UC Berkeley

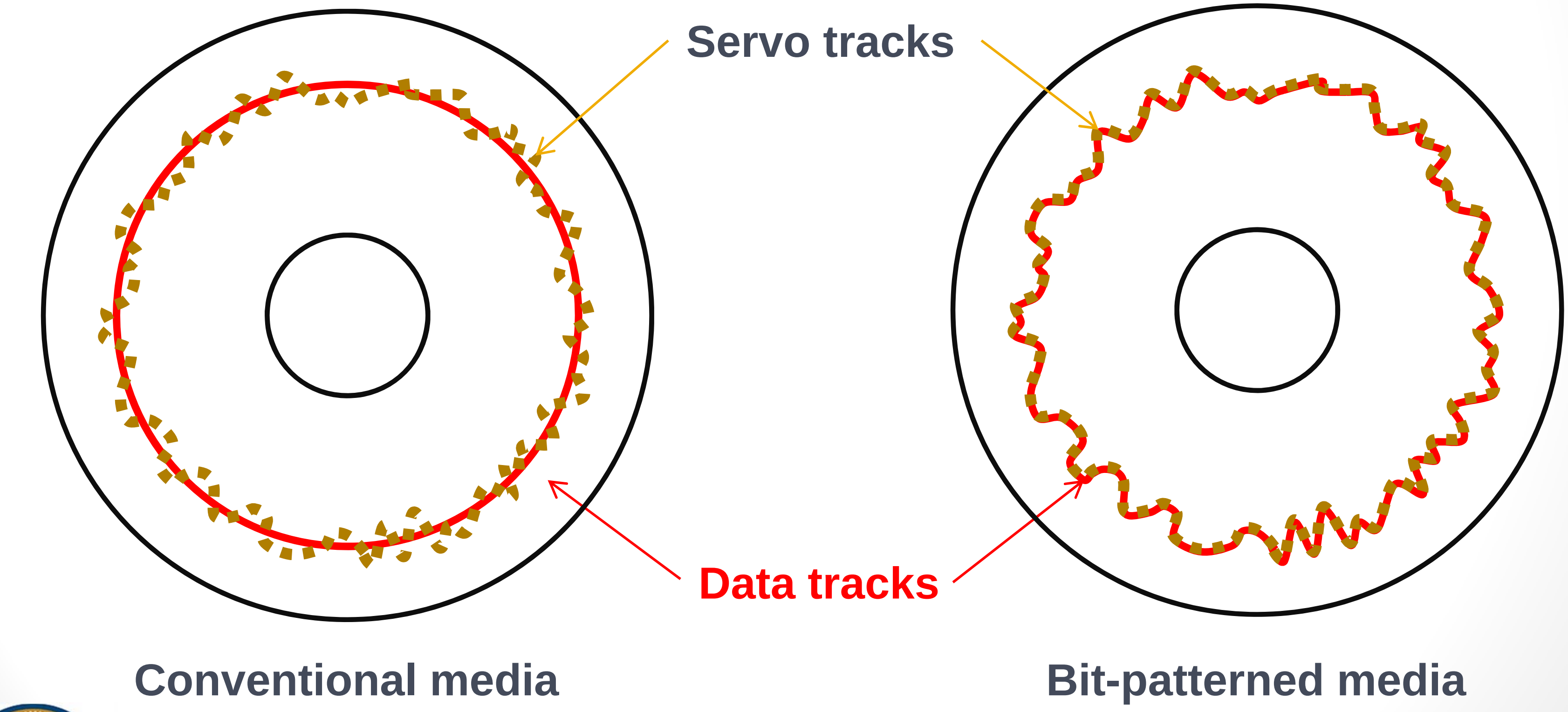
Adaptive Feedforward Repetitive Run-Out Tracking in Bit Patterned Recording

Behrooz Shahsavari, Ehsan Keikha,
Fu Zhang, Omid Bagherieh, Roberto Horowitz,

CML Sponsors Meeting

Repeatable Runout in Bit Patterned Recording

- Conventional media: data is written on concentric circular tracks
- Bit patterned media: data should be written on tracks with predetermined shapes, which are created by lithography on the disk.

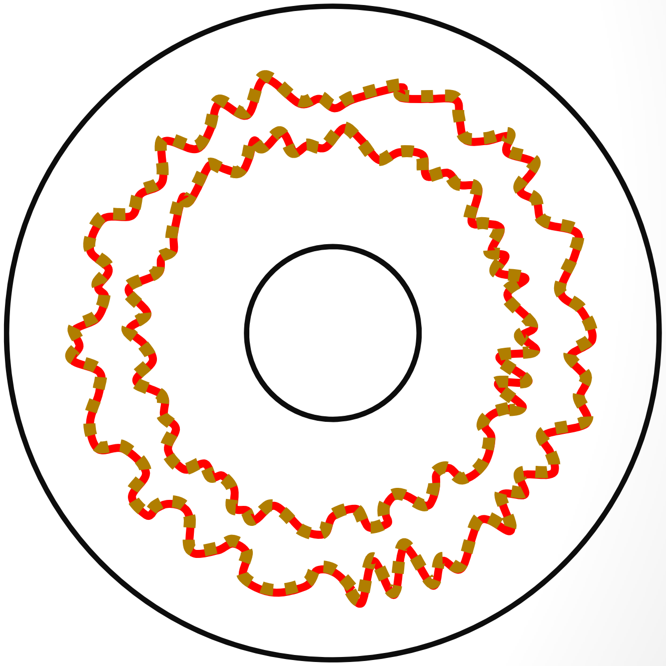


Objective

- The goal of this project is to control the voice coil motor (VCM) such that the read/write head follows **unknown** repeatable runout (RRO).

