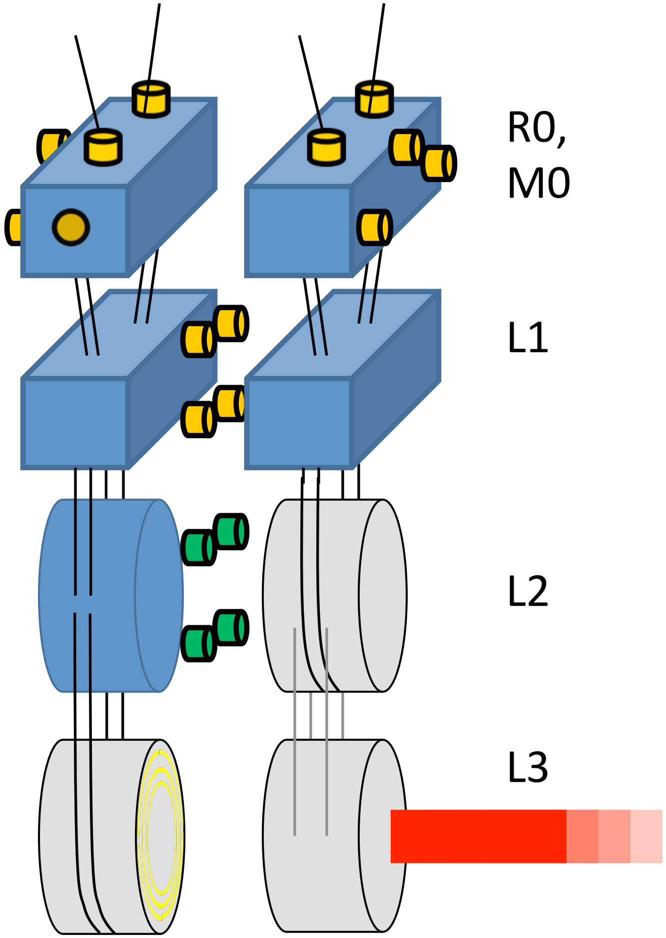
Suspension Control with Thoughts on Modern Control

Brett Shapiro

19 May 2015

Can modern control be useful for the suspension systems?

Quadruple Suspension (Quad)



Purpose
- Input Test Mass (ITM, TCP)
- End Test Mass (ETM, ERM)

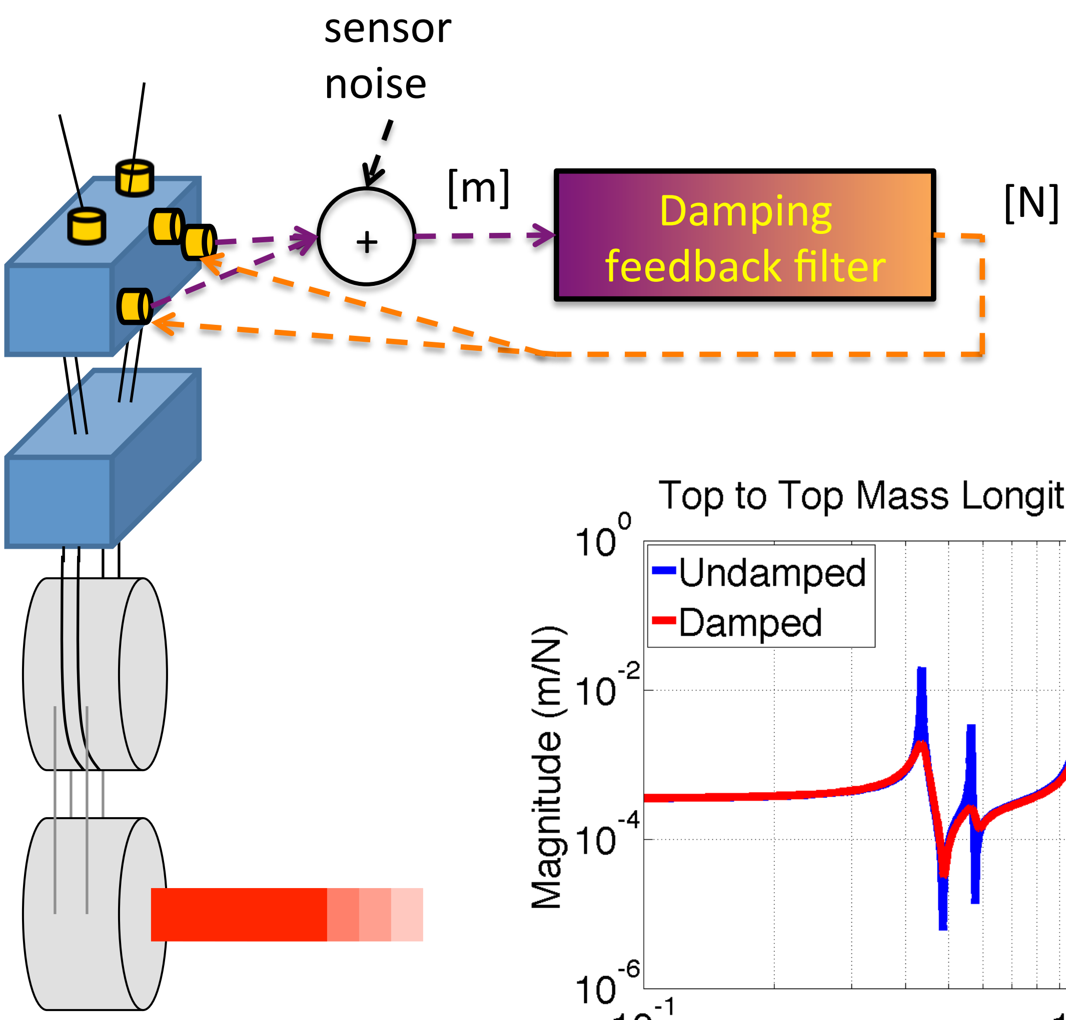
Location
- End Test Masses, Input Test Masses

Control
- Local – damping at M0, R0
- Global – LSC & ASC at all 4

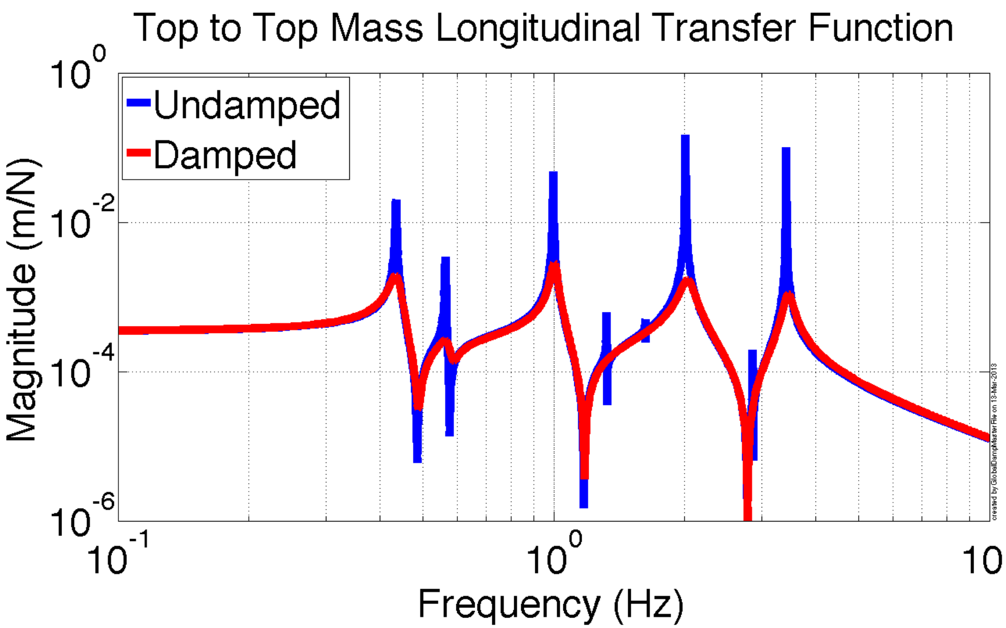
Sensors/Actuators
- BOSEMs at M0, R0, L1
- AOSEMs at L2
- Optical levers and interf. sigs. at L3
- Electrostatic drive (ESD) at L3

Documentation
- Final design review - T1000286
- Controls arrang. – E1000617

Suspension damping feedback

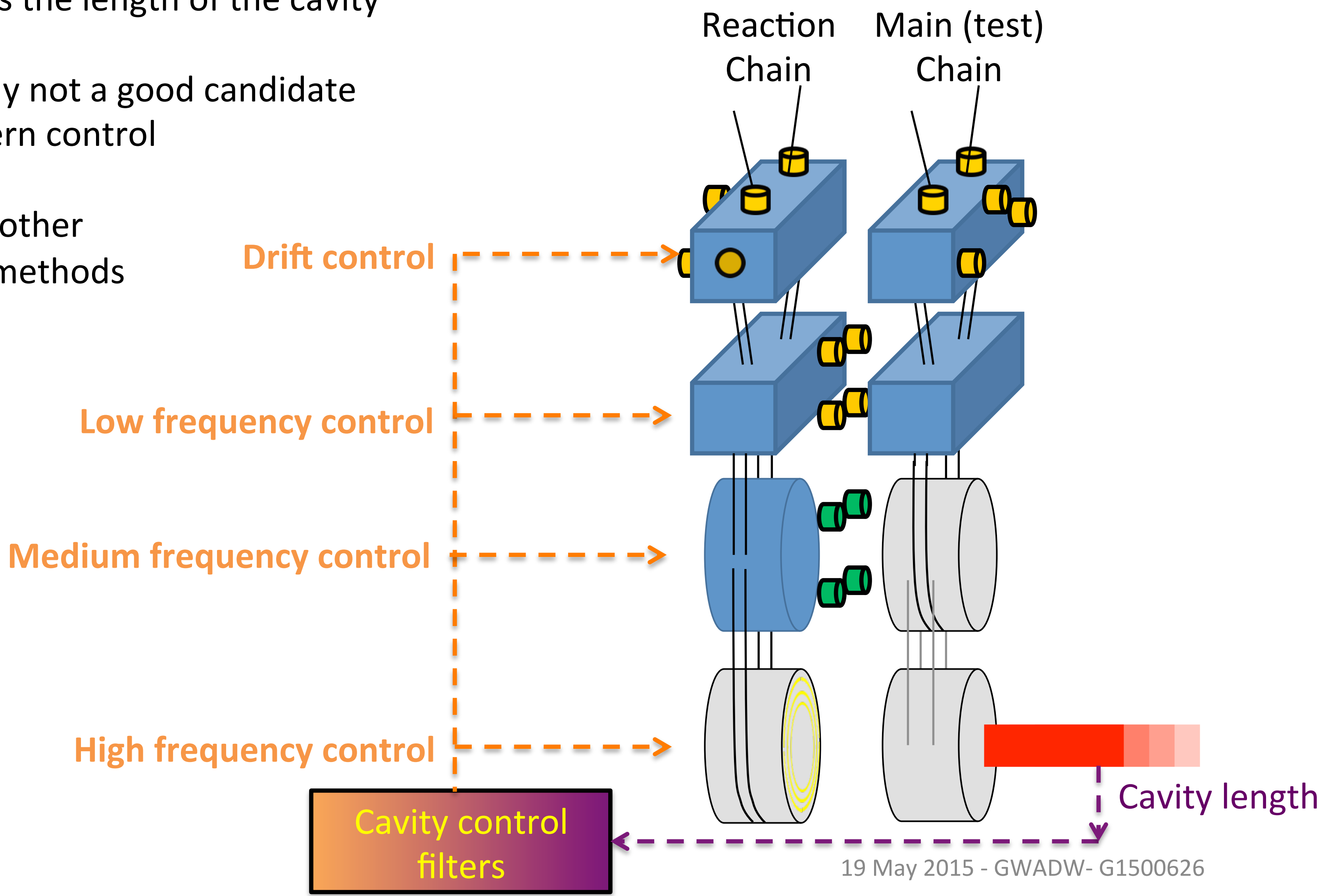


- Used to reduce the Qs of the suspension modes
- Good candidate for modern control



