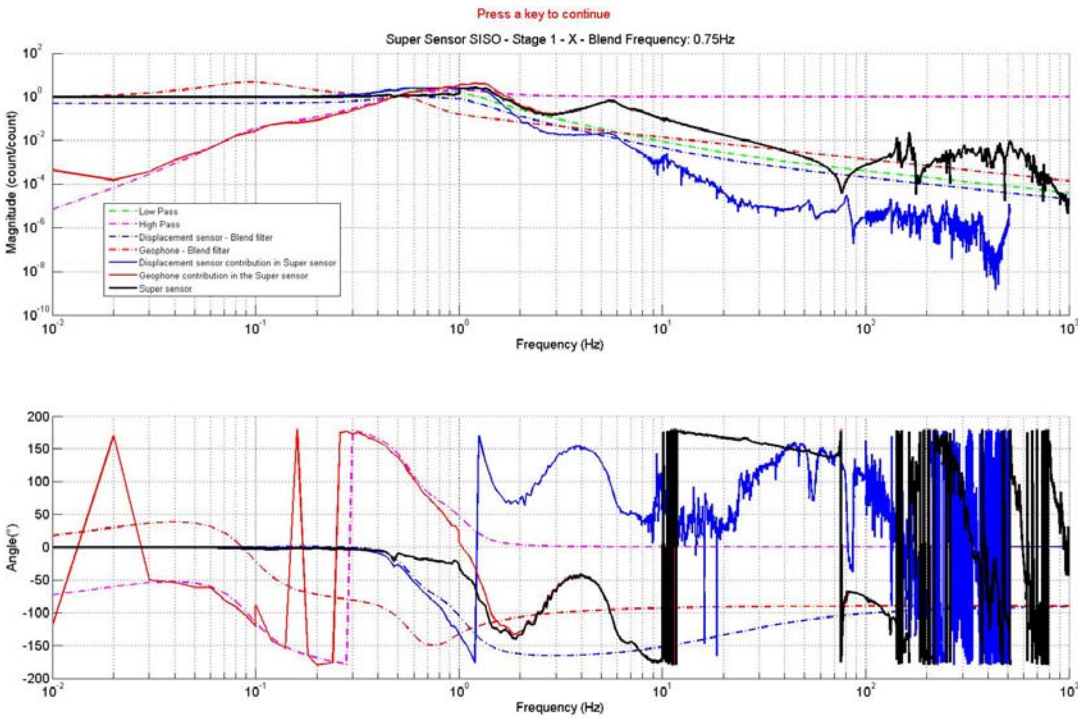

Figure 25 - Blend filters

Work to do:
- **Edit initial data**
- **Edit save_parameters file**
- **Edit title_C2C_Damped_str**
- **Edit save_data_Super_Sensors_Damped**
- **Edit save_Blend_Filters**

Step_9_C2D_Blend_Filters_LASTI_Prototype_BSC_ISI

Step 9 digitizes the blend filters computed during step 8. In details, this scripts:
- Loads continuous Blend filters
- Digitizes Blend filters
- **Save digitized blend filters in a txt filen in the svn at:** /svncommon/seisvn/seismic/BSC-ISI/LASTI/Prototype/Phase_3/Digitized_Filters

Work to do:
- **Edit save_parameters file**
- **Edit Blend_Filters**

Step_10_Isolation_Loops_ST2_Z_RX_RY_LASTI_Prototype_BSC_ISI

During steps 10, 11, 12 and 13, isolation filters are tuned. Step 10 and 11 enables to design the isolation Filters in the vertical directions (Z, RX, RY) of stage 2 and stage 1 respectively. Step 12 and 13 enables to design the isolation filters in the vertical directions (Z, RX, RY) of stage 2 and stage 1 respectively.

If isolation and damping filter are engaged, the control drive transfer functions are given by:

$$U_1/Exc = [TF_{C2C}CPS_{CS} * CPS_{Blend\ CS} + TF_{C2C}GEO_{CS} * GEO_{Blend\ CS}] * Controller_{S1} + TF_{C2C}GEO_{CS} * Damping_{S1}$$

$$U_2/Exc = [TF_{C2C}CPS_{FN} * CPS_{Blend\ FN} + TF_{C2C}GEO_{FN} * GEO_{Blend\ FN}] * Controller_{S2} + TF_{C2C}GEO_{FN} * Damping_{S2}$$

When the isolation and the damping filters are engaged, the MIMO open loop is given by:

$$Open\ Loop_{[30,30]} = TF_{C2C\ [30,12]} * Cont\ Damp_{[12,30]}$$

with

$$Cont\ Damp = \begin{bmatrix} CPS_{Blend\ CS} * Controller_{S1} & STS_{Blend\ CS} * Controller_{S1} + Damp\ L4C & L4C_{Blend\ CS} * Controller_{S1} + Damp\ L4C & 0 & 0 \\ 0 & 0 & 0 & CPS_{Blend\ CS} * Controller_{S2} & GS13_{Blend\ FN} * Controller_{S2} + Damp\ GS13 \end{bmatrix}$$

where the sub blocks of Cont Damp are diagonal matrices.

In details, this scripts:

- loads the damped super sensors transfer functions, the damped C2C transfer functions, the undamped symmetrized transfer functions, the blend filters, damping filters
- Computes the gain of the Isolation filters once the operator places poles and zeros. Filters are designed in a SISO configuration.
- Computes the SISO and MIMO super sensor transfer functions and the SISO and the MIMO C2C transfer functions with the isolation filters of Stage 2 (Z, RX and RY directions) engaged.
- **Saves the results in the svn at**
    - Saves data in the svn at svncommon/seisvn/seismic/BSC-ISI/LASTI/Prototype/Phase_3/Data/Transfer_Functions/Simulations/Super_Sensors
    - Saves figures in the svn at svncommon/seisvn/seismic/BSC-ISI/LASTI/Prototype/Phase_3/Data/Transfer_Functions/Simulations/Super_Sensors
    - Save Isolation Filters in the svn at /svncommon/seisvn/seismic/BSC-ISI/LASTI/Prototype/Phase_3/Continuous_Filters

Step_10_Isolation_Loops_ST2_Z_RX_RY_LASTI_Prototype_BSC_ISI calls:

- Plot_Super_Sensors_Isolation_SISO_BSC_ISI: This function is used when tuning the isolation Filters. It plots the plant (Super sensors), the Isolation Filters, the open Loop, the SISO sensitivity, the SISO closed loop for each DOF. The figure is automatically saved
- Plot_Super_Sensors_Isolation_MIMO_BSC_ISI: This function is used when tuning the isolation Filters. It plots the plant (Super sensors), the Isolation Filters, the open Loop, the MIMO sensitivity, the MIMO closed loop for each DOF. The figure is automatically saved.
- Plot_Full_MIMO_Suppression_BSC_ISI: This function plots the Full MIMO Suppression when the damping and the isolation filters are engaged.
- Plot_Isolation_Full_MIMO_BSC_ISI: This functions plots the TF of the Undamped, damped and controlled ISI

Work to do:

- **Edit save_parameters file**
- **Edit Initial Data**
- **Edit Blend_Filters, Damping_Filters**
- **Edit save_Isolation_Filters_ST2_V**
- **Edit save_Isolation_Filters_ST2_V**
- **Edit UUG_Isolation**
- **Edit Plant_Coarse_ID**

The following figure presents:

- In blue, the TF of the undamped structure (Super sensors)
- In dashed black, a coarse plant fit that is used to create the isolation filter
- In green, the isolation filter
- In black, the SISO open loop
- In magenta, the SISO sensitivity SISO relative to the MIMO response of the system when the damping filters of stage 2 are engaged
- In red, the closed loop SISO
- Dashed lines refer to the transfer function once the boost filter is introduced

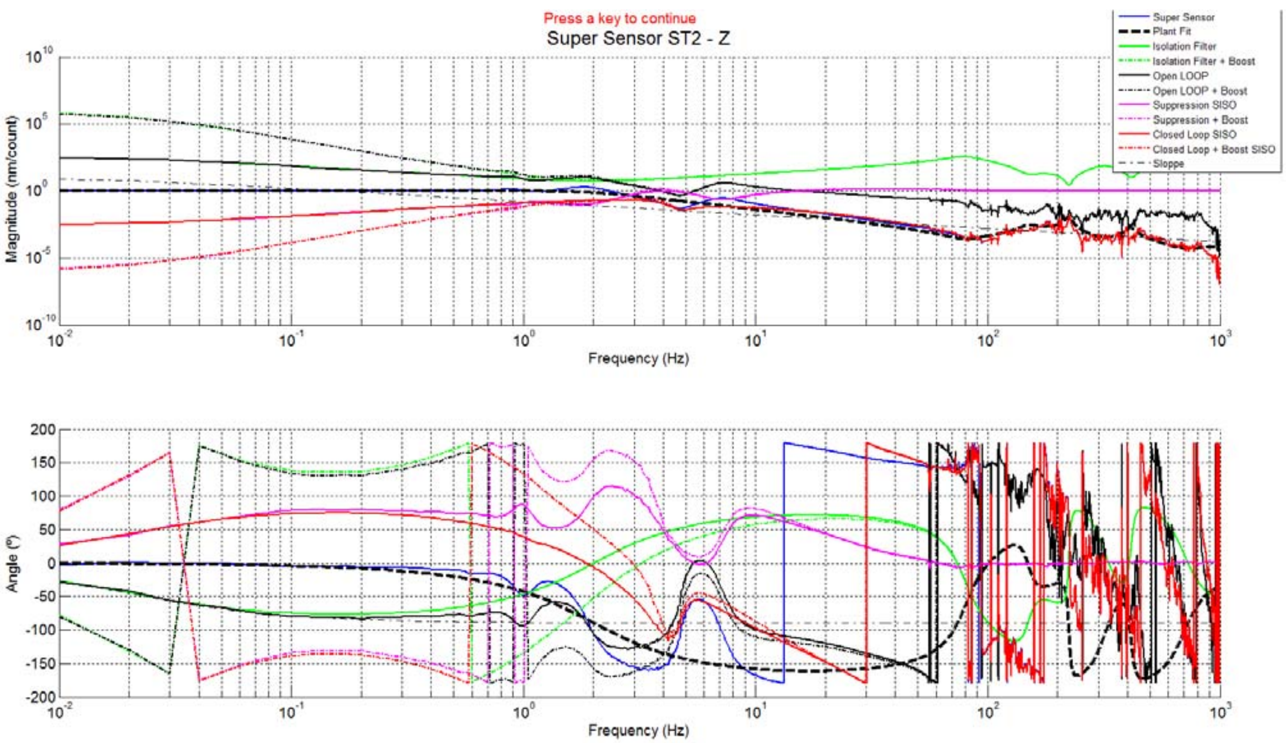

Figure 26 - Tuning Isolation loops on stage 2 - Z

The following figure compares the SISO and the MIMO response of the super sensor (Stage 2 - Z).

