Iterative modeling & control design

Tying together the threads

Modern model-based control needs two things

- A local performance objective using the nominal model
  - A robustness measure involving the rough model error

Modeling with closed-loop data tells us about the model mismatch

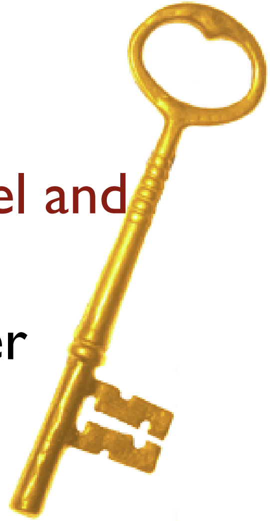
- Performance-based control design comes with a performance expectation

$$\left\| \begin{pmatrix} \left(1+P(e^{j\omega})C(e^{j\omega})\right)^{-1} & \left(1+P(e^{j\omega})C(e^{j\omega})\right)^{-1}C(e^{j\omega}) \\ P(e^{j\omega})\left(1+P(e^{j\omega})C(e^{j\omega})\right)^{-1} & P(e^{j\omega})\left(1+P(e^{j\omega})C(e^{j\omega})\right)^{-1}C(e^{j\omega}) \end{pmatrix} \right\|_\infty \leq \gamma$$

$$\left| \frac{P(e^{j\omega}) - \hat{P}(e^{j\omega},\theta)}{\hat{P}(e^{j\omega},\theta)} \times \frac{C(e^{j\omega})\hat{P}(e^{j\omega},\theta)}{1+C(e^{j\omega})\hat{P}(e^{j\omega},\theta)} \right| < 1$$

$$\left| H(e^{j\omega}) \times \frac{P(e^{j\omega}) - \hat{P}(e^{j\omega},\theta)}{\hat{P}(e^{j\omega},\theta)} \times \frac{C(e^{j\omega})\hat{P}(e^{j\omega},\theta)}{1+C(e^{j\omega})\hat{P}(e^{j\omega},\theta)} \times \frac{1}{1+C(e^{j\omega})P(e^{j\omega})} \right| < \epsilon$$

Key principles



Model-based control performance depends on nominal model and model quality

- Nominal model and model quality depend on the controller operating when the data is collected
  - Can we match up these two issues?
  - Can we set up the problem so that the successive controllers cause successive model to become more appropriate?

Try to link all the frequency-domain formulae to share a common objective

- But first some clues ...

First-cut controller $C_0(z)$

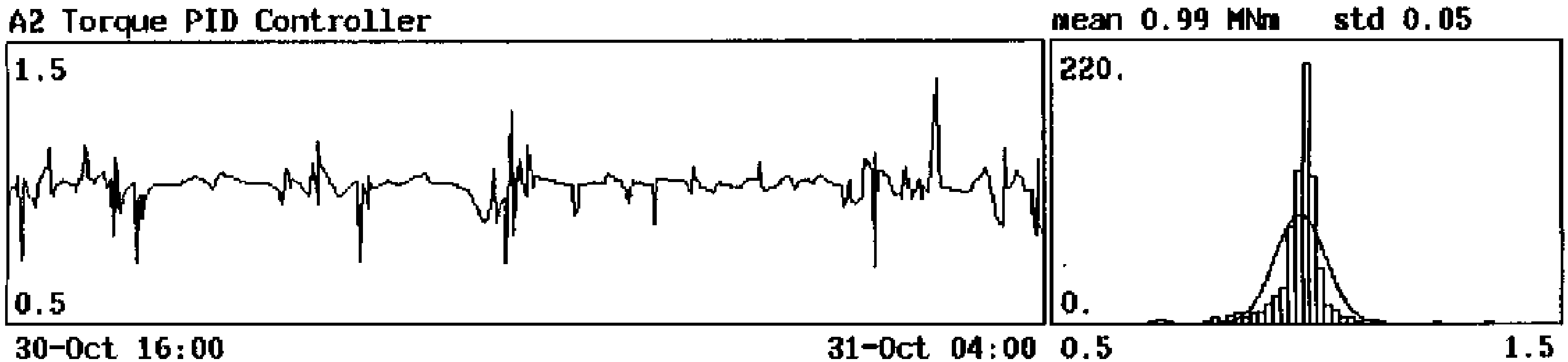
Design experimental reference signal *r(t)*, collect data

