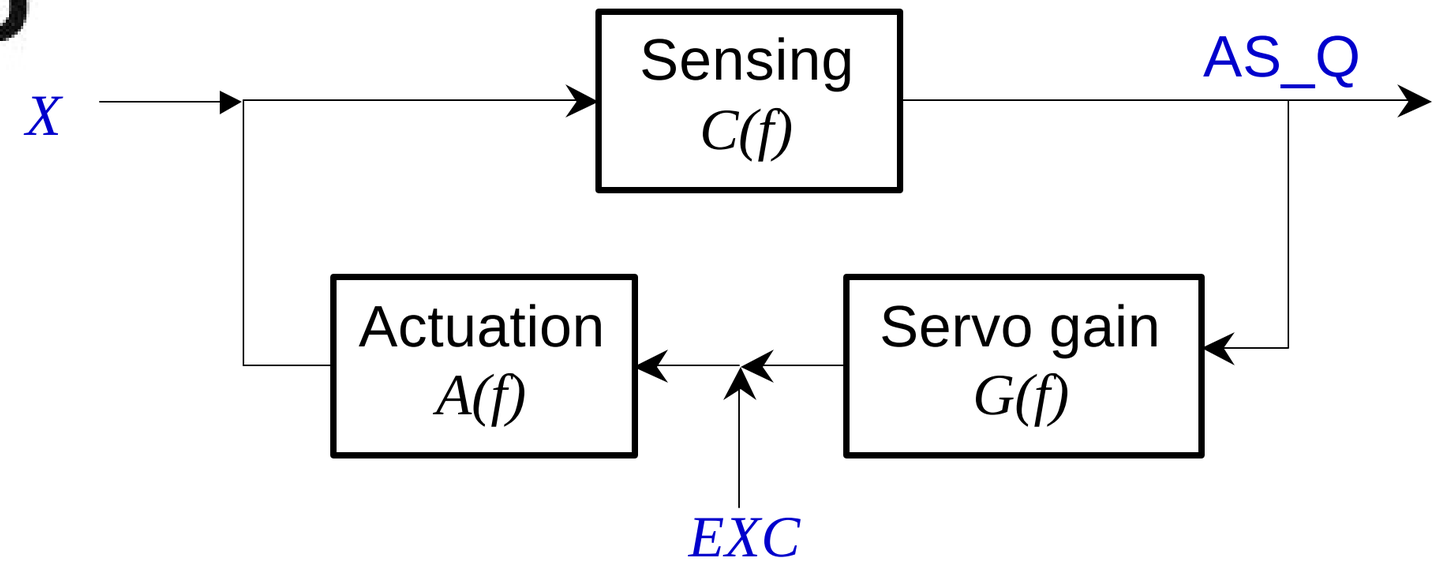
Autocalibration

M. Landry
LIGO Hanford

and the calibration team

Autocalibrator

- TCL/Expect script for one-click calibration
- Typically performed during commissioning studies, and roughly once per day during S2
- Employs Sigg's DTT and Matlab
- Easy web access of data products



what we're after: $\frac{\mathrm{AS\_Q}}{X} = \frac{C}{1 - G\,A\,C}$

what we measure: $\frac{\mathrm{AS\_Q}}{\mathrm{EXC}} = \frac{A\,C}{1 - G\,A\,C}$

To get the response function to an externally-induced displacement, just divide by *A(f)*