Issues

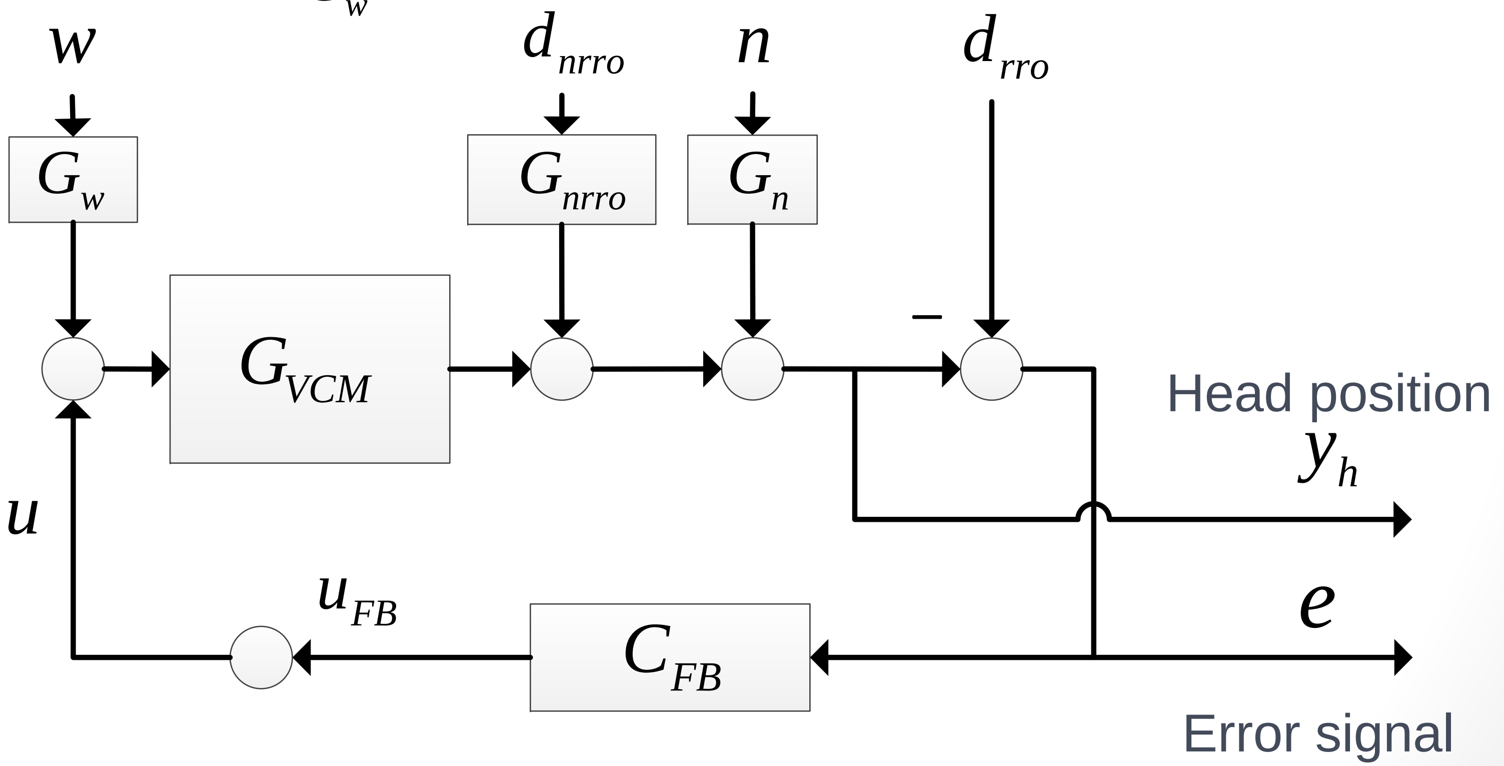
- RRO frequency spectrum is unknown
- RRO frequency spectrum can spread up to very high frequencies; therefore, will be amplified by the servo controller
- System dynamics is changing from drive to drive, and by temperature variation.
- RRO is changing in both circumferential and radial direction



Bit-patterned media

Controller Architecture

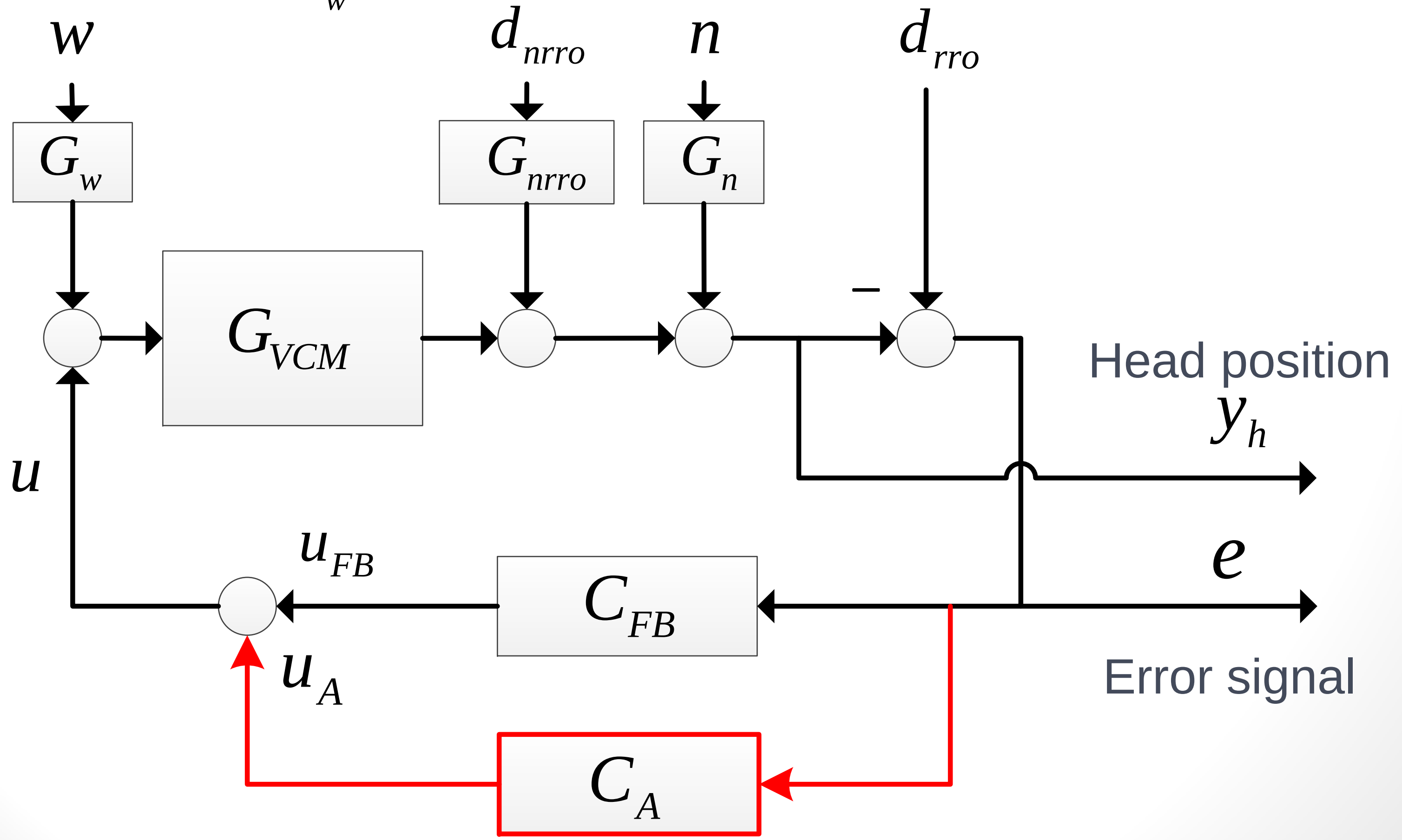
- Feedback controller, C_{FB} , attenuates the following noises
  - NRRO: G_{nrro}
  - Meas. Noise: G_n
  - Windage: G_w



