Equalization Tuning of the PCIe Physical Layer by Using Machine Learning in Industrial Post-silicon Validation

F. E. Rangel-Patiño[1,2], A. Viveros-Wacher[2], C. Rajyaguru[3], E. A. Vega-Ochoa[2], S. D. Rodriguez-Saenz[2], J. L. Silva-Cortes[2], H. Shival[3], and J. E. Rayas-Sánchez[1]

[1]Department of Electronics, Systems, and Informatics, ITESO – The Jesuit University of Guadalajara, Tlaquepaque, Jalisco, 45604 Mexico

[2]Intel Corp. Zapopan, Jalisco, 45019 Mexico

[3]Intel Corp. Folsom, CA, 95630 USA

Outline

- Introduction to PCIe PHY tuning
- A machine learning (ML) proposal
- Unsupervised and supervised ML
- PHY tuning and optimization
- Clustering and GPR modeling results
- PHY optimization formulation and results
- Conclusions

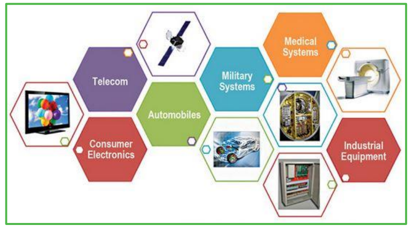
IMS
Connecting Minds. Exchanging Ideas.

The Journey of PCIe

Servers

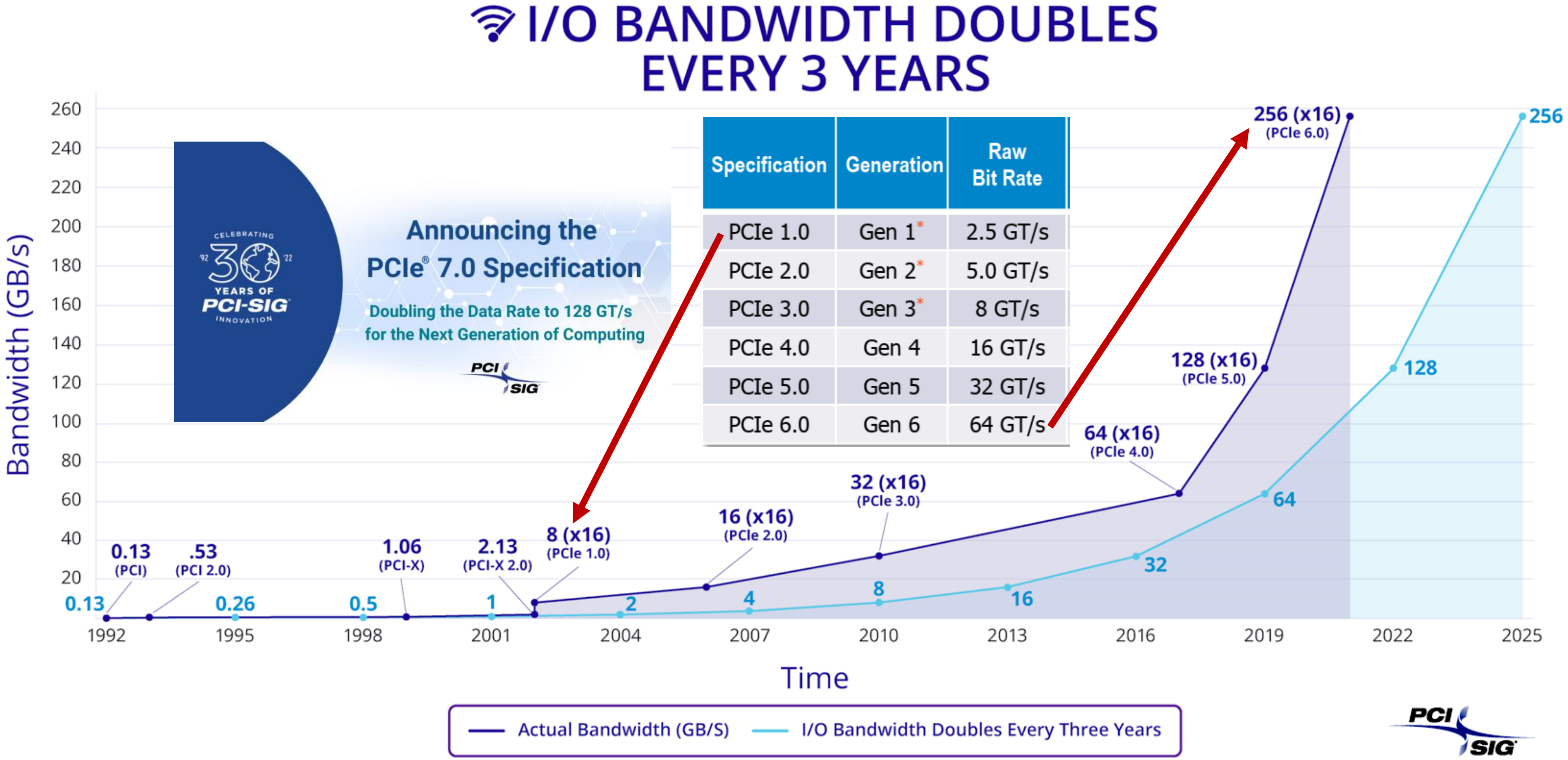
Storage

Communications



Embedded

Continuous improvement in data rates and usage models



Specification	Generation	Raw Bit Rate
PCIe 1.0	Gen 1*	2.5 GT/s
PCIe 2.0	Gen 2*	5.0 GT/s
PCIe 3.0	Gen 3*	8 GT/s
PCIe 4.0	Gen 4	16 GT/s
PCIe 5.0	Gen 5	32 GT/s
PCIe 6.0	Gen 6	64 GT/s

PCI SIG

Usage of PCIe technology continues to grow driven by BW demand

PCIe Tuning Complexity

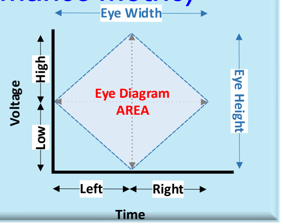
- PHY tuning is very time consuming
- Typically based on exhaustive search methods to find the "best" Tx and Rx EQ settings

1 Identify tuning "knobs"

PHY tuning may involve hundreds of Tx/Rx knobs to be adjusted

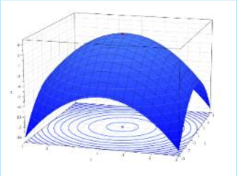
2 Define the FOM (i,e Performance metric)

Define a metric to qualify the recipe ingredients, which will be used by the optimization algorithms

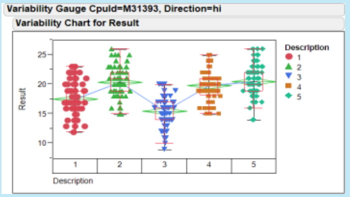


3 Knobs optimization to maximize FOM

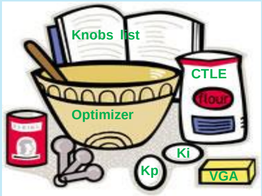
Find the best knobs values combination that optimizes the FOM