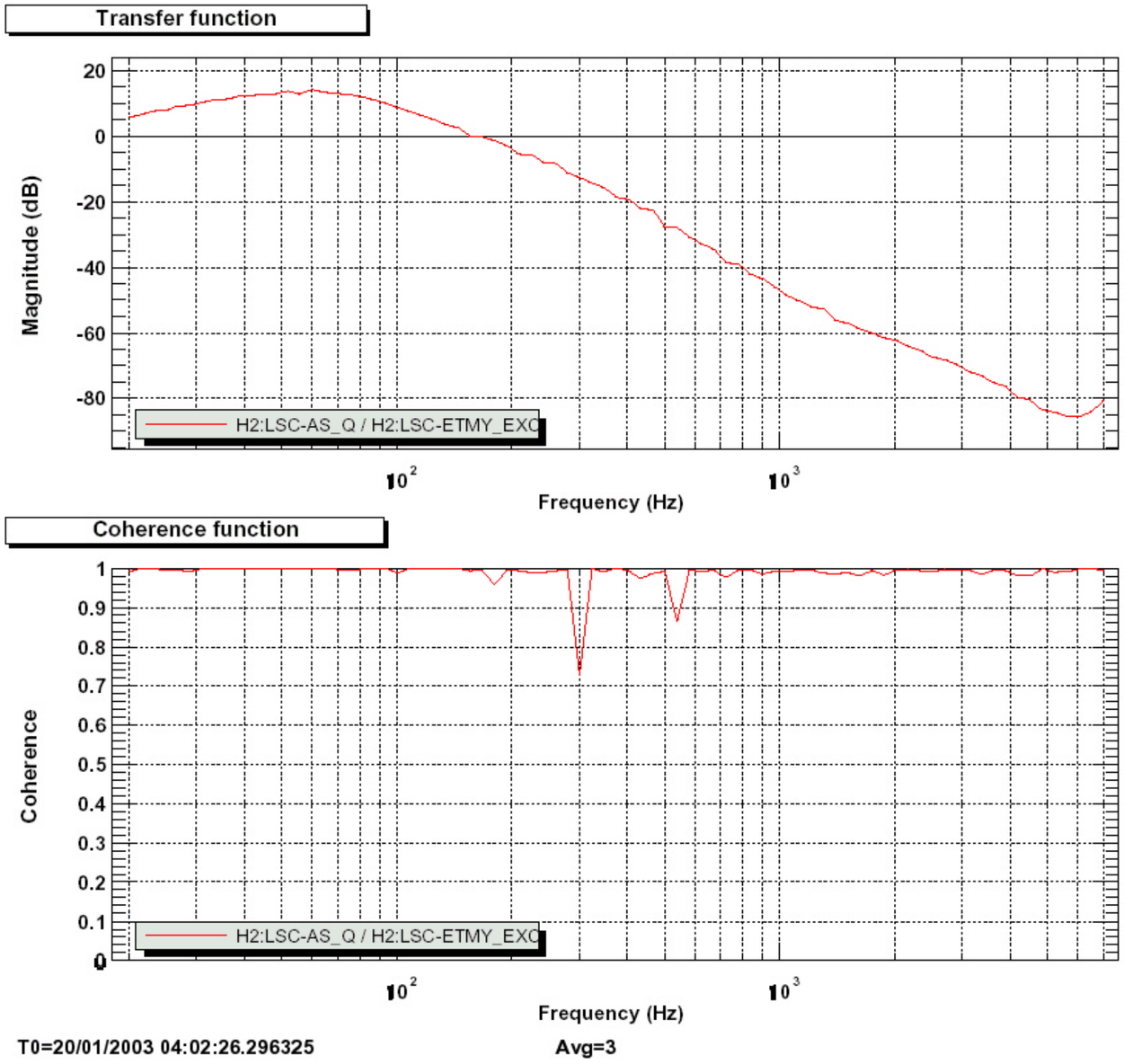
A swept-sine excitation traces out the full transfer function

(P.Shawhan, G020064-00)

Autocal ingredients:

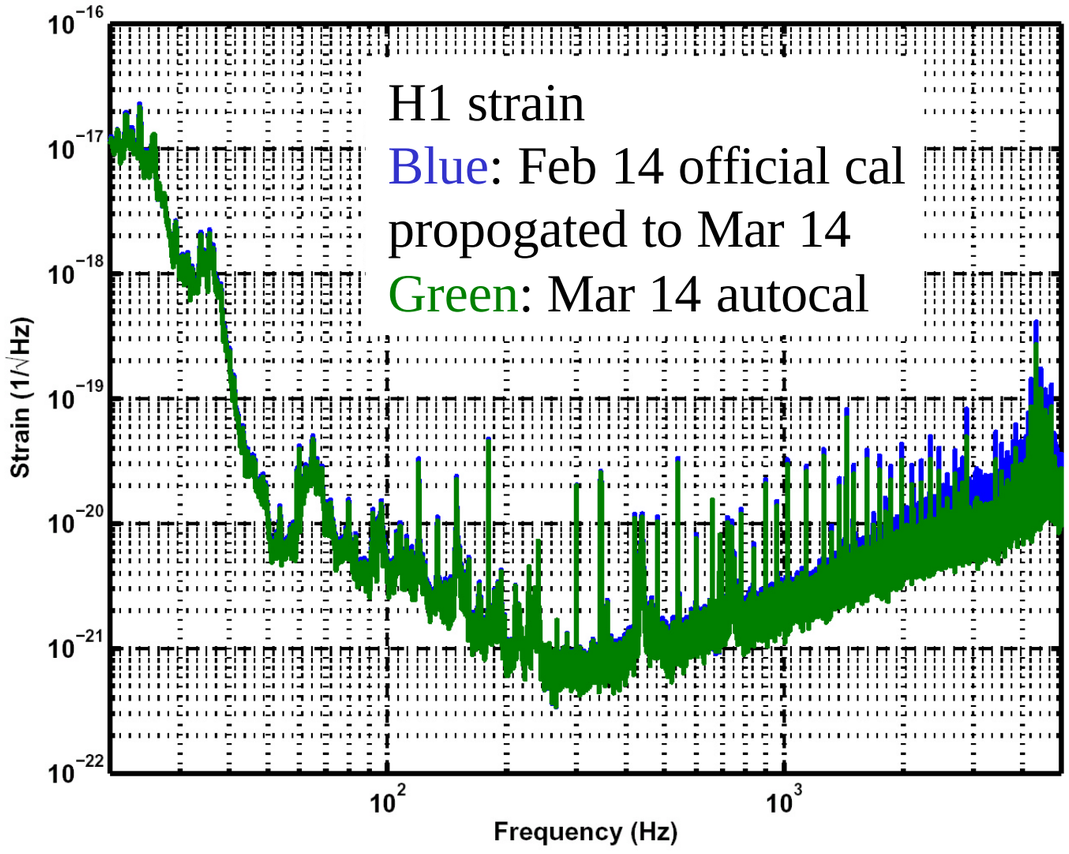
- DC calibration
- Swept sine in the form of AS_Q/ETM_EXC
- AS_Q amplitude spectral density
- Actuation function A(f)

