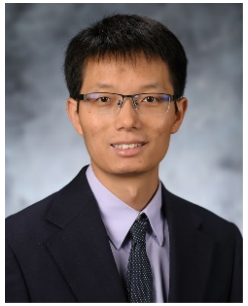
High-Dimensional Uncertainty Quantification via Active and Rank-Adaptive Tensor Regression

Zichang He
Department of Electrical & Computer Engineering
University of California, Santa Barbara, CA
E-mail: zichanghe@ucsb.edu

Zichang He

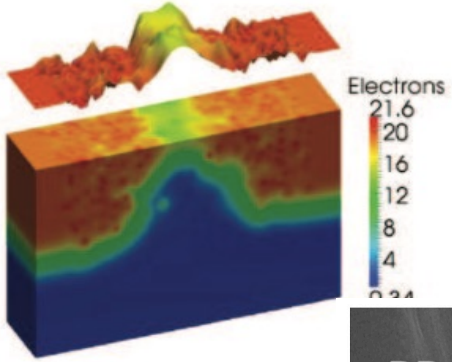


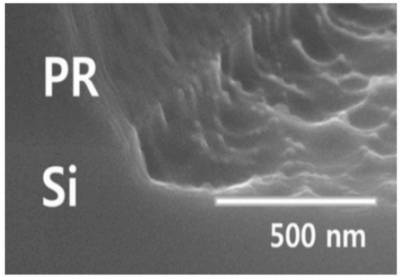
Zheng Zhang

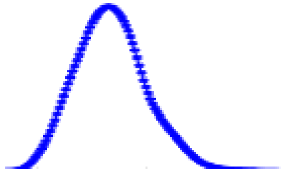
IEEE EPEPS, 2020

Motivation: Uncertainty quantification

Process variations



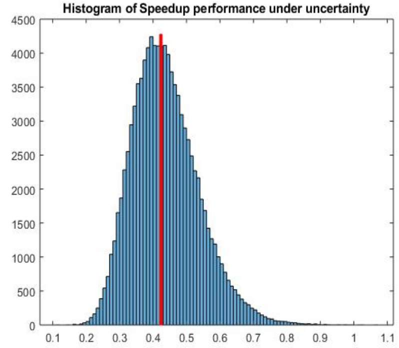


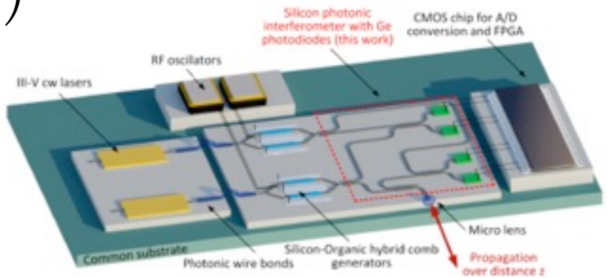


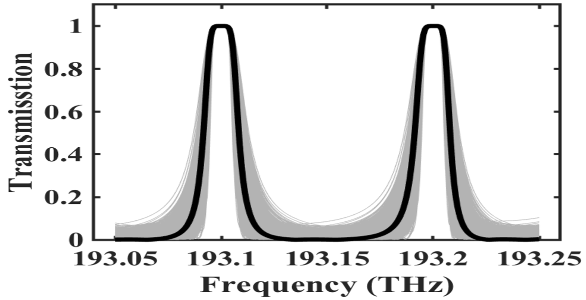
Device/ Circuit simulator

$\boldsymbol{\xi} \in \mathbb{R}^d \longrightarrow y(\boldsymbol{\xi})$

Performance uncertainties







Detailed simulations are usually expensive!

Stochastic spectral methods

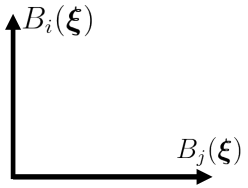
Given process variation random parameters

$$\boldsymbol{\xi} = [\xi_1, \ldots, \xi_d]$$

We want to find a surrogate model such that[1]

$$y(\boldsymbol{\xi}) \approx \sum_{j=1}^{N} X_i B_i(\boldsymbol{\xi})$$

- $B(\boldsymbol{\xi})$ is a predefined orthogonal and normalized polynomial basis.



- *X* is the unknown coefficient.
- In practical, we need to truncate $B(\boldsymbol{\xi})$, e.g. bounded by a certain polynomial order.

To construct an accurate surrogate with fewer samples.

[1] Xiu, Dongbin. *Numerical methods for stochastic computations: a spectral method approach*. Princeton university press, 2010.