Controller Architecture

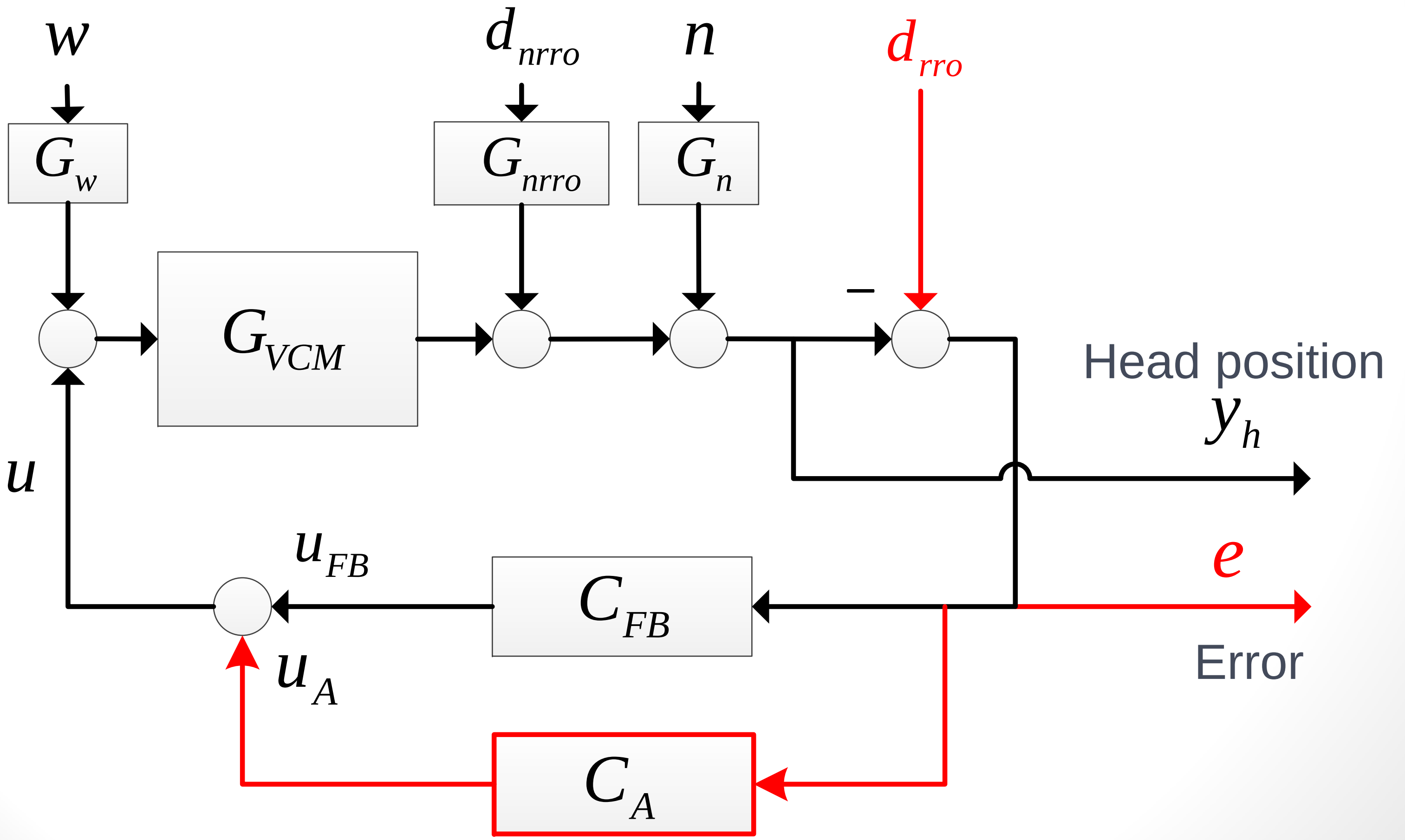
- Feedback controller, C_{FB}
  - NRRO: G_{nrro}
  - Meas. Noise: G_n
  - Windage: G_w

- Adaptive controller, C_A, is added in a "Plug-in" fashion to **track**
  - RRO: d_{rro}



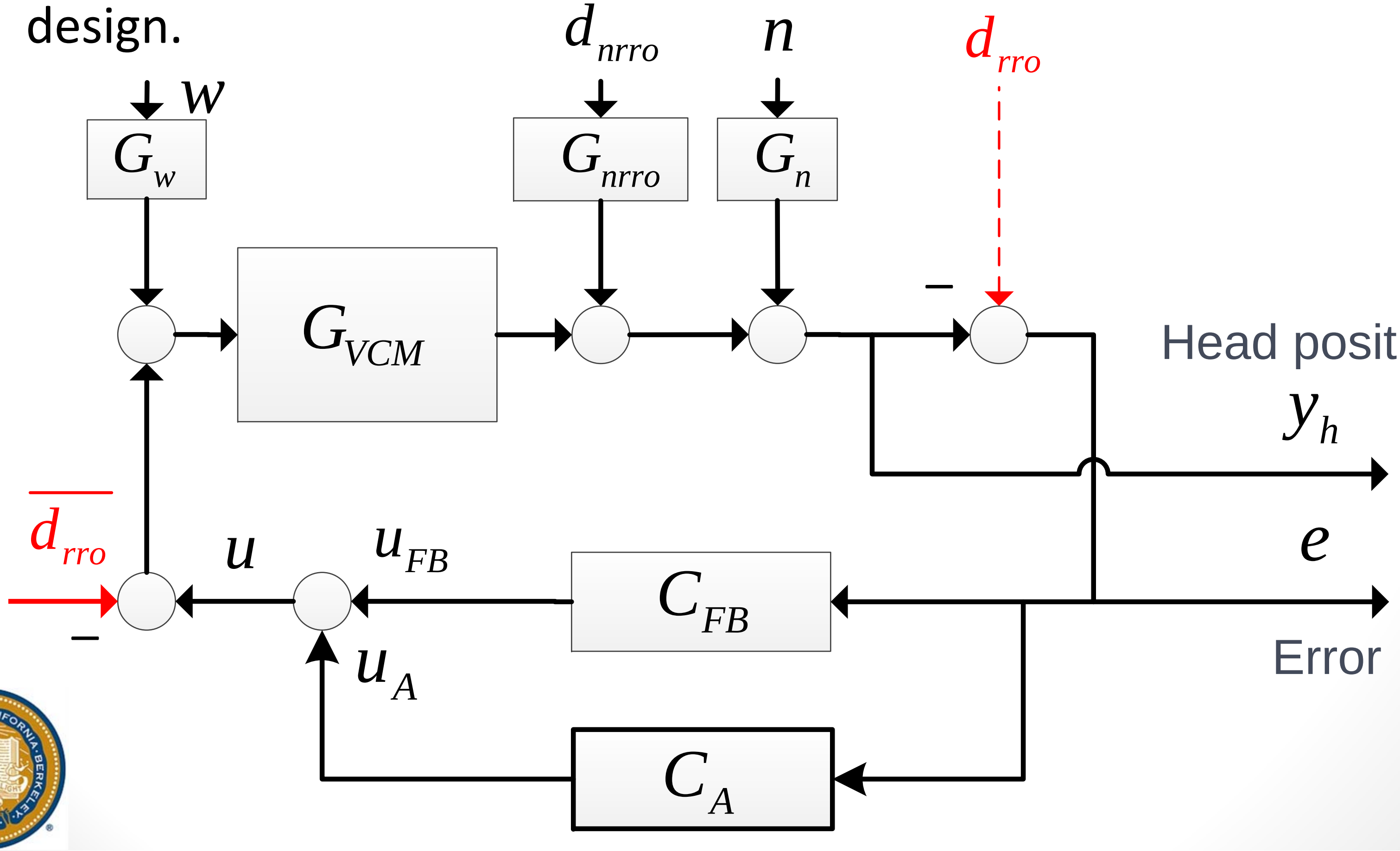
Controller Architecture

- We aim to design an adaptive controller, C_A, such that the error signal, e, is minimized. In other words, the head position y_h is following the RRO, d_{rro}.