4 Confirm runs to verify results



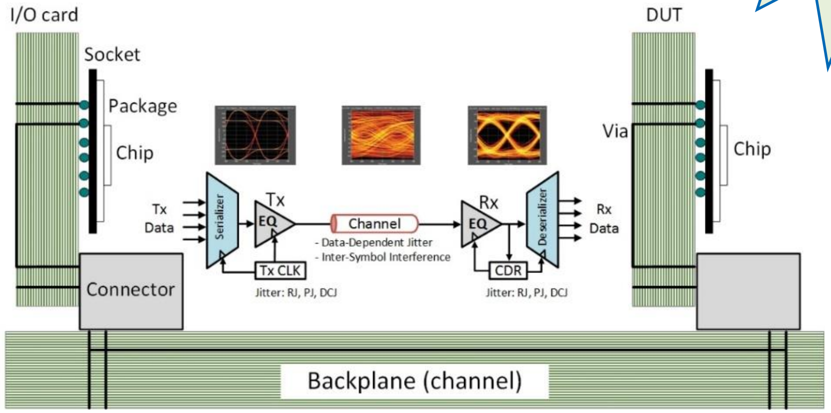
5 Recipe is done

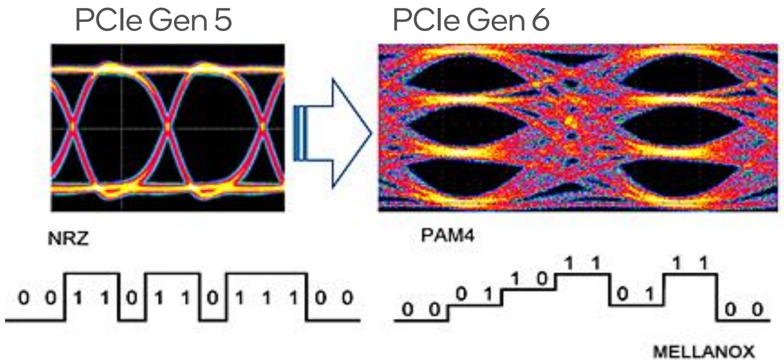


Challenge: make PHY tuning inexpensive by reducing the number of lab measurements

Margins variability

Higher data rates

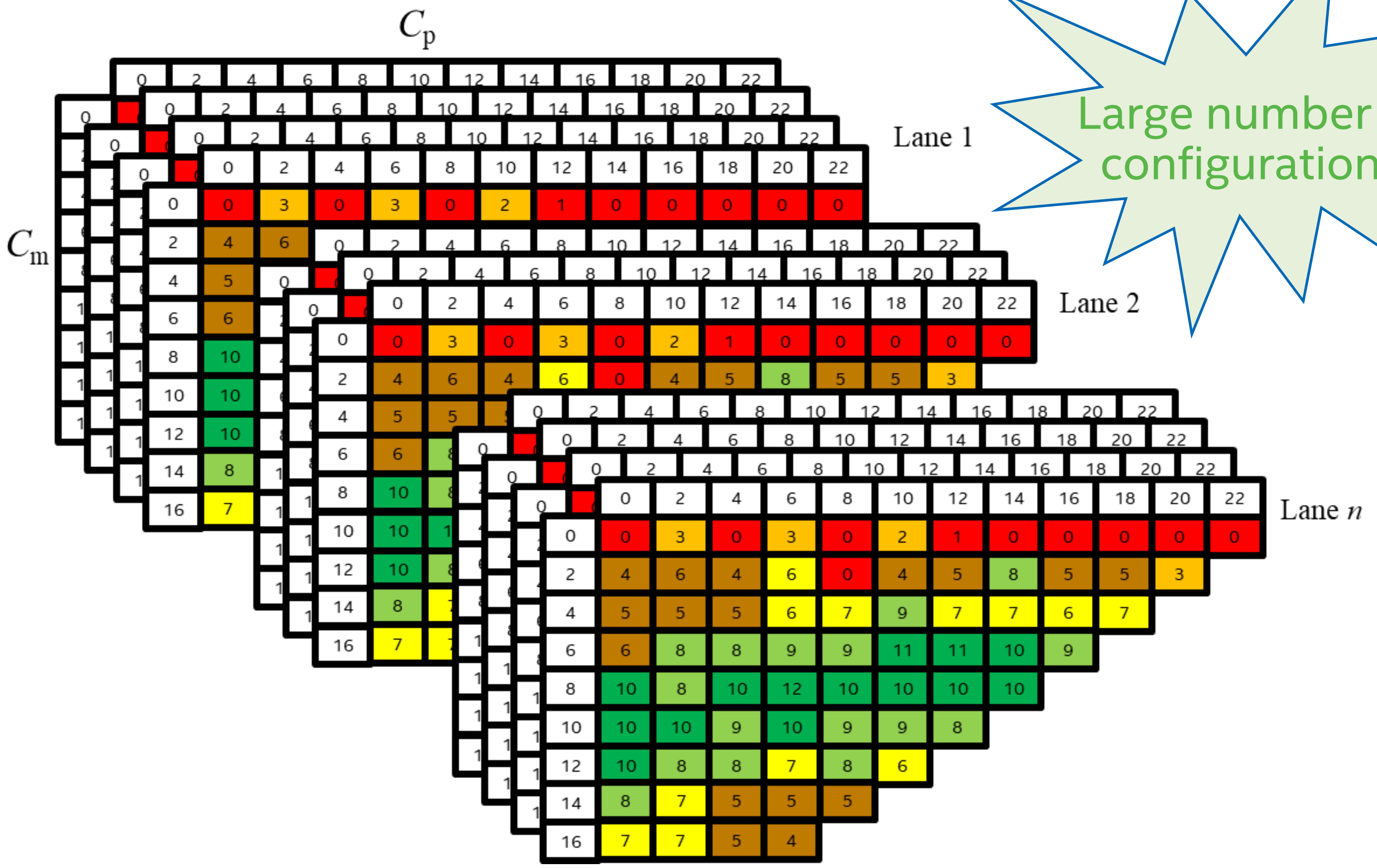




Challenge: provide higher margins on a wide variety of devices and channels, along with process, voltage and temperature (PVT) variation

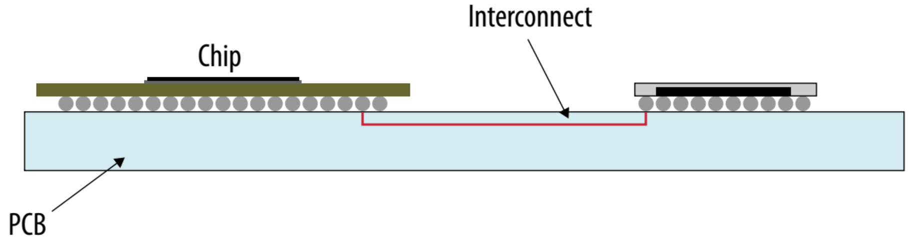
PCIe Tuning Complexity (cont.)

Many EQs maps, obtained from lab measurements, are typically employed for different lanes, channels, PVT, etc.