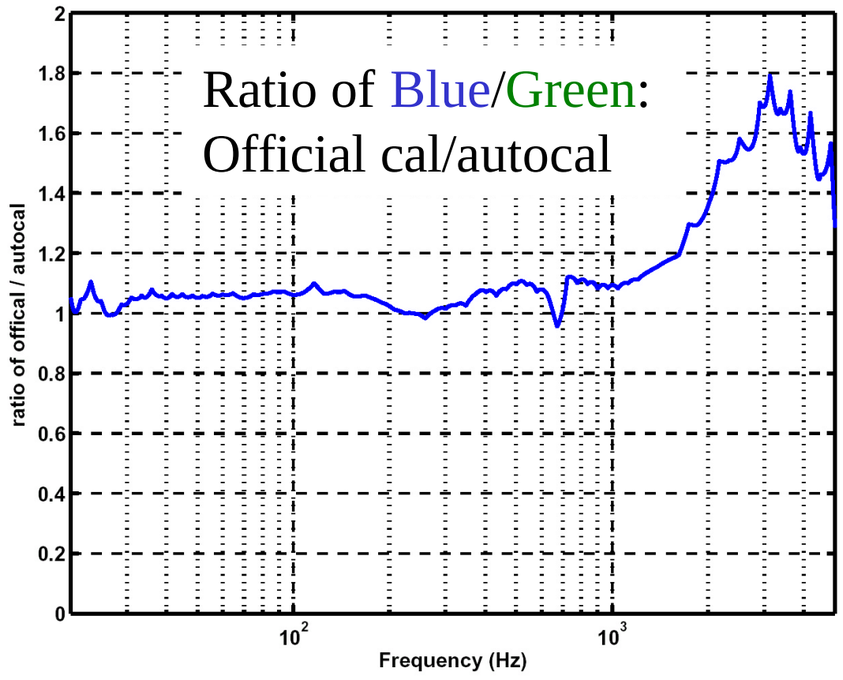
Transfer function is interpolated and divided out of AS_Q spectra