Challenges in surrogate modeling

1. Curse of dimensionality:

   Exponential complexity of needed samples

   - Stochastic collocation: $O(p+1)^d$
   - Regression: $O\binom{p+d}{d} \approx O(d^p)$

Compressive sensing (Li et al.),
Hyperbolic regression (Roy et al.),
ANOVA (Zhang et al.)
…

2. Sampling method

   Don't have a golden-thumb for sampling

Our solution:

✓ Reduce # of variables to linear complexity $O(dr(p+1))$

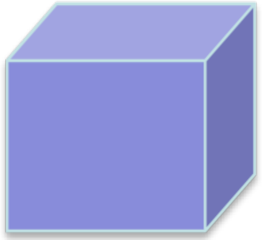
✓ An exploration-exploitation balanced sampling method

Tensor background

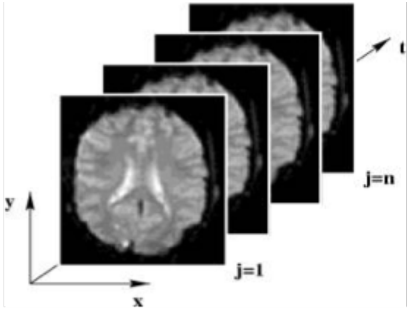
- matrix: 2-D data array

$$\mathbf{A} = [a_{i_1 i_2}] \in \mathbb{R}^{n_1 \times n_2}$$

- 3-D tensor



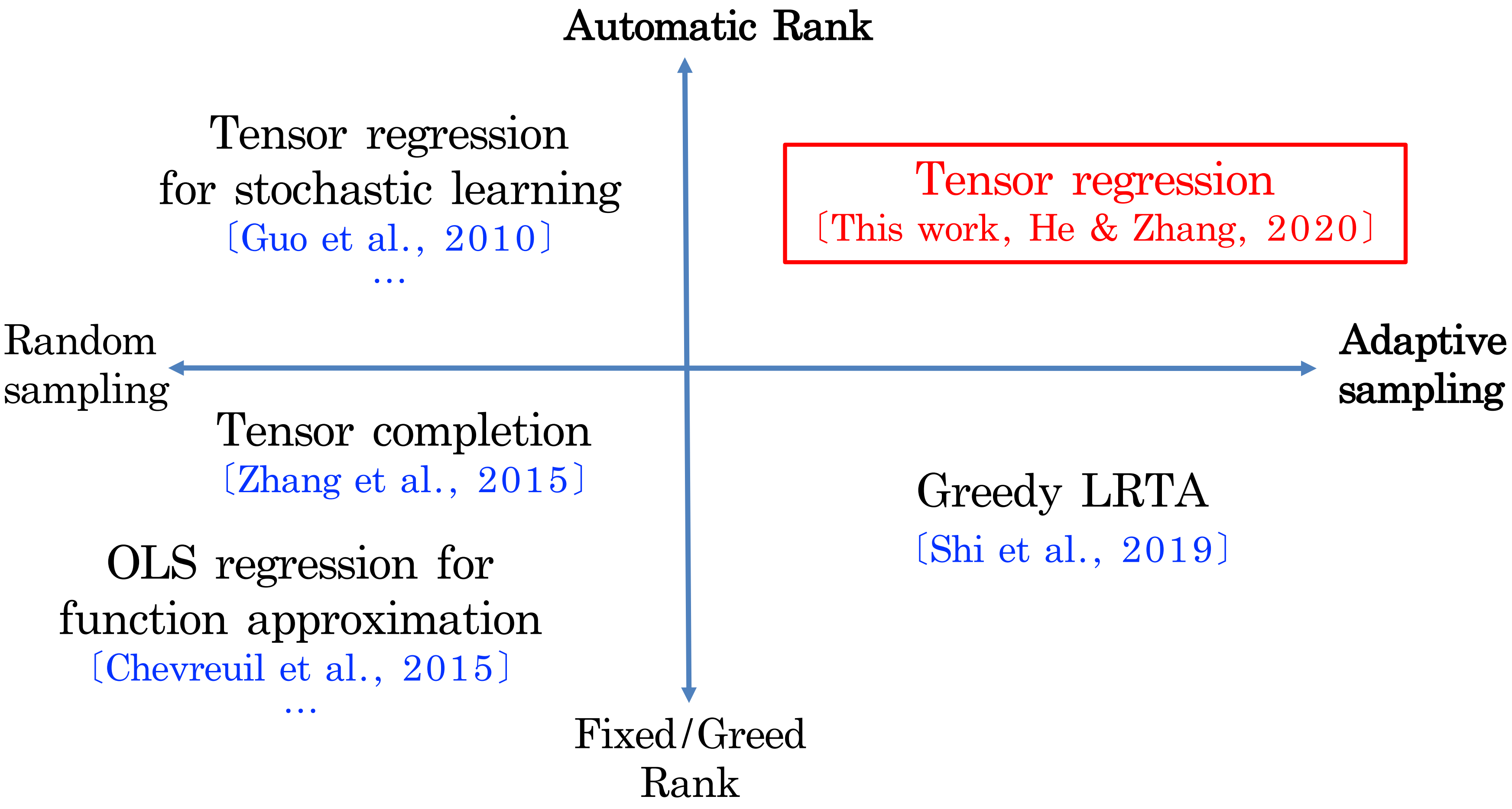
$$\mathcal{A} = [a_{i_1 i_2 i_3}] \in \mathbb{R}^{n_1 \times n_2 \times n_3}$$