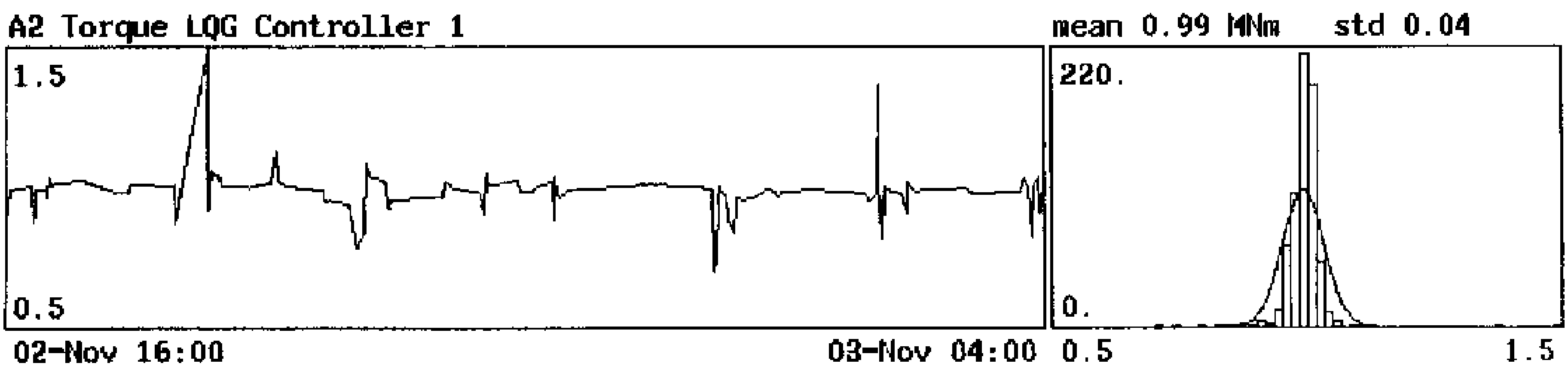
- Fit a process model with data selection and filtering to reflect objectives
  - Design a Linear Quadratic Gaussian controller with Loop Transfer Recovery (LQG/LTR) - a level of computed robustness
    - Expect the best but challenge the assumptions
      - Run the controller validation test

The control design comes with expected performance measures

- Spectra of controlled signals
  - Deviation from expected tells us about model quality for that controller

PID performance vs $C_0(z)$





We expected much lower variance, less spectral coloring

Aha!

Let's look at the spectrum of the torque from the LQG/LTR controlled system

According to my calculations $\hat{\Phi}_t(\omega) = \left| \dfrac{\hat{H}(e^{j\omega})}{1 + \hat{P}(e^{j\omega})C_0(e^{j\omega})} \right|^2$

But it really is measured to be $\Phi_t(\omega) = \left| \dfrac{H(e^{j\omega})}{1 + P(e^{j\omega})C_0(e^{j\omega})} \right|^2$

Frequency-weight the next LQG control design criterion to accommodate this

$$
\begin{aligned}
J_{\text{LQG}} &= \frac{1}{N} \sum_{k=1}^{N} \left\{ [F(z)t_k]^2 + \lambda\, [F(z)s_k]^2 \right\} \\
&= \frac{1}{2\pi} \int_{-\pi}^{\pi} \left\{ \left| F(e^{j\omega}) \right|^2 \Phi_t(\omega) + \lambda \left| F(e^{j\omega}) \right|^2 \Phi_s(\omega) \right\} d\omega
\end{aligned}
$$

Aha! Choose frequency-weighting $F(z) = \dfrac{H(z)}{1 + P(z)C_0(z)} \times \dfrac{1 + \hat{P}(z)C_0(z)}{\hat{H}(z)}$