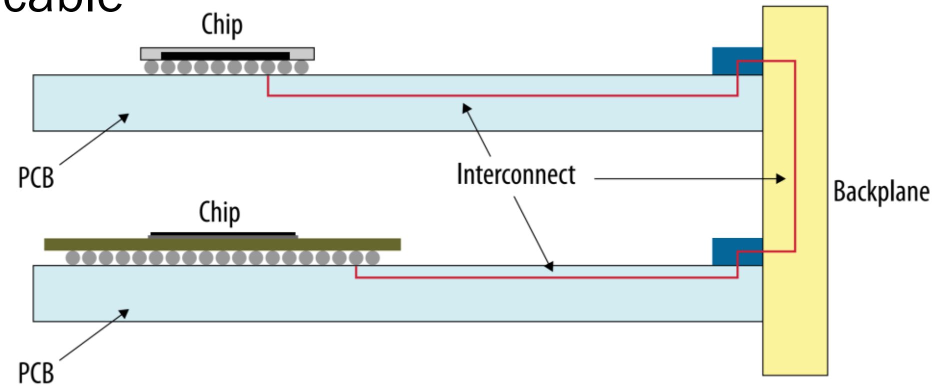


Large number of configurations

Short to mid range interconnect chip to chip within a PCB

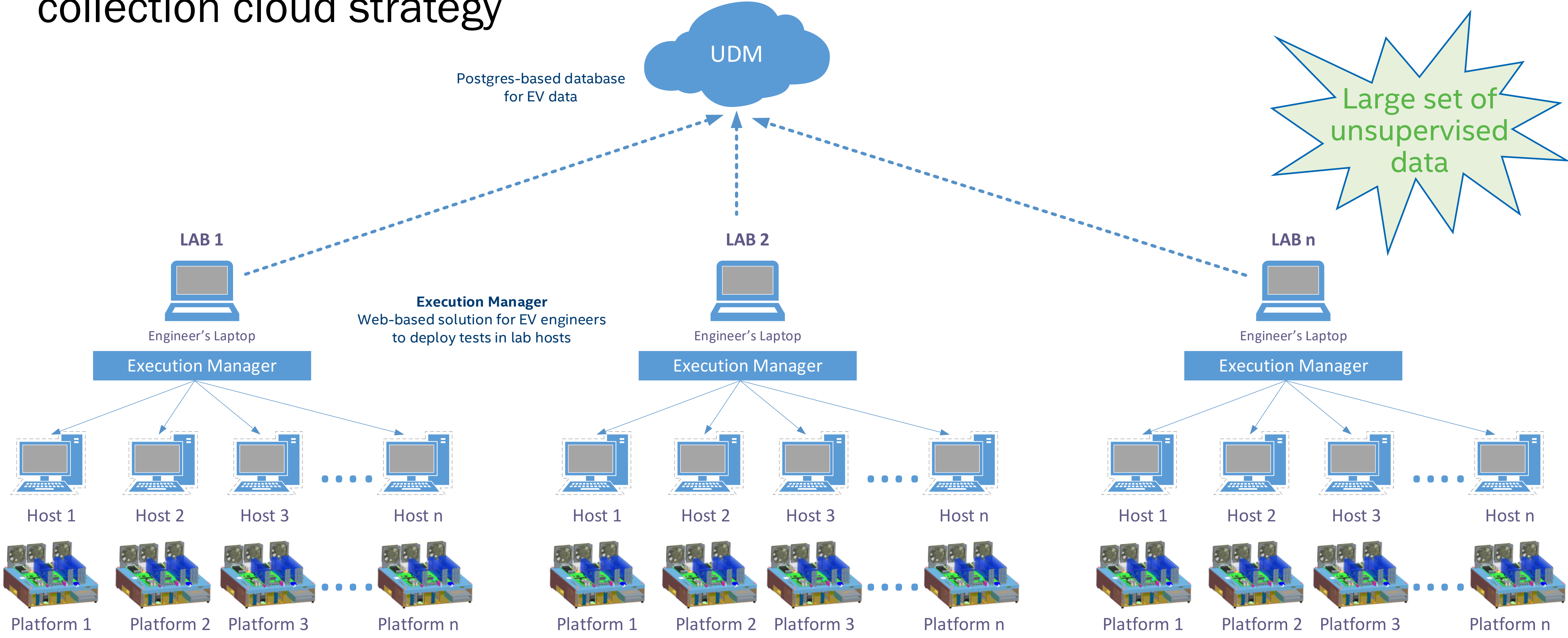


Mid to long range interconnect chip to chip across a backplane/midplane or a cable



Post-Silicon Validation Cloud Tools

To maximize resources globally, electrical validation is moving to a data-collection cloud strategy

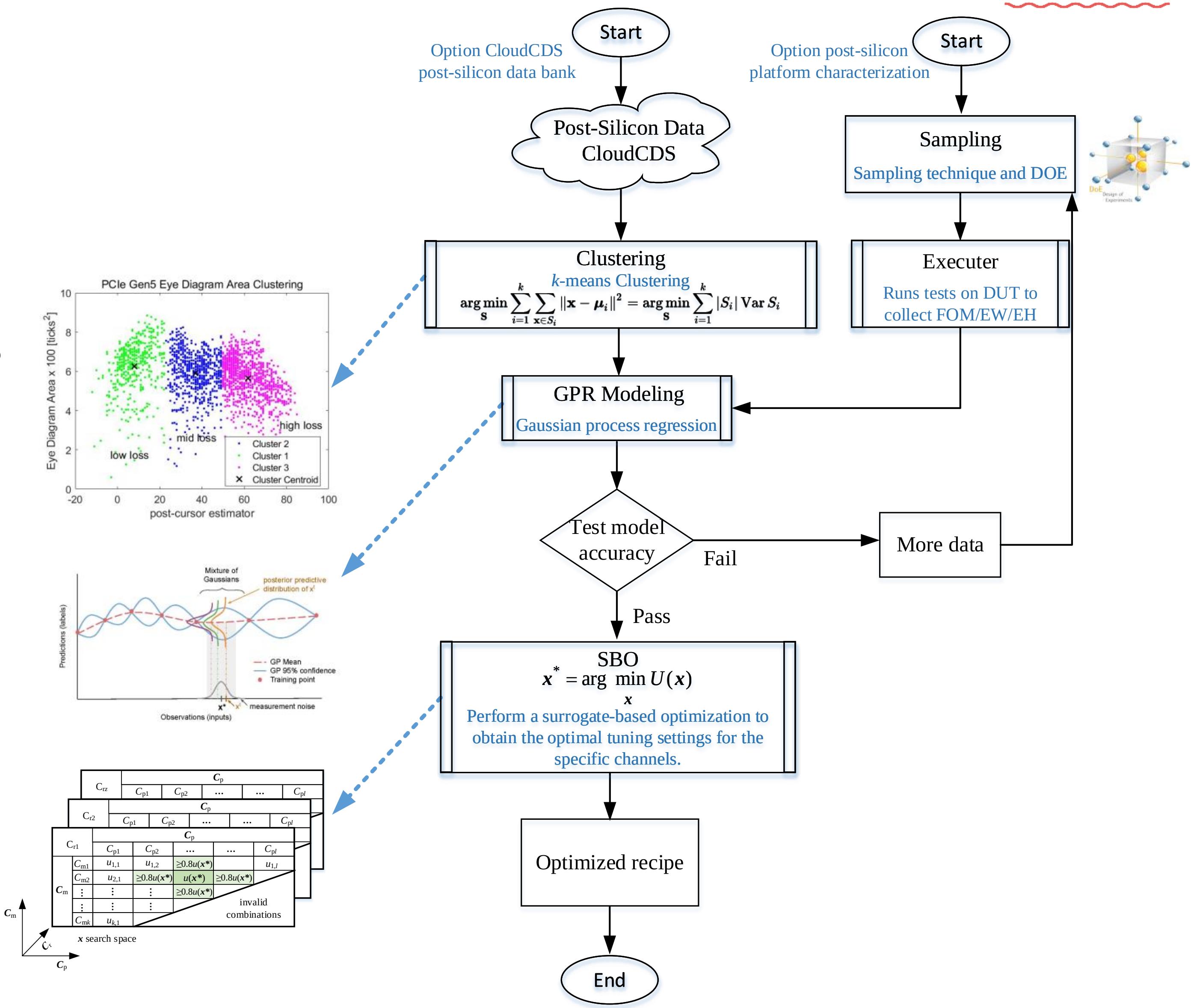


A Machine Learning Proposal

- We propose a combination of unsupervised and supervised ML techniques to tune high-speed PCIe5 NRZ and PCIe6 PAM4 designs on a real post-Si platform
- Our proposal helps to:
  – Eliminate manual PHY tuning efforts
  – Reduce engineering and debugging costs
  – Accelerate post-silicon validation
- We aim at increasing accuracy and robustness of the solution by using massive and diverse post-Si validation cloud data