- General case: d-dimensional tensor

$$\mathcal{A} = [a_{i_1 \cdots i_d}] \in \mathbb{R}^{n_1 \times \cdots \times n_d}$$

Existing works

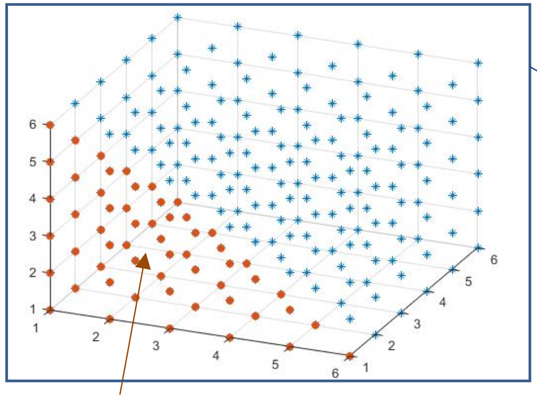


Some existing works in electronic design automation

A predefined tensor rank is usually unknown to the user

Low-rank approximation to coefficients

$$y(\boldsymbol{\xi}) \approx \sum_{j=1}^{N} X_i B_i(\boldsymbol{\xi})$$



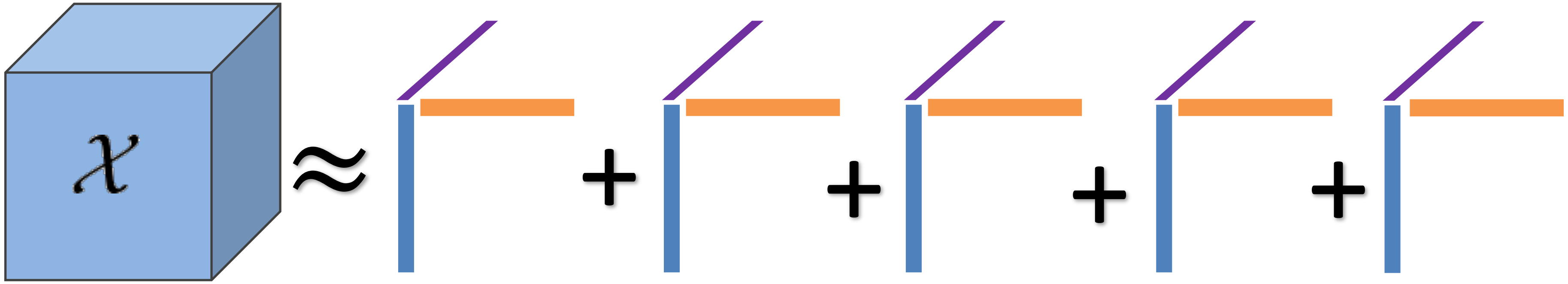
Full basis by tensor product

Total degree truncation in gPC

Ours: Full basis tensor $\mathcal{B}(\xi)$ + low-rank coefficient tensor $\mathcal{X}$.

$$y(\boldsymbol{\xi}) \approx \langle \mathcal{X}, \mathcal{B}(\boldsymbol{\xi}) \rangle$$

$$\mathcal{X} \approx \sum_{r=1}^{R} \mathbf{u}_r^{(1)} \circ \mathbf{u}_r^{(2)} \cdots \circ \mathbf{u}_r^{(d)} = [\![\mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \ldots, \mathbf{U}^{(d)}]\!]$$ (Linear complexity)