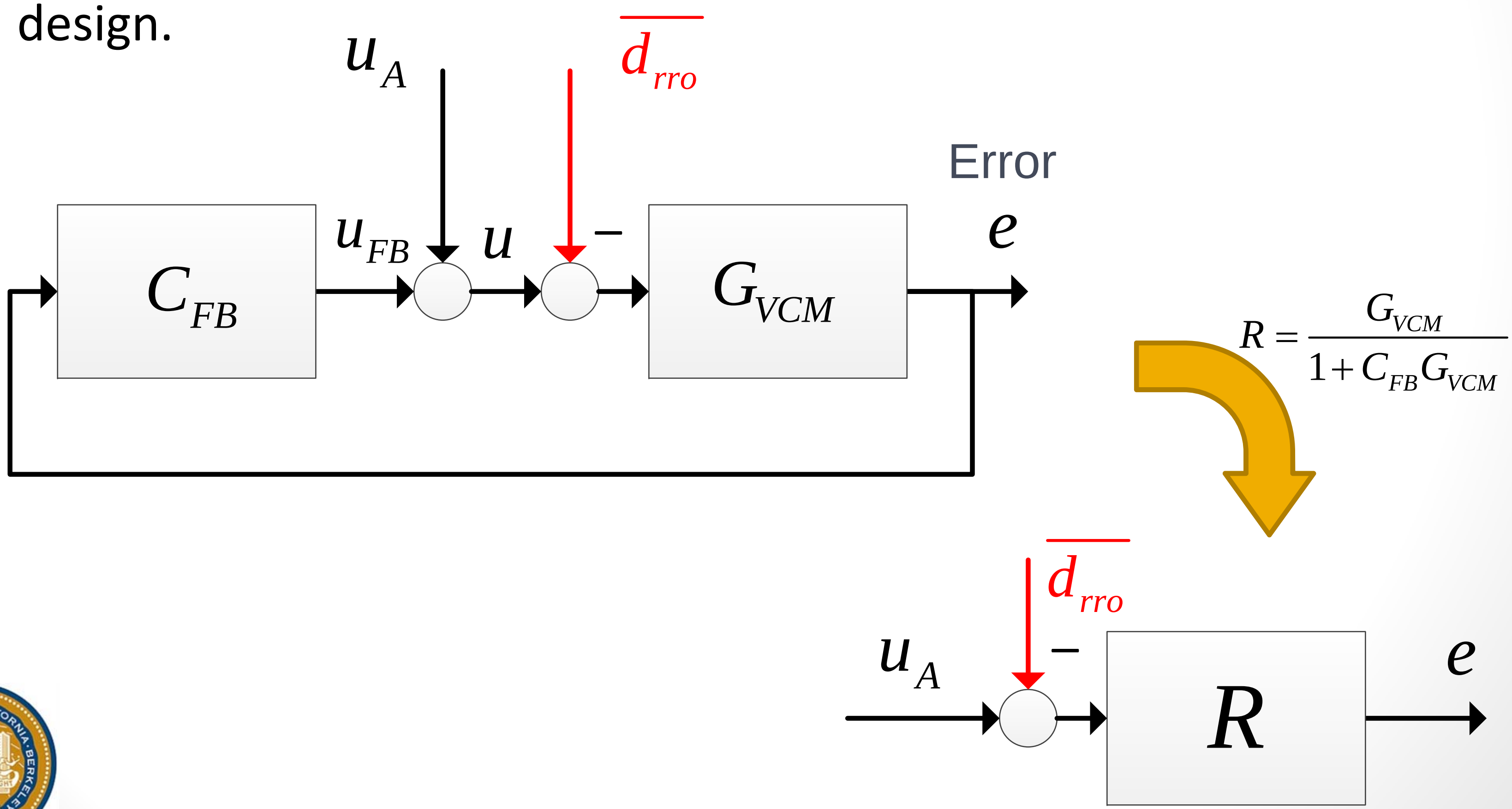
Controller Architecture

- We replace the unknown RRO, d_{rro}, by another unknown periodic sequence, $\overline{d}_{rro}$, that is added to u.
- We assume that the noises are attenuated by the feed-back controller C_{FB}; therefore, can be ignored in feedforward control design.

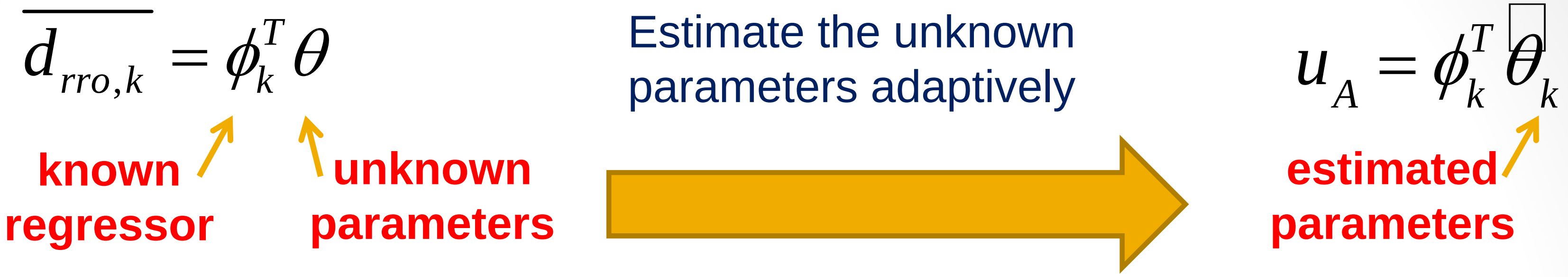


Controller Architecture

- We replace the unknown RRO, d_{rro} , by another unknown periodic sequence, $\overline{d}_{rro}$, that is added to u.
- We assume that the noises are attenuated by the feed-back controller C_{FB} ; therefore, can be ignored in feedforward control design.



Adaptive Control Algorithm



$$\overline{d_{rro,k}} = \phi_k^T \theta$$

known regressor

unknown parameters

Estimate the unknown parameters adaptively

$$u_A = \phi_k^T \overset{\square}{\theta}_k$$

estimated parameters

We propose a new adaptive control algorithm based on the following two key ideas:

1. The PES is converted to an **auxiliary error variable,** in order to use in the parameter adaptation algorithm.
2. A novel **adaptive step size** is proposed to increase the convergence rate and boost the steady state performance.

