2017 DESIGN AND VERIFICATION™ DVCON CONFERENCE AND EXHIBITION UNITED STATES

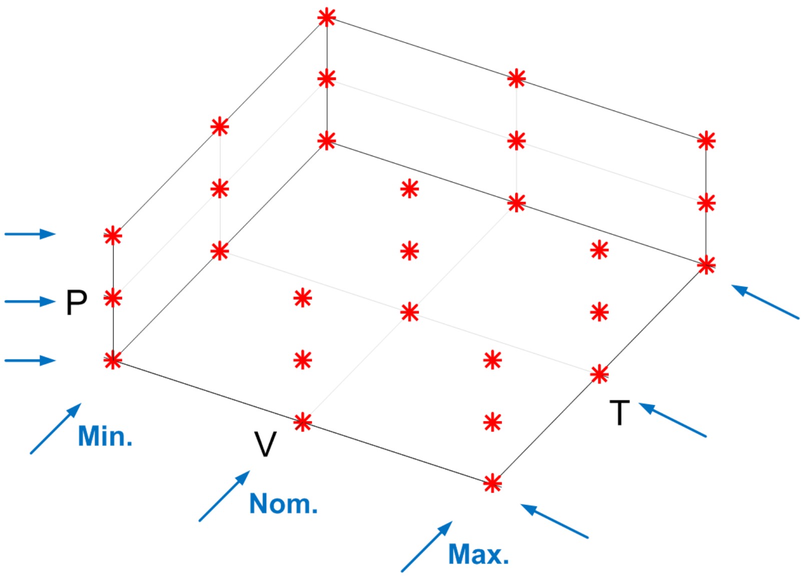
Machine Learning based PVT Space Coverage and Worst Case Exploration In Analog and Mixed-Signal Design Verification

Honghuang Lin, Zhipeng Ye, Asad Khan

TEXAS INSTRUMENTS

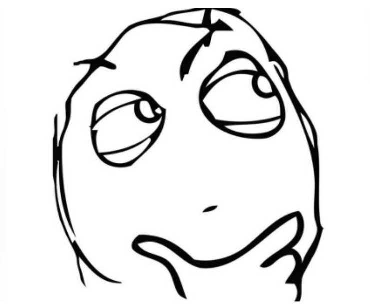
Motivation

- Verifying under PVT variations → essential in AMS verifications
- Traditional method: extreme-case corners



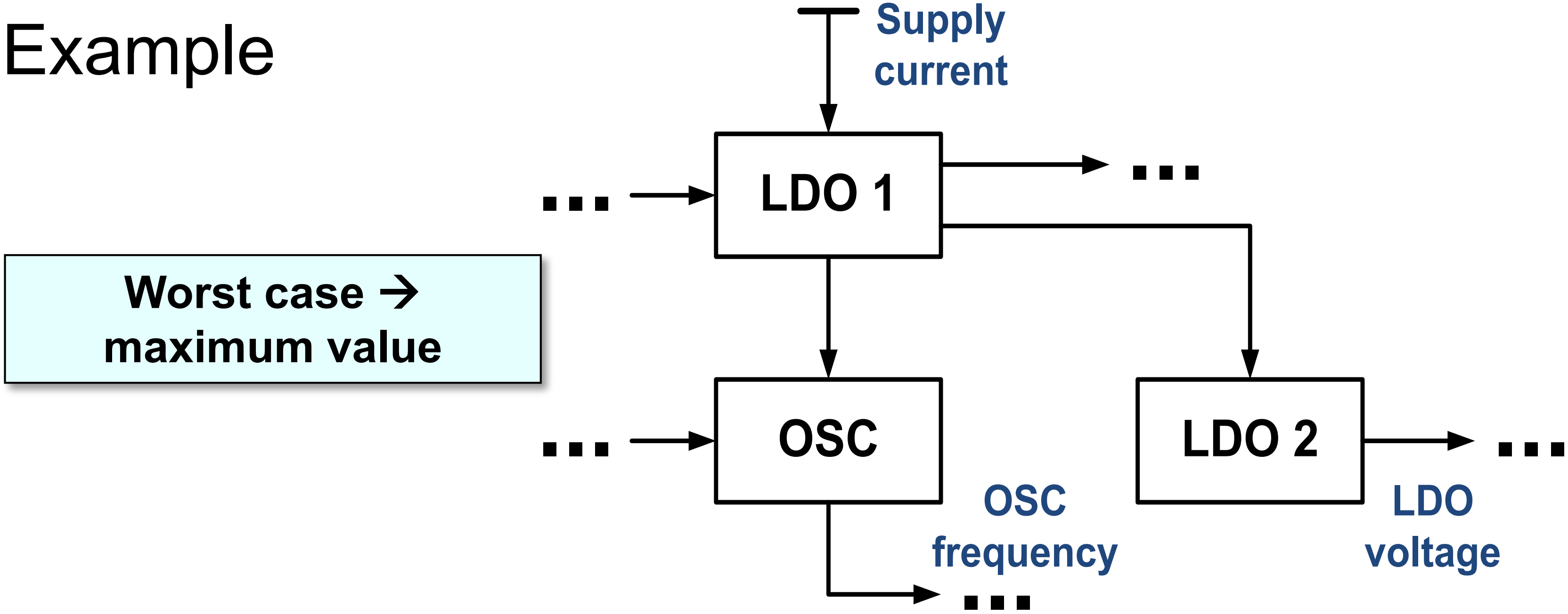
Can the corners always capture the worst case?