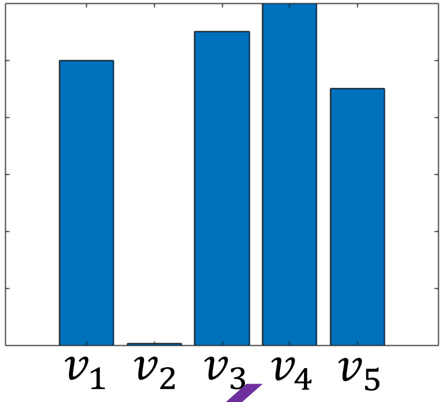


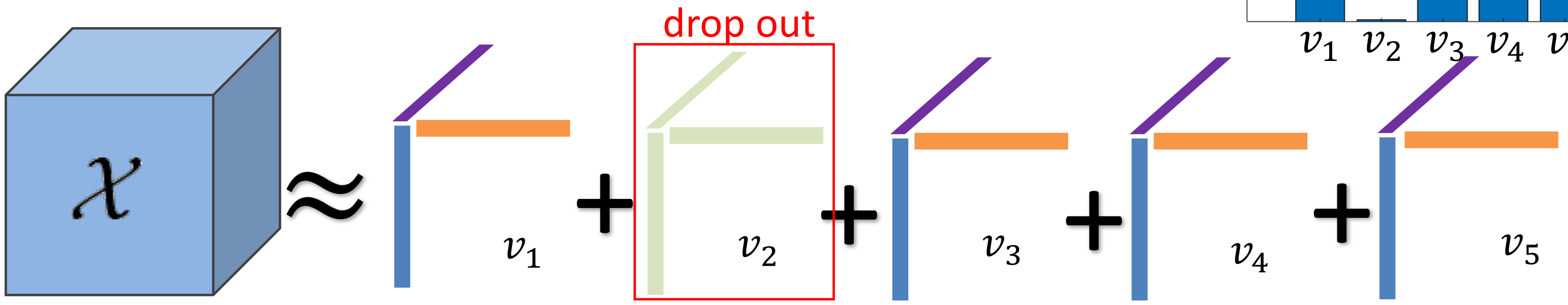
Contribution 1: Group-sparsity regularizer

$$\min_{\{\mathbf{U}^{(k)}\}_{k=1}^{d}} h(\mathcal{X}) + \lambda g(\mathcal{X})$$

$$h(\mathcal{X}) = \frac{1}{2}\sum_{n=1}^{N}\left(y_n - \langle [\![\mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \ldots, \mathbf{U}^{(d)}]\!], \mathcal{B}_n \rangle\right)^2$$

$$g(\mathcal{X}) = \|\mathbf{v}\|_q, \mathbf{v} = \left(\sum_{k=1}^{d} \|\mathbf{u}_r^{(k)}\|_2^2\right)^{\frac{1}{2}}, \quad q \in (0,1].$$





Solver to regression

$$\min_{\{\mathbf{U}^{(k)}\}_{k=1}^{d}} f(\mathcal{X}) = h(\mathcal{X}) + \lambda g(\mathcal{X}).$$

$g(X)$: non-differentiable & non-convex

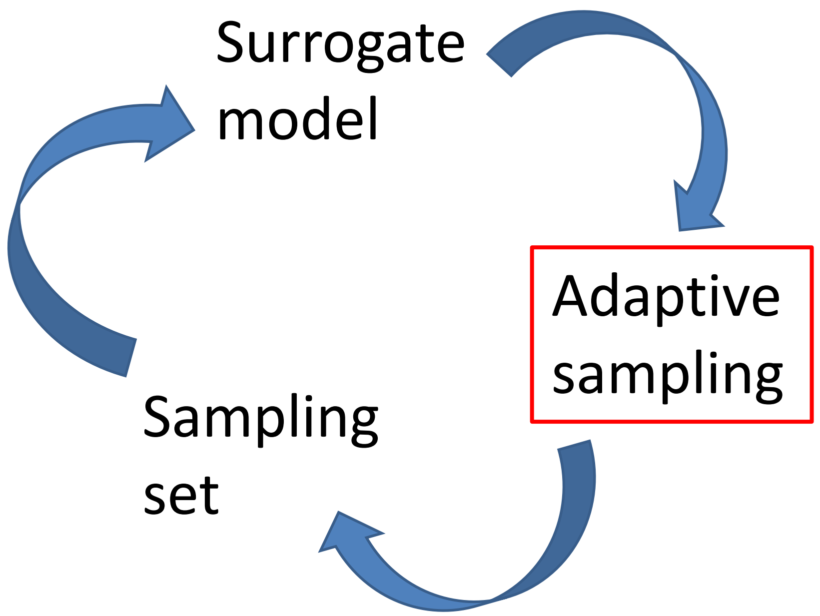
Variational inequality[1] →

Upper bound $\hat{g}(X)$: differentiable & convex

Then $f(X)$ can be solved by alternating algorithms efficiently

[1] Jenatton et al. "Structured sparse principal component analysis", in Proc. Artif. Intell. Statist., 2010