Autocalibration G030092-00W

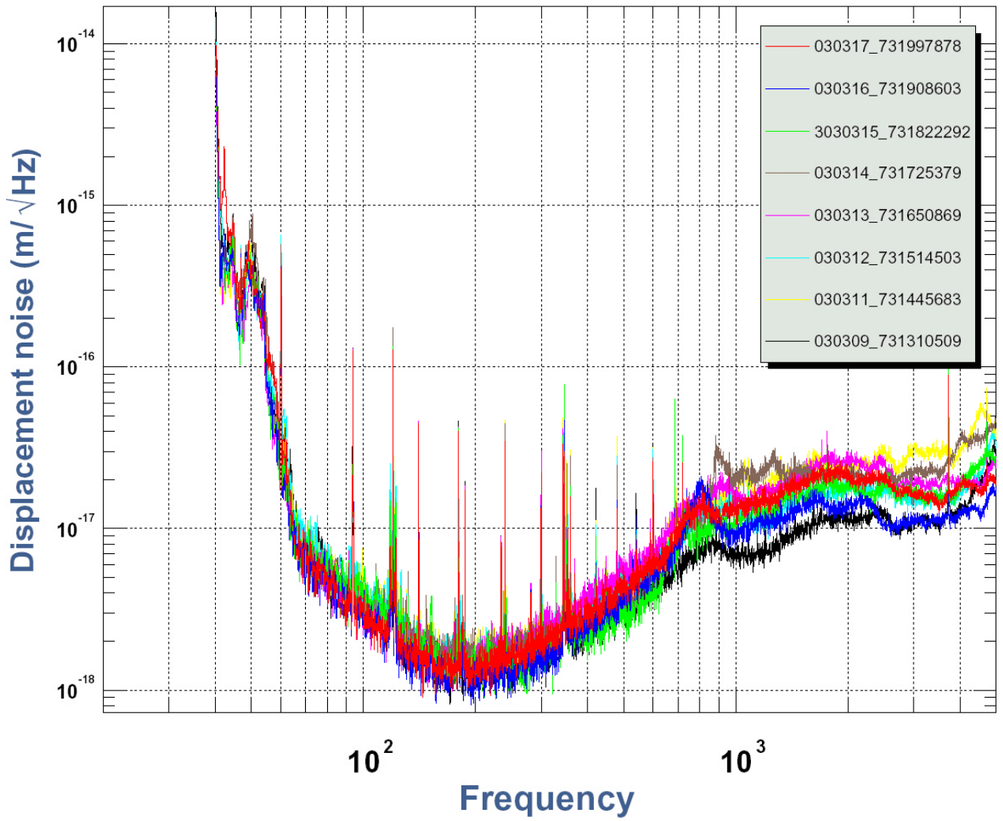
Comparison of Autocalibration spectra to input to ‘official’ calibration propogated in time