Do the corners provide a good coverage of the continuous PVT space?



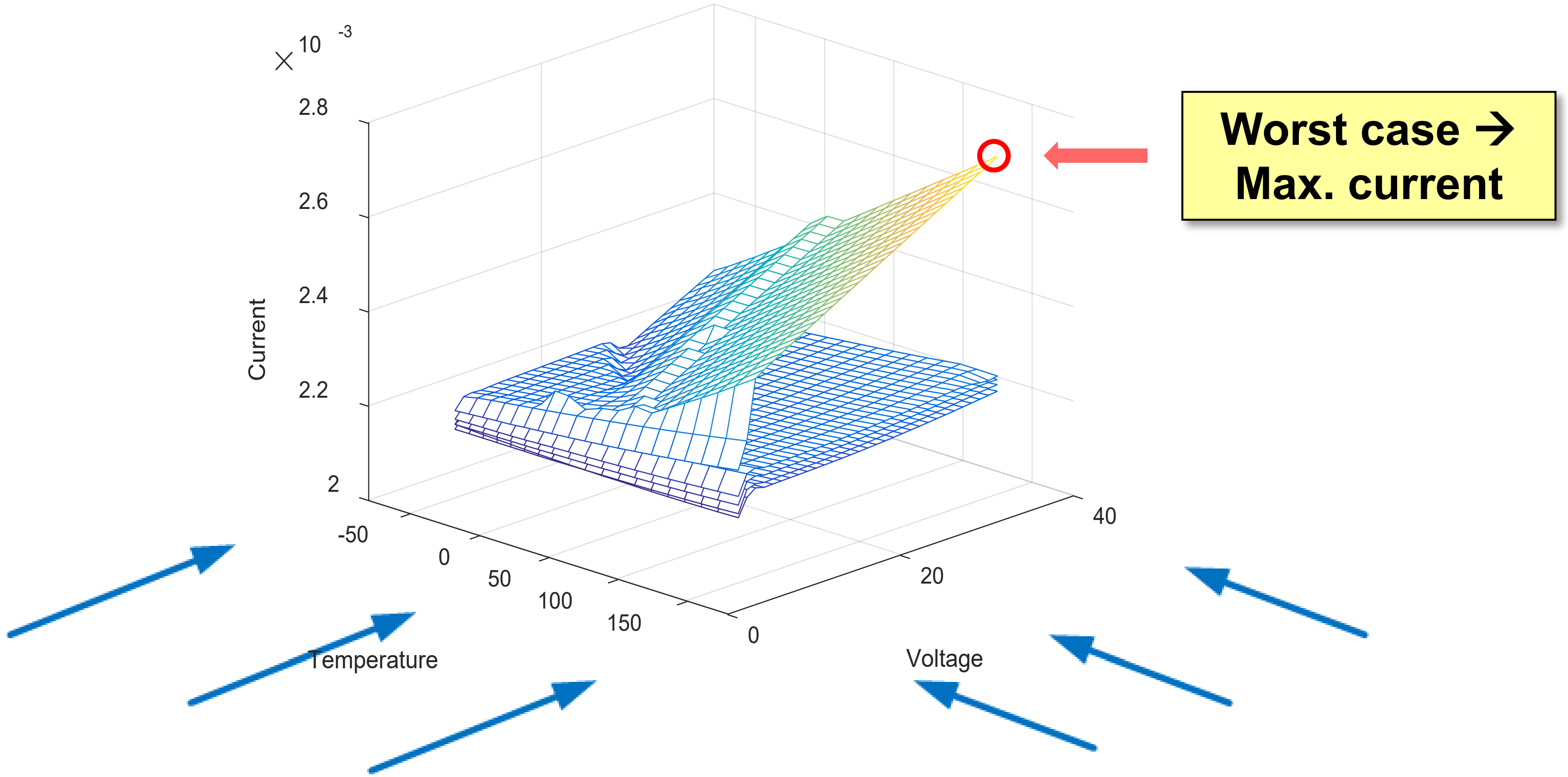
Example (1)

- Analog signals → continuous, nonlinear
- Extreme corners may not be sufficient in capturing the worst case
- Example