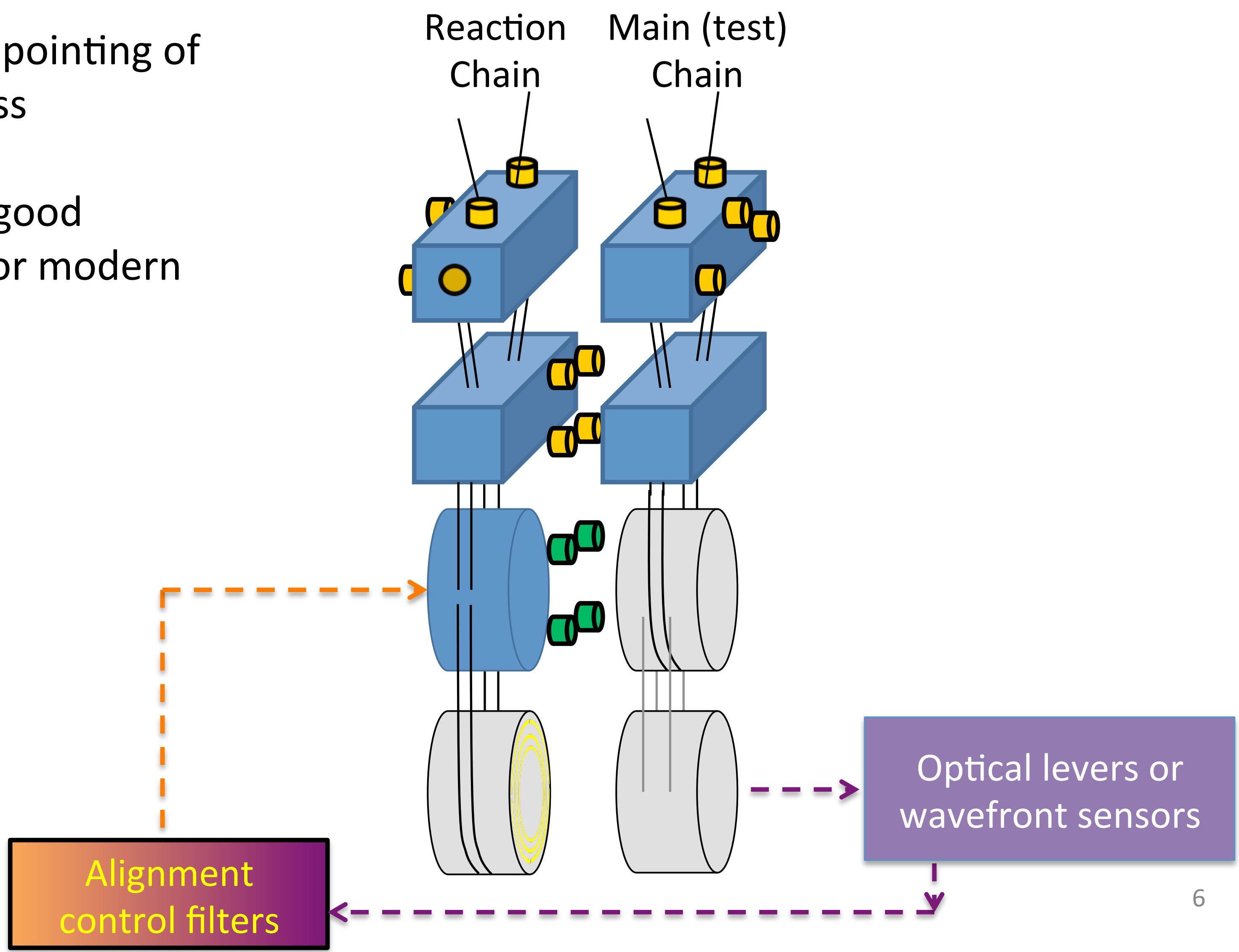
Suspension cavity control

- Controls the length of the cavity
- Probably not a good candidate for modern control
- Maybe other optimal methods

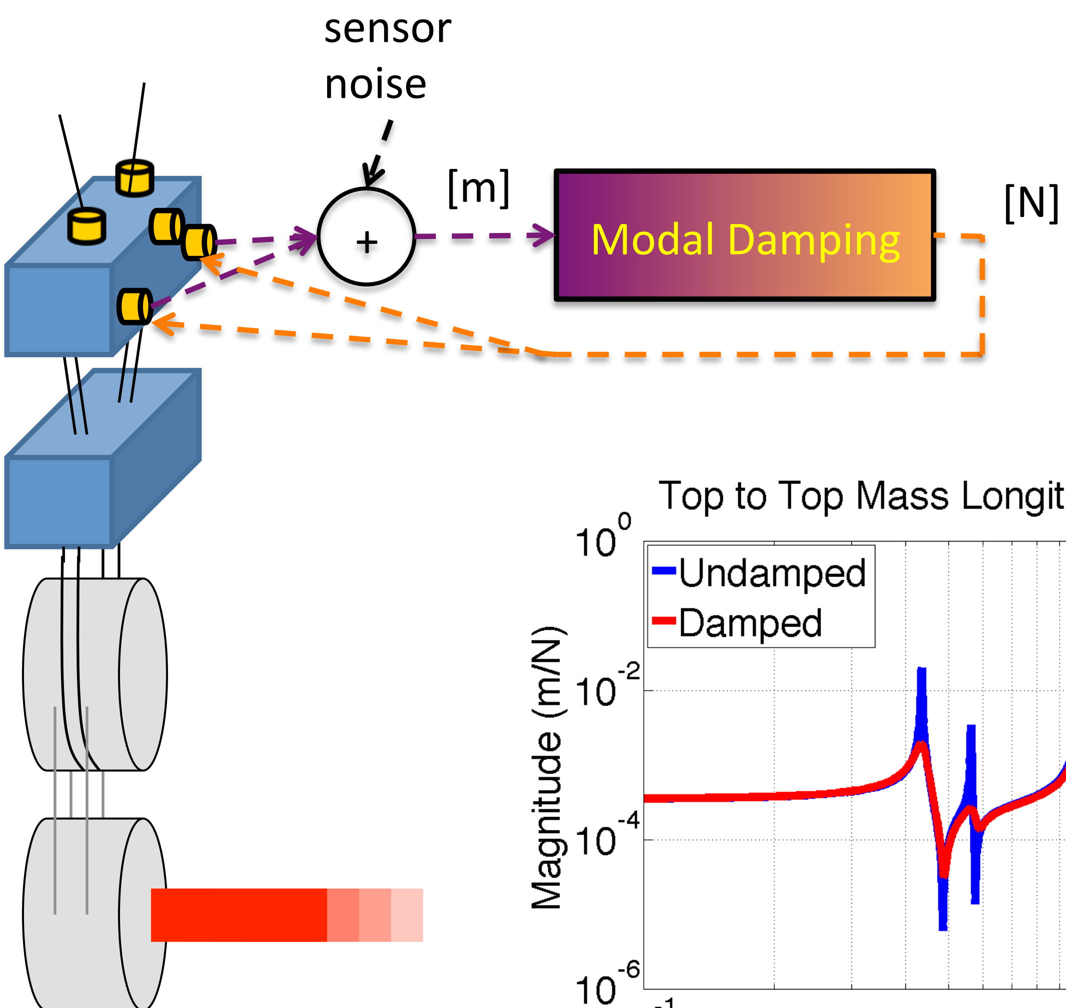


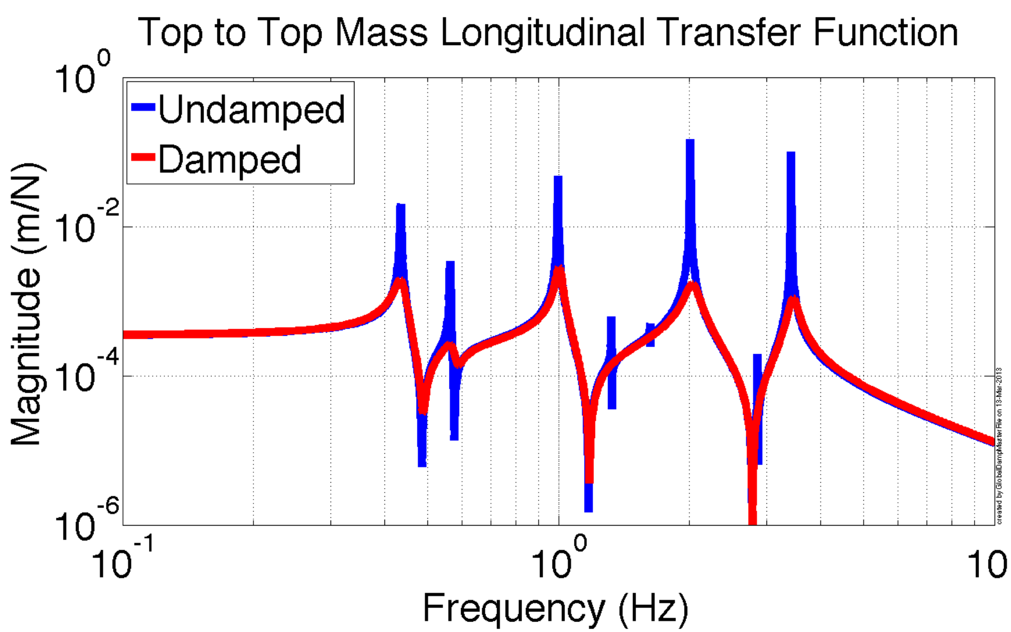
Suspension angular control

- Control the pointing of the test mass
- Might be a good candidate for modern control