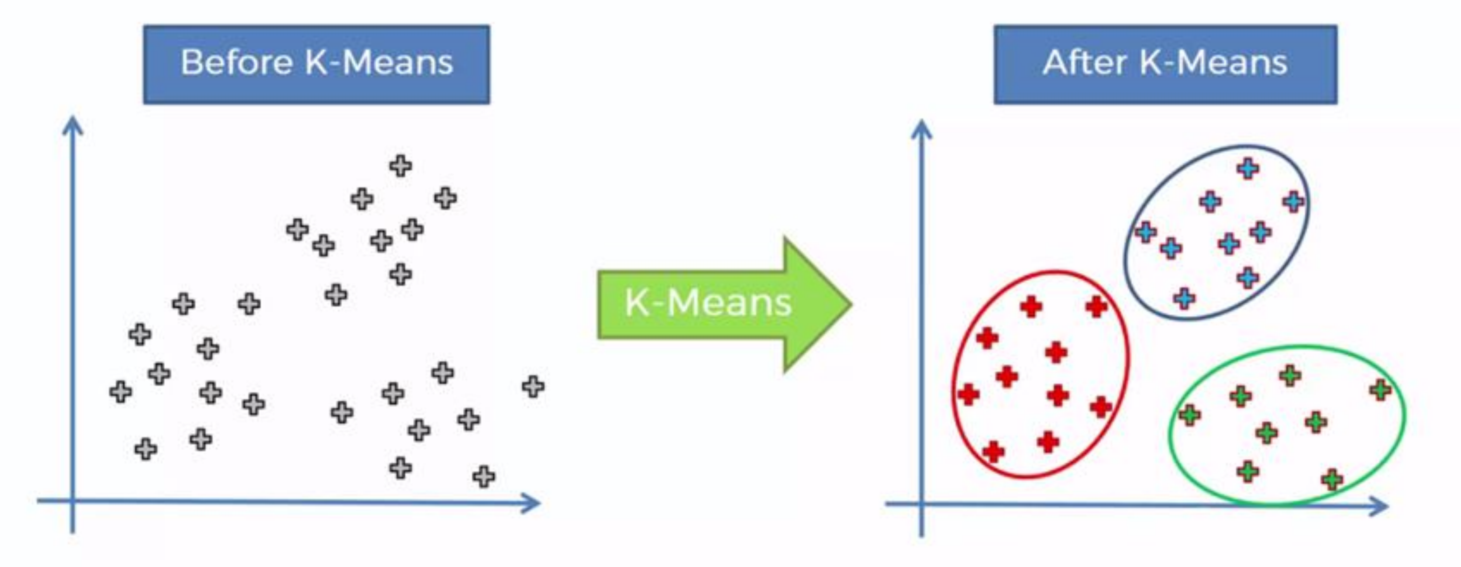
A Machine Learning Proposal (cont.)

- We use unsupervised ML modeling to cluster prior SMV and EQ data
- The clustered data segments are used to train supervised ML models
- These models approximate margins for a given combination of PHY parameters
- The models are used to find optimal PHY tuning parameters through a surrogate-based optimization (SBO)



Unsupervised ML: Clustering

- Unsupervised ML algorithms learn patterns from untagged data
- Clustering is the task of grouping similar data points
- We use the k-means clustering algorithm that partitions n observations into k clusters



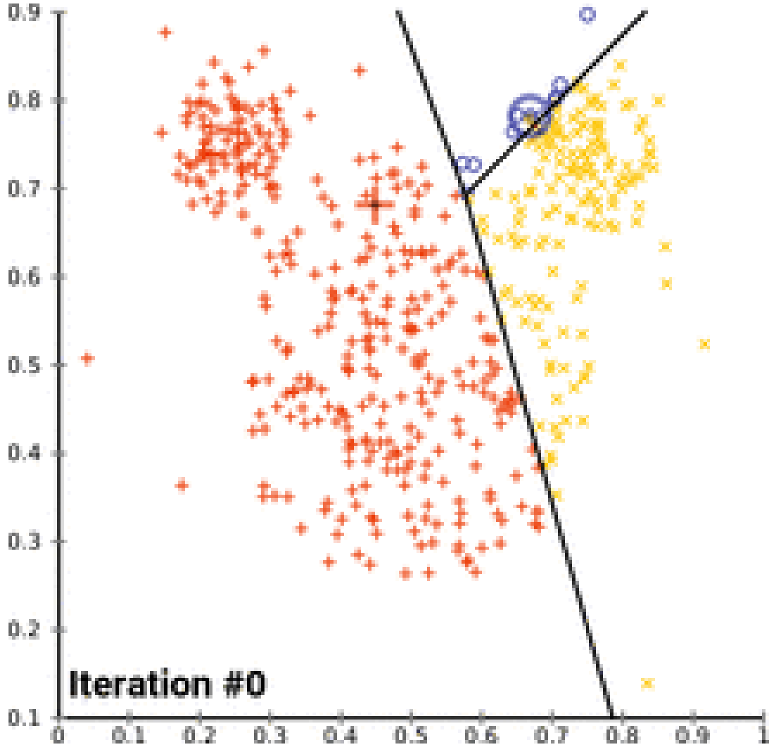
Unsupervised ML: Clustering (cont.)

Given a set of observations $(x_1, x_2, \ldots, x_n)$, where each observation is a d-dimensional real vector, k-means clustering aims to partition the n observations into k ($\leq n$) sets $S = \{S_1, S_2, \ldots, S_k\}$ to minimize the within-cluster variance

We solve:

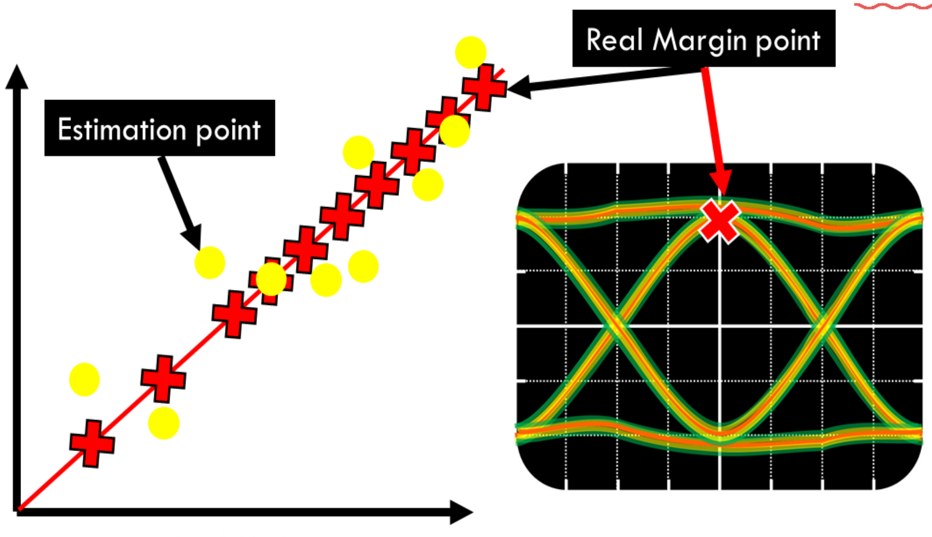
$$\underset{\mathbf{S}}{\arg\min} \sum_{i=1}^{k} \sum_{\mathbf{x} \in S_i} \|\mathbf{x} - \boldsymbol{\mu}_i\|^2 = \underset{\mathbf{S}}{\arg\min} \sum_{i=1}^{k} |S_i| \operatorname{Var} S_i$$