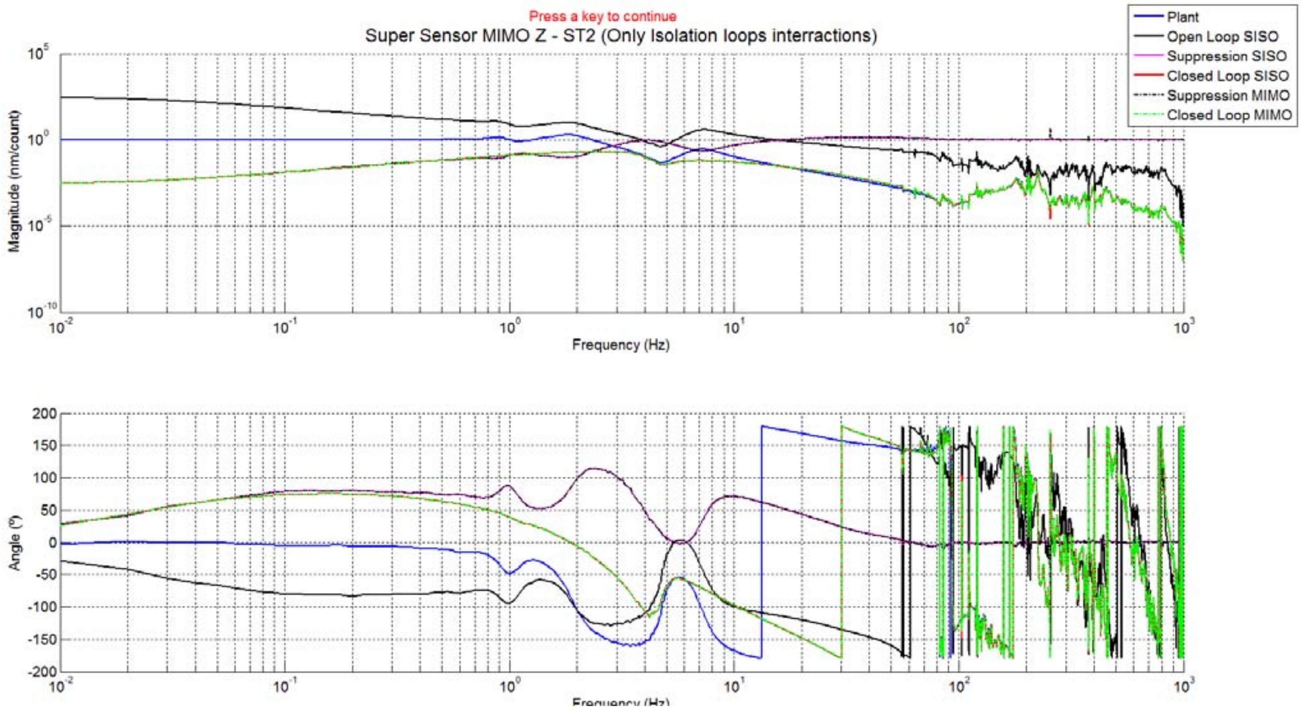

Figure 27 - Comparison SISO vs MIMO - Super sensor transfer function - ST2 act to ST2 Super sensor Z

The following figure shows the transfer function from actuator of stage 2 to the GS13 in the Z direction (MIMO calculation).

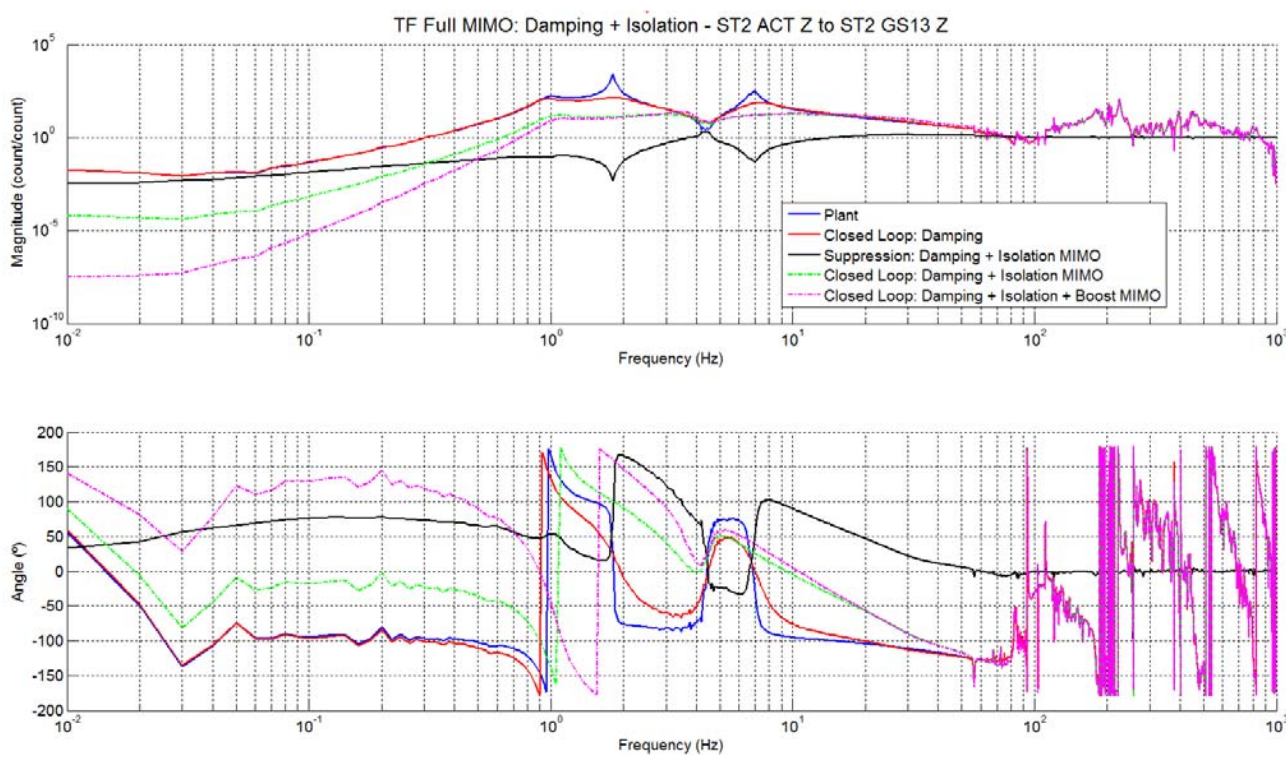

Figure 28 - Transfer function MIMO - ST2 Act to ST2 GS13 – Z

Step_11_Isolation_Loops_ST1_Z_RX_RY_LASTI_Prototype_BSC_ISI

During step 11, Isolation filters of stage 1 in the Z, RX and RY are designed. In details, this scripts:

- loads the damped and isolated (Stage 2 Z, RX and RY) super sensors transfer functions, the C2C transfer functions, the blend filters, damping filters, Isolation filters
- Computes the gain of the Isolation filters once the operator places poles and zeros. Filters are designed in a SISO configuration.
- Computes the SISO and MIMO super sensor transfer functions and the SISO and the MIMO C2C transfer functions with the isolation filters of Stages 1 and 2 (Z, RX and RY directions) engaged.
- **Saves the results in the svn at**
    - Saves data in the svn at svncommon/seisvn/seismic/BSC-ISI/LASTI/Prototype/ Phase_3/Data/Transfer_Functions/Simulations/Super_Sensors
    - Saves figures in the svn at svncommon/seisvn/seismic/BSC-ISI/LASTI/Prototype/Phase_3/Data/Transfer_Functions/Simulations/Super_Sensors
    - Save Isolation Filters in the svn at /svncommon/seisvn/seismic/BSC-ISI/LASTI/ Prototype/Phase_3/Continuous_Filters

Step_12_Isolation_Loops_ST2_X_Y_RZ_LASTI_Prototype_BSC_ISI

During step 12, Isolation filters of stage 2 in the X, Y and Z are designed. In details, this scripts:

- loads the damped and isolated (Stages 1 and 2 Z, RX and RY) super sensors transfer functions, the C2C transfer functions, the blend filters, damping filters, Isolation filters
- Computes the gain of the Isolation filters once the operator places poles and zeros. Filters are designed in a SISO configuration.
- Computes the SISO and MIMO super sensor transfer functions and the SISO and the MIMO C2C transfer functions with the isolation filters of Stages 1 and 2 (Z, RX and RY directions) engaged and Stage 2 (Z, RX and RY directions)
- **Saves the results in the svn at**
    - Saves data in the svn at svncommon/seisvn/seismic/BSC-ISI/LASTI/Prototype/ Phase_3/Data/Transfer_Functions/Simulations/Super_Sensors
    - Saves figures in the svn at svncommon/seisvn/seismic/BSC-ISI/LASTI/Prototype/Phase_3/Data/Transfer_Functions/Simulations/Super_Sensors
    - Save Isolation Filters in the svn at /svncommon/seisvn/seismic/BSC-ISI/LASTI/ Prototype/Phase_3/Continuous_Filters

Step_13_Isolation_Loops_ST1_X_Y_RZ_LASTI_Prototype_BSC_ISI

During step 12, Isolation filters of stage 2 in the X, Y and Z are designed. In details, this scripts:

- loads the damped and isolated (Stages 1 and 2 Z, RX and RY + Stage 2 Z, RX and RY) super sensors transfer functions, the C2C transfer functions, the blend filters, damping filters, Isolation filters
- Computes the gain of the Isolation filters once the operator places poles and zeros. Filters are designed in a SISO configuration.
- Computes the SISO and MIMO super sensor transfer functions and the SISO and the MIMO C2C transfer functions with the isolation filters of Stages 1 and 2 (Z, RX, RY, Z, RX and RY directions) engaged
- **Saves the results in the svn at**
    - Saves data in the svn at svncommon/seisvn/seismic/BSC-ISI/LASTI/Prototype/Phase_3/Data/Transfer_Functions/Simulations/Super_Sensors
    - Saves figures in the svn at svncommon/seisvn/seismic/BSC-ISI/LASTI/Prototype/Phase_3/Data/Transfer_Functions/Simulations/Super_Sensors
    - Save Isolation Filters in the svn at /svncommon/seisvn/seismic/BSC-ISI/LASTI/Prototype/Phase_3/Continuous_Filters

The figure below shows the design of the stage 1 isolation filter in the X direction. The super sensor in blue is calculated once the isolation filters [Stages 1 and 2 (Z, RX, RY) and stage 2 (X, Y, RZ)] are engaged.

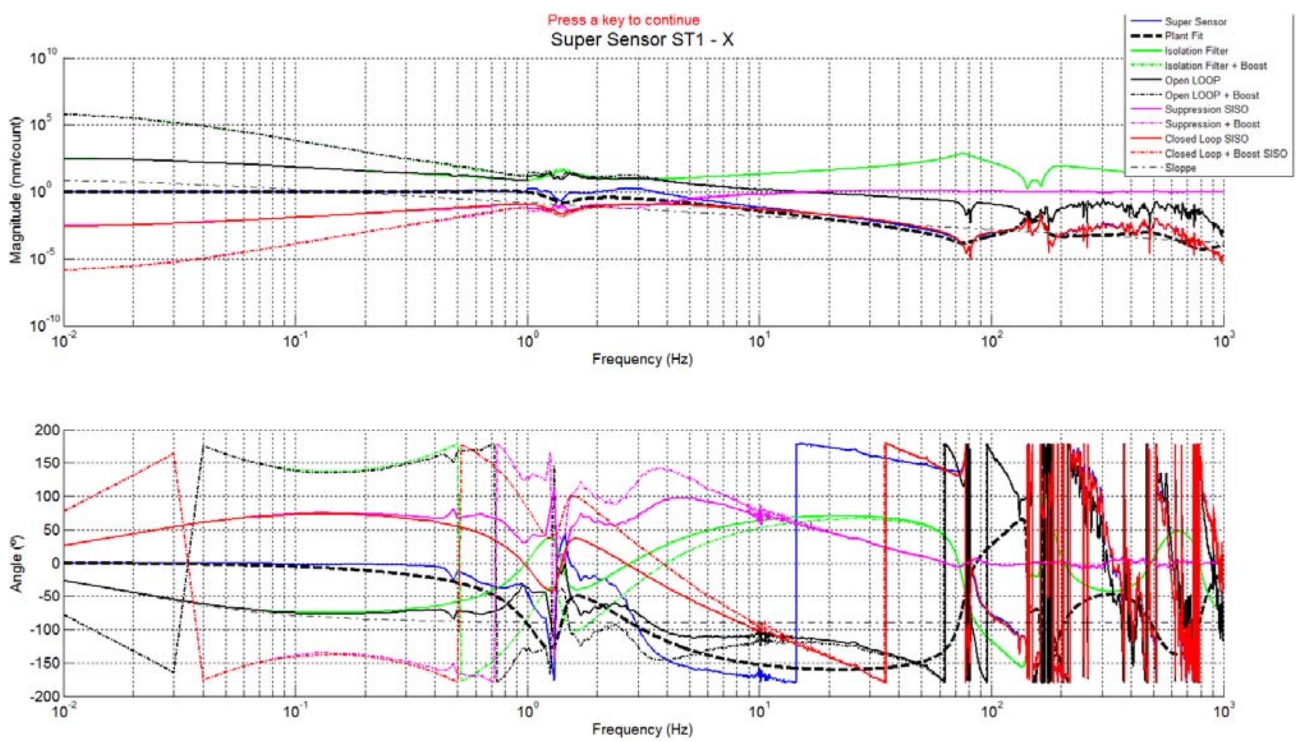

Figure 29 - Tuning Isolation loops on stage 1 - X

The figure below shows the SISO and the MIMO response of the super sensor (stage 1 direction X) when all isolation filters are engaged.