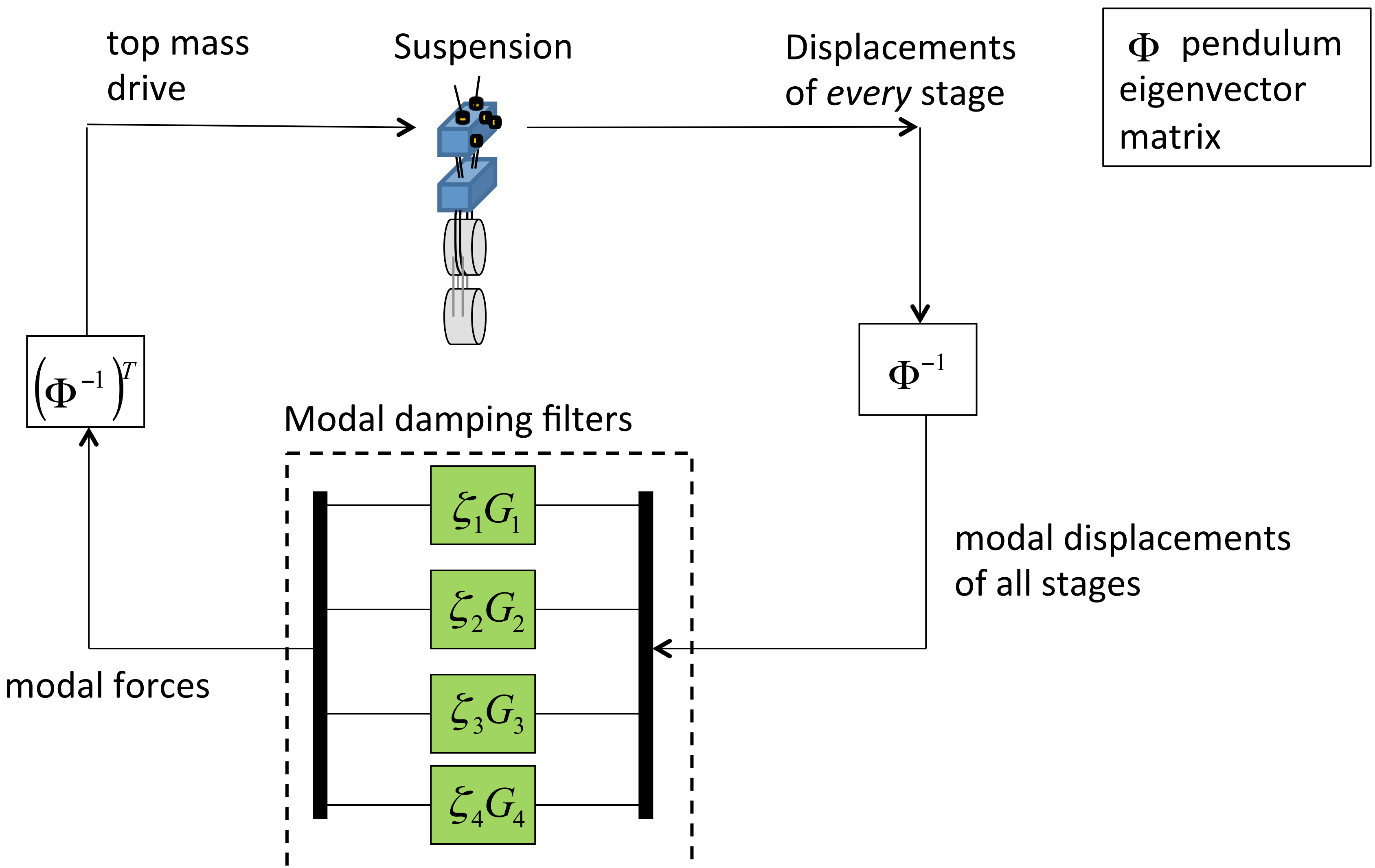


Modern Control Example - Damping

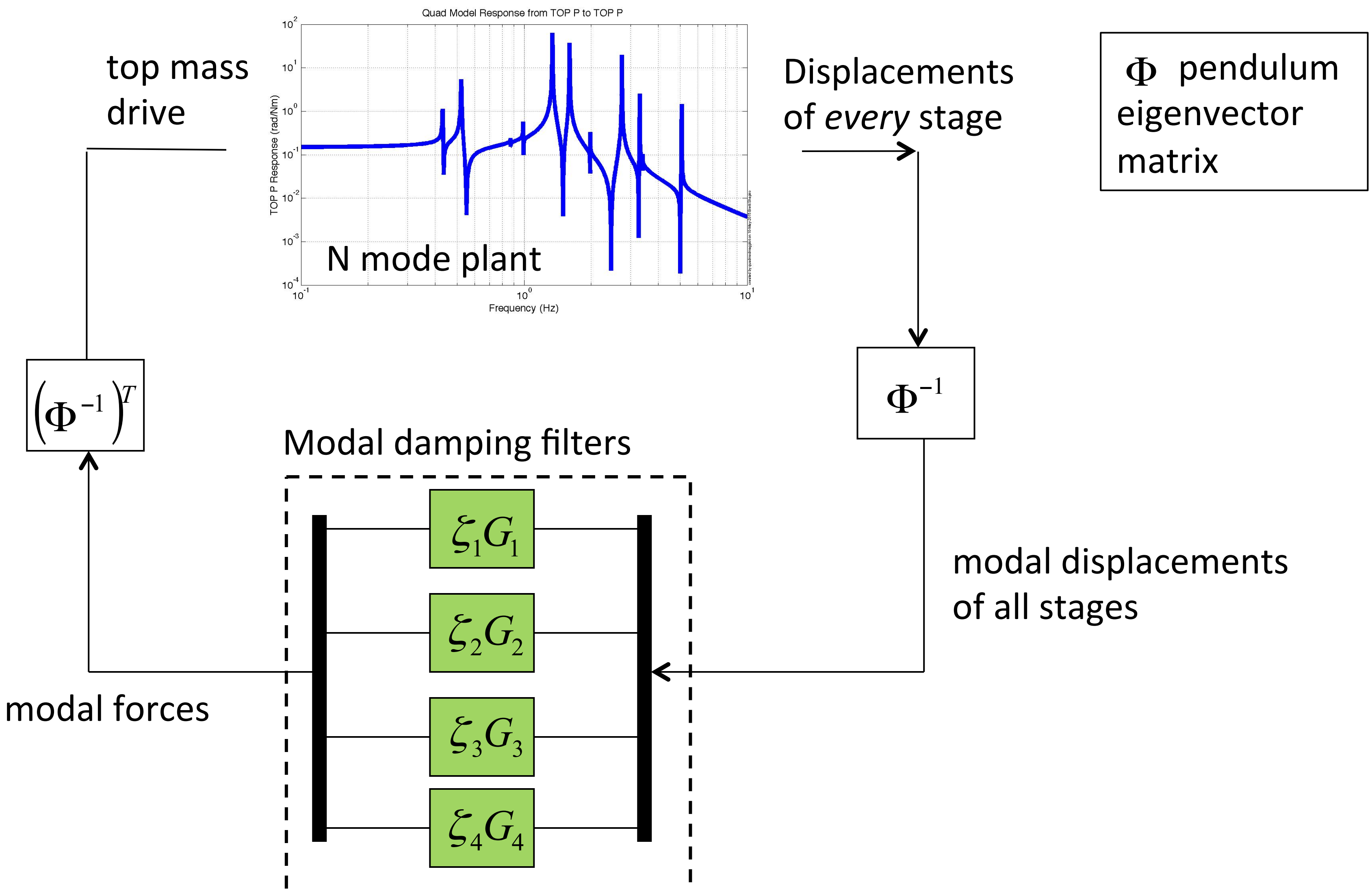




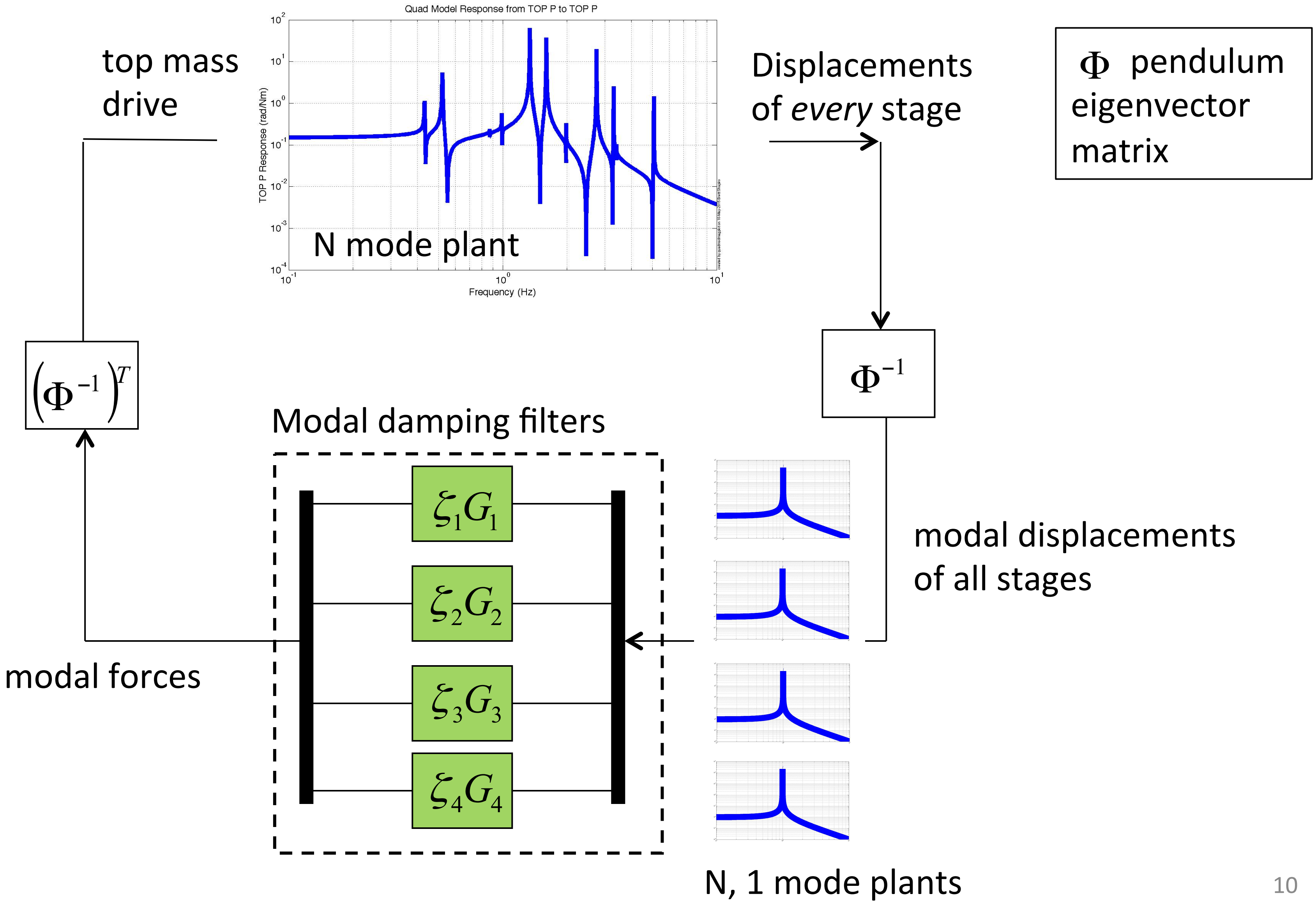
Modal Damping with State Estimation



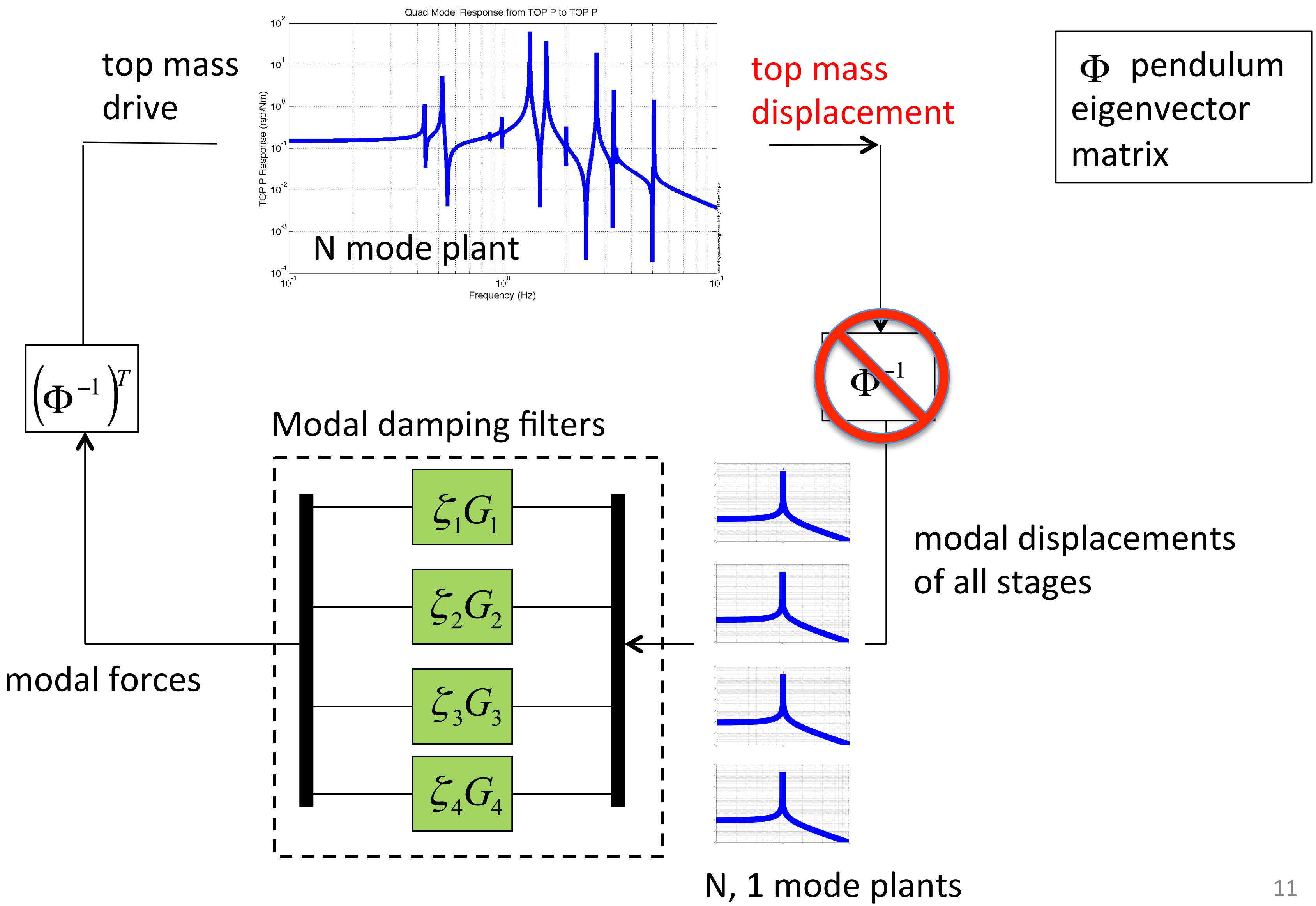
Modal Damping with State Estimation



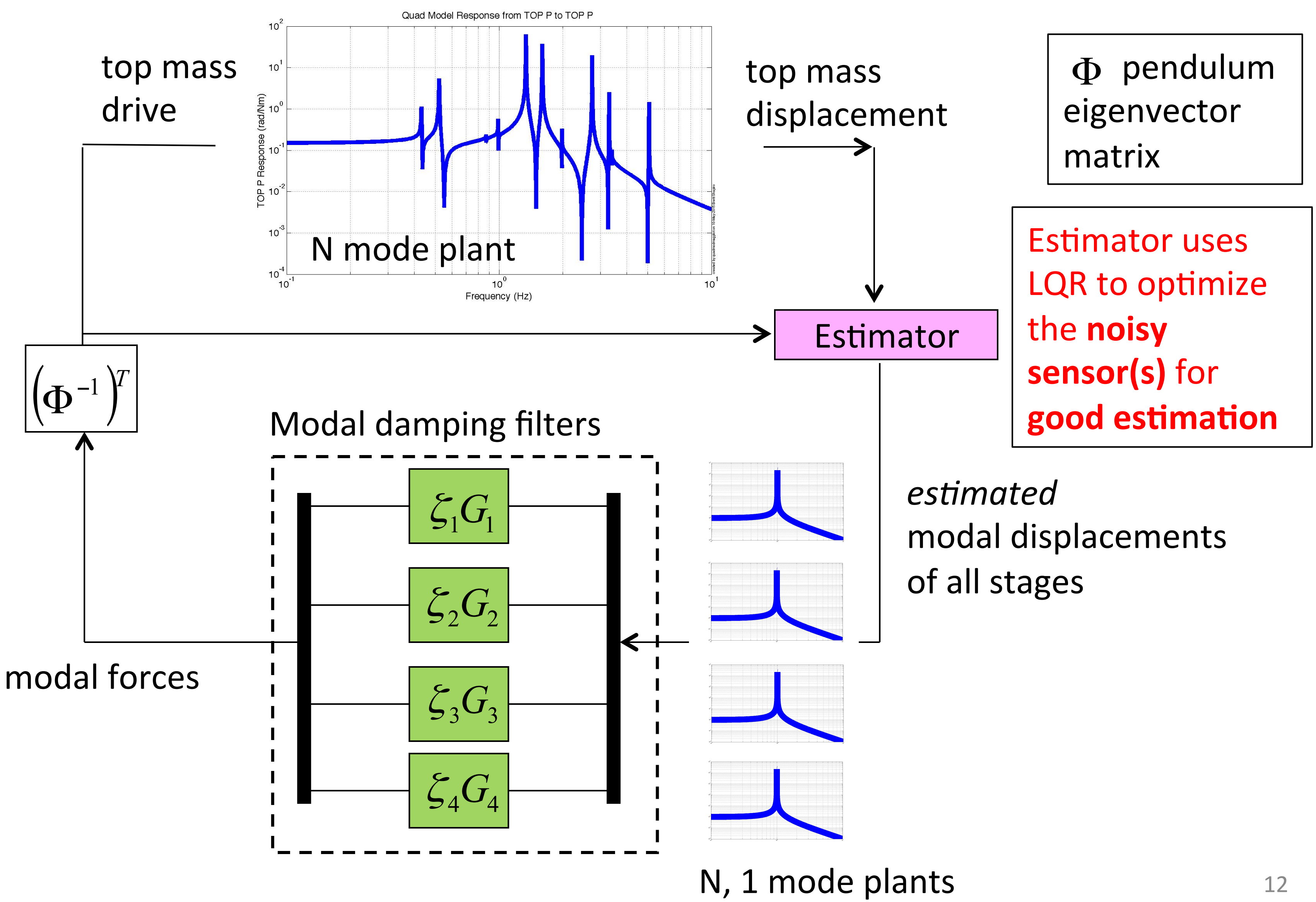
Modal Damping with State Estimation



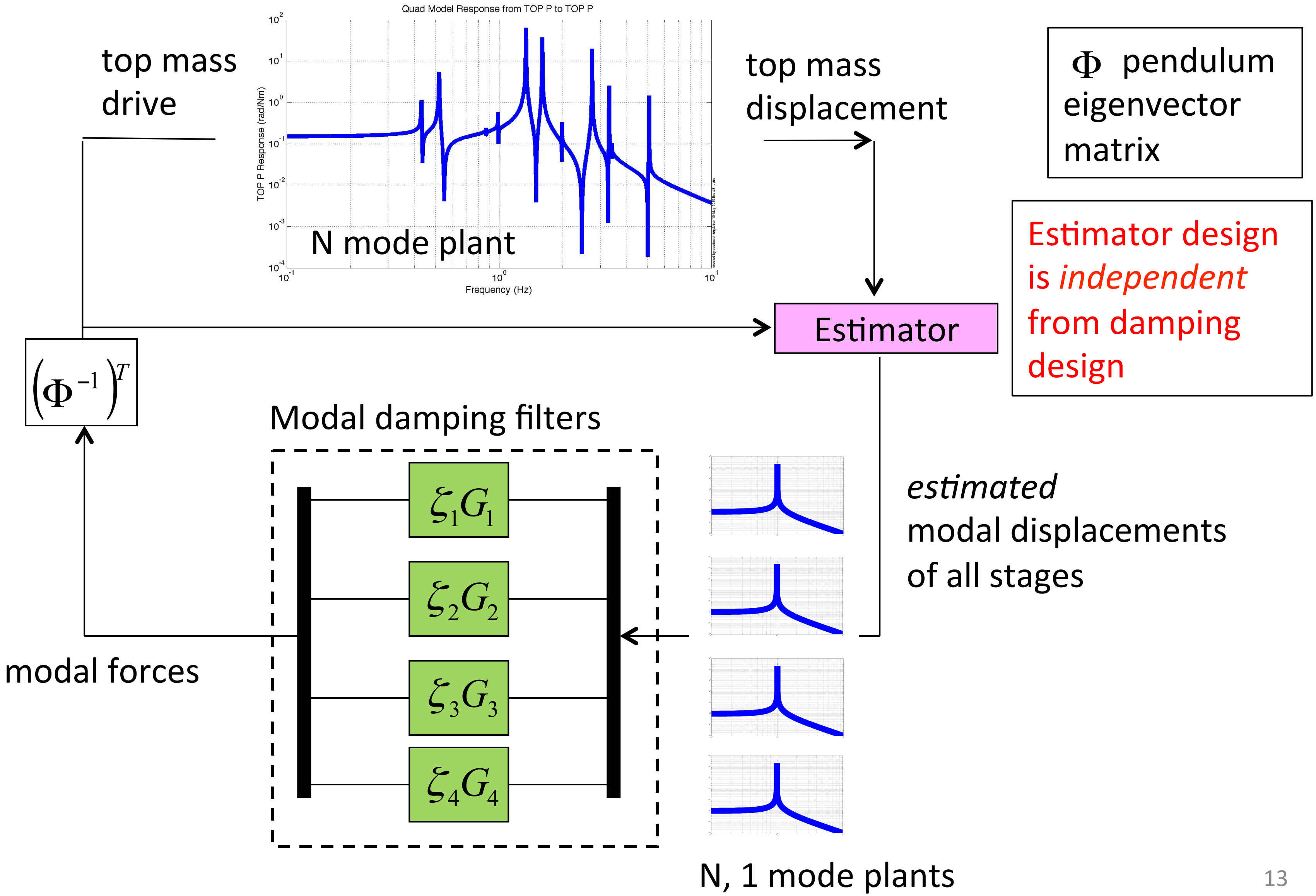
Modal Damping with State Estimation



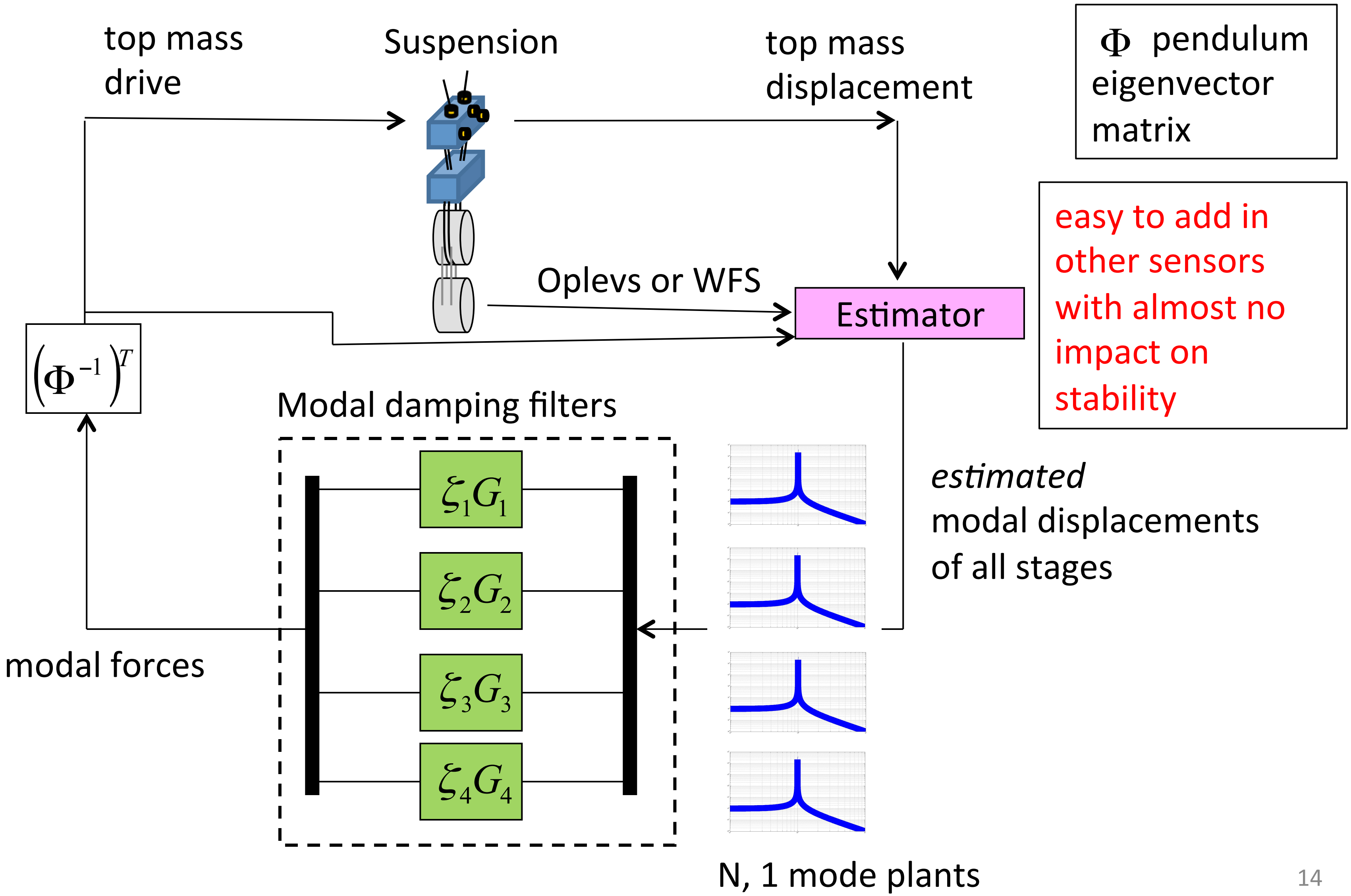
Modal Damping with State Estimation



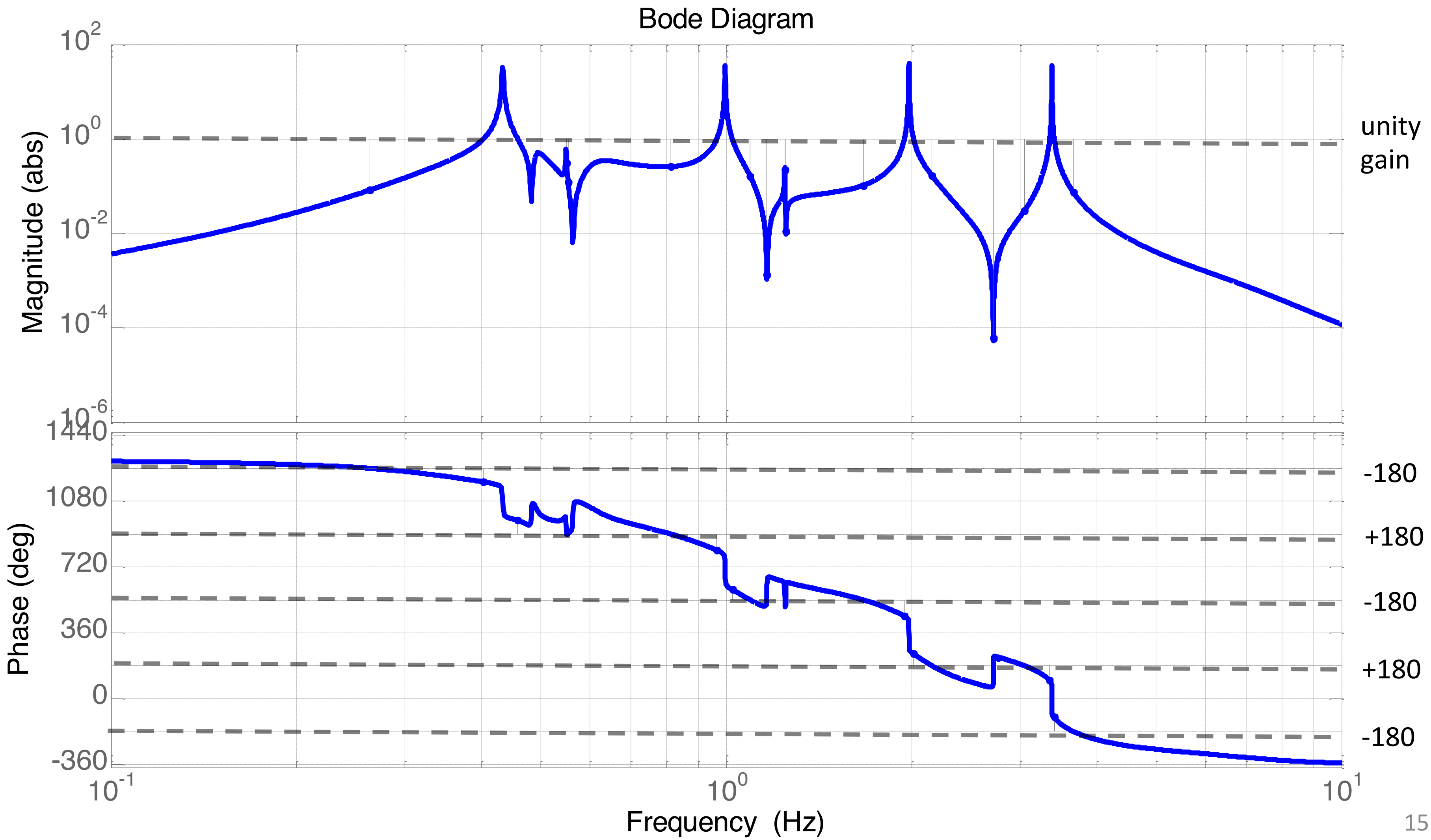
Modal Damping with State Estimation



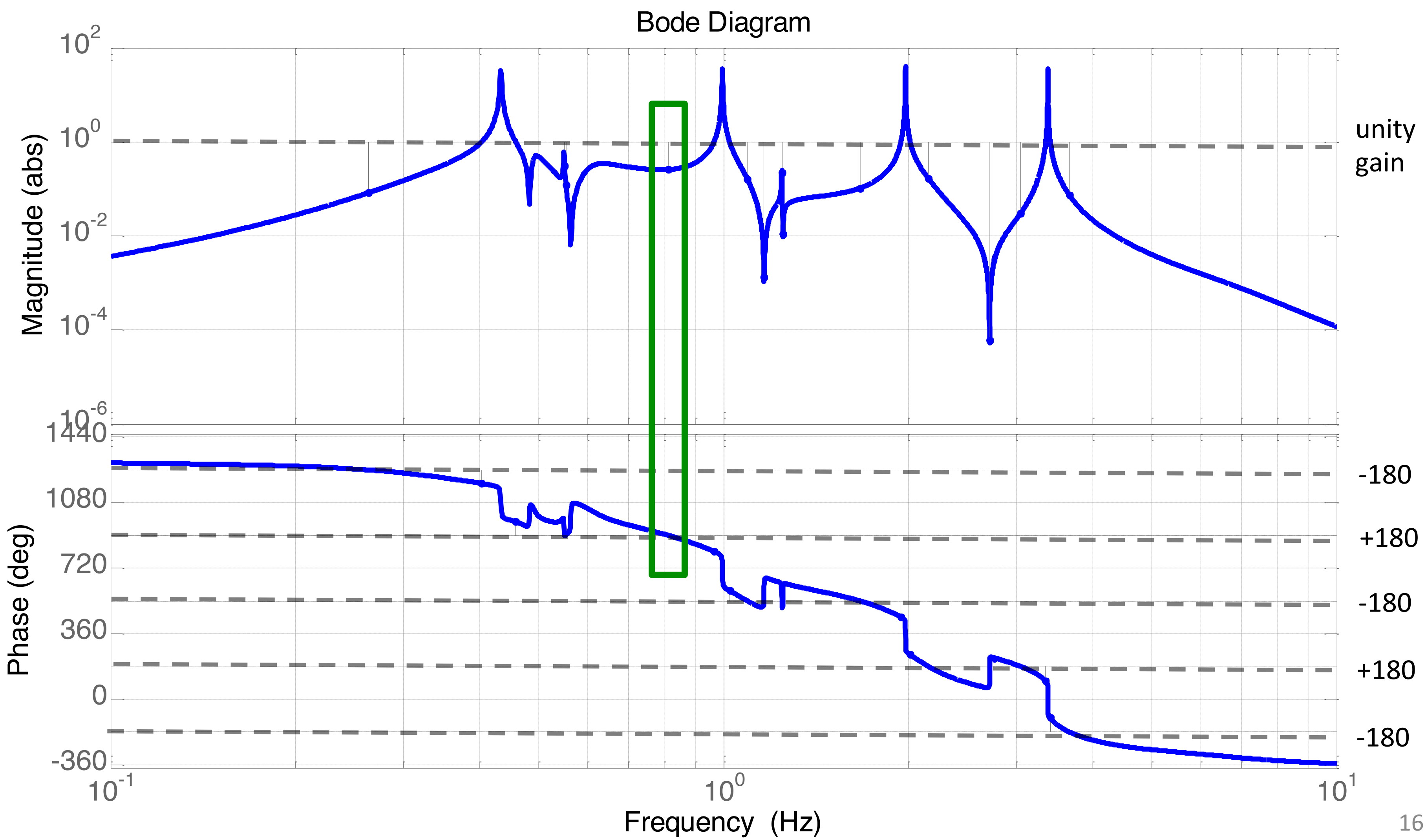
Modal Damping with State Estimation



Modal Damping Loop Gain – This is Stable!?



Modal Damping Loop Gain – This is Stable!?



Humans (loop shaping)

✓

- Distinguishing good and bad results
- Adjusting to the unexpected
- ‘human readable’ designs

✗

- MIMO systems with many coupled loops
- Have individual biases and strategies

Modern controls

✓

- MIMO systems
- Not subject to human biases and strategies
- Quantitatively ‘optimal’ solutions

✗

- ‘Optimal’ only in the context of the cost function
- Unmodeled dynamics
- ‘human readable’ designs

Conclusion – every job has its tool

Back Ups

Concerns I often hear about Modern control, and my responses to those concerns

This is just a first attempt to answer these concerns. I think we should agree on the responses as a group, citing examples. By doing so we'll be well on our way to exploiting the best each technique has to offer.

Concerns often heard about Modern Control (MC)

1. We're just moving the complexity of the design to the cost function.

2. It is hard to predict how a change in the cost will impact the loop.

3. The cost function does not permit the designer to include as much intuition as classical techniques like loop shaping.

4. We don't know what optimal really means so we end up just turning the knobs of the cost function to get a result the seems OK.

5. Stability is not guaranteed in practice.

6. Only a few people know how to use it.

7. We'll lose understanding of our own control systems. Too much complexity is buried in algorithms.