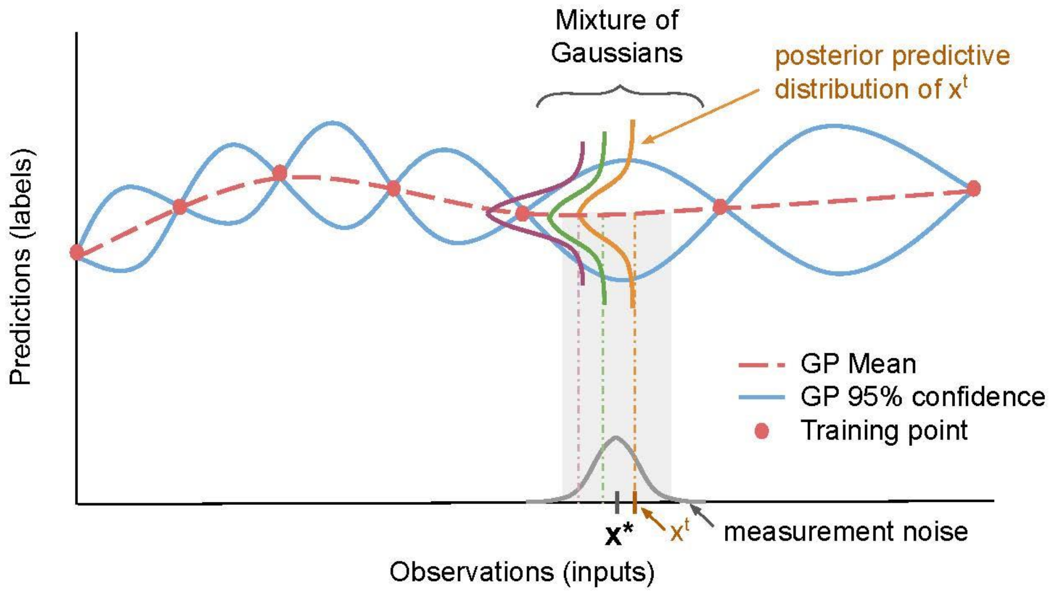
where $\boldsymbol{\mu}_i$ is the mean of points in S_i



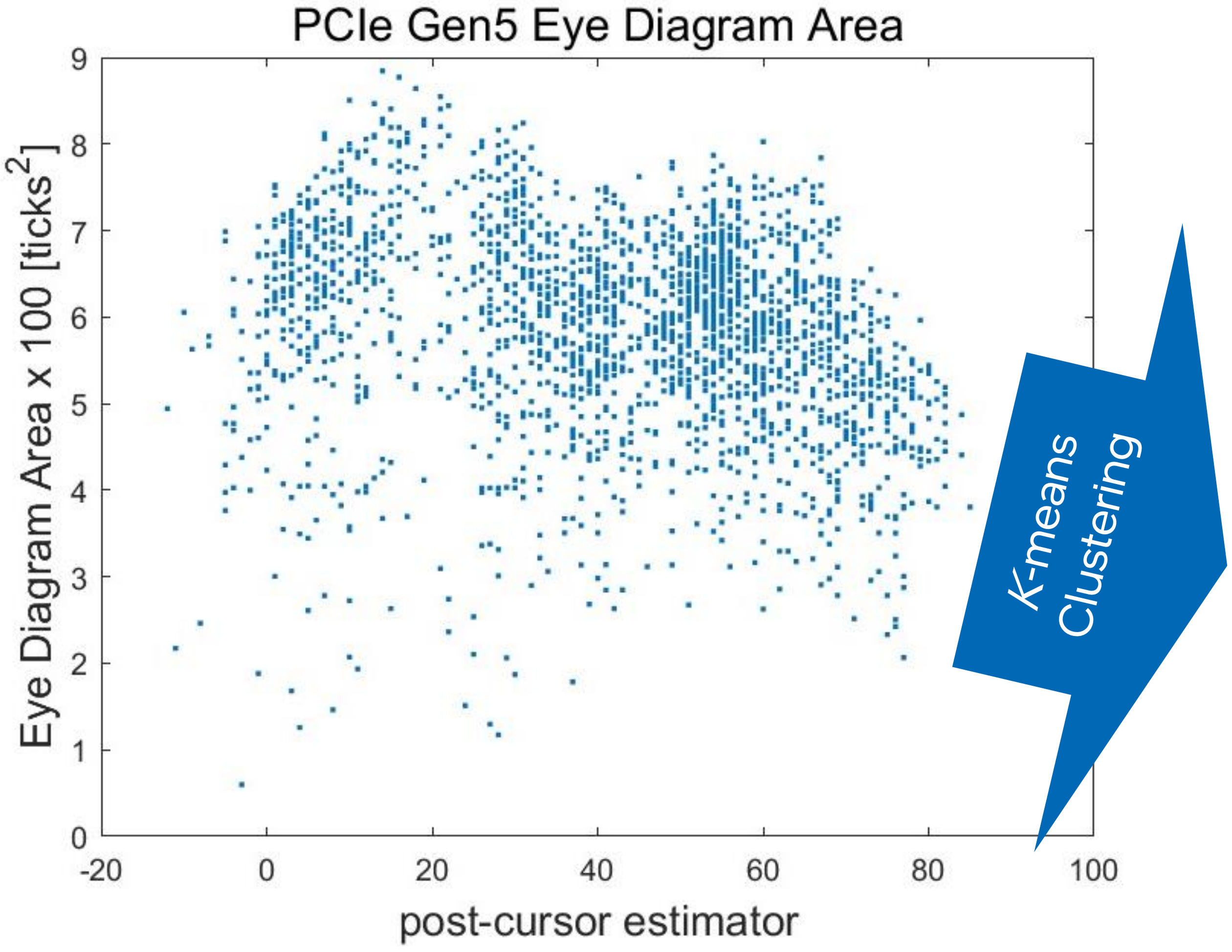
Supervised ML: GPR Modeling

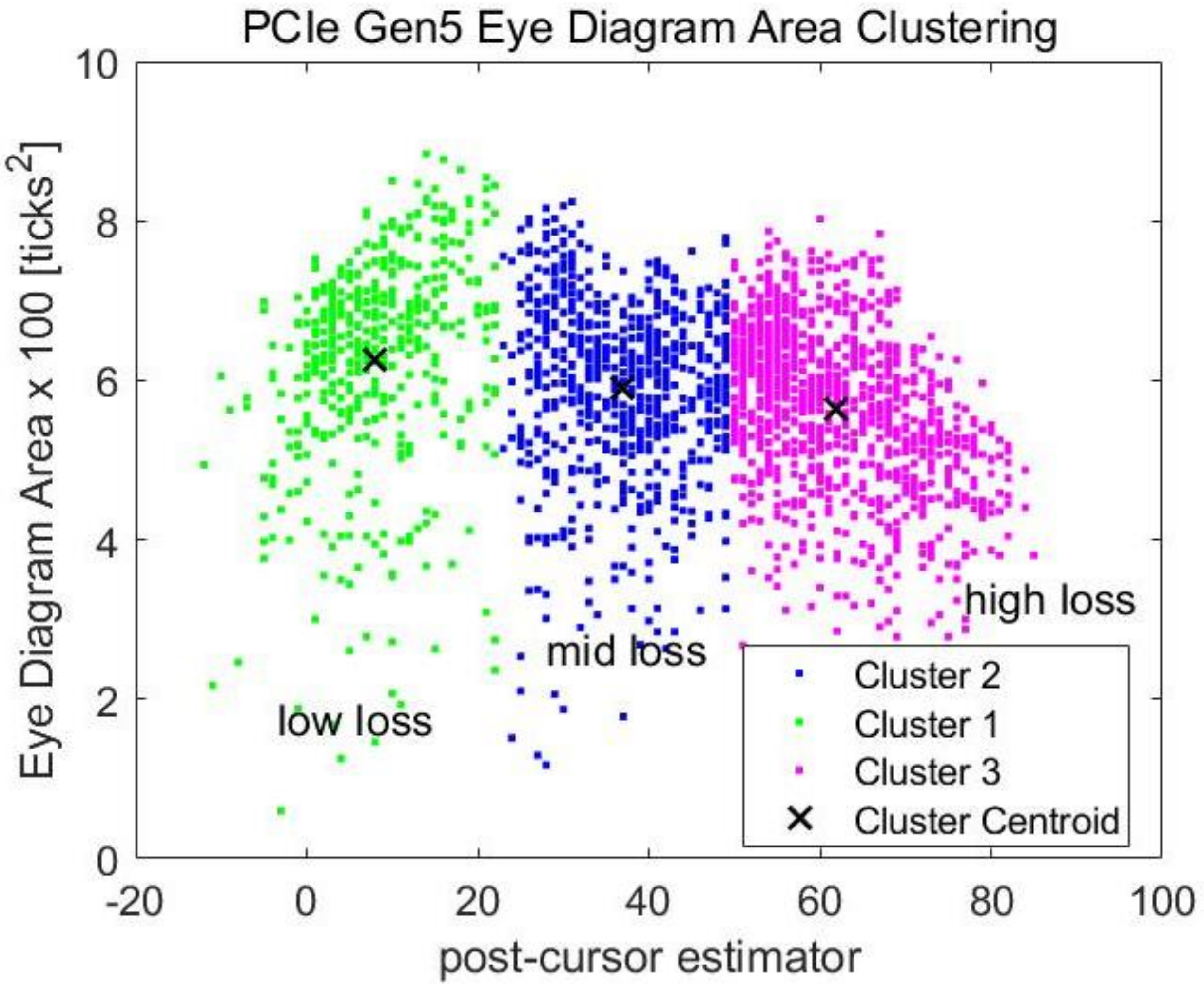
- We use supervised machine learning regression to model eye margins
- Given the large statistical fluctuations in post-Si electrical validation measurements, we use Gaussian process regression (GPR)
- GPR aims to predict not only the outputs based on inputs, but also their variability (probability distribution)