Contribution 2: Adaptive sampling method

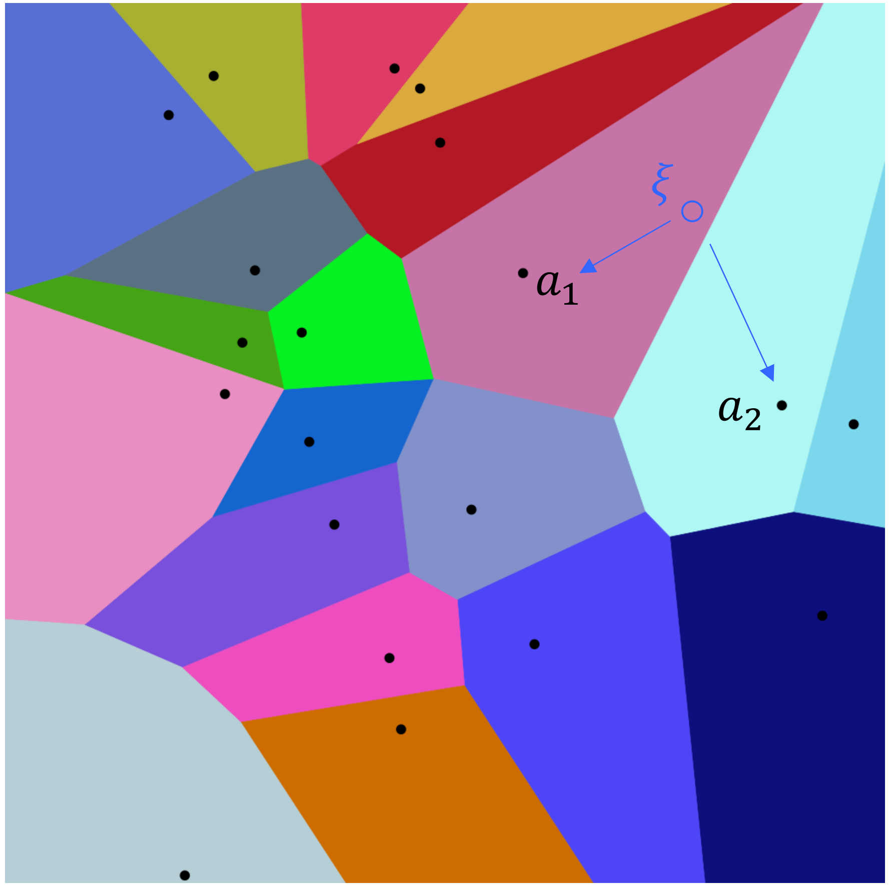


Balance between Exploration & Exploitation

- Samples should spread over the sampling space.
- More samples should focus on the critical regions.

Adaptive sampling: Exploration

Step 1: Estimate and select a Voronoi cell



[Picture Source: Wikipedia]

Voronoi diagram

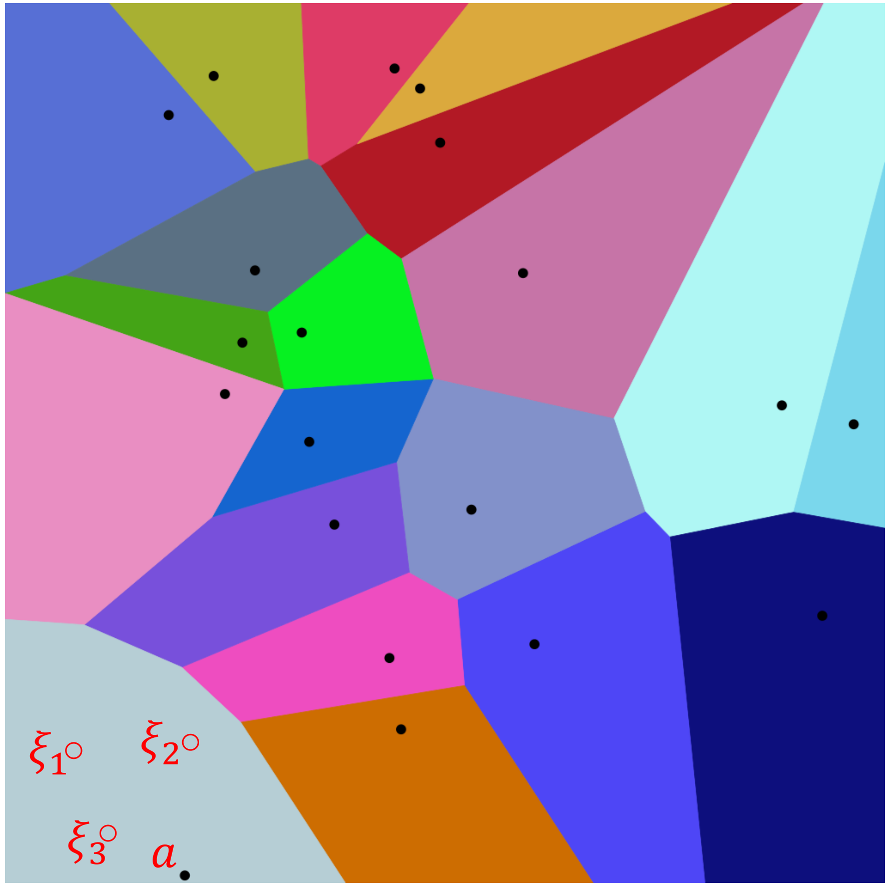
Voronoi cell $C(a_i)$ covers the region that are closest to a_i.