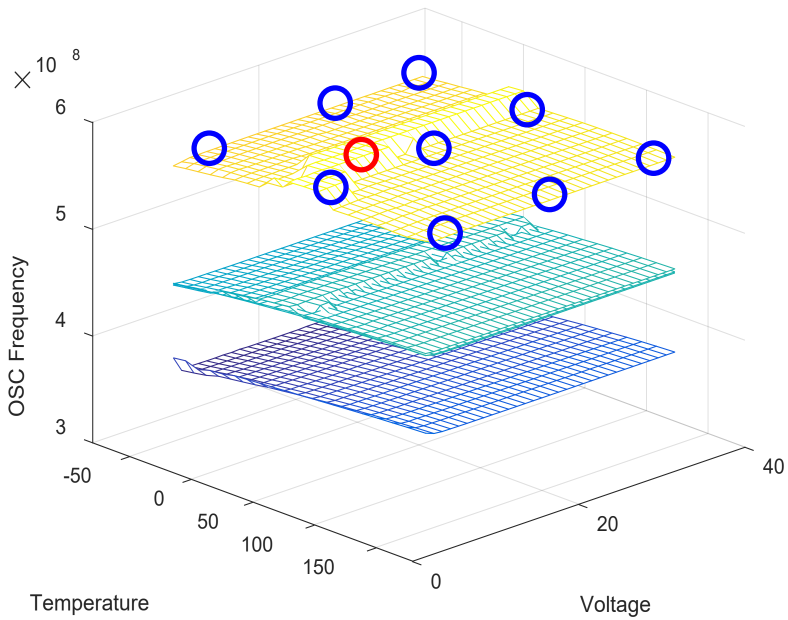
Example (2)

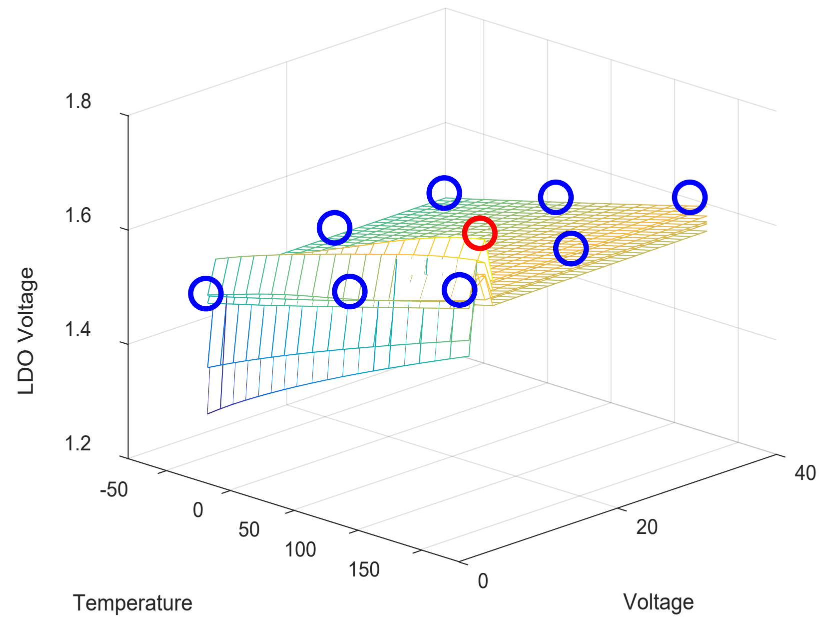
- Supply current: extreme corners work



Example (3)