My responses to those concerns

1. **“We’re just moving the complexity of the design to the cost function.”**

   In some cases yes. But there are cases where MC simplifies the problem. Modal damping is an example, which is actually a mix of MC and traditional loop shaping. The MC decouples the system into independent modes, which are then controlled using traditional loop shaping. In other cases, MC might be no more or less complex, yet provide better results. In general, keeping the cost function simple helps a lot.

2. **“It is hard to predict how a change in the cost will impact the loop.”**

   Can be yes. Using simple cost functions makes it easier to predict what will happen. If the cost function needs to be so complicated that it is disconnected from the result, then MC may not be the right tool for the problem.

3. **“The cost function does not permit the designer to include as much intuition as classical techniques like loop shaping.”**

   Adding some frequency dependence helps here. Keeping it simple also helps preserve intuition. If the frequency domain shape needs to very complicated, or frequency weighting is required for more than 2 or 3 loops, MC might not be the best choice (at least on its own).

My responses to those concerns

4. **“We don’t know what optimal really means so we end up just turning the knobs of the cost function to get a result the seems OK.”**

   As before, keeping it simple helps preserve intuition. Also, sometimes it works well simply to scan through the space of cost function values and pick the ones that work best. Modal damping was implemented in this way, by writing an ‘outer’ cost function that optimized the MC cost function with things we actually care about like DARM noise and Q values. Might it seem silly to have a cost function for a cost function, but it does work very well in this case.

5. **“Stability is not guaranteed in practice.”**

   Stability is an issue with any feedback method (even feedforward sometimes). If you start with a good model in the modern controller, any adjustments needed to ensure stability will be minor.