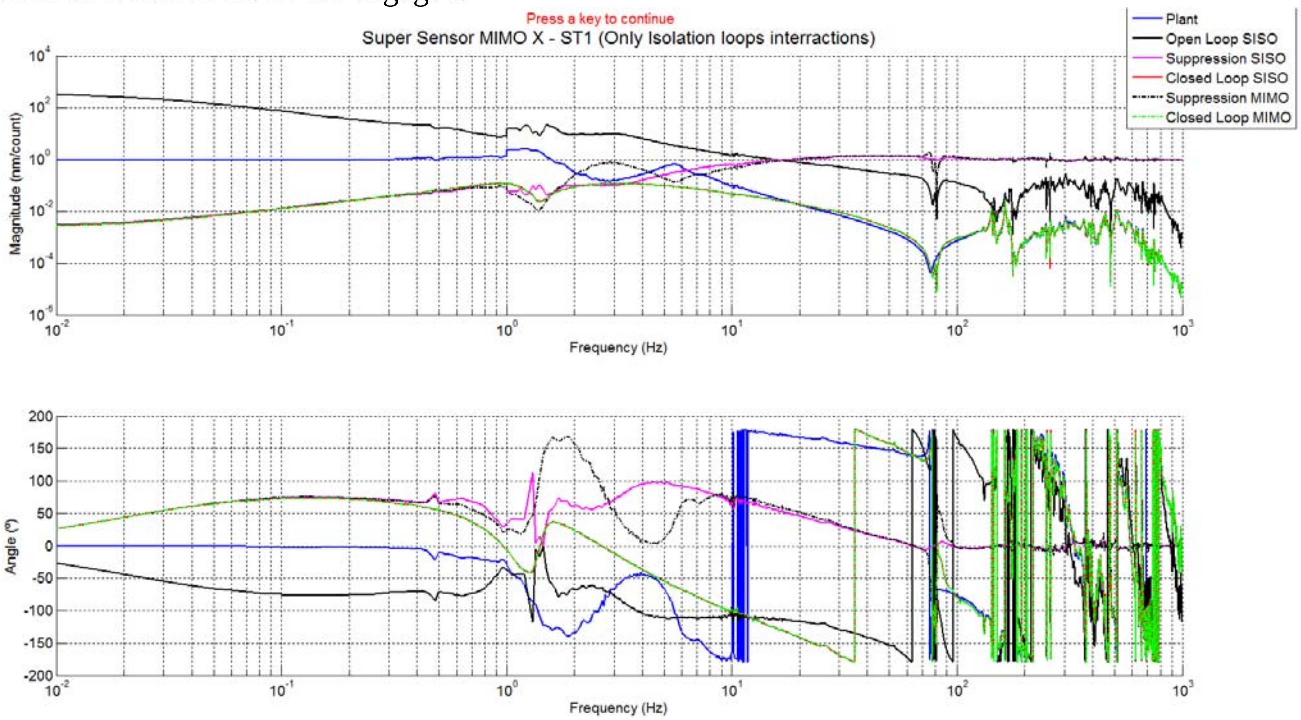

Figure 30 - Comparison SISO vs MIMO - Super sensor transfer function – ST1 act to ST1 Super sensor X

The figure below shows the transfer function from actuator X of stage 1 to the L4C in the X-direction when the system is undamped, damped and isolated (with and without boost).

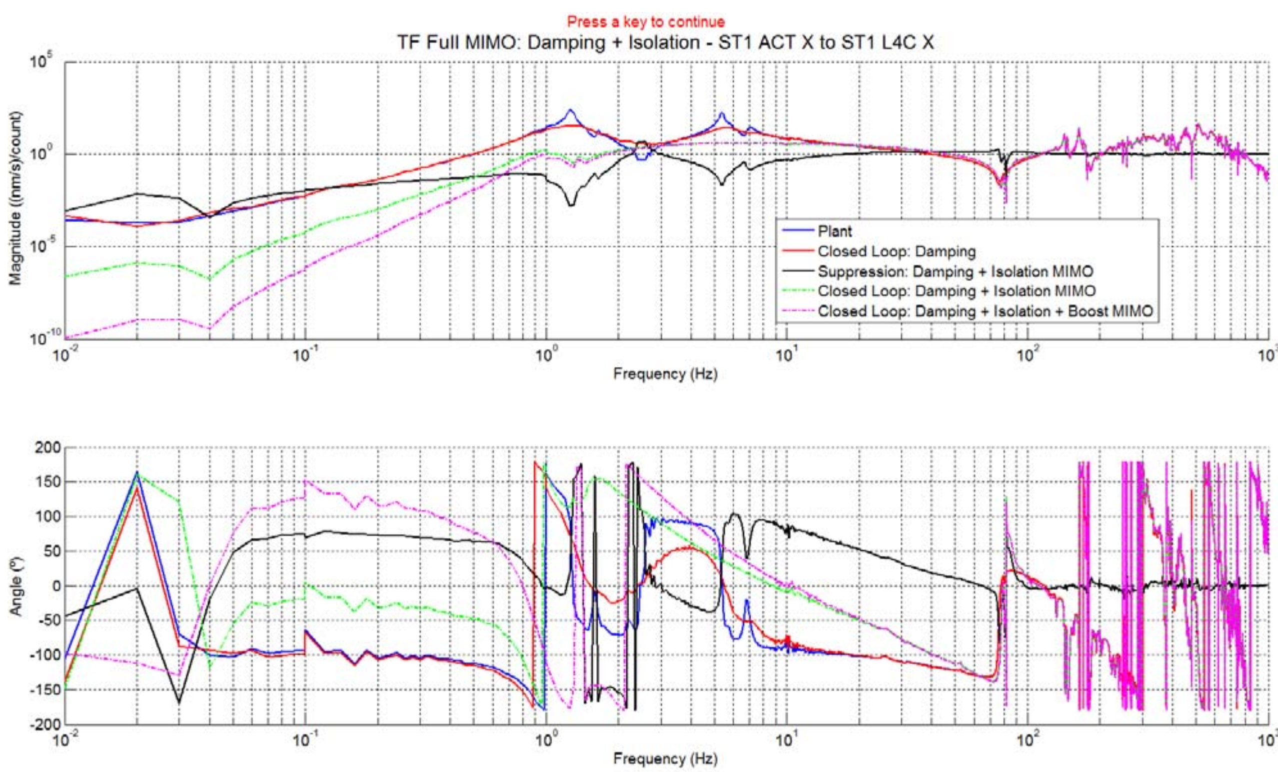

Figure 31 - Transfer function MIMO – ST1 Act to ST1 L4C – X

Step_14_Open_Loop_check_LASTI_Prototype_BSC_ISI

This step is used to show the evolution of the open loop during the design of the control. In details, this script:

- Loads the isolation filters
- Loads the open loops computed during the design. These open loops are computed with 0,3,6 or 9 isolation filters already engaged
- Computes the open loop of each DOF considering no isolation filters are engaged (first filter to be engaged)
- Computes the open loop of each DOF considering 11 isolation filters are already engaged (last filter to be engaged)

The four following figures compare the open loops when:

- The isolation filters is the first filter to be engaged
- The isolation filter is engaged respecting the design order
- The isolation filter is the last to be engaged

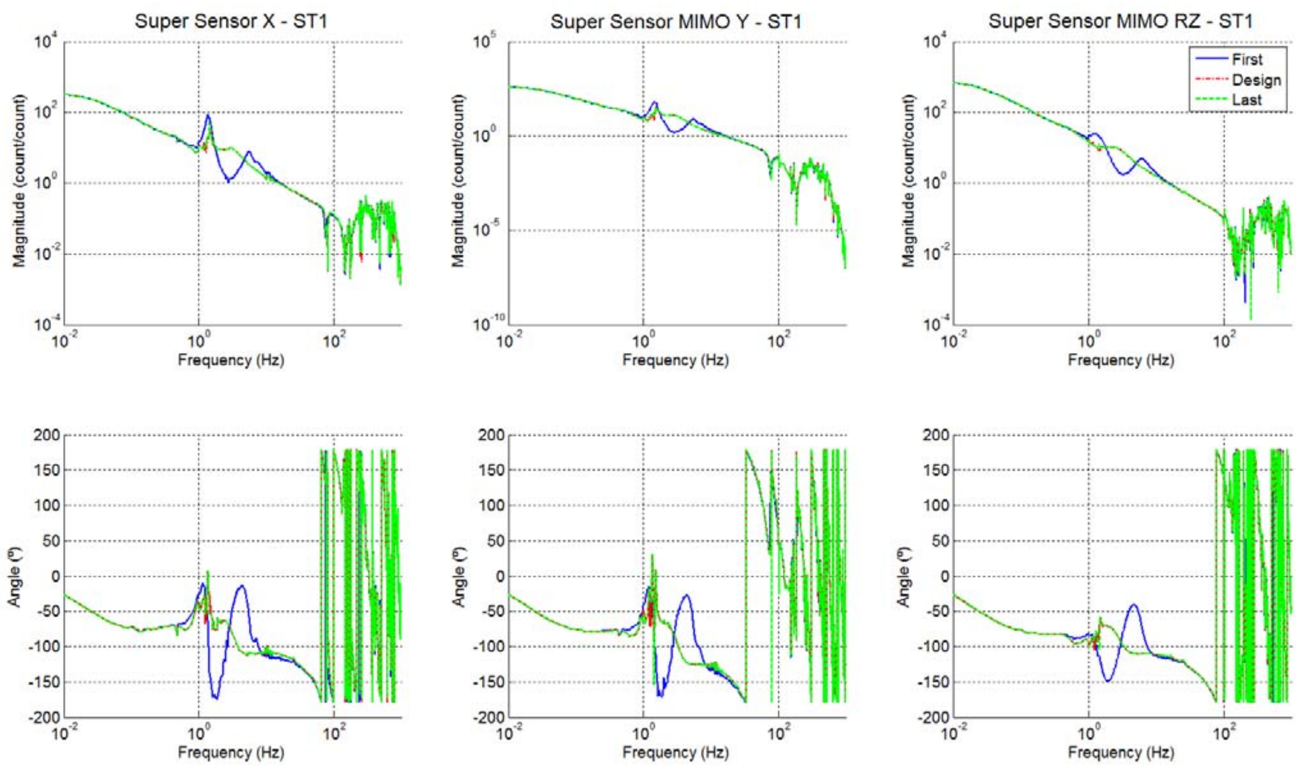

Figure 32 - Evolution of the open loops - Stage 1 - X Y RZ

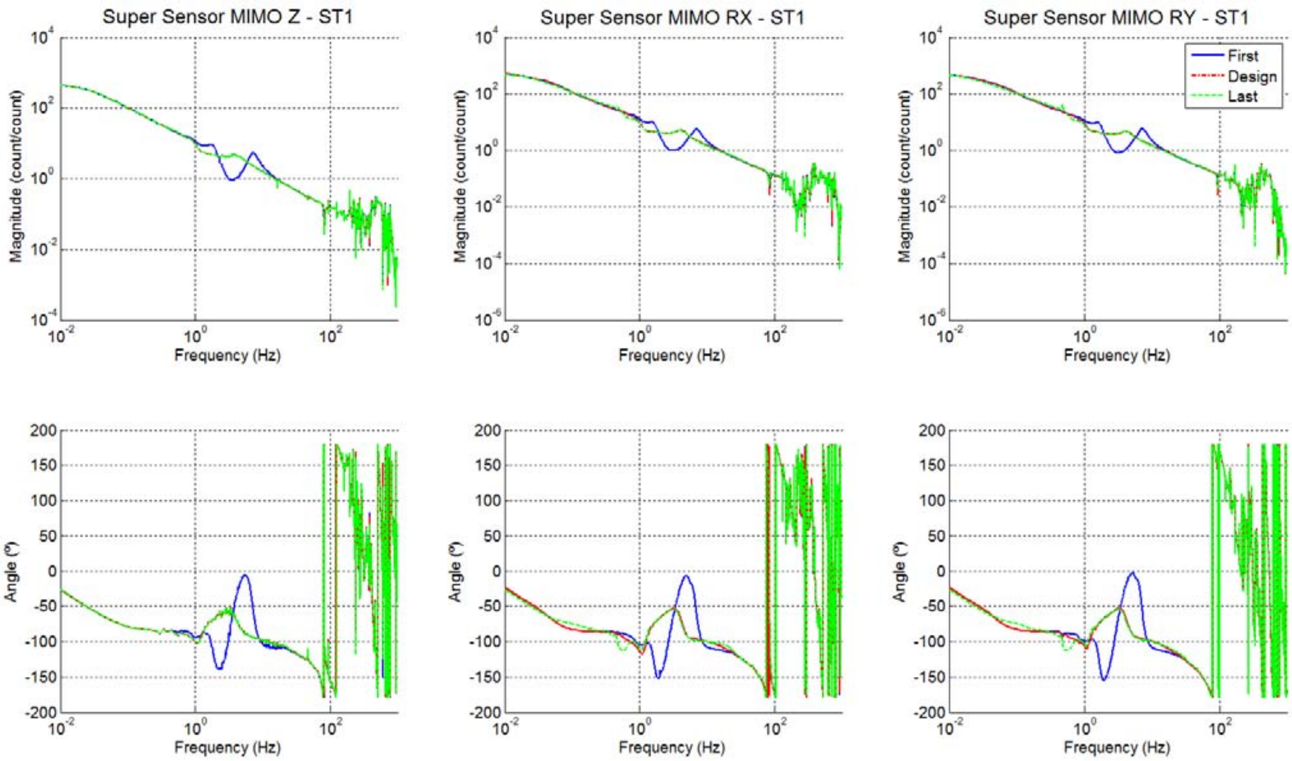

Figure 33 - Evolution of the open loops - Stage 1 – Z RX RY

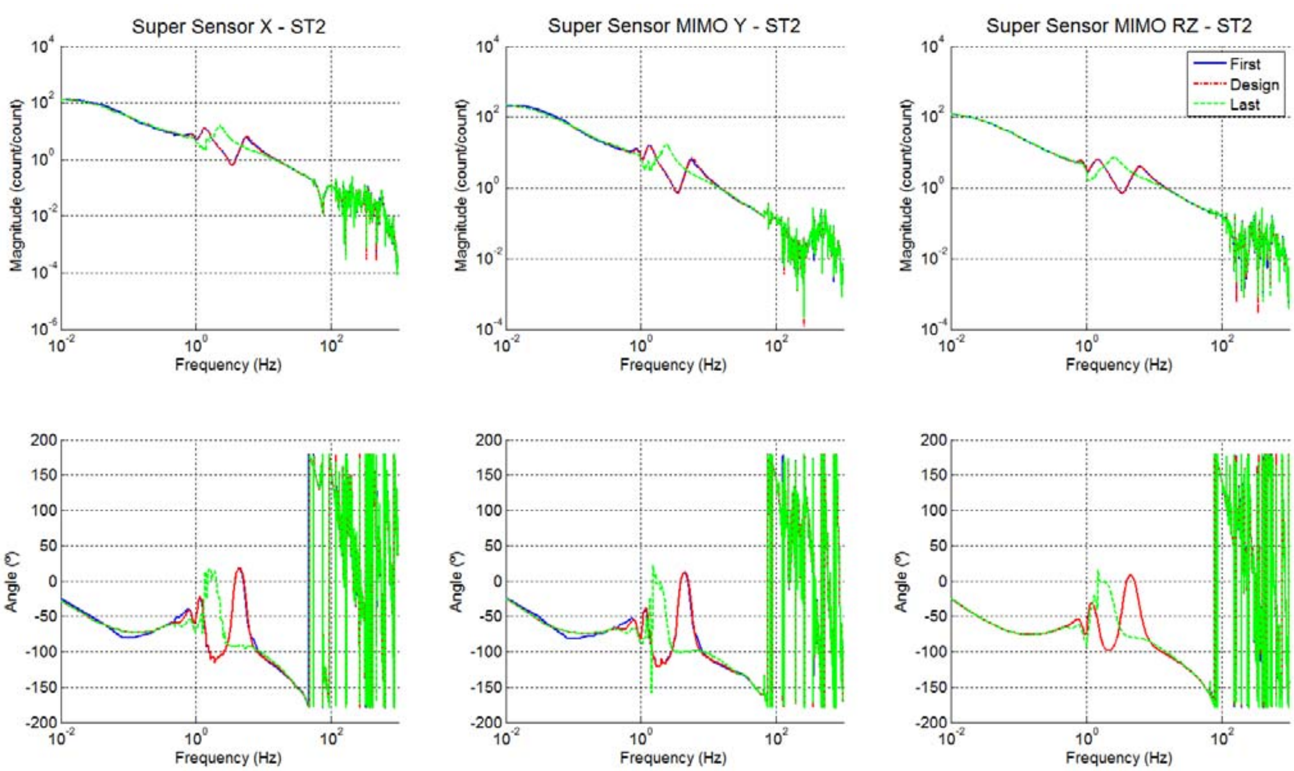

Figure 34 - Evolution of the open loops - Stage 2 - X Y RZ

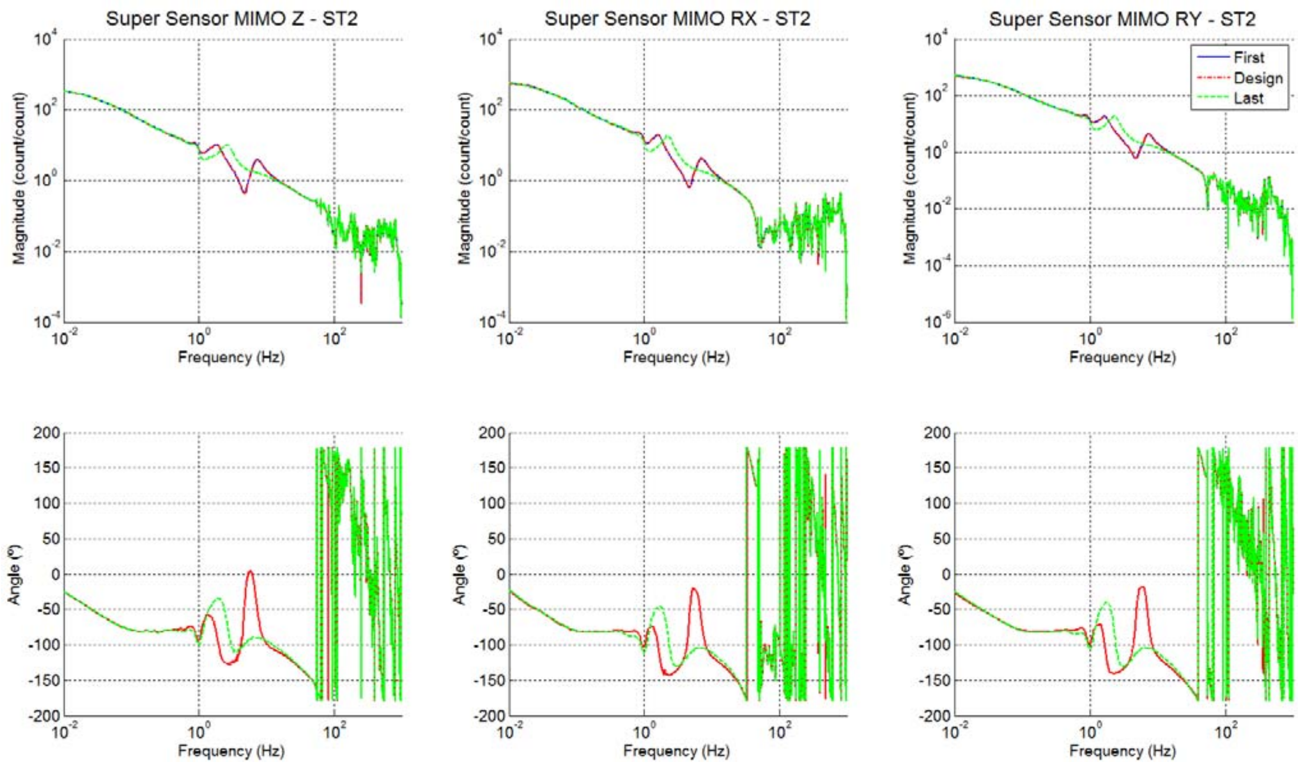