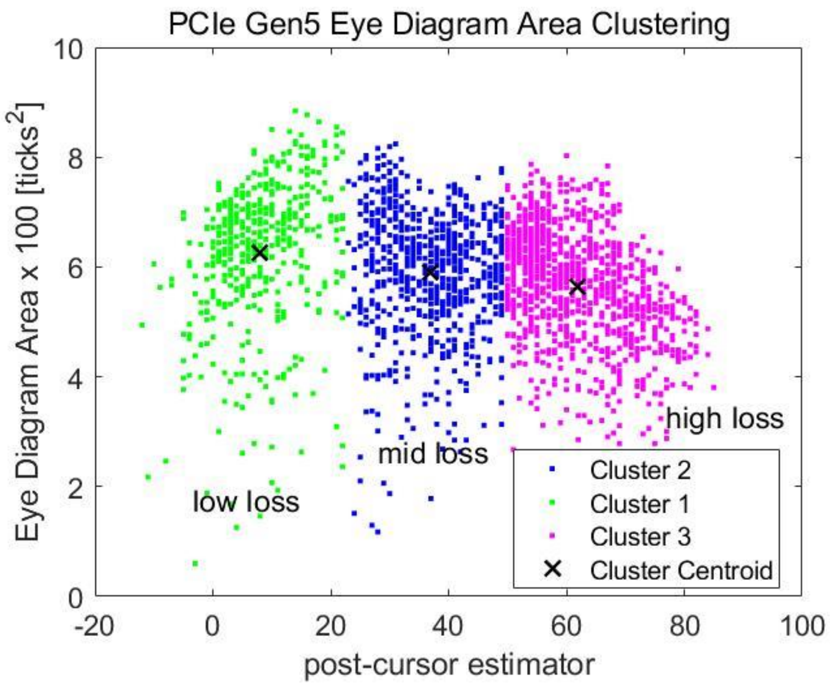


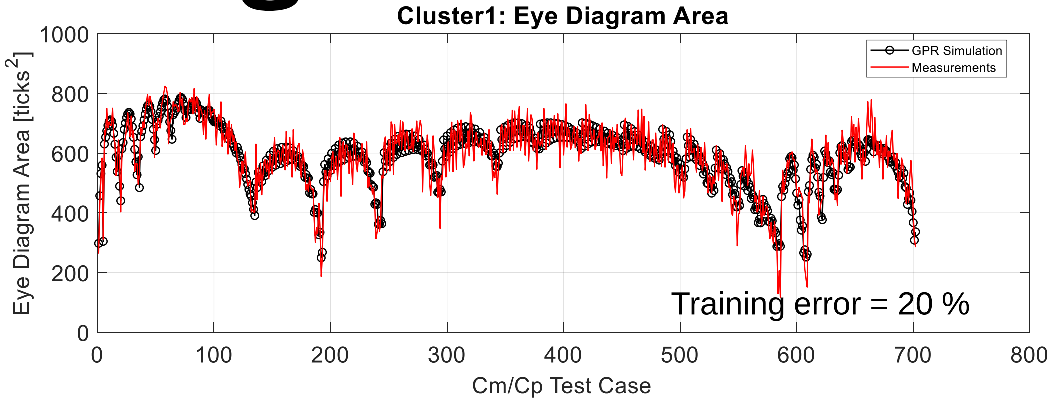
Clustering Results

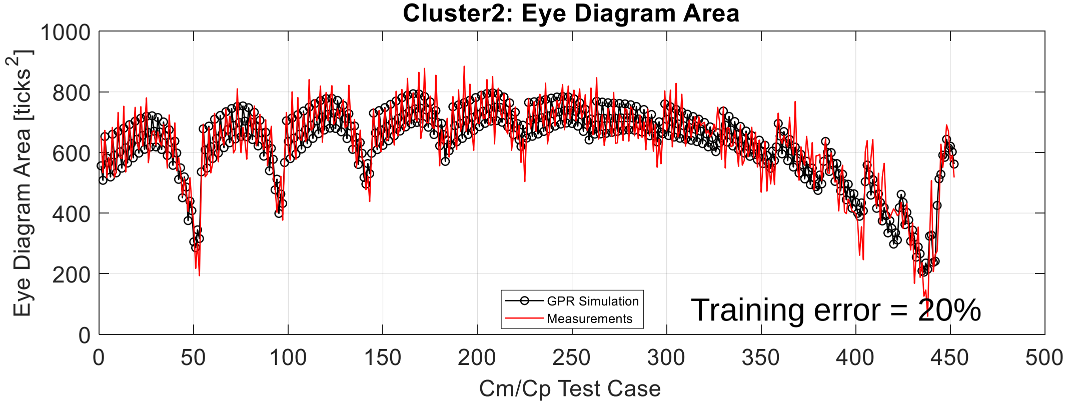


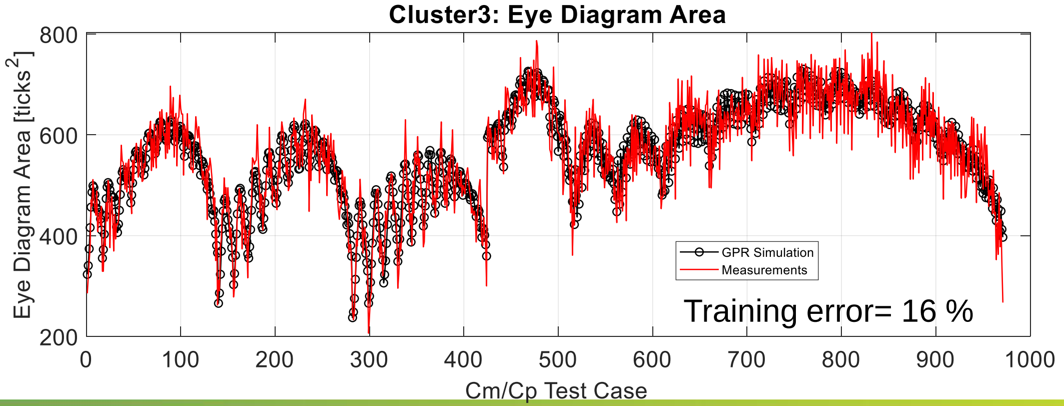


GPR Modeling Results

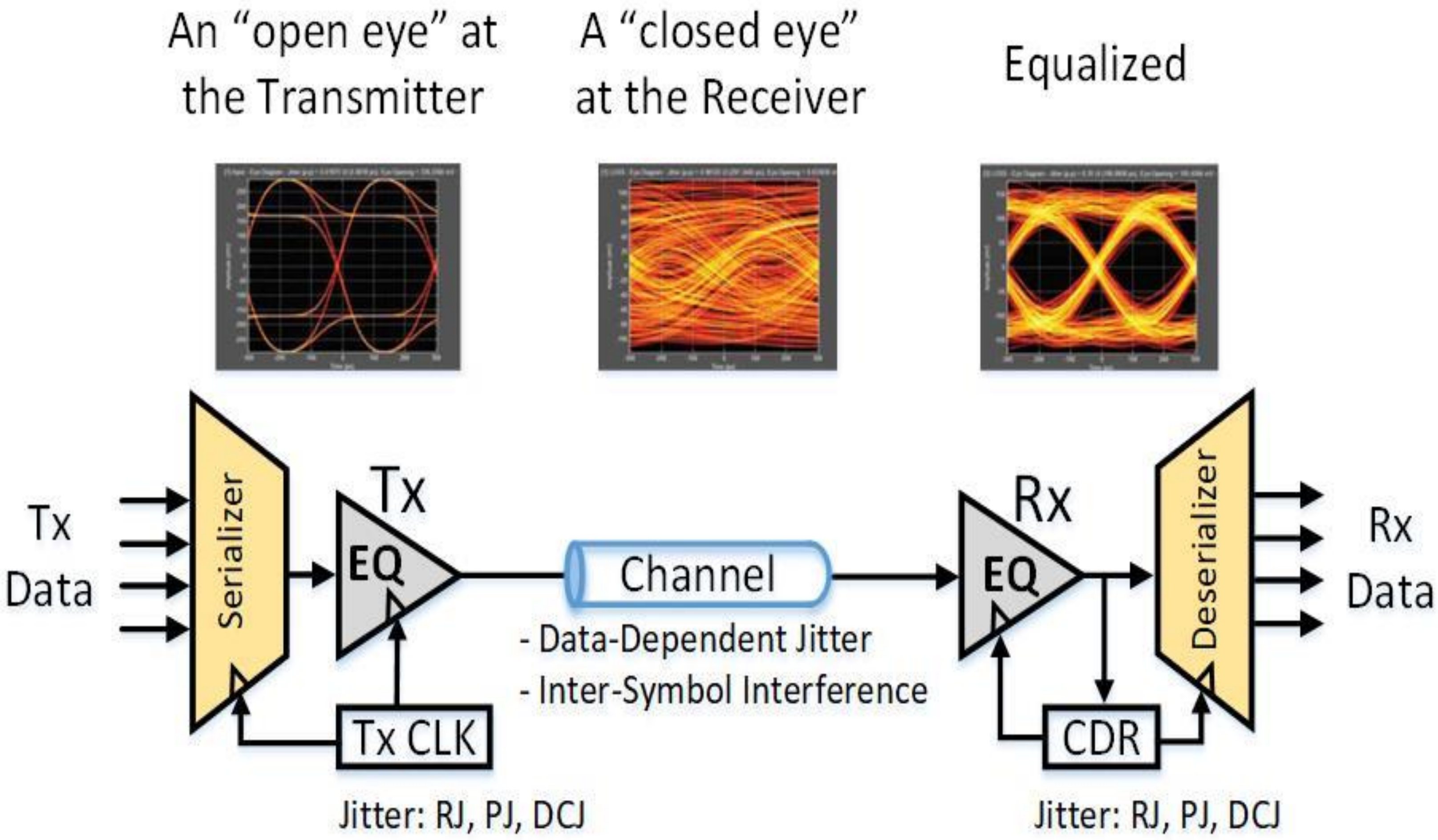








Equalization: PCIe PHY Tuning

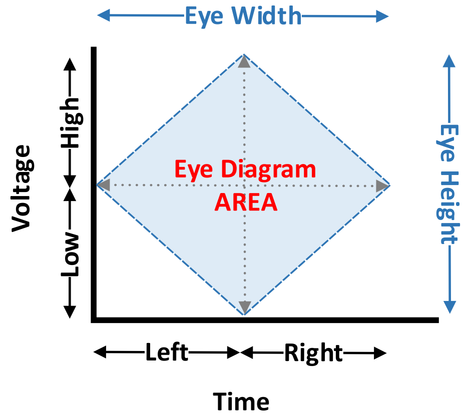


Equalization settings need to be optimized for best link performance

SBO Exploiting GPR Models

- $\boldsymbol{R}_{\mathrm{m}} \in \Re^2$: GPR response using functional margins

  $$\boldsymbol{R}_{\mathrm{m}} = \boldsymbol{R}_{\mathrm{m}}(\boldsymbol{x},\boldsymbol{\psi},\boldsymbol{\delta}) = \left[e_{\mathrm{w}}(\boldsymbol{x},\boldsymbol{\psi},\boldsymbol{\delta}) \quad e_{\mathrm{h}}(\boldsymbol{x},\boldsymbol{\psi},\boldsymbol{\delta})\right]^T$$