- OSC frequency and LDO output voltage
  - Extreme corners miss the worst cases

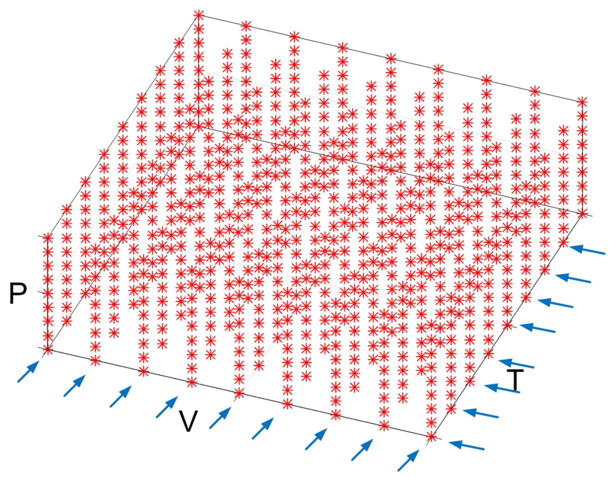


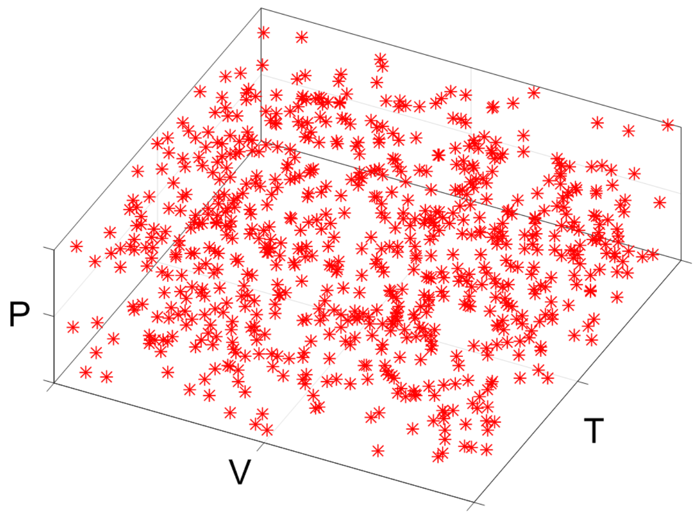


Extreme corners are not reliable

Motivation (Cont.)

- Straight-forward solutions





Fine discretization → coverage via spatial similarity

Monte Carlo → coverage in statistical sense

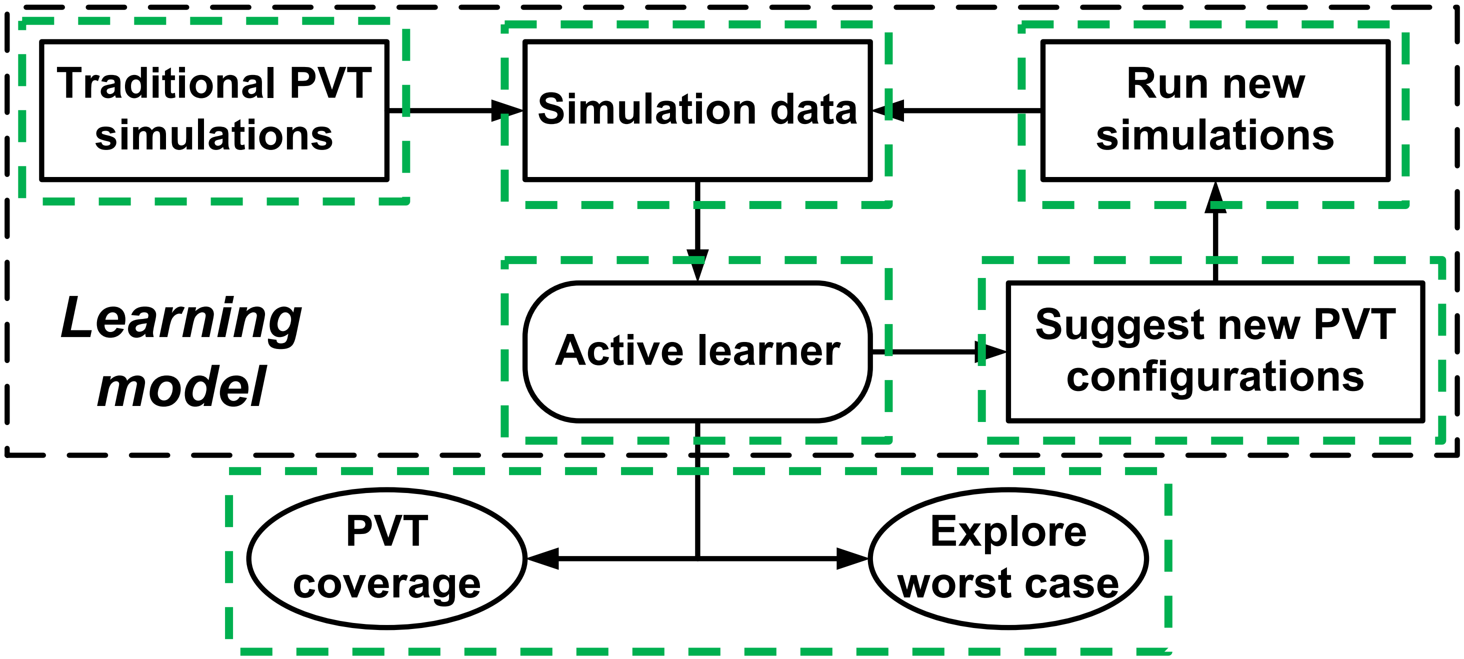
Large amount of simulations → can be very expensive

Proposed Method

- From the data-driven perspective
  - Simulation results → data
  - Modeling, prediction → machine learning
- Coverage of the PVT space
  - Prediction of unknown PVT configurations → classification / regression
- Sample selection
  - Best choice based on "learnt knowledge" → active learning

Overview

- Machine learning model
  - PVT coverage → reliability evaluation
  - Worst case exploration → efficient verification