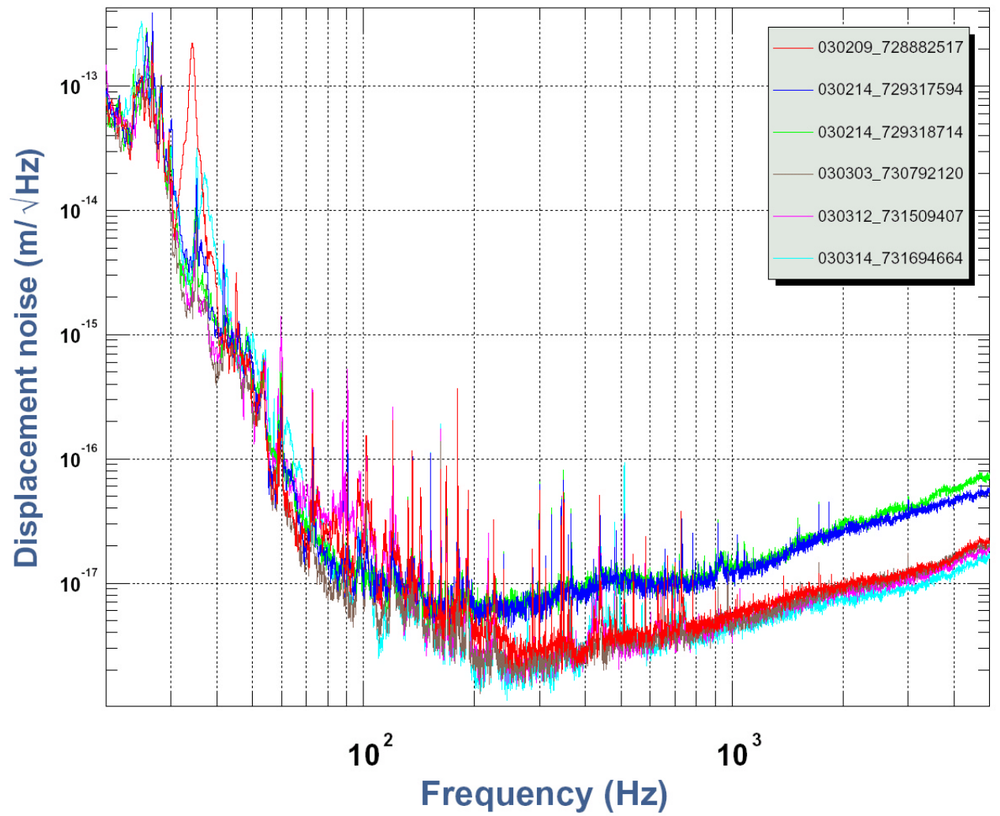


Time variations in autocal S2 output

L1 S2 autocal displacement spectra



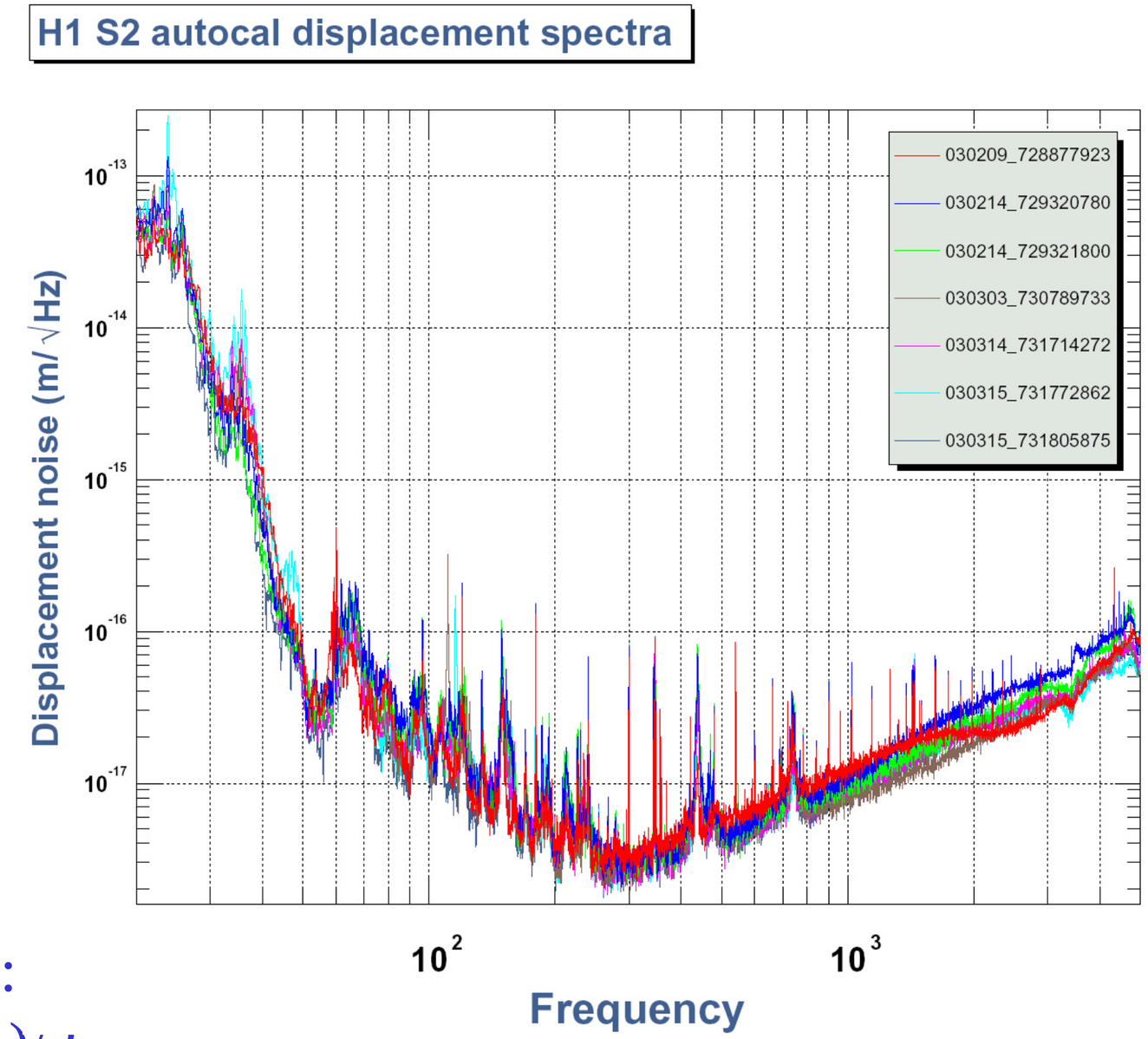
H2 S2 autocal displacement spectra



Displacement noise:
L_x-L_y, not (!) (L_x-L_y)/2

Autocalibration G030092-00W

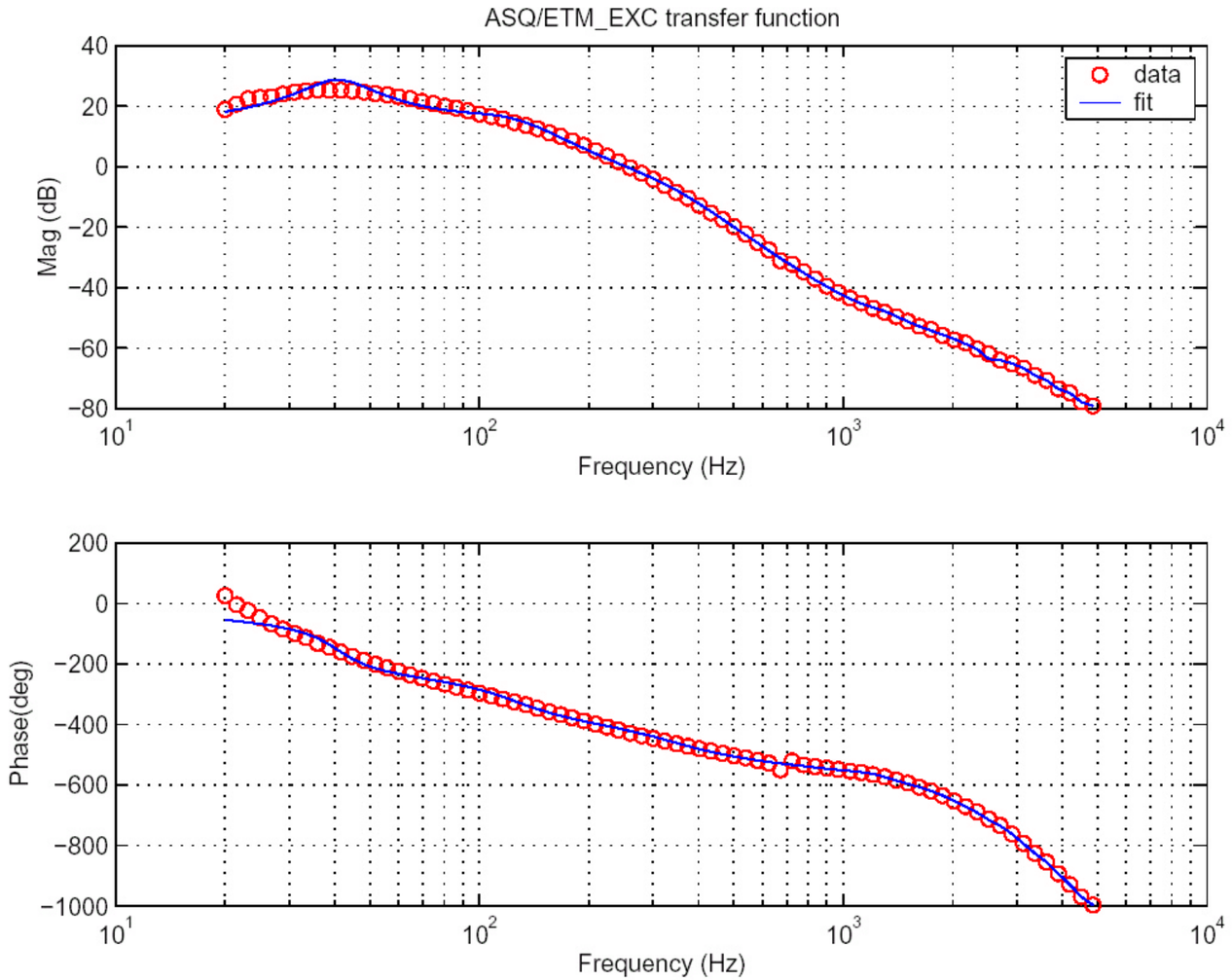
Time variations in autocal S2 output



Displacement noise:
L_x-L_y, not (!) (L_x-L_y)/2

Autocalibration G030092-00W

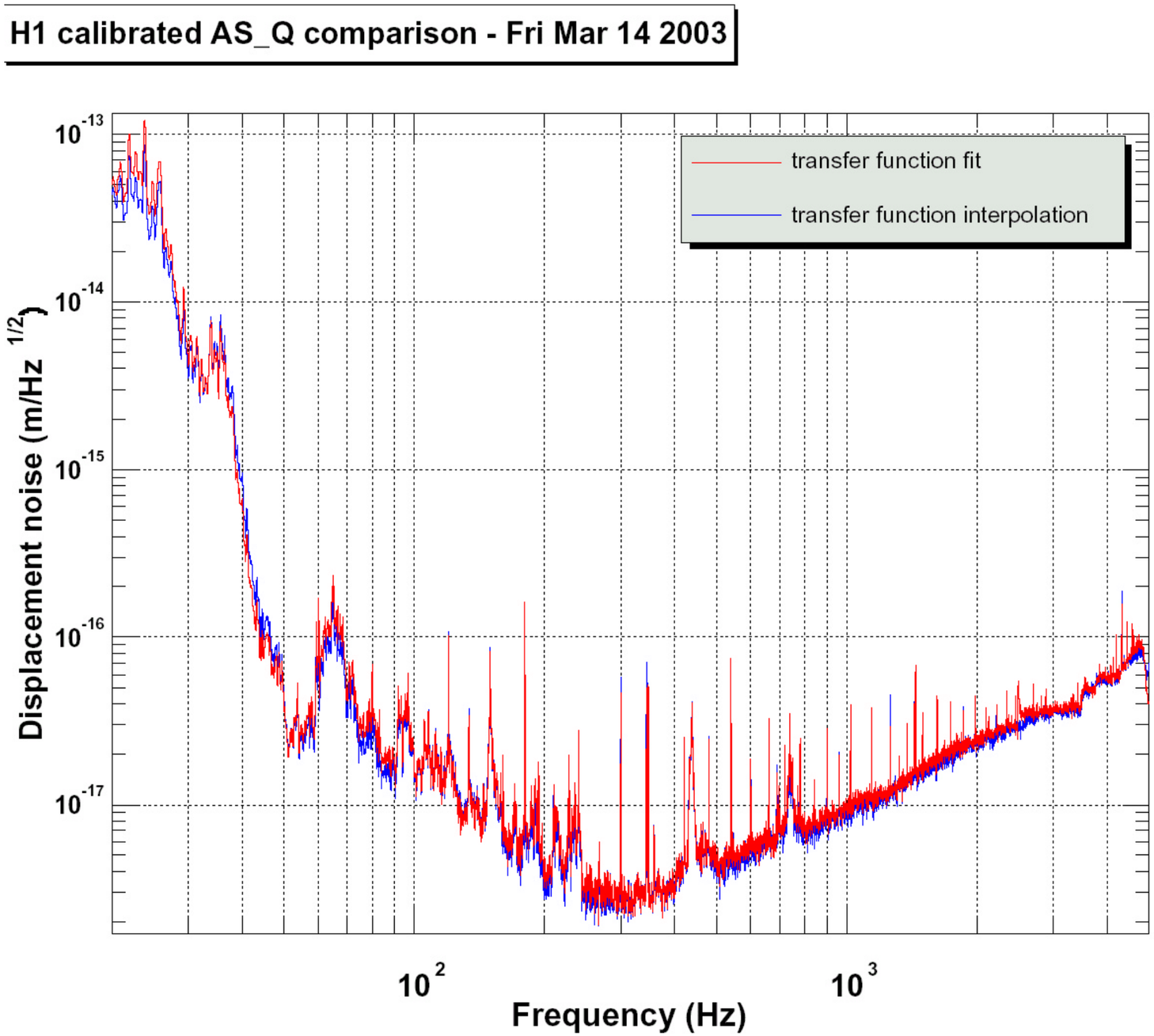
Transfer function fits and dtt records



Autocalibration G030092-00W

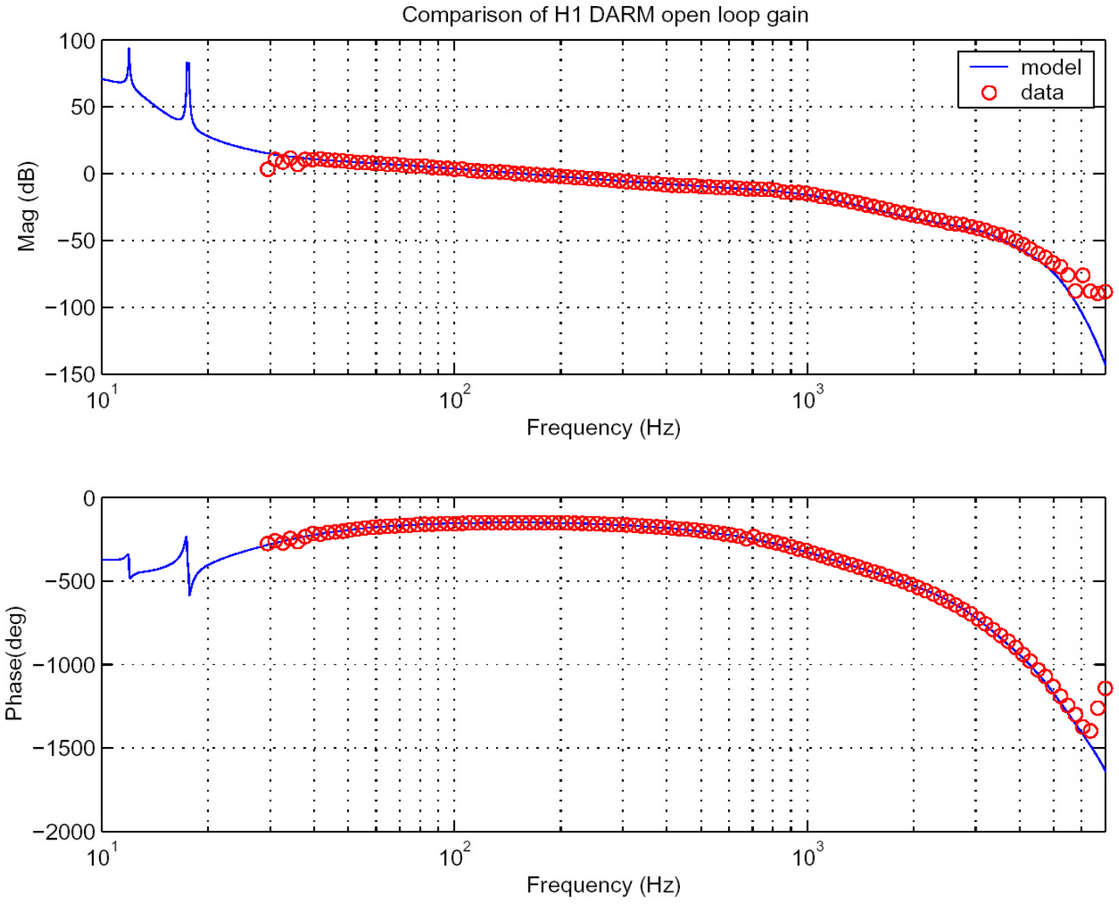
```
<LIGO_LW Name="Calibration[0]">
  <Param Name="Channel" Type="string">H1:LSC-AS_Q</Param>
  <Time Type="GPS">731714272</Time>
  <Param Name="Reference" Type="string">check</Param>
  <Param Name="Unit" Type="string">m</Param>
  <Param Name="Gain" Type="double">1.926178e-10</Param>
  <Param Name="Poles" Type="doubleComplex" Dim="18">
262.757317 5846.466145