- We solve by direct optimization

  $$\boldsymbol{x}^* = \arg\min_{\boldsymbol{x}} U(\boldsymbol{x})$$ where $\boldsymbol{x}$ has the EQ settings

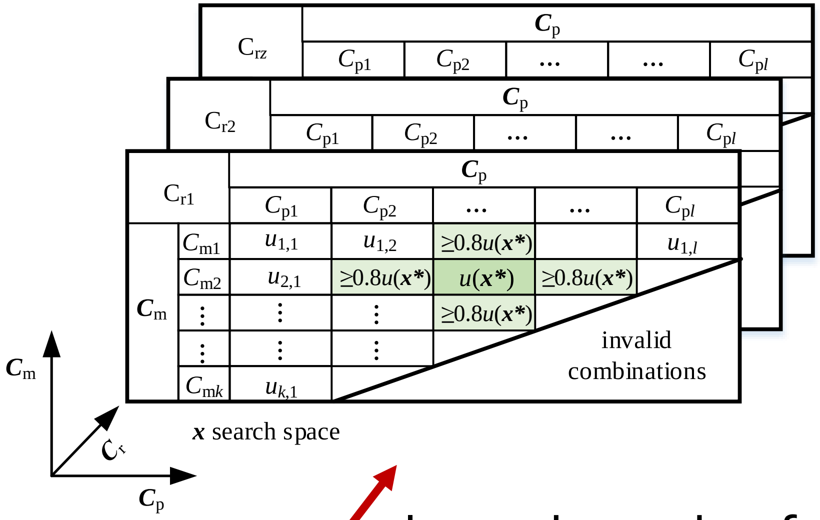
- Our unconstrained objective function is

$$U(\boldsymbol{x}) = -[e_w(\boldsymbol{x})][e_h(\boldsymbol{x})] + L(\boldsymbol{x})\left[\frac{|u(\boldsymbol{x}^{(0)})|}{\max\{\boldsymbol{l}(\boldsymbol{x}^{(0)})\}}\right]$$

where $\boldsymbol{x}^{(0)}$ is the starting point and $L(\boldsymbol{x})$ is a corner limits penalty function defined as

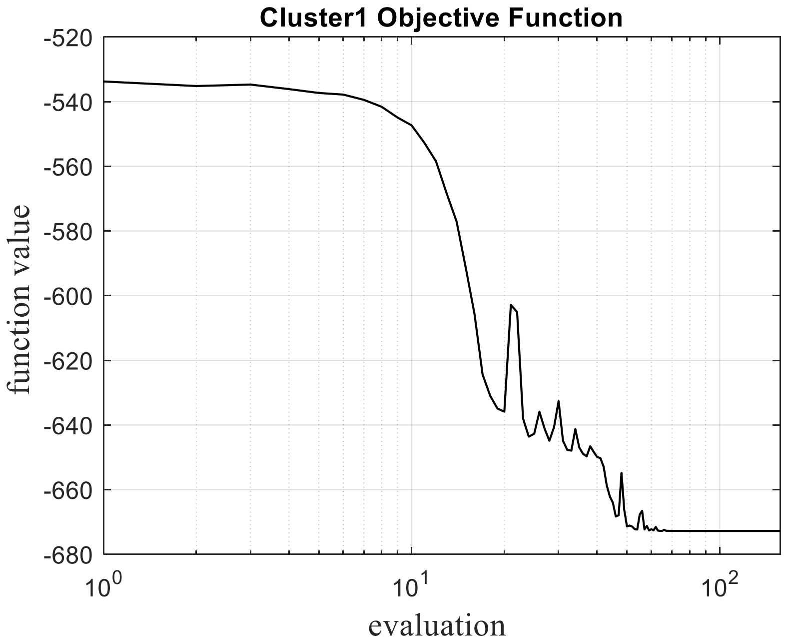
$$L(\boldsymbol{x}) = \max\{0, \max\{\boldsymbol{l}(\boldsymbol{x})\}\}$$