Use this to design $C_1(z)$ using the same model

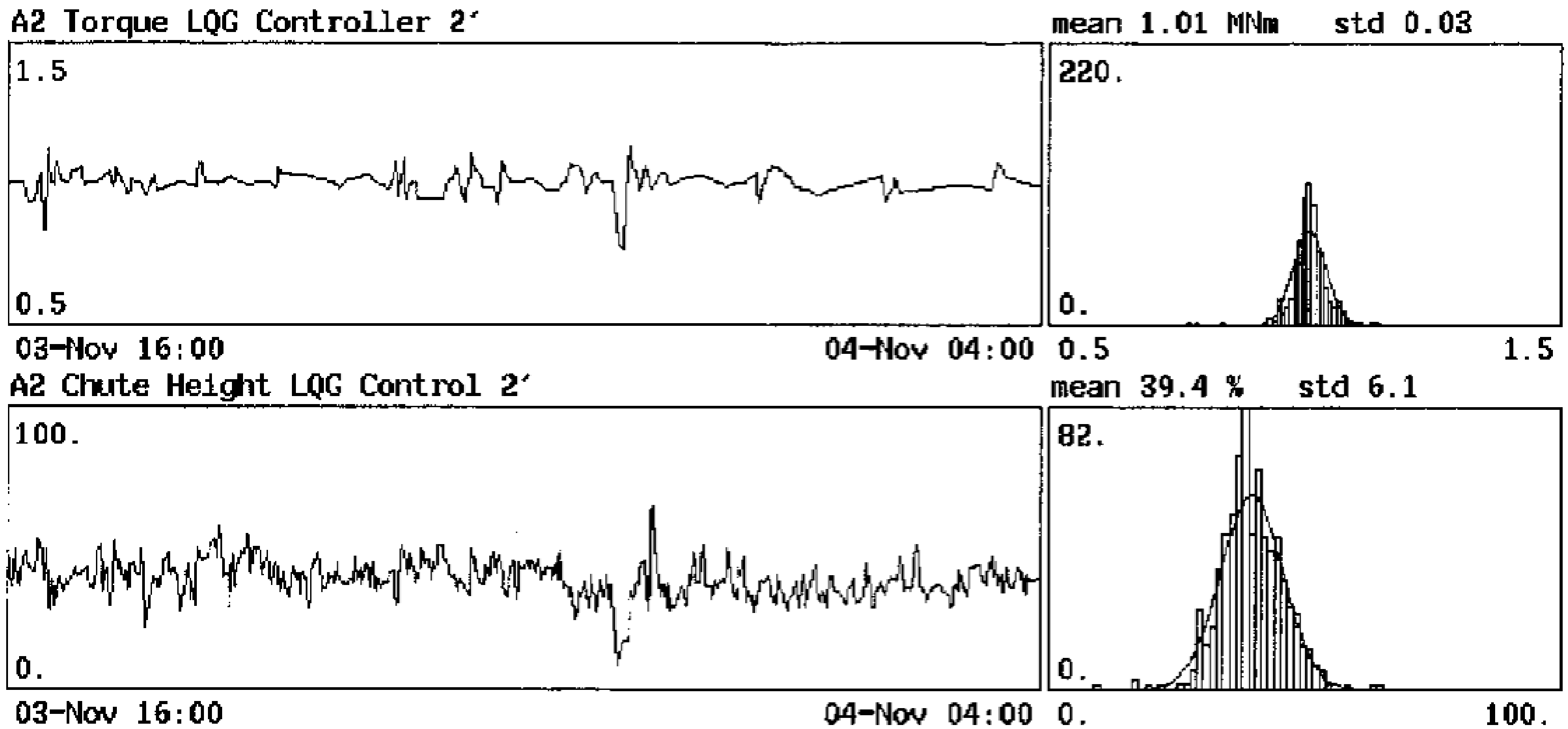
A sequence of control adjustments

- Model
  - Control design
    - Refined control using data-based frequency-weighting
    - •
    - •
- Re-model the process with data from latest controlled data
  - Redesign controller
    - Refine controller
    - •
    - •
- •
- •