My responses to those concerns

6. **"Only a few people know how to use it."**

   I think this and the next points are the most serious concerns. My response for this one is that if we want our systems to work at peak performance, we need to make all the tools available to us. People who work on high performance systems should make the effort to become familiar with MC methods. In most cases MATLAB makes them pretty easy to use. A deep understanding of the (usually difficult) math behind them isn't necessary for effective use.

7. **"We'll lose understanding of our own control systems. Too much complexity is buried in algorithms."**

   This is legitimately something too watch out for. In some cases MC can simplify the control without losing much understanding of what's going on, as in modal damping. In other cases MC can obscure the physics of what's going on. In those cases we should use loop shaping methods, unless MC solves some essential problem loop shaping can't. Thus, the point made earlier in these slides to use the right tool in the right place.

Summarizing what Modern Control is good and bad at and when it is useful for us

Summarizing what is modern control good at

- MIMO systems
  - Will utilize many signals seamlessly and *take advantage* of built in couplings
- Generating *theoretically* stable solutions
  - Even for unstable systems (Para. Inst. perhaps?)
- Very simple to use software, e.g. MATLAB
- Not being subject to human biases

Summarizing what is modern control bad at?

- 'Optimal' is only in the context of the cost function (inputs vs outputs).
  - Loss of human intuition
  - No nonlinear effects like actuation limits
- Stability requires very good models
  - Will not handle unmodeled dynamics at all, e.g. bounce and violin modes
- The controller is no more complex than the plant
  - Though you can augment the plant with any frequency dependence in the cost functions

So when is modern control useful?

- When the usual methods aren't meeting the requirements

and /or

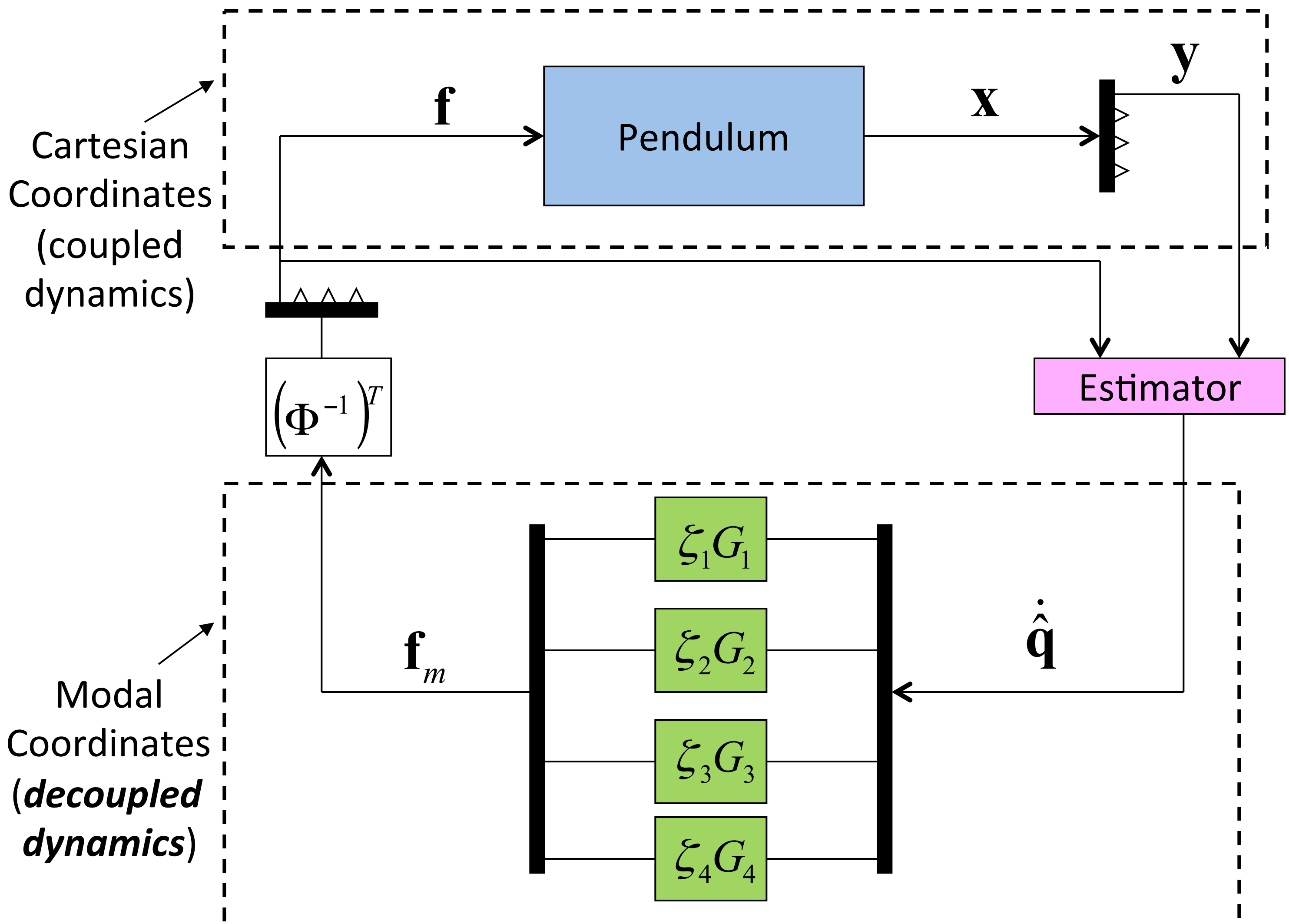
- When MC can simplify a MIMO system

and (in all cases)