Adaptive Control Algorithm

$$\overline{d_{rro,k}} = \phi_k^T \theta$$

known regressor **unknown parameters**

We want to estimate the unknown parameters adaptively

$$u_A = \phi_k^T \overset{\square}{\theta}_k$$

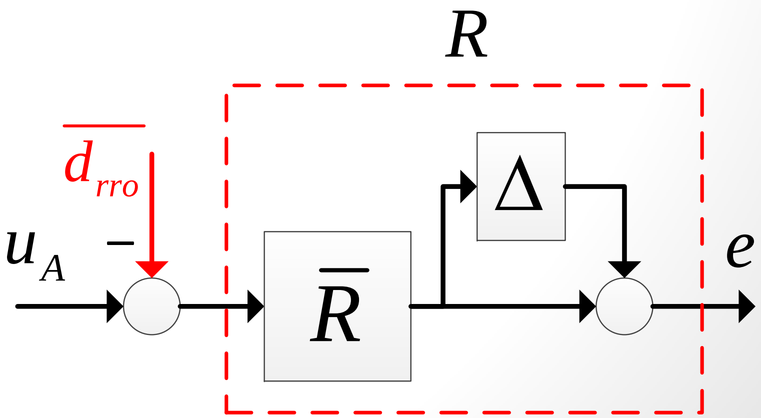
estimated parameters

- Auxiliary error

$$\overline{e_k} = e_k - \widetilde{e_k} = \underbrace{R\left(\phi^T\hat{\theta}_k - \phi^T\theta\right)}_{\text{measurable}\quad e_k} - \underbrace{\left[\bar{R}\left(\phi^T\widehat{\theta_k}\right) - \left(\bar{R}\phi_k\right)^T\hat{\theta}_k\right]}_{\text{known}\quad \widetilde{e_k}}$$

- Parameters update

$$\hat{\theta}_{k+1} = \hat{\theta}_k + \boxed{\mu_k}\left(\bar{R}\phi_k\right)\overline{e_k}$$



Adaptive Step Size – Key ideas

$$\hat{\theta}_{k+1} = \hat{\theta}_k + \boxed{\mu_k}\left(\bar{R}\phi_k\right)\overline{e_k}$$

- The step size in adaptation is a function of “Auxiliary Error” convergence.
- As we get closer to the real parameters, the step size becomes smaller.
- Finally, the algorithm stops when a certain performance is achieved.
- The adaptation starts again whenever the error becomes large (e.g. we move to another track with a different RRO).

$$V_k^h = \frac{1}{h}\sum_{i=k-h+1}^{k} \bar{e}_i^2 \quad \longleftarrow$$

Mean squared error approximation

$$\bar{\mu}_k = \rho\left(V_k^h - V^d\right) \quad \longleftarrow$$

Desired performance

$$\mu_k = \begin{cases} \min(\bar{\mu}_k, \mu_{allowed}) & \left(\bar{\mu}_k > 0 \text{ and } \mu_{k-1} > 0\right) \text{ or } \left(\bar{\mu}_k > \mu^{ub}\right) \\ 0 & \text{otherwise} \end{cases}$$

Adaptive Step Size

$$\bar{\mu}_k = \rho\left(V_k^h - V^d\right)$$

ρ: scalar gain V^d: desired PES variance

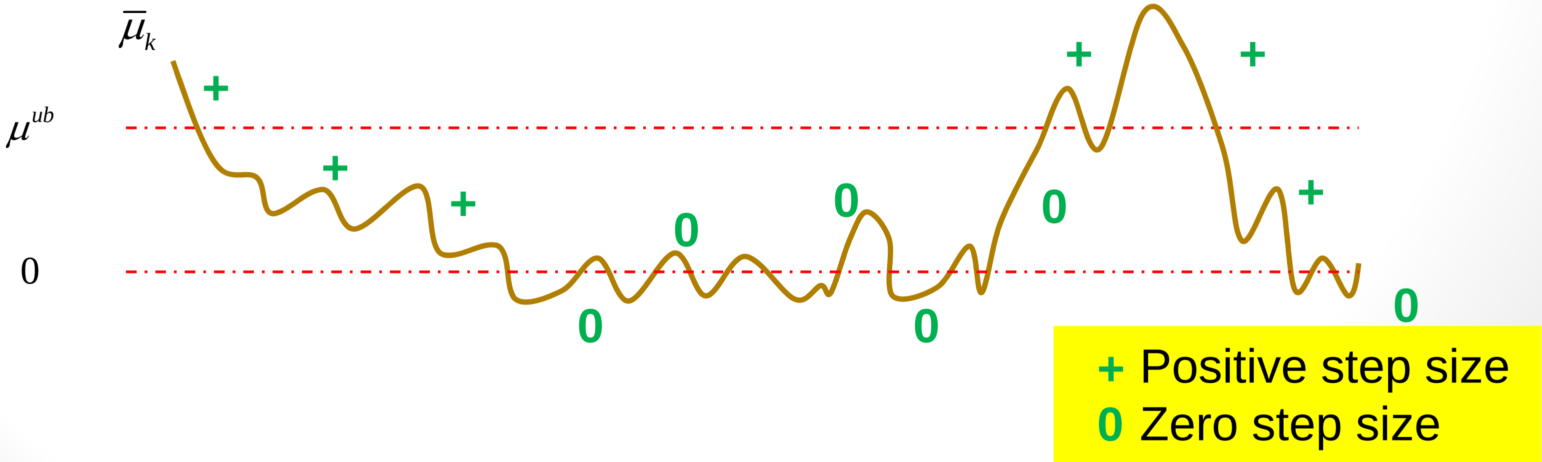
$$V_k^h = \frac{1}{h}\sum_{i=k-h+1}^{k} \bar{e}_i^2$$

V_k^h: aprox. variance of aux. error

$$\mu_k = \begin{cases} \min(\bar{\mu}_k, \mu_{allowed}) & \left(\bar{\mu}_k > 0 \text{ and } \mu_{k-1} > 0\right) \text{ or } \left(\bar{\mu}_k > \mu^{ub}\right) \\ 0 & \text{otherwise} \end{cases}$$

$\mu_{allowed}$: maximum step size to guarantee convergence

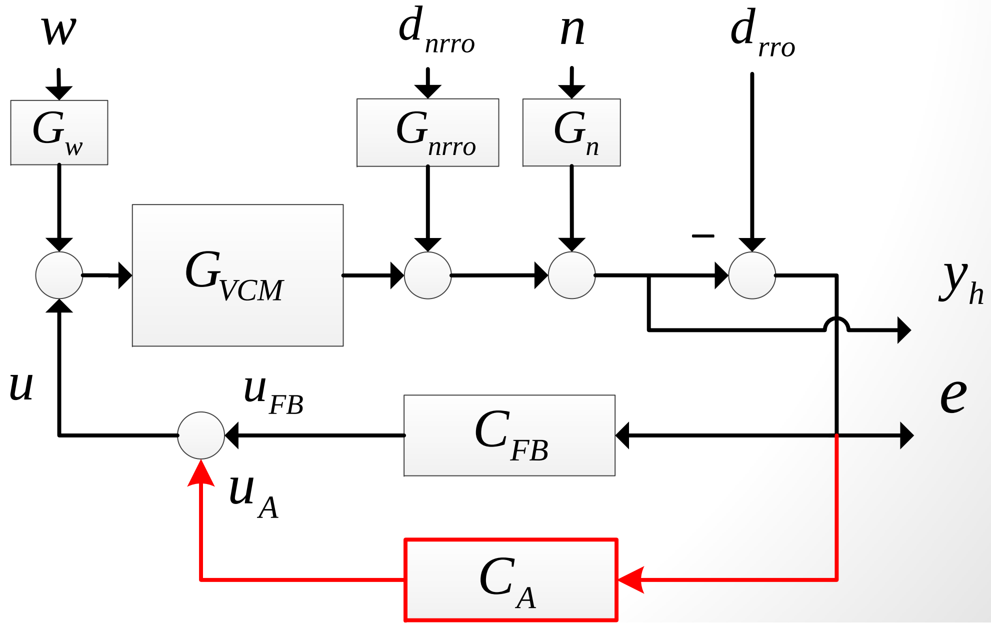
μ^{ub} : design parameter defining the dead-zone width