Sugar mill controllers

- PID controller $C_{-1}(z)$
- Model with PID derived data $\hat{P}_0(z)$
  - LQG/LTR un-frequency-weighted control $C_0(z)$
  - Frequency-weighted controller adjustment $C_1(z)$
  - Frequency-weighted controller adjustment $C_1'(z)$
- New model identified with $C_1(z)$ $\hat{P}_1(z)$
  - LQG/LTR un-frequency-weighted control $C_2(z)$
  - Frequency-weighted controller adjustment $C_2'(z)$
- Stop at controller $C_2(z)$

Ultimate performance



Some conclusions

There is much to be gained from examining the modeling and controller design as a joint problem

Modeling is fairly expensive

- So it makes sense to re-use the model via controller tuning

The closed-loop data are really informative about controller performance

- which, in turn, is informative about model quality
  - Iterative solution is necessary

It is possible to tune the controller without a model at all

- Iterative Feedback Tuning
  - Based on gradient calculation from data

Iterative feedback tuning

Controller is parametrized by a set of numbers ρ $C_\rho(z)$

Try to optimize the choice of ρ using experimental data

Estimate the gradient of the performance criterion

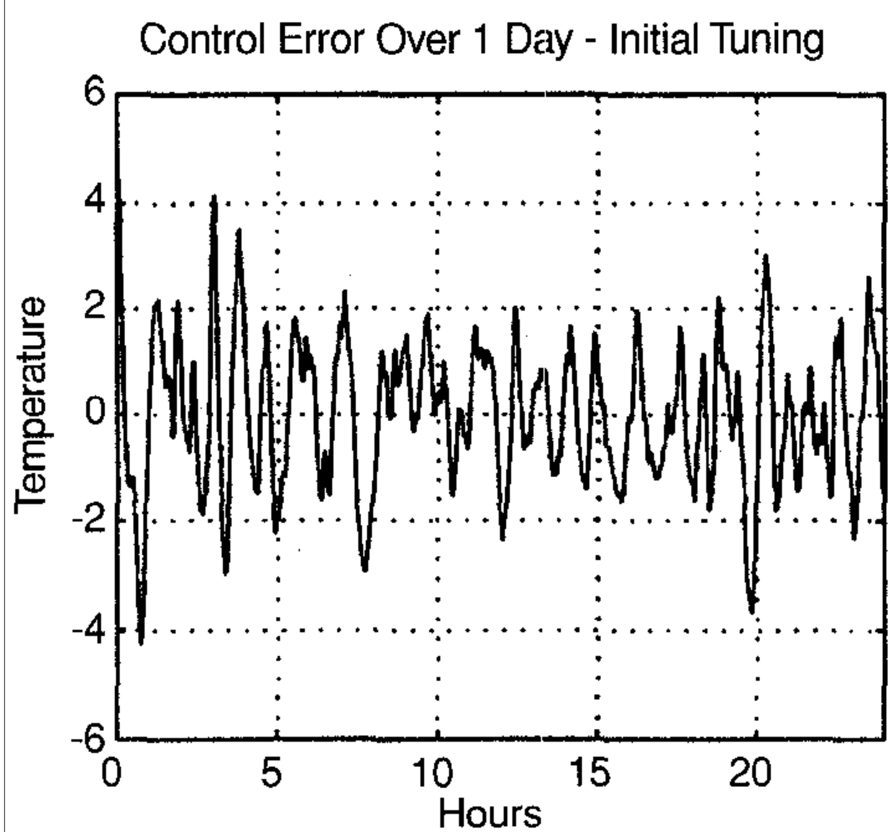
Gradient is calculated by filtering closed-loop signal through filters such as