- When the cost functions are simple

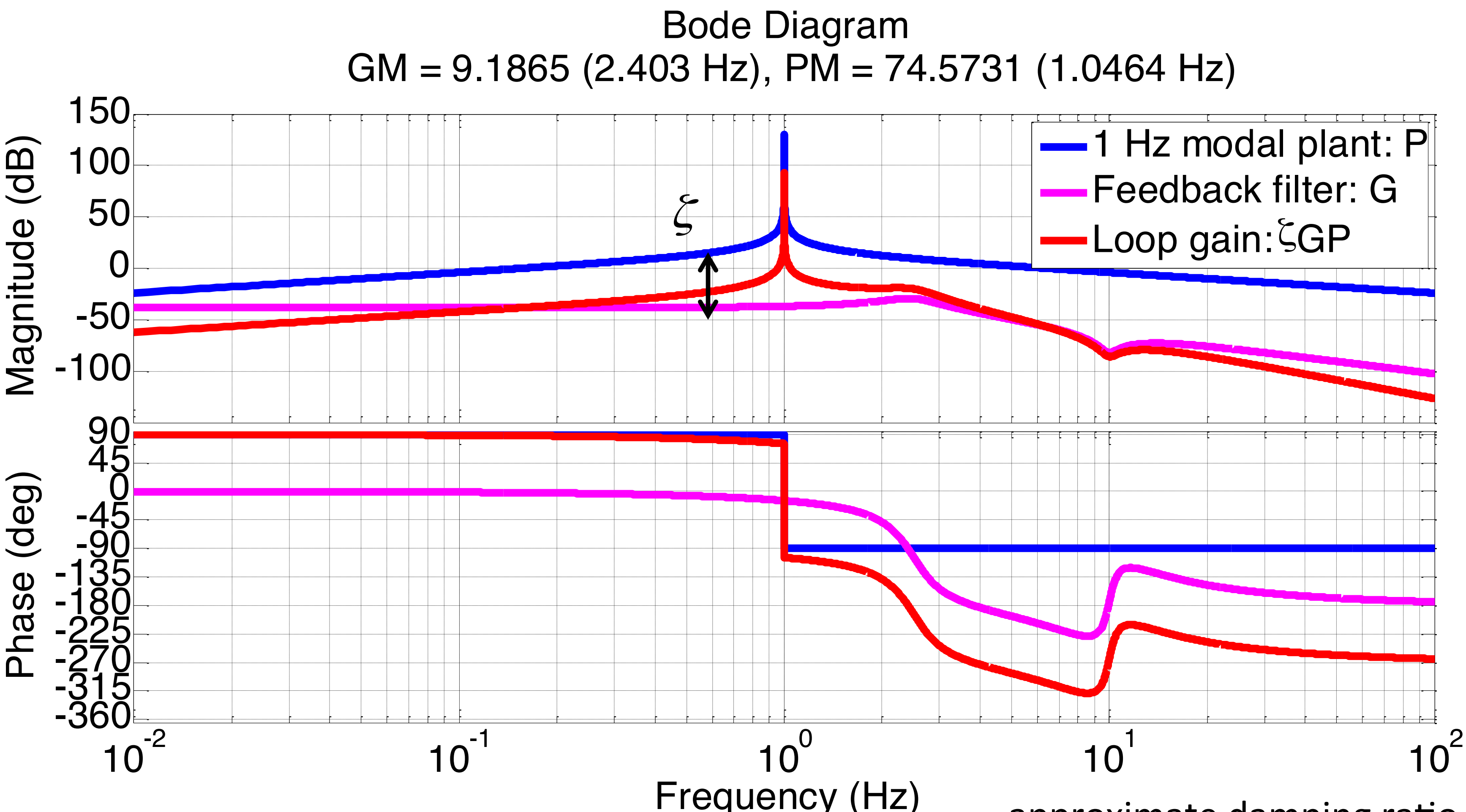
Modal Damping Notes

Modal Damping with State Estimation



- $\mathbf{x} = \Phi \mathbf{q}$
- Φ = pendulum eigenvector matrix
- $\mathbf{x}$ = Cartesian coordinates
- $\mathbf{y}$ = sensor sig.
- $\hat{\mathbf{q}}$ = estimated modal coord.
- $\mathbf{f}$ = Cartesian damping forces
- $\mathbf{f}_m$ = modal damping forces
- $G_i = i_{\text{th}}$ mode damping filter
- $\zeta_i = i_{\text{th}}$ mode damping gain

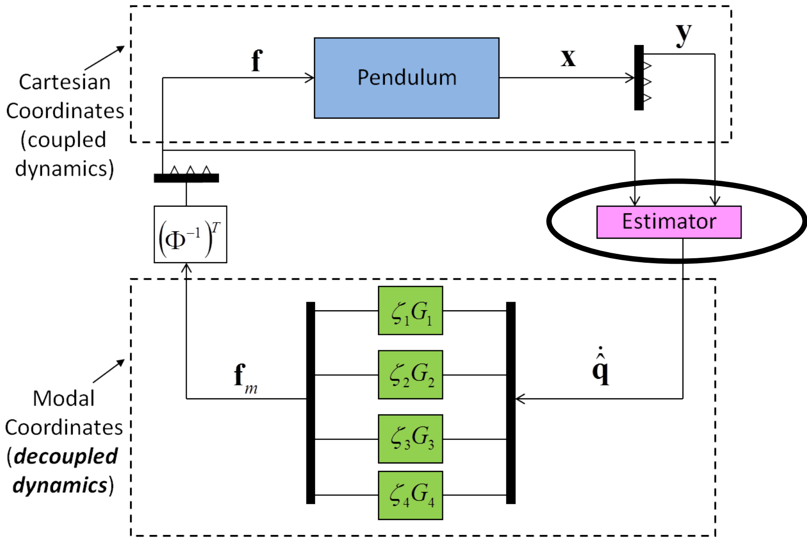
Modal Feedback Design



approximate damping ratio

$$G = \frac{s^2 + 2\pi s + (20\pi)^2}{\left[s^2 + 5\sin(10^\circ)\omega_n s + (2.5\omega_n)^2\right]\left[s^2 + 20\pi s + (20\pi)^2\right]}, \quad 0 \le \zeta < 1, \quad P = \frac{s}{s^2 + \omega_n^2}, \quad \omega_n = 2\pi$$

Estimator Design



$$\begin{bmatrix} \dot{\hat{\mathbf{q}}} \\ \ddot{\hat{\mathbf{q}}} \end{bmatrix} = \mathbf{A}_m \begin{bmatrix} \hat{\mathbf{q}} \\ \dot{\hat{\mathbf{q}}} \end{bmatrix} + \mathbf{B}_m \begin{bmatrix} \mathbf{f} \\ 0 \end{bmatrix} - \mathbf{L}_m \left(\mathbf{C}_m \begin{bmatrix} \hat{\mathbf{q}} \\ \dot{\hat{\mathbf{q}}} \end{bmatrix} - \mathbf{y} \right)$$

$\mathbf{A}_m, \mathbf{B}_m \rightarrow$ Pendulum model

$\mathbf{C}_\mathrm{m} \rightarrow$ Sensor matrix

$\mathbf{L}_\mathrm{m} \rightarrow$ Estimator feedback matrix

Linear Quadratic Regulator (LQR) design

$$J = \int_0^\infty \left(\begin{bmatrix} \tilde{\mathbf{q}}^T & \dot{\tilde{\mathbf{q}}}^T \end{bmatrix} \mathbf{Q} \begin{bmatrix} \tilde{\mathbf{q}} \\ \dot{\tilde{\mathbf{q}}} \end{bmatrix} + \mathbf{z}_m^T \mathbf{R} \mathbf{z}_m \right) dt$$

$$\mathbf{L}_m = \underset{\mathbf{L}_m}{\arg\min}(J)$$

$\tilde{\mathbf{q}} = \hat{\mathbf{q}} - \mathbf{q} =$ estimation error

$\mathbf{z}_m = -\mathbf{L}_m^T \begin{bmatrix} \tilde{\mathbf{q}} \\ \dot{\tilde{\mathbf{q}}} \end{bmatrix} \rightarrow$ sensor noise amplification

?

Q, R

Weight cost of estimation error

Weight cost of using noisy sensor

Choosing Q and R: Not Unique

$$J = \int_0^\infty \left(\begin{bmatrix} \tilde{\mathbf{q}}^T & \dot{\tilde{\mathbf{q}}}^T \end{bmatrix} \mathbf{Q} \begin{bmatrix} \tilde{\mathbf{q}} \\ \dot{\tilde{\mathbf{q}}} \end{bmatrix} + \mathbf{z}_m^T \mathbf{R} \mathbf{z}_m \right) dt$$

$$\mathbf{L}_m = \arg\min_{\mathbf{L}_m}(J)$$

$\mathbf{Q}$

$$\begin{bmatrix} \tilde{q}_1^T & \cdots & \dot{\tilde{q}}_{n-1}^T & \dot{\tilde{q}}_n^T \end{bmatrix} \begin{bmatrix} 0 & & & 0 \\ & \cdots & & \\ & & m_{n-1}^{-2} & \\ 0 & & & m_n^{-2} \end{bmatrix} \begin{bmatrix} \tilde{q}_1 \\ \cdots \\ \dot{\tilde{q}}_{n-1} \\ \dot{\tilde{q}}_n \end{bmatrix}$$

$$\mathbf{R} = \begin{bmatrix} R_1 & & & 0 \\ & R_2 & & \\ & & \cdots & \\ 0 & & & R_m \end{bmatrix}$$

m_i = modal mass of mode i

m_i^{-1} = modal velocity impulse response amplitude

R is still to be determined

Solving the R matrix for MIMO Modal Damping

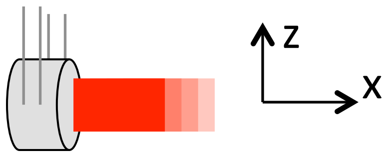
Try a bunch of R matrices and see what works best

$$J_R(R) = \max_i \left(T_{s,i}^2\right) + \max_i \left(N_i^2\right)$$
$$R = \arg\min(J_R)$$

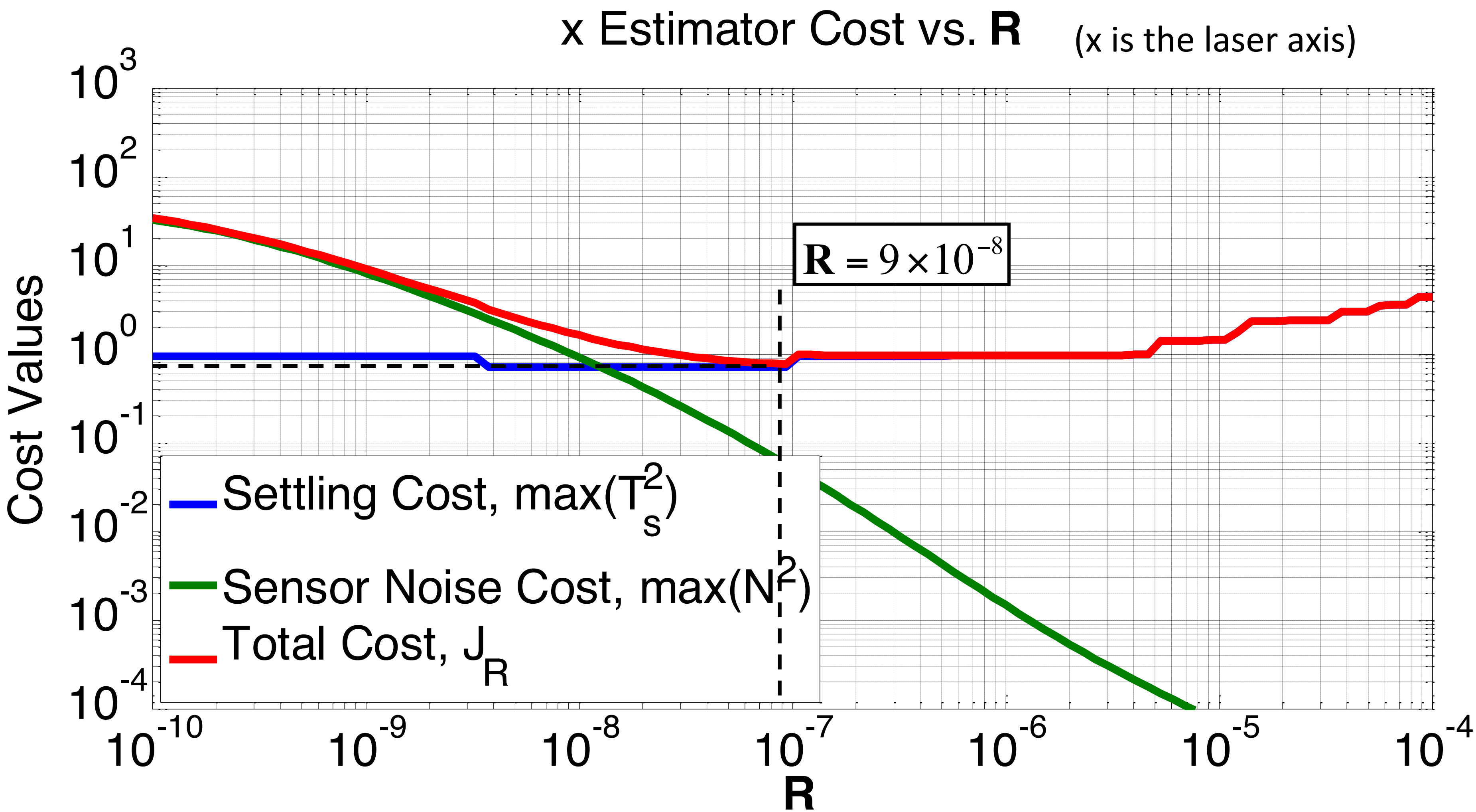
Measure 'best' with an auxiliary cost function.