+ Positive step size
0 Zero step size

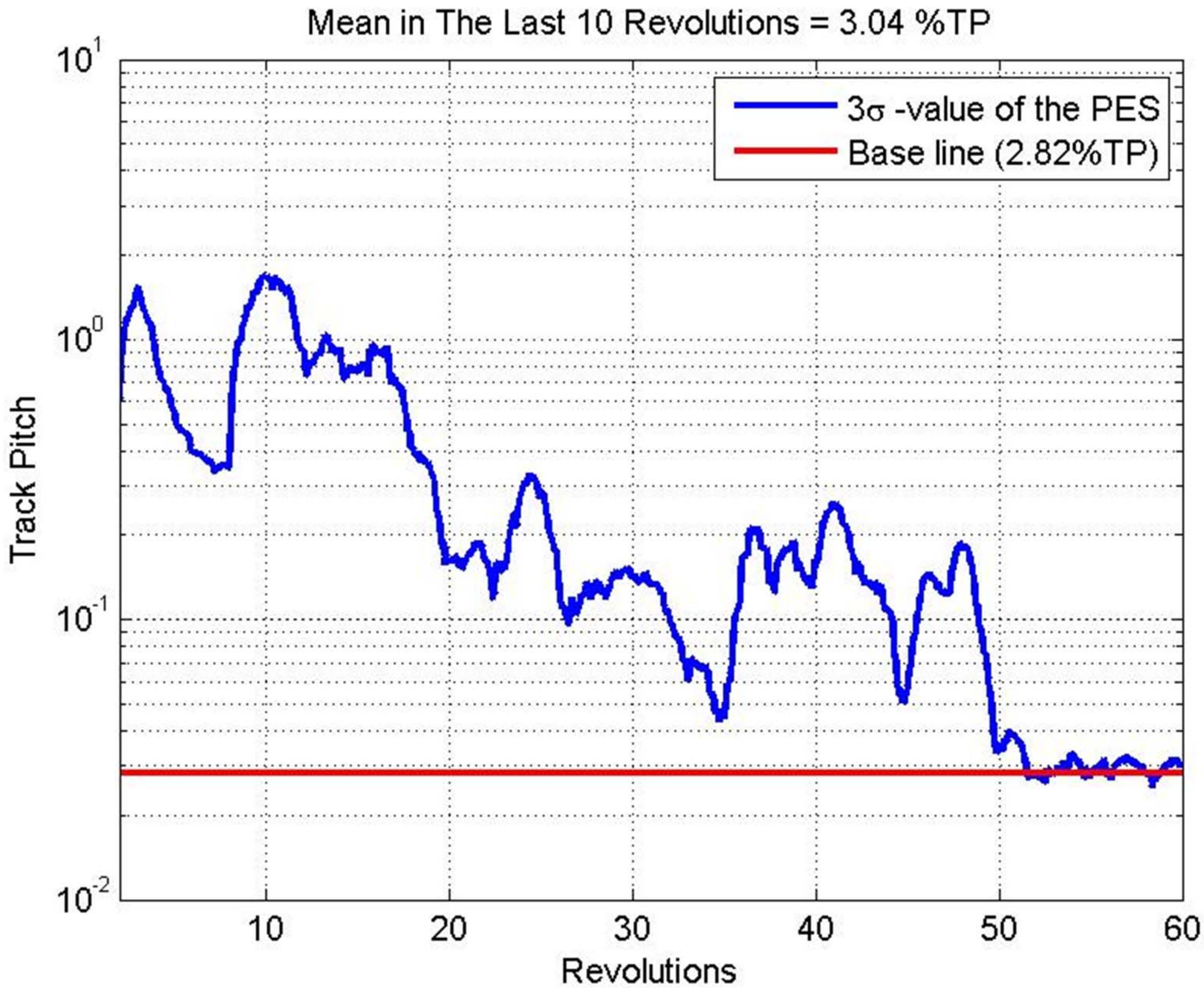
Simulation Results

- G_w, G_n , G_{nrro} and d_{rro} are modeled based on the real PES, and G_{VCM} and C_{FB} are modeled based on real frequency responses, all from a drive provided by HGST, a Western Digital company.
- Artificial RRO, that contains frequencies up to the 90th multiple of fundamental (spindle) frequency, is added to the real PES (from HGST and Seagate).
- RRO harmonics are divided into low, mid, and high frequency regions and their adaptation is scheduled in time.