PVT Space Coverage

- PVT configuration: $\{\boldsymbol{x}_i\}, i = 1, 2, \dots, n$
- Resulting performance: t_i
- Actual mapping: $\not{f} : \boldsymbol{x} \rightarrow t$
- Approximate with f such that errors are small, $\epsilon_i = |t_i - f(\boldsymbol{x}_i)|, i = 1, 2, \dots, n$
- $PVT\ coverage = \dfrac{size\ of\ \{\boldsymbol{x} | t_{min} \leq f(\boldsymbol{x}) \leq t_{max}, \boldsymbol{x} \in \Omega\}}{size\ of\ \{\boldsymbol{x} | \boldsymbol{x} \in \Omega\}}$

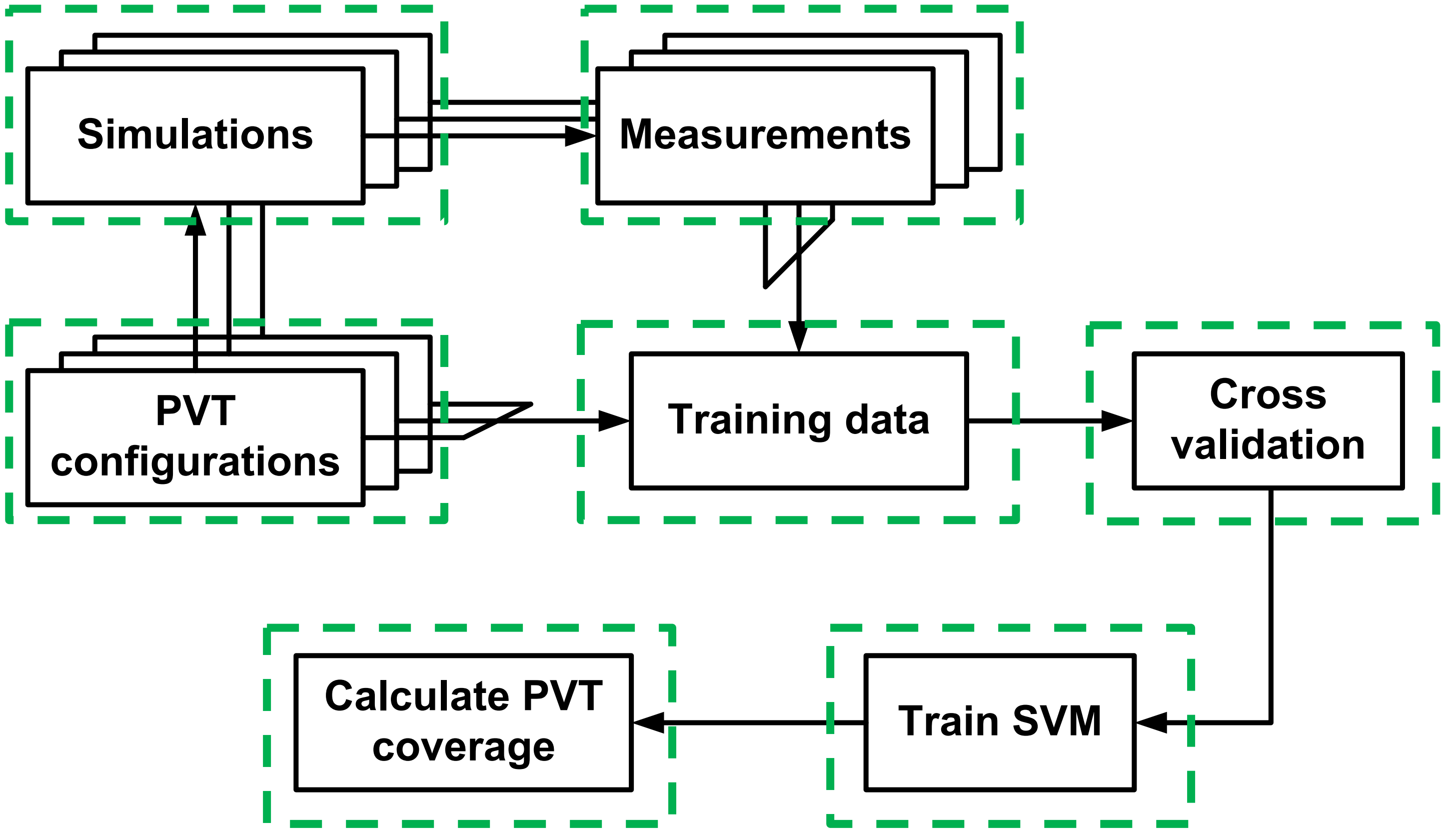
Can be solved by general purpose machine learning techniques ← **Support vector machines (SVM)**