Figure 35 - Evolution of the open loop - Stage 2 – Z RX RY

Step_15_C2D_Isolation_Filters_LASTI_Prototype_BSC_ISI

Step 15 digitizes the isolation filters computed during steps 10, 11, 12 and 13. In details, this scripts:

- Loads continuous isolation filters
- Digitizes the isolation filters
- **Save digitized isolation filters in a txt filen in the svn at:** /svncommon/seisvn/seismic/BSC-ISI/LASTI/Prototype/Phase_3/Digitized_Filters

Work to do:

- **Edit save_parameters file**
- **Edit Blend_Filters**

Step_16_Tilt_decoupling_LASTI_Prototype_BSC_ISI

Not done

6 Measuring performances

Powerspectra can be measured either using Matlab or DTT. Matlab scripts use DAQ channels whereas DTT doesn't. There are two types of Matlab scripts:

- Scripts that get the data, compute, display and save powerspectra in .mat and .fig file (These scripts can be used as long as the data are stored in the front-end)
- Scripts that read and display powerspectra already computed

Matlab functions that evaluate powerspectra are located at:
/svncommon/seisvn/seismic/BSC-ISI/LASTI/Scripts/Data_Collection

For instance, the function that plots GS13 powerspectra is called Powerspectra_GS13_BSC_ISI. Input arguments are:

- GPS_TIME_Start defines when the measurements started
- F_resolution is the frequency resolution in Hz
- Average is the number of average
- Overlap_per_cent is the overlap in % (for 50%, set this argument to 50)
- Display (set 1 if you want to display powerspectra)
- save_path defines where the figure will be saved
- figure_1_name is the figure name

Comparison between the simulated and measured transfer functions when the damping and the isolation filters are engaged

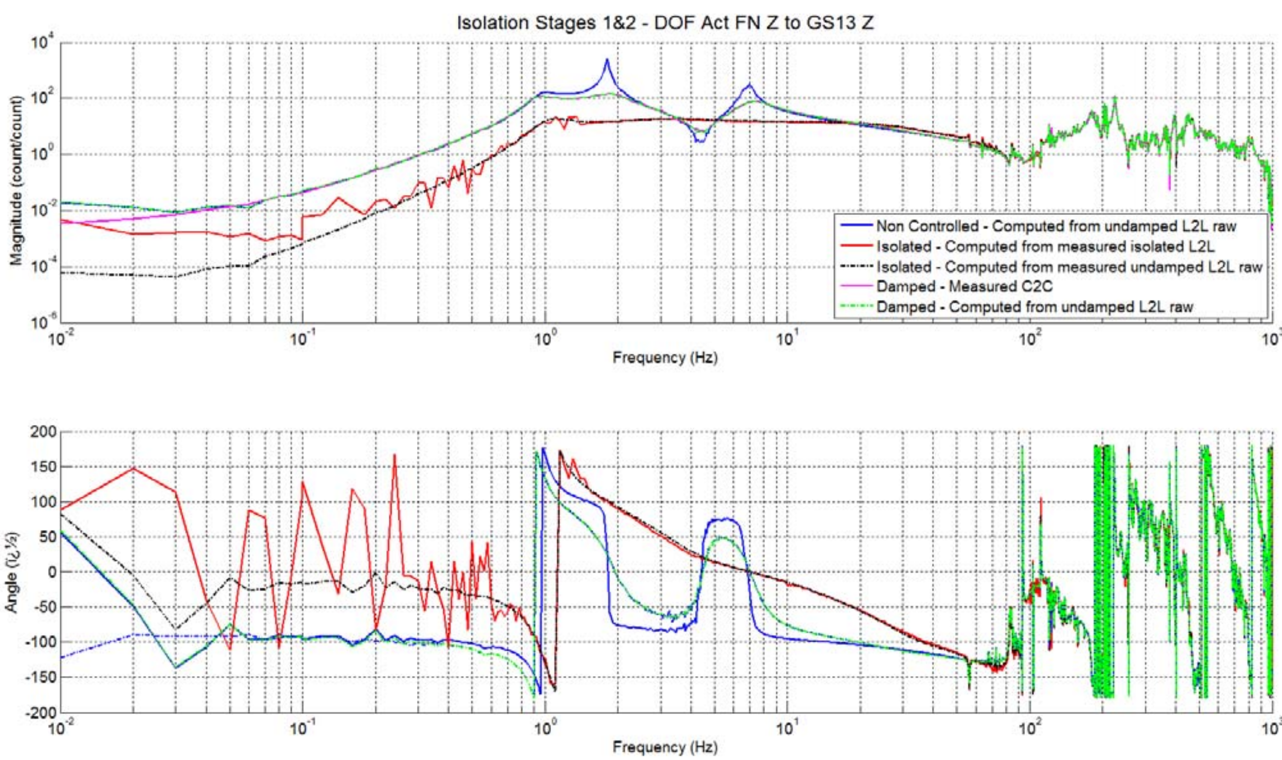

Figure 36 - Comparison Simulation vs experiments - Damping and isolation filters engaged
TF Cartesian to Cartesian - ST2 Z to ST2 GS13 Z

7 Work to do to improve the scripts