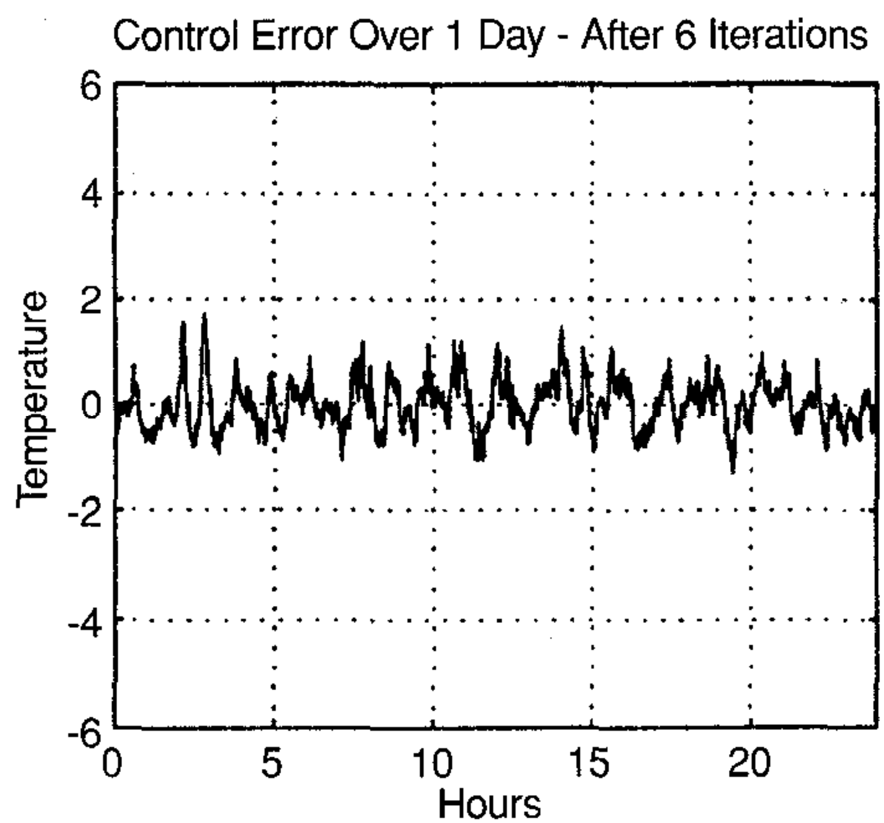
$$\frac{\partial C_\rho(z)}{\partial \rho} = F(z)$$