SVM

- A powerful ***nonlinear*** regression tool
- $f(\boldsymbol{x}) = \boldsymbol{w} \cdot \boldsymbol{x} + b$ → Kernel method for nonlinearity
- $Minimize\ \frac{\|\boldsymbol{w}\|^2}{2} + C \sum_{i=1}^{n} (\xi_i + \xi_i^*)$,
- $subject\ to \begin{cases} t_i - \boldsymbol{w} \cdot \boldsymbol{x}_i - b \le \epsilon + \xi_i \\ \boldsymbol{w} \cdot \boldsymbol{x}_i + b - t_i \le \epsilon + \xi_i^* \\ \xi_i, \xi_i^* \ge 0 \end{cases}$

SVM based PVT Coverage

- Calculation flow of the SVM based PVT space coverage



Simulation Reduction (1)

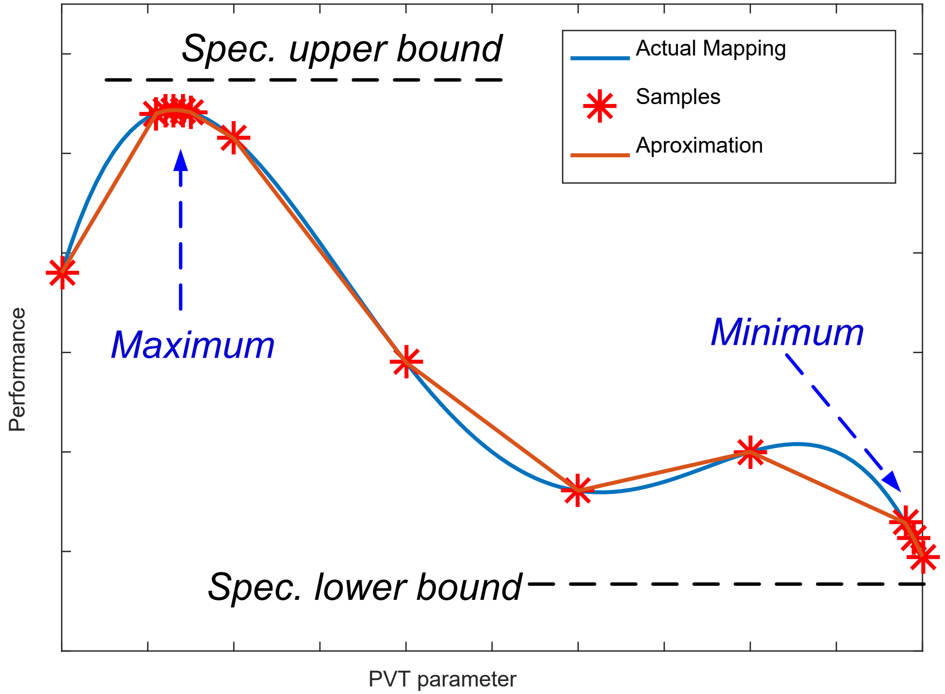
- SVM → What if training data is reduced?
  - Intuition: less data → lower accuracy? Not always right!
- Sparse solution → concept of support vectors
- Dual form: $f = \sum_{i=1}^{n}(\alpha_i - \alpha_i^*)\,(\boldsymbol{x}_i \cdot \boldsymbol{x}) + b$
- $\boldsymbol{x}_i$ with non-zero $(\alpha_i - \alpha_i^*)$ → support vectors → have contribution to the model
- Reach the same or similar SVM model with fewer samples → possible!

Simulation Reduction (2)

- For the purpose of verification → Pass or Fail

Really need to run simulations intensively?