262.757317 -5846.466145
1883.352262 3089.598291
1883.352262 -3089.598291
-123.148589 3741.662130
-123.148589 -3741.662130
-942.362939 3496.954352
-942.362939 -3496.954352
2049.448491 0.000000
4.345046 2491.244582
4.345046 -2491.244582
-1696.391788 0.000000
-167.558718 1244.442055
-167.558718 -1244.442055
-151.343554 0.000000
25.335462 0.000000
0.076000 0.760000
0.076000 -0.760000
</Param>
  <Param Name="Zeros" Type="doubleComplex" Dim="20">
-0.066280 -5140.037721
-0.066280 5140.037721
437.348413 -4179.824070
437.348413 4179.824070
101.388529 -3724.825689
101.388529 3724.825689
458.252756 -3137.264643
458.252756 3137.264643
-5.934789 -2488.112243
-5.934789 2488.112243
649.993263 -2013.739651
649.993263 2013.739651
147.856315 -1261.558605
147.856315 1261.558605
154.182601 -310.014768
154.182601 310.014768
42.762852 -119.289024
42.762852 119.289024
7.743098 -40.174473
7.743098 40.174473
</Param>
  <Param Name="Default" Type="boolean">1</Param>
</LIGO_LW>
```

Comparison of fitted Transfer function Method to interpolation



Autocalibration G030092-00W

Upcoming: model-based autocal



Fritschel model

```
function [openloopgain,sensing,calASQ] = DARMmodel(darmin)
% DARMmodel   %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Evaluates a model of the interferometer DARM feedback loop %%
% Usage:                                                      %%
%         >> darm = DARMparams;                               %%
%         >> [olg,sense,calibasq] = DARMmodel(darm);          %%
%                                                             %%
%       Returns the open loop gain (OLG, unitless) and the    %%
%       sensing function (SENSE, in counts/strain), each in   %%
%       a 3 column array, where the first column is the       %%
%       frequency vector, the second column is the amplitude, %%
%       and the third column is the phase (in rads)           %%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
global darm
darm = darmin;

fl = darm.fl;
fu = darm.fu;
fs = darm.fs;
%ff = logspace(log10(fl),log10(fu),darm.npt);
ff = linspace(fl,fu,darm.npt);
ugf = darm.ugf;
iugf = min(find(ff>ugf));

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%             Analog portion of loop                %%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

fc = darm.cavpole;          % cavity pole
cavpole = zpk([],-2*pi*fc,2*pi*fc);
etmpend = pendulum(darm);

%%%%%%%%%%%      AA & AI filtering   %%%%%%%%%%%%%%%
[z,p,k] = ellip(5,4,60,2*pi*7570,'s');
ai = zpk(z,p,k);
```

-
-
-

Summary

- Quasi-independent check on full calibration procedure
- Quick and mostly painless, performed daily
- Have to understand the deviations at high frequency
- Commissioning tool and static cal: transfer function fits
- Next: extremely fast calibration using unity gain finder/model