Adjust controller using gradient

$$\rho_{k+1} = \rho_k - \alpha \frac{\partial J_{\mathrm{LQG}}}{\partial \rho}$$

IFT performance - distillation column



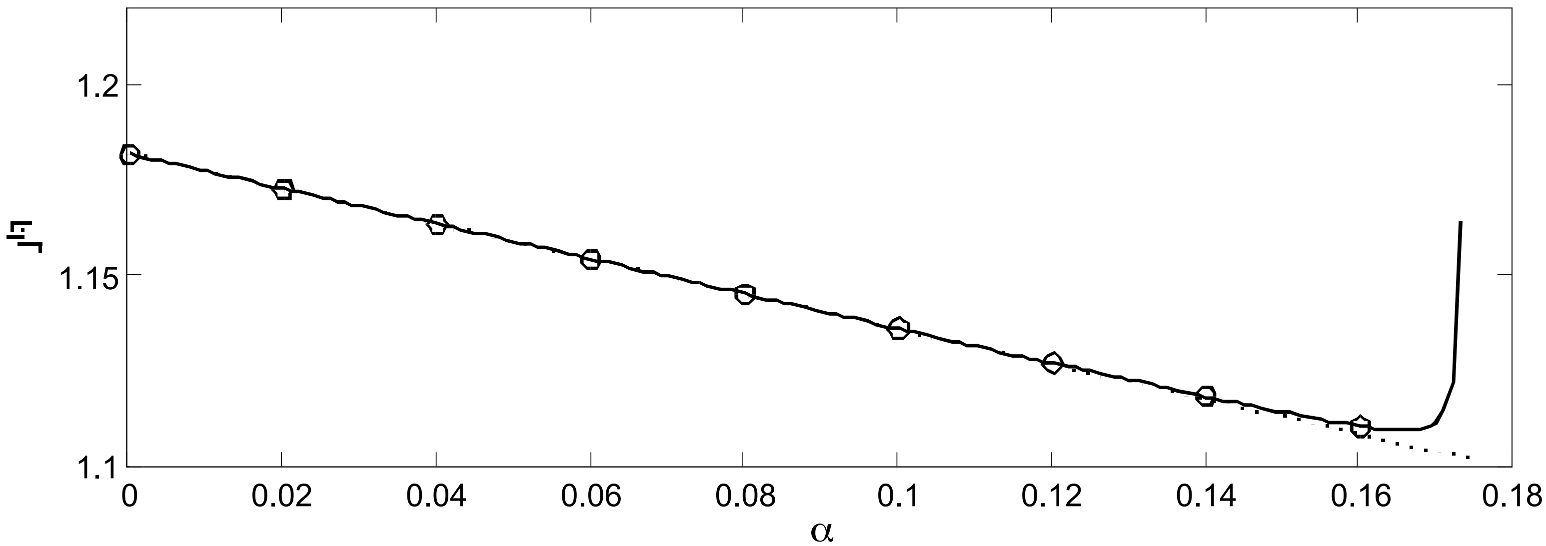


From: Hjalmarsson, Gevers, Gunarsson, Lequin, *CSM*, 1998

Many reports of applications success

Local minimizers, model-free adaptation

Caution in controller tuning



- Care is needed in moving between controllers
  - Guarantees of stability and performance would be helpful
    - But we need to look at more than the performance alone

Vinnicombe's ν-gap metric

$$\delta_\nu(C_1, C_2) = \max_\omega \left| \left(1 + |C_1(e^{j\omega})|^2\right)^{1/2} \left(C_1(e^{j\omega}) - C_2(e^{j\omega})\right) \left(1 + |C_2(e^{j\omega})|^2\right)^{1/2} \right|$$

$$\delta_\nu(C_1, C_2) \in [0, 1]$$

Frequency-domain distance measure between systems/controllers

There is an additional phase condition

If stabilizes *P* define the stability margin

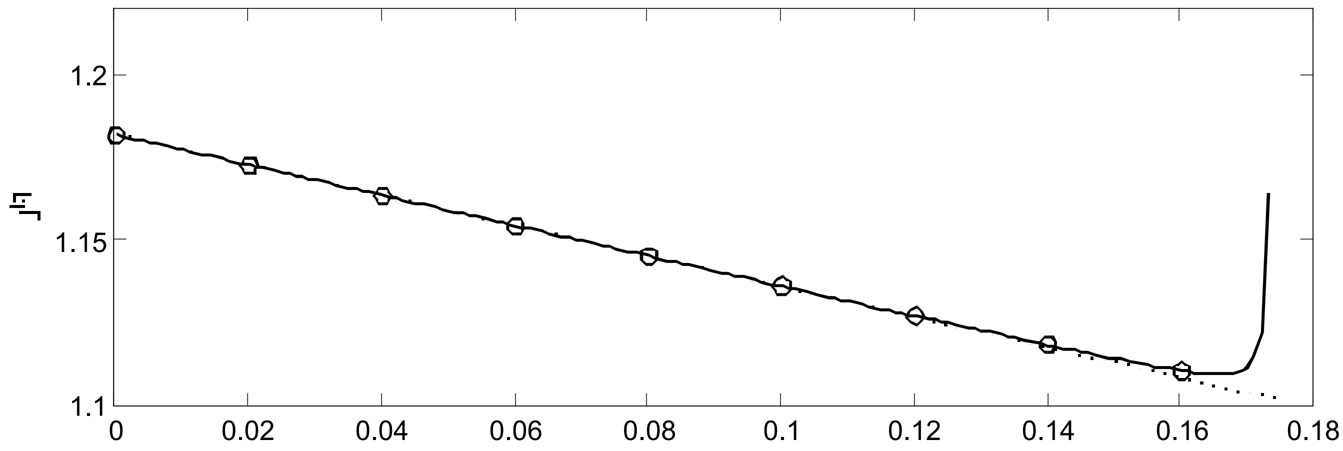
$$b_{P,C_1} = \left[\max_\omega \left| \begin{pmatrix} P(1+C_1P)^{-1}C_1 & P(1+C_1P)^{-1} \\ (1+C_1P)^{-1}C_1 & (1+C_1P)^{-1} \end{pmatrix} \right| \right]^{-1}$$