- $T_{s,i} = \dfrac{\text{Stage 4 settling time for DOF } i}{\text{10 seconds}}$
- $N_i = \dfrac{\text{Stage 4 sensor noise for DOF } i \text{ at 10 Hz}}{\text{Stage 4 noise requirement for DOF } i \text{ at 10 Hz}}$

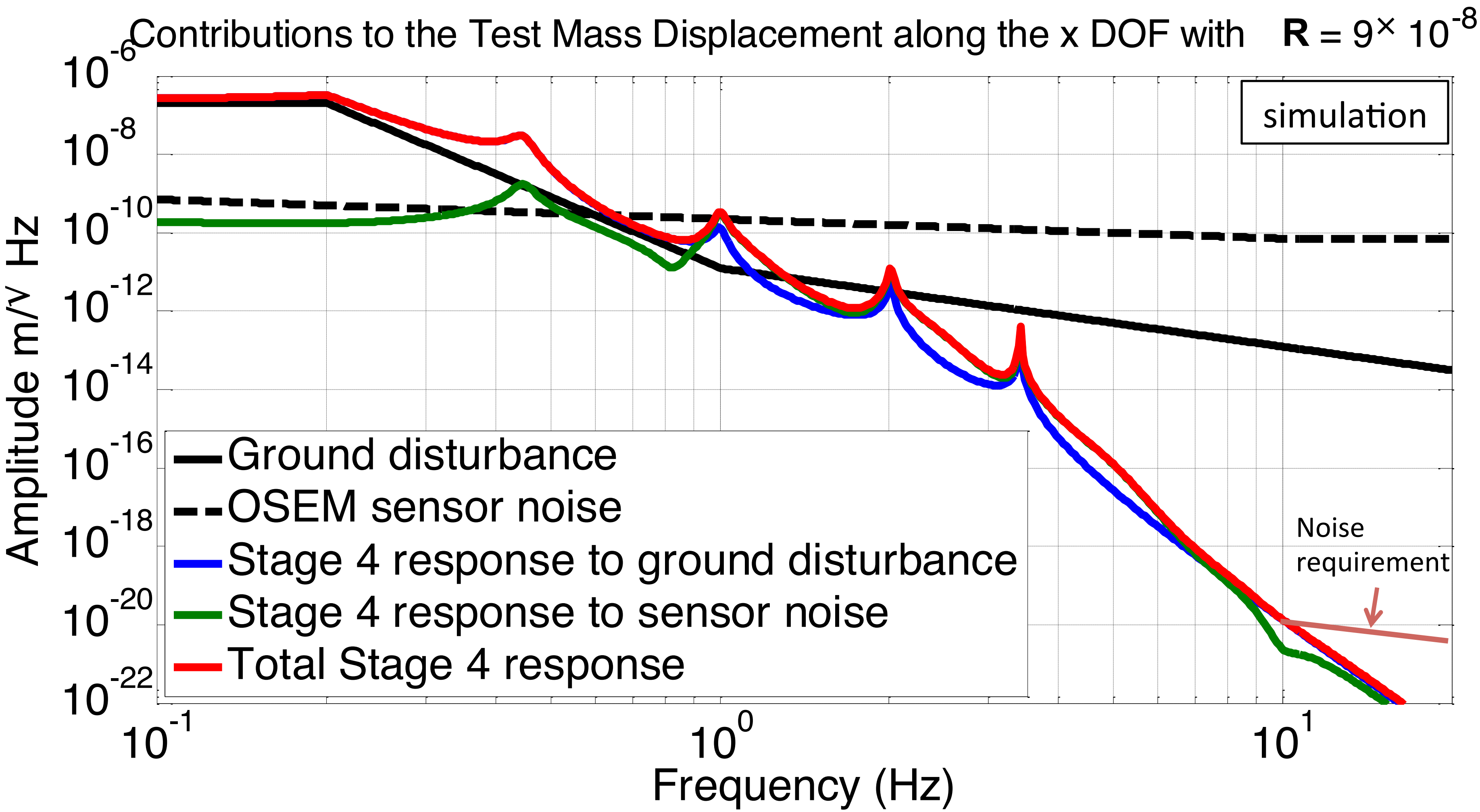
DOFs *i* are: x, y, z, yaw, pitch, roll



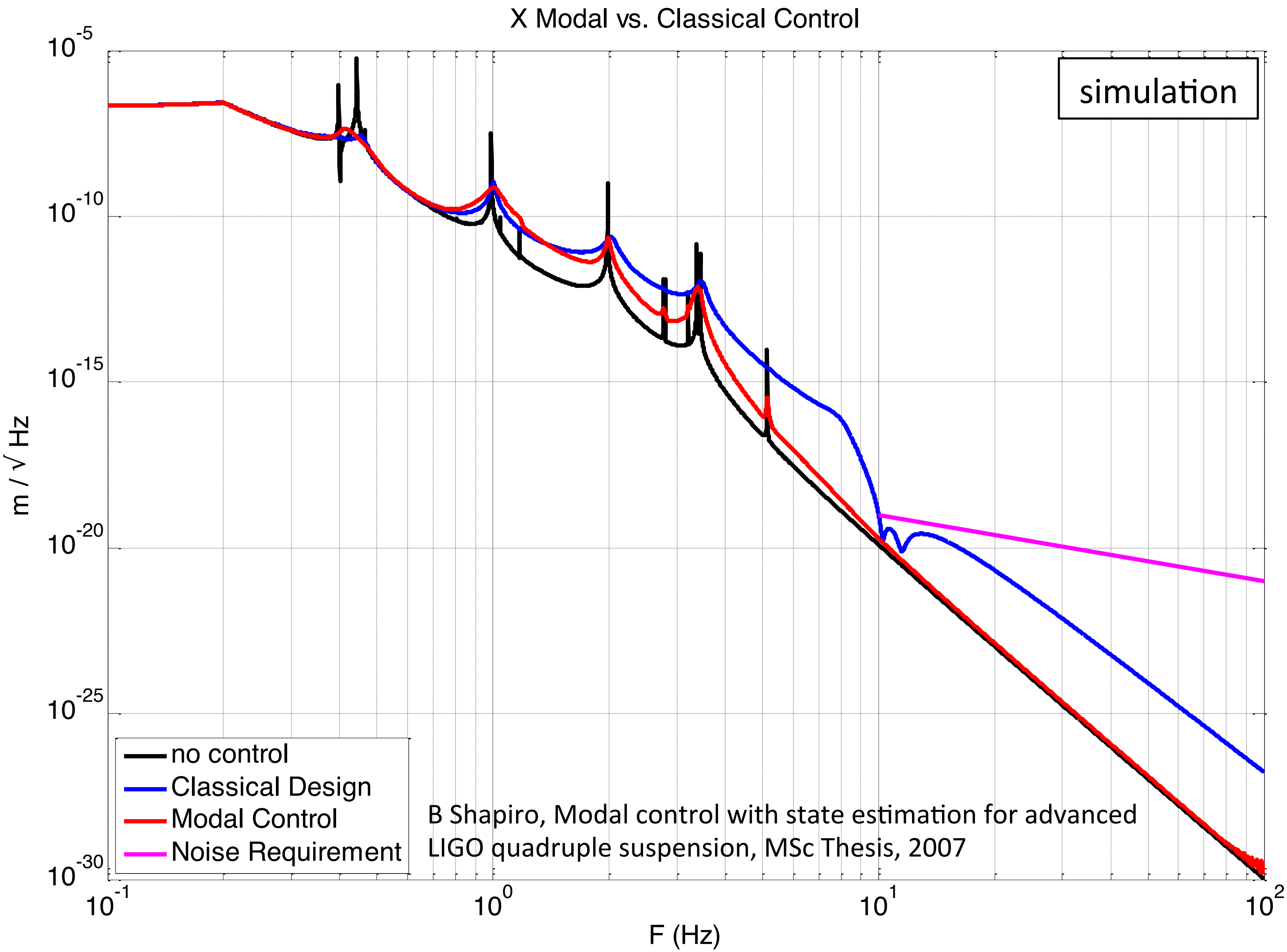
Modal Estimation Cost



Optimal Noise Amplification



Comparison of MD Noise Performance

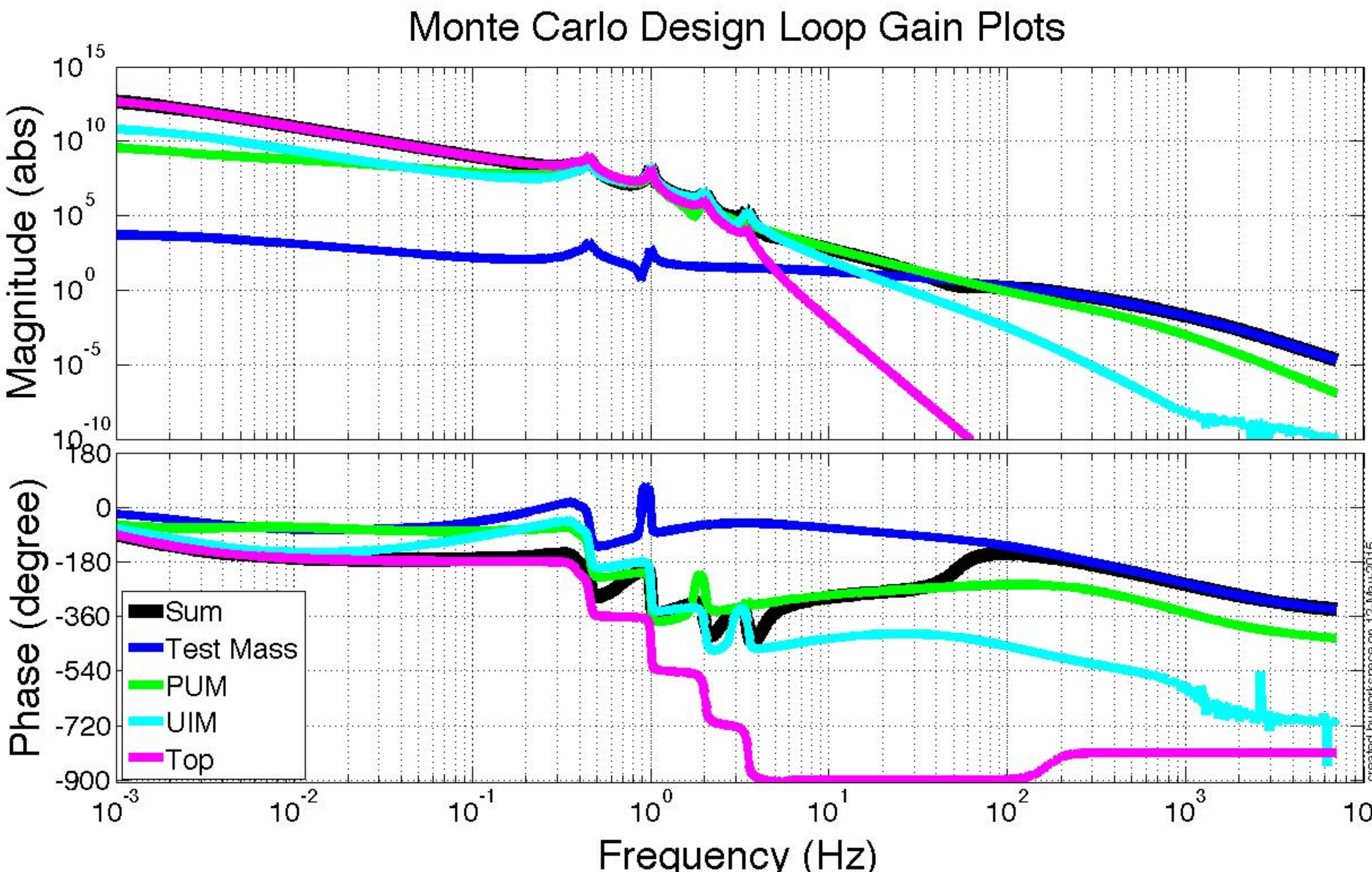


Generalized Cost Functions for Cavity Length Control

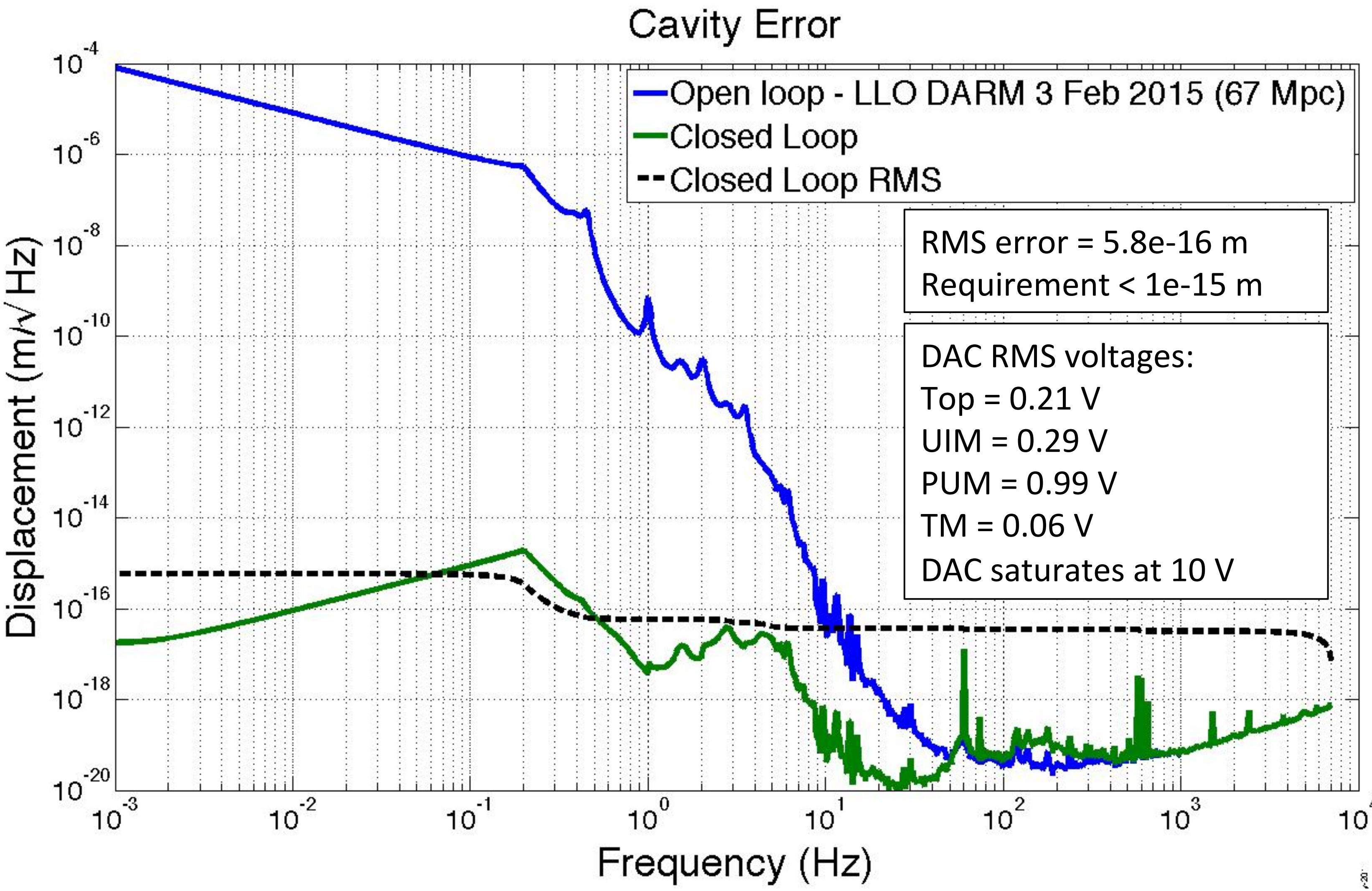
References: G1400853, G1400567, SWG log 11297