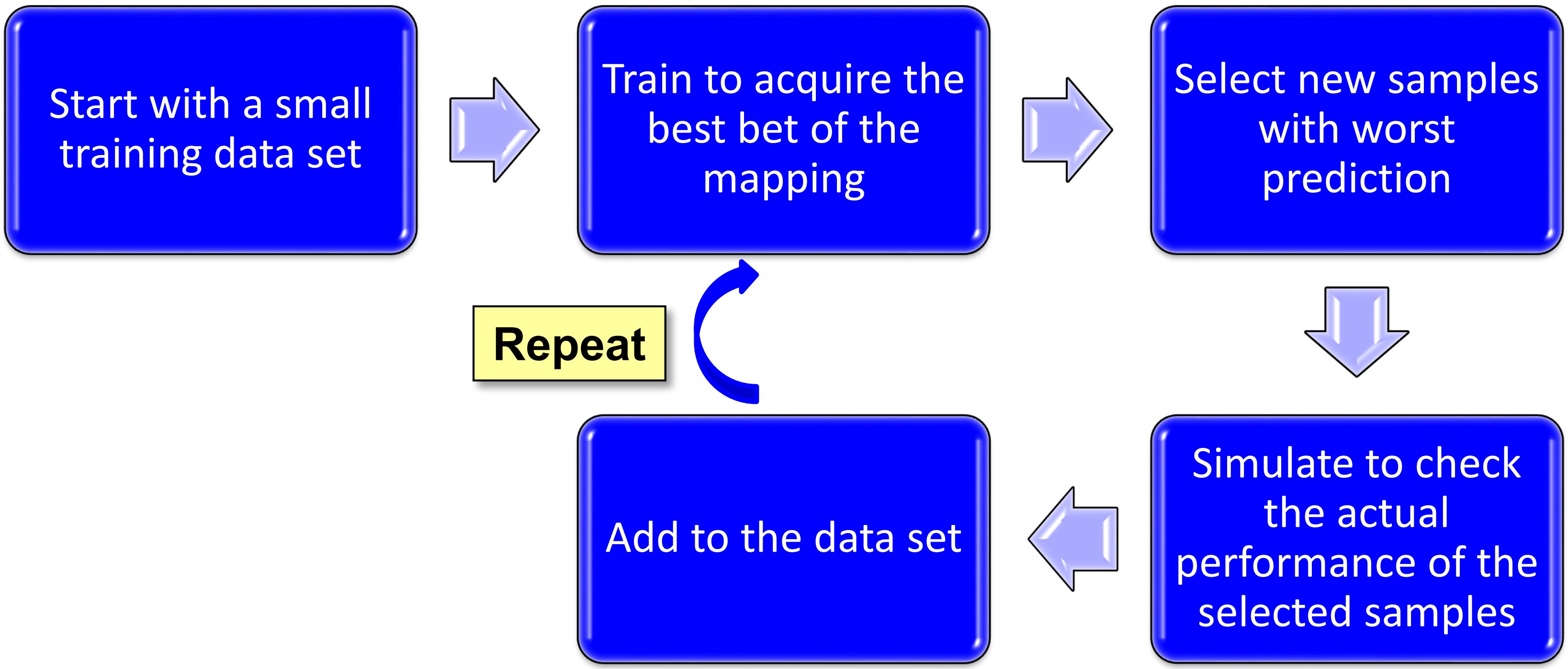
Ideally: identify the worst case



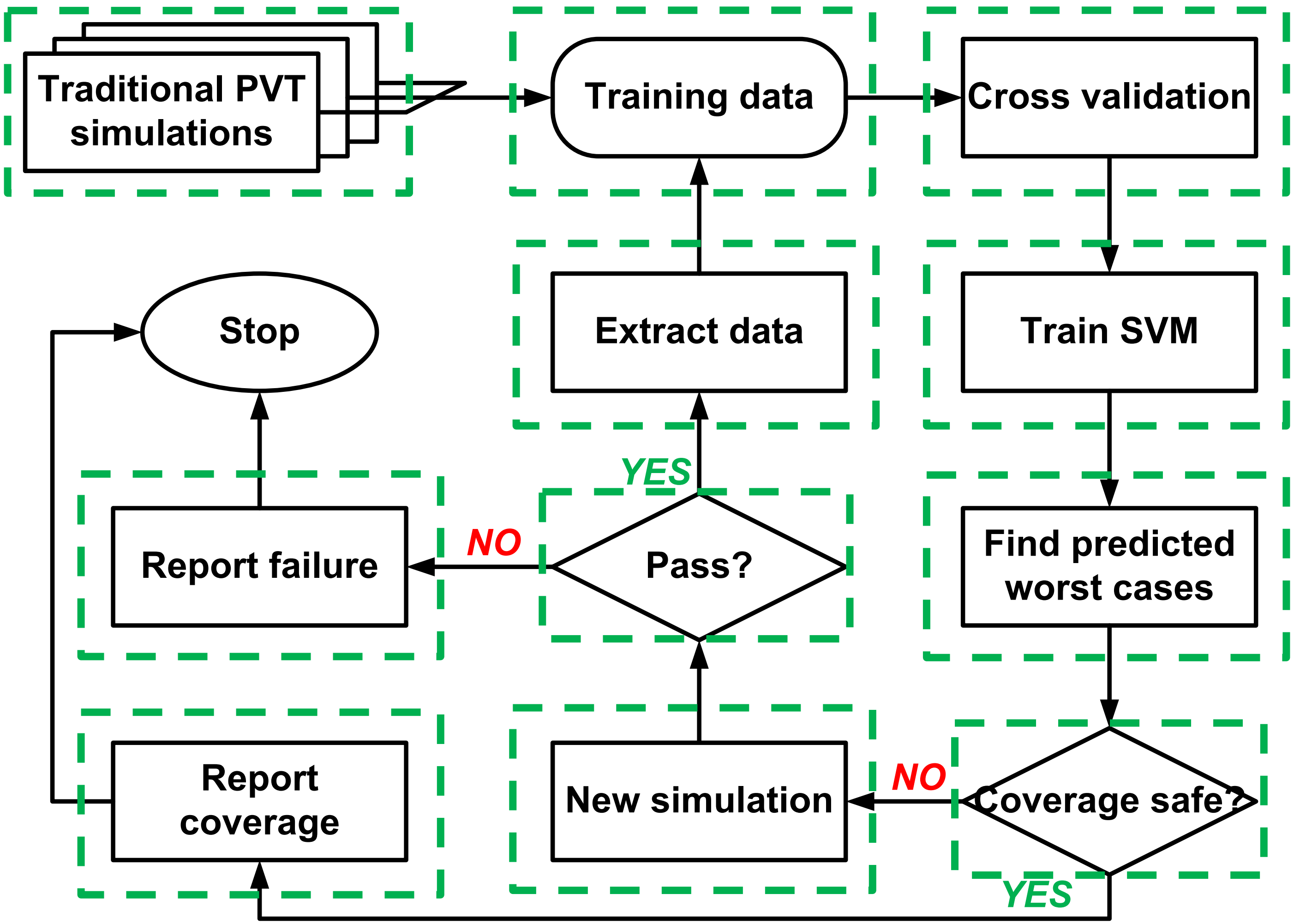
More practically, sample more simulations near worst case locations