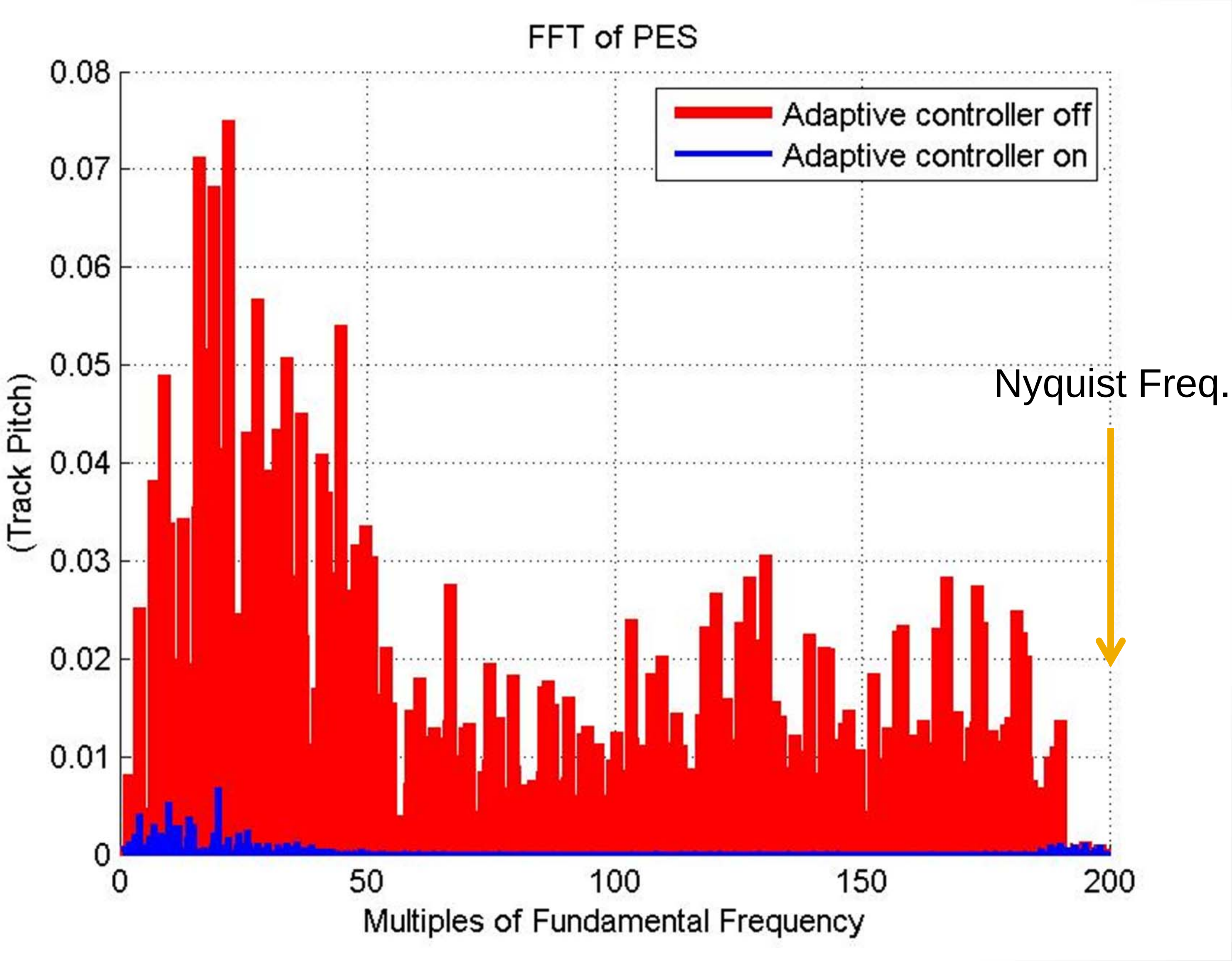


Simulation Results

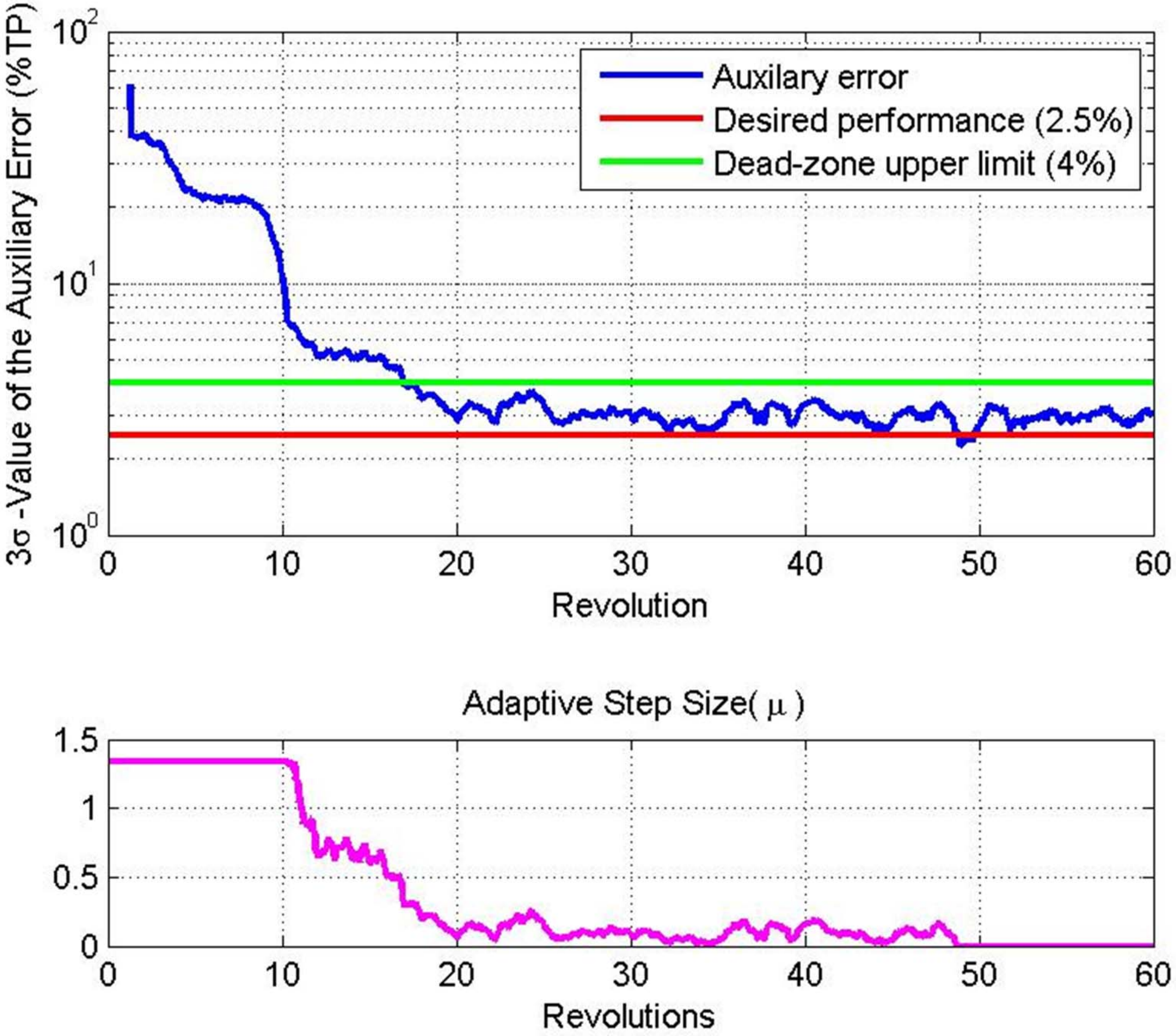
- Tracking Performance



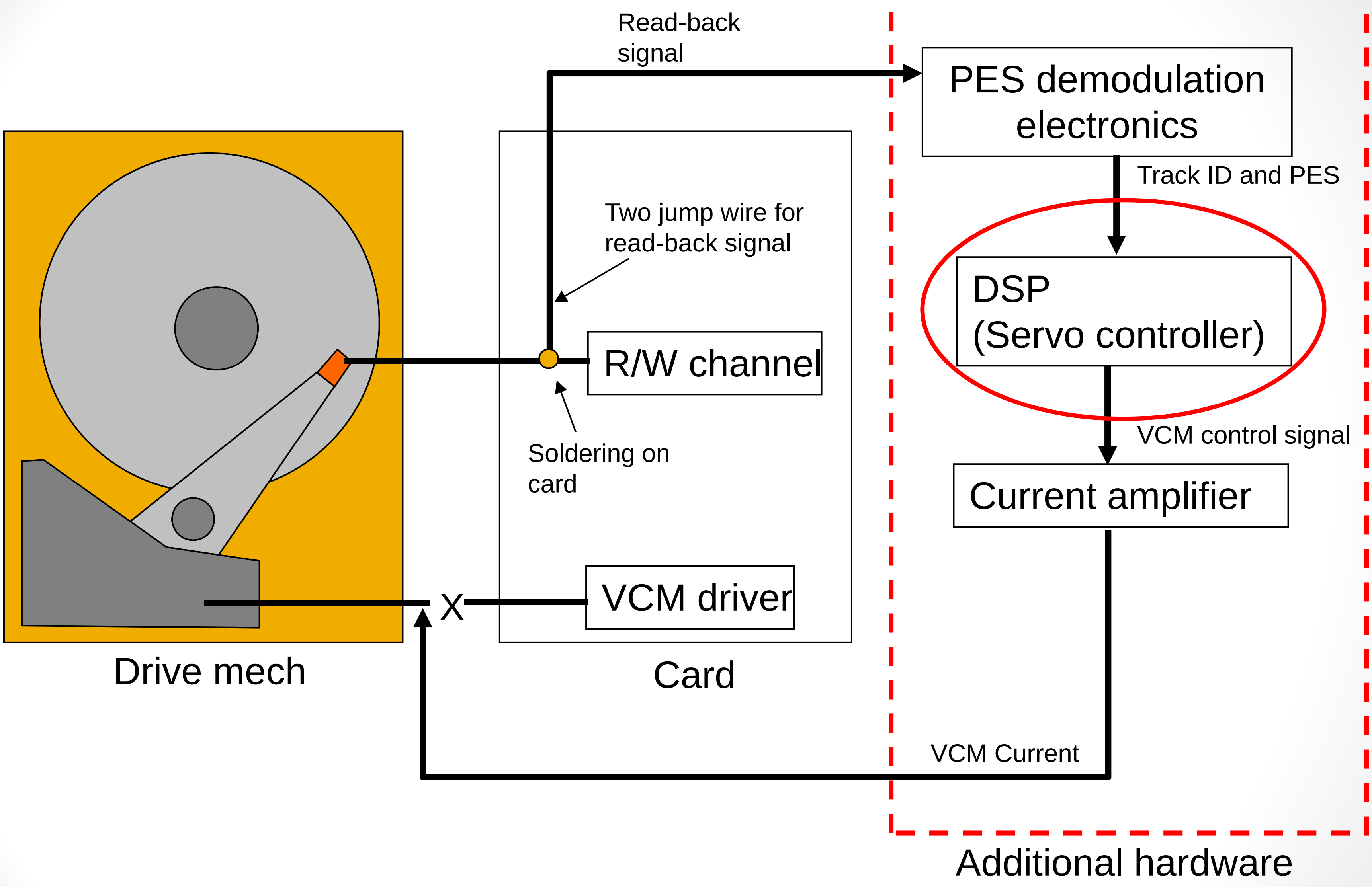
Simulation Results



Simulation Results – Adaptive Step size



Experimental setup



Read-back signal

PES demodulation electronics

Track ID and PES

Two jump wire for read-back signal

DSP (Servo controller)

R/W channel

VCM control signal

Soldering on card

Current amplifier

X

VCM driver

Drive mech

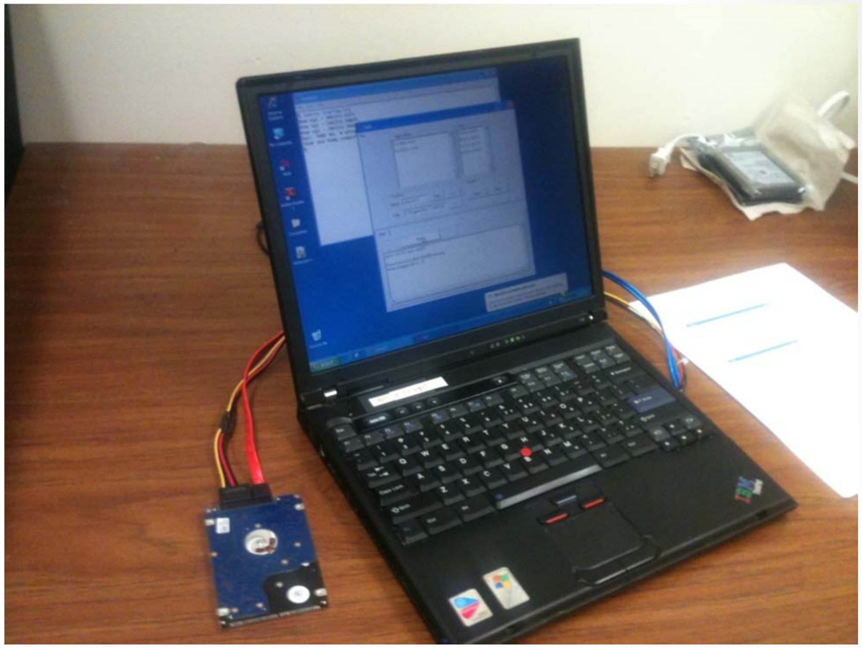
Card

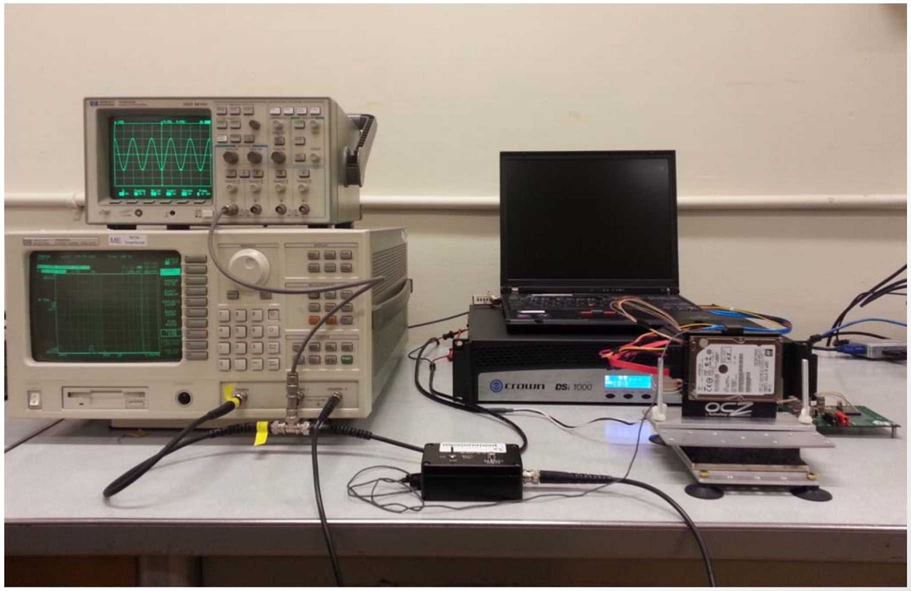
VCM Current

Additional hardware

HDD Toolkit

- Power up
- Initialization
- Reset
- Seek to ID/OD/MD
- Digital PES available
- External controller signal can be injected



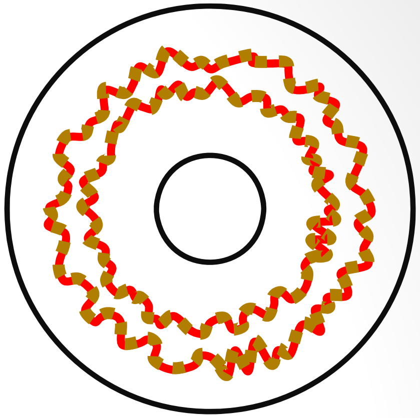