The volume of a cell can estimate the sampling density.

The diagram can be estimated by Monte Carlo samples.

Adaptive sampling: Exploitation

Step 2: Select a sample from one Voronoi cell



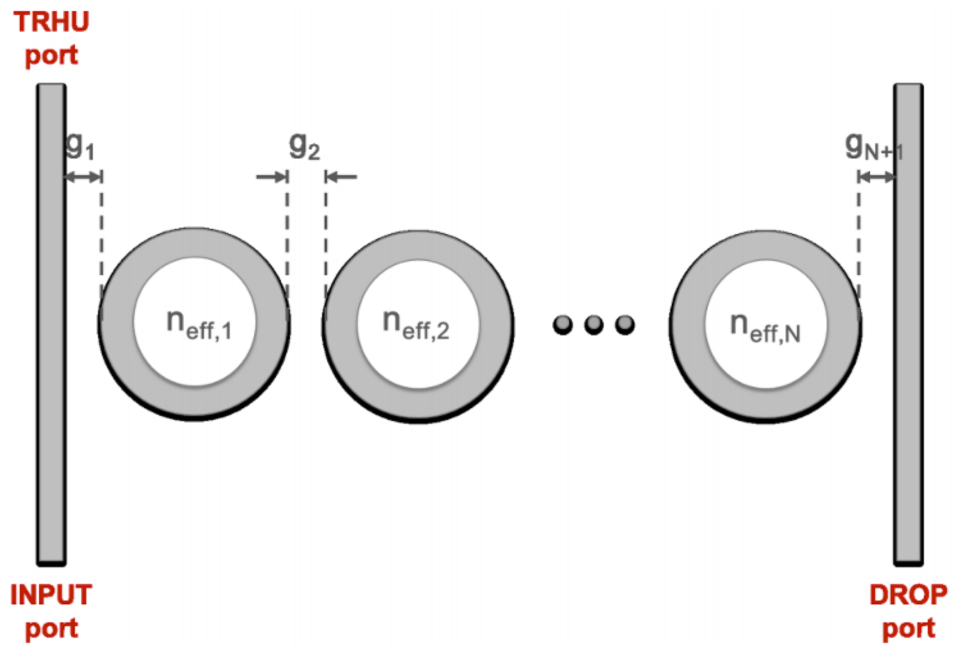
[Picture Source: Wikipedia]