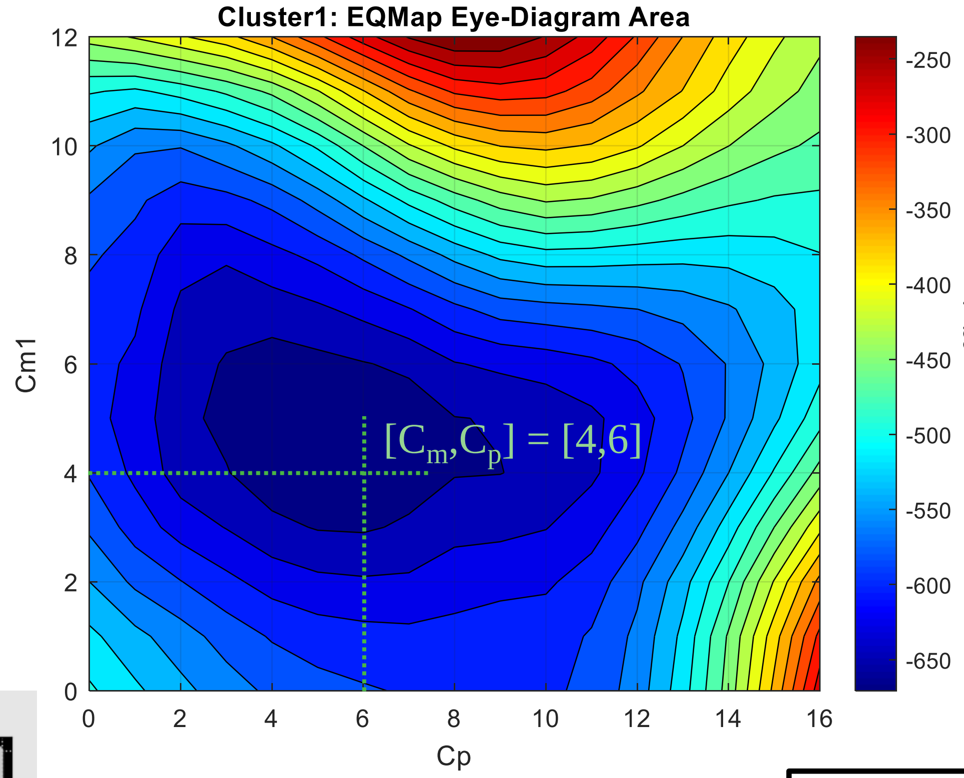
with $$\boldsymbol{l}(\boldsymbol{x}) = 0.8u(\boldsymbol{x})\begin{bmatrix}1 & 1\\ 1 & 1\end{bmatrix} - \begin{bmatrix}u(\boldsymbol{x}-[1\,0]^{\mathrm{T}}) & u(\boldsymbol{x}+[1\,0]^{\mathrm{T}})\\ u(\boldsymbol{x}-[0\,1]^{\mathrm{T}}) & u(\boldsymbol{x}+[0\,1]^{\mathrm{T}})\end{bmatrix}$$

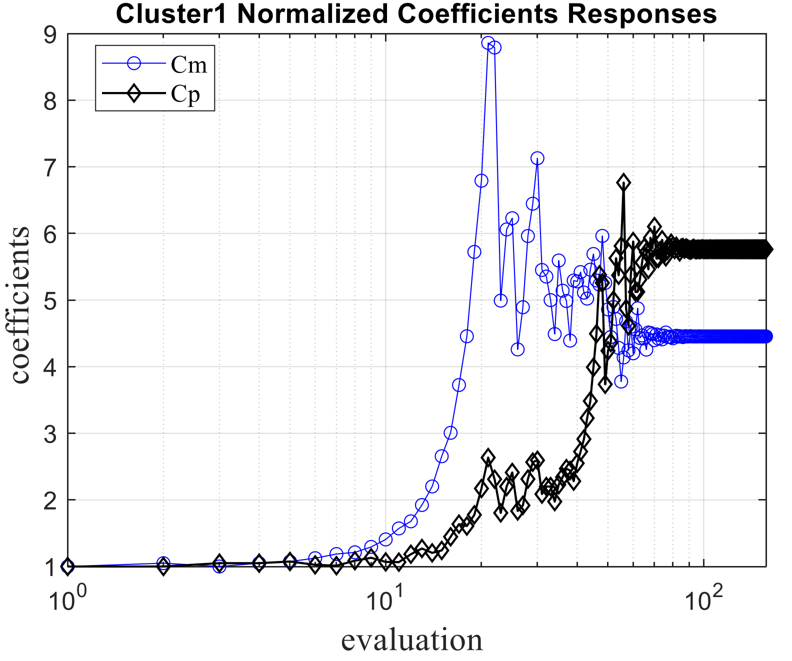


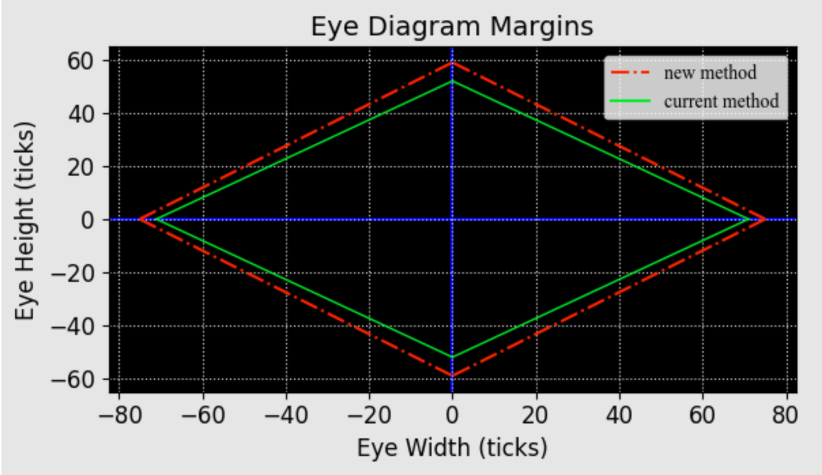
lower bounds of $0.8u(\boldsymbol{x}^*)$ in the vicinity of $\boldsymbol{x}^*$

SBO PCIe PHY Tuning Results









- 20% increase in eye diagram area (measured)
- Current method takes days, while new method takes a few hours

PCIe Gen5 – FIR EQ Opt. Summary

Cluster 1: $\boldsymbol{x}^* = [4 \quad 6]^T$, $U(\boldsymbol{x}^*) = -672.77$

Cluster 2: $\boldsymbol{x}^* = [4 \quad 8]^T$, $U(\boldsymbol{x}^*) = -719.35$

Cluster 3: $\boldsymbol{x}^* = [5 \quad 4]^T$, $U(\boldsymbol{x}^*) = -688.73$

Conclusions

- We propose a new methodology for PCIe link equalization
- Our methodology can be applied to other interfaces for PHY optimization
- We use ML techniques to cluster post-silicon data from different channels and feed those clusters to a GPR-based metamodel for each channel
- We use SBO to find the optimal PHY settings
- A significant increase in eye diagram margins is achieved, accelerating the PHY tuning process