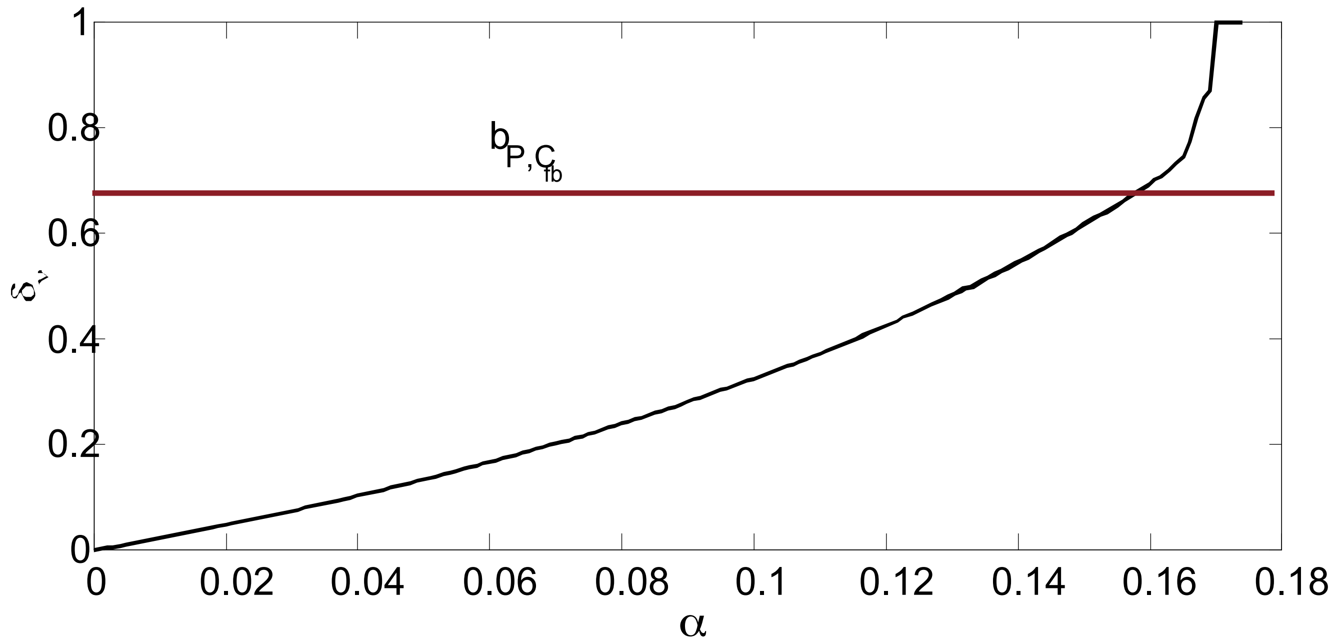
$$b_{P,C_1} \in [0, 1]$$

Then we have the following guarantee

$$\arcsin b_{P,C_2} \geq \arcsin b_{P,C_1} - \arcsin \delta_\nu(C_1, C_2)$$

What went wrong?





Conclusions

There is an emerging group of frequency domain formulae and expressions which assist in dealing with linking

- Control stability and performance robustness
  - Model approximation bias and variance
    - Controller adjustment using gradient methods
      - Stability and performance guarantees

Underlying much of this is the analysis of closed-loop experimental data prior to modeling and after controller implementation

- It makes sense to view the eventual control design as providing guidance to the testing and validation of models
  - Likewise the margin analysis yields limits to controller tuning
    - Closed-loop data and vigorous tests are the key