Nonlinearity measure γ

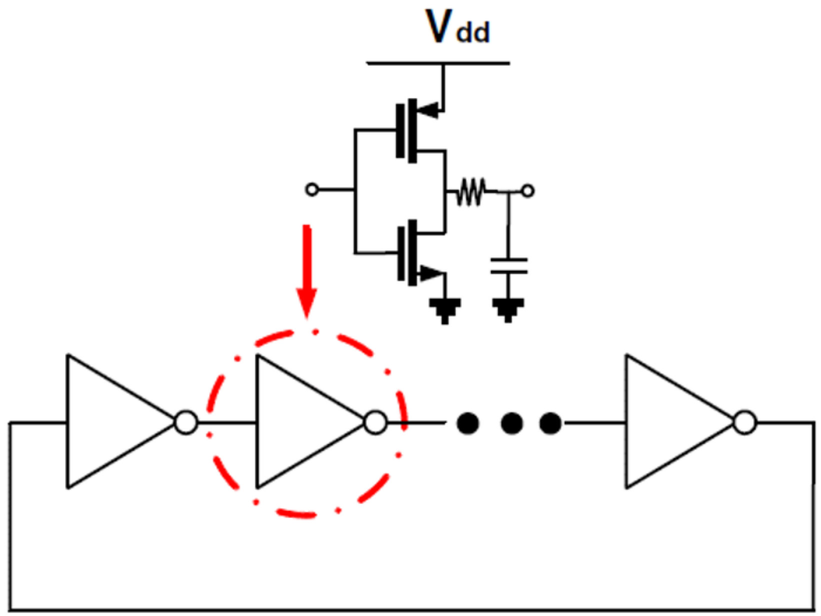
$$\gamma(\boldsymbol{\xi}) = |\hat{y}(\boldsymbol{\xi}) - \hat{y}(\mathbf{a}) - \nabla\hat{y}(\mathbf{a})^T(\boldsymbol{\xi} - \mathbf{a})|$$

The selected sample will be the most nonlinear one in a least-sampled region

Numerical Experiments

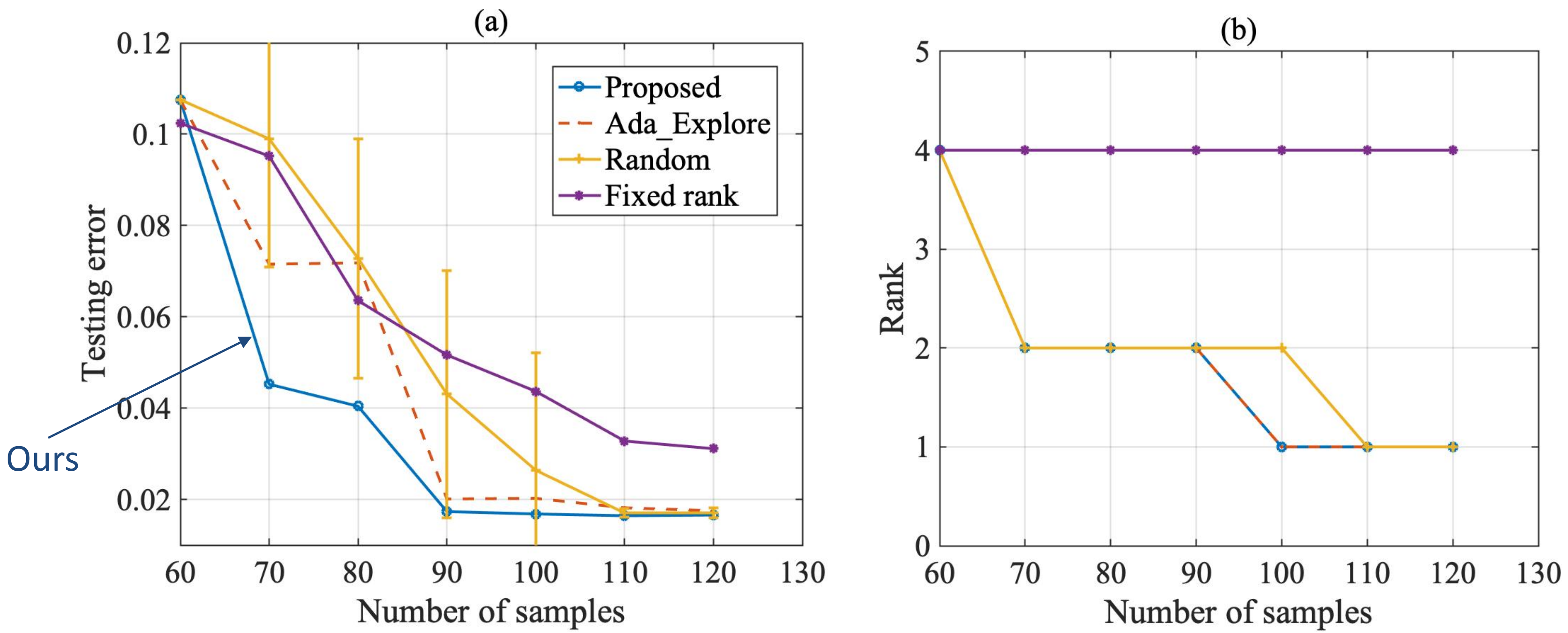


Photonic band-pass filter
(19 random parameters)



CMOS ring oscillator
(57 random parameters)