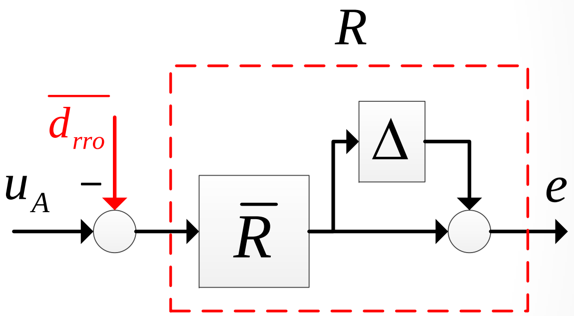
Controller implementation

- A real time embedded system has been developed on the DSP hardware using CCS.
- This system includes: initialization of system's interrupt, memory management and peripherals IO (timer, SPI module, PLL module etc.)
- The system successfully reads the digital PES from the HDD and sends a control signal through DAC.
- The interaction between the HDD, DSP system and DAC has been tested. The RRO following controller, which only relies on the PES signal, is ready to be implemented.
- Code has been optimized significantly to save computation time.

Future Work

- Develop adaptive algorithms for 2D RRO variation.





- Adaptive compensation for mismatch (temperature and manufacturing)

- Implement and evaluate the algorithms

Thank you

Any Questions?