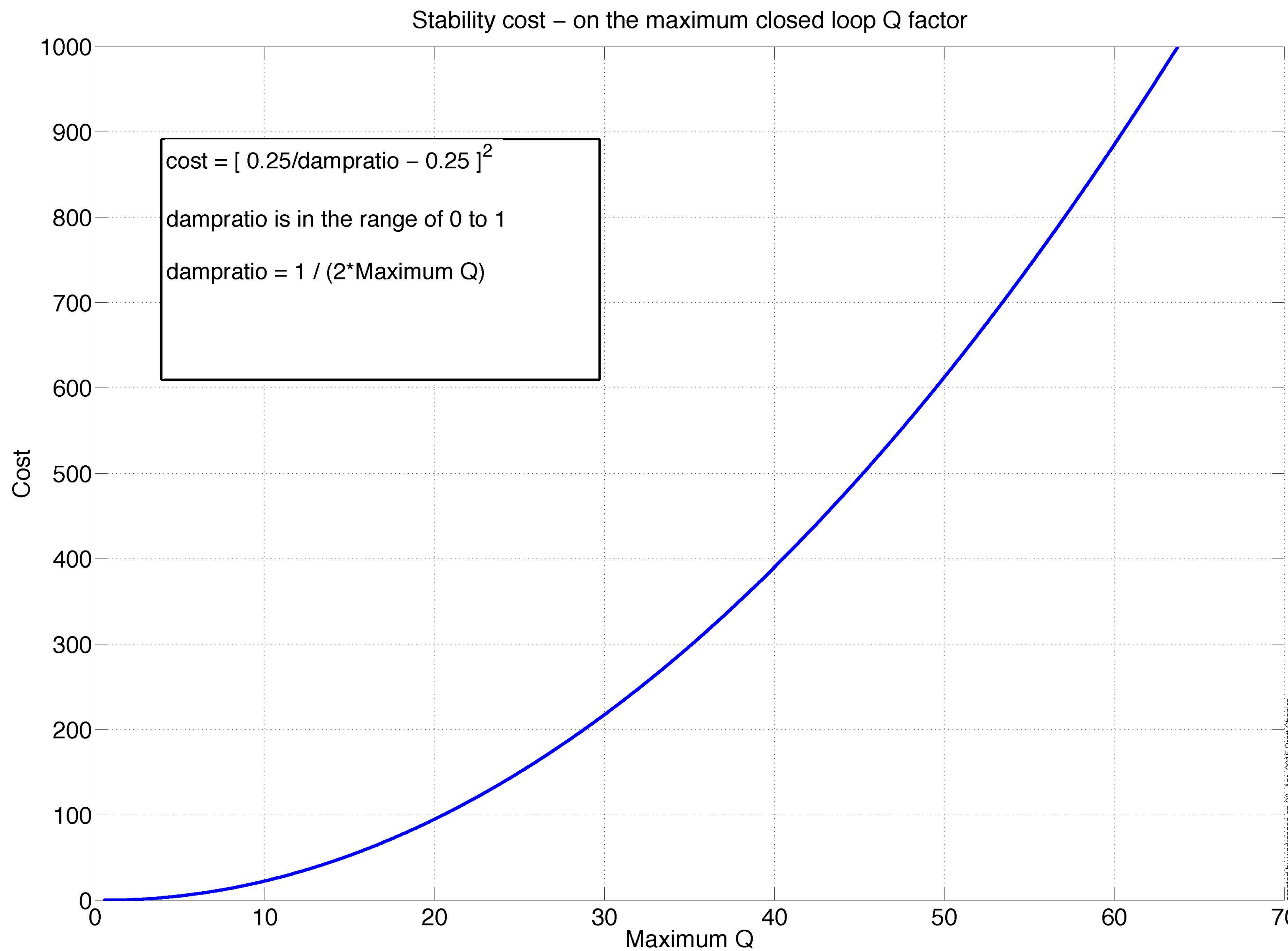
Monte Carlo Design

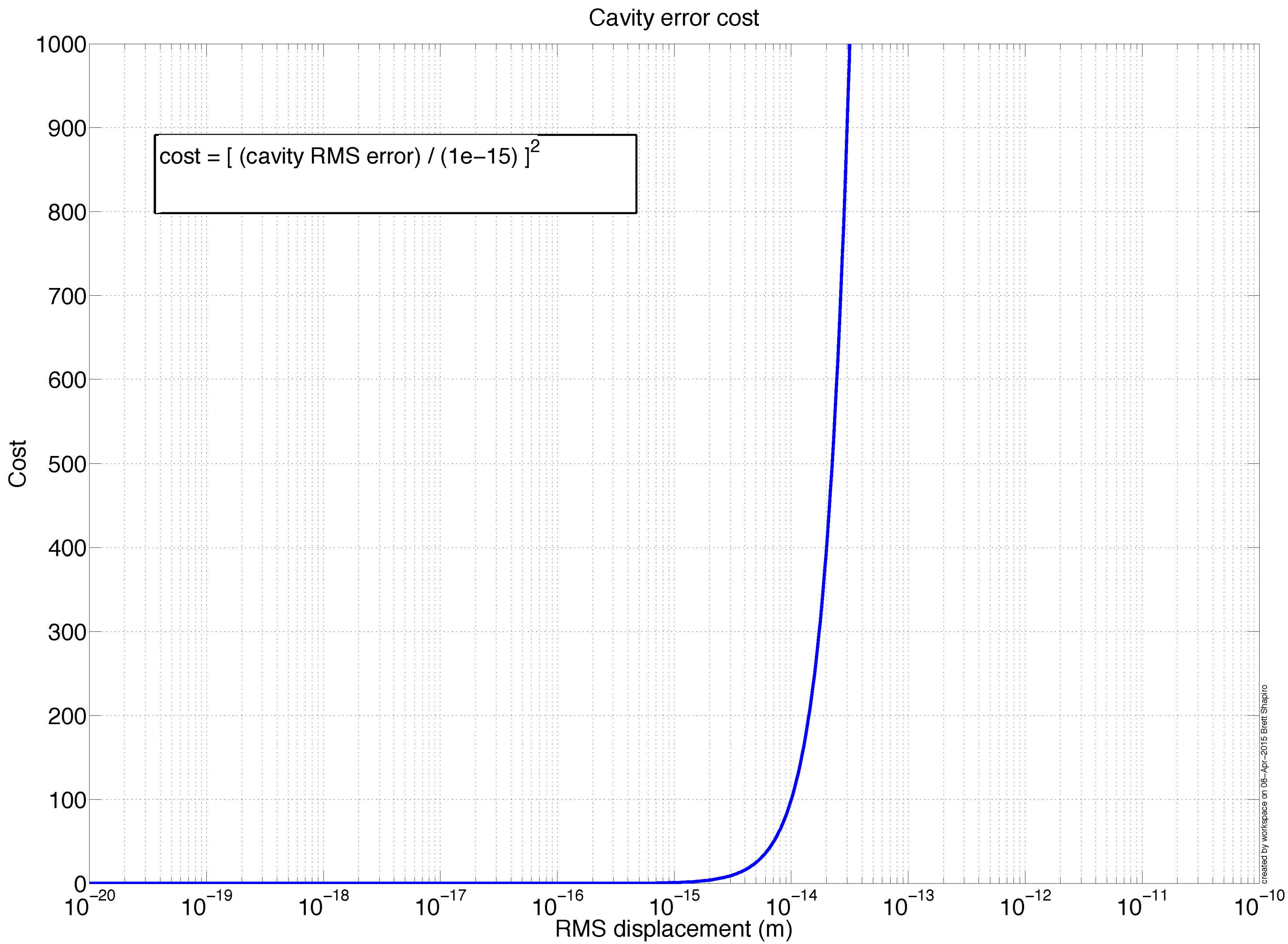
12 million iterations – about 1 day of computation time in MATLAB



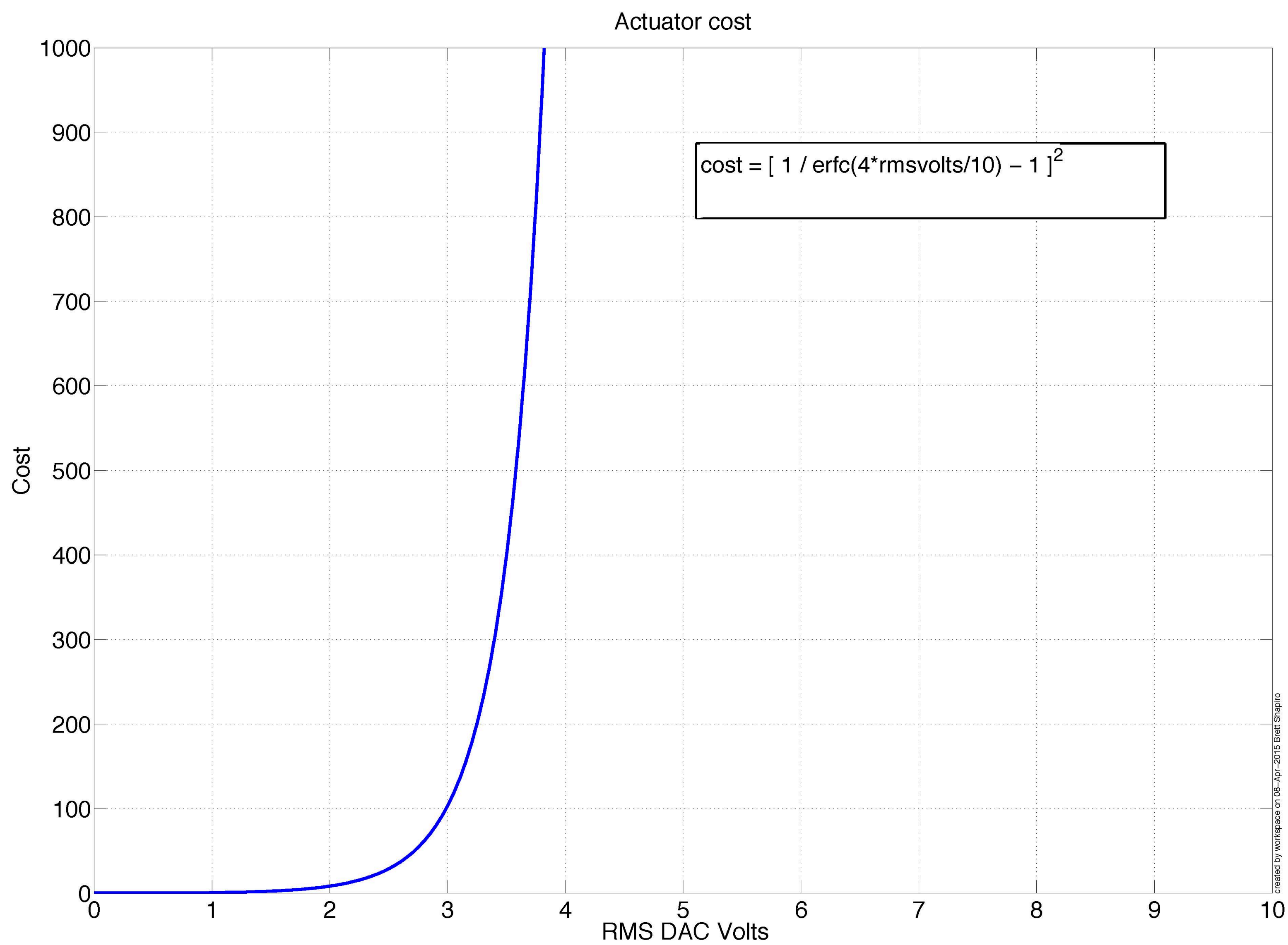
Cavity Performance (simulation)




Stability cost – on the maximum closed loop Q factor
Cost
Maximum Q
1000
900
800
700
600
500
400
300
200
100
0
0
10
20
30
40
50
60
70
cost = [0.25/dampratio – 0.25]^2
dampratio is in the range of 0 to 1
dampratio = 1 / (2*Maximum Q)
created by workspace on 06–Apr–2015 Brett Shapiro



Cavity error cost
cost = [(cavity RMS error) / (1e−15)]^2
Cost
RMS displacement (m)
1000
900
800
700
600
500
400
300
200
100
0
10^−20
10^−19
10^−18
10^−17
10^−16
10^−15
10^−14
10^−13
10^−12
10^−11
10^−10
created by workspace on 08−Apr−2015 Brett Shapiro



Actuator cost
cost = [1 / erfc(4*rmsvolts/10) − 1]^2
Cost
RMS DAC Volts
0
100
200
300
400
500
600
700
800
900
1000
0
1
2
3
4
5
6
7
8
9
10
created by workspace on 08−Apr−2015 Brett Shapiro