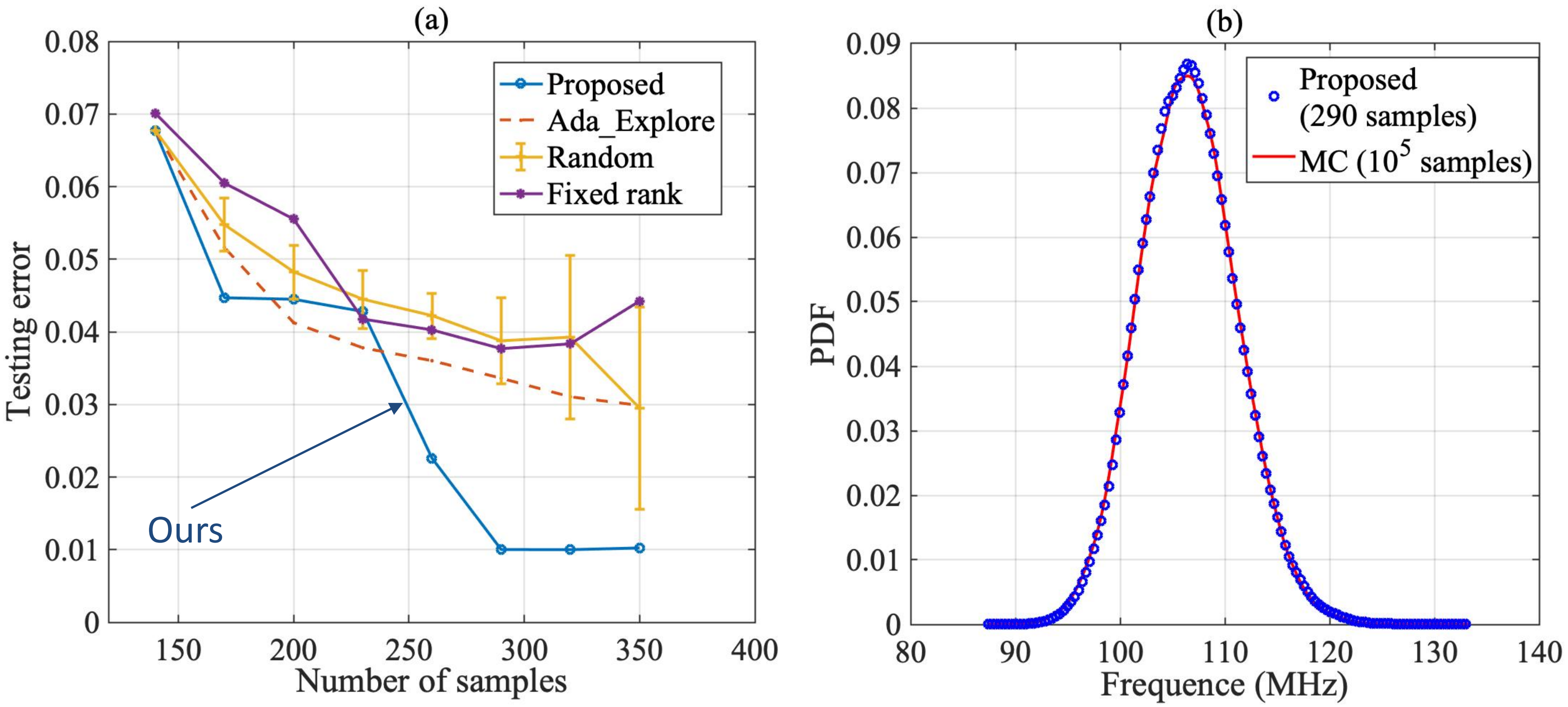
Photonic band-pass filter



10^5 MC simulations approximated by ~ 100 samples in tensor model ($\sim 10^3$x speedup)

Approximation results will be satisfied if the rank is well estimated.

CMOS ring oscillator



10^5 MC simulations approximated by ~ 300 samples in tensor model (~ 350x speedup)

Proposed adaptive sampling is effective.

Experiment: CMOS ring oscillator

Compare with a standard gPC expansion of a total degree scheme: fewer samples & better accuracy

	Proposed	**Total-degree gPC**	**MC**
# of variables	**855**	1711	N/A
# of samples	**290**	1711	10^5
Mean	106.28	106.58	106.53
Deviation	4.616	6.81	4.641
Error	**1%**	4.84%	N/A

Take-home message

A tensor regression model for high-dimensional UQ

Two technical contributions:

- Automatic rank determination via group sparsity regularization
- Adaptive sampling via Voronoi diagram

Validation on a photonics filter and a CMOS ring oscillator (up to 1000x speedup)