Active Learning

- Smarter way to sample?

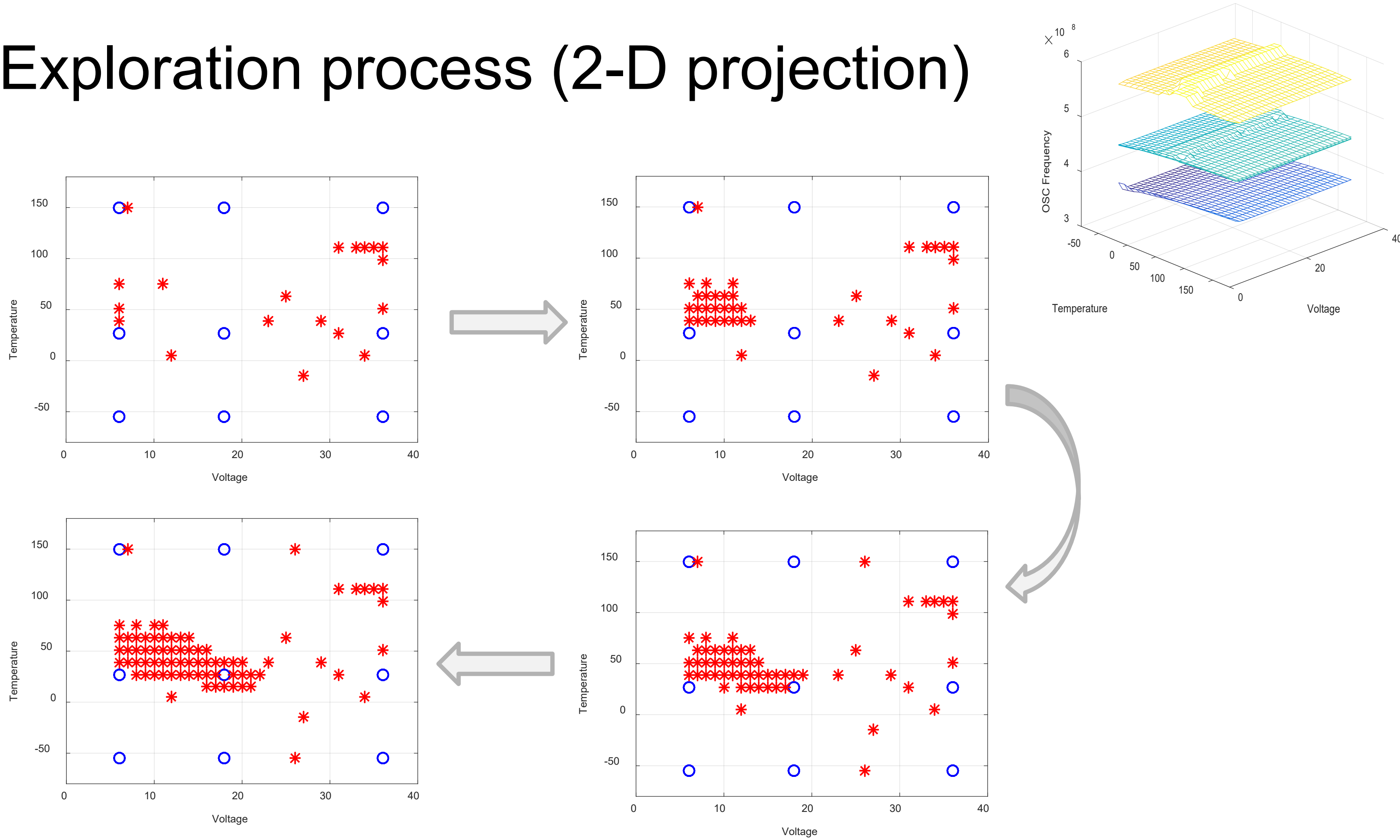


Worst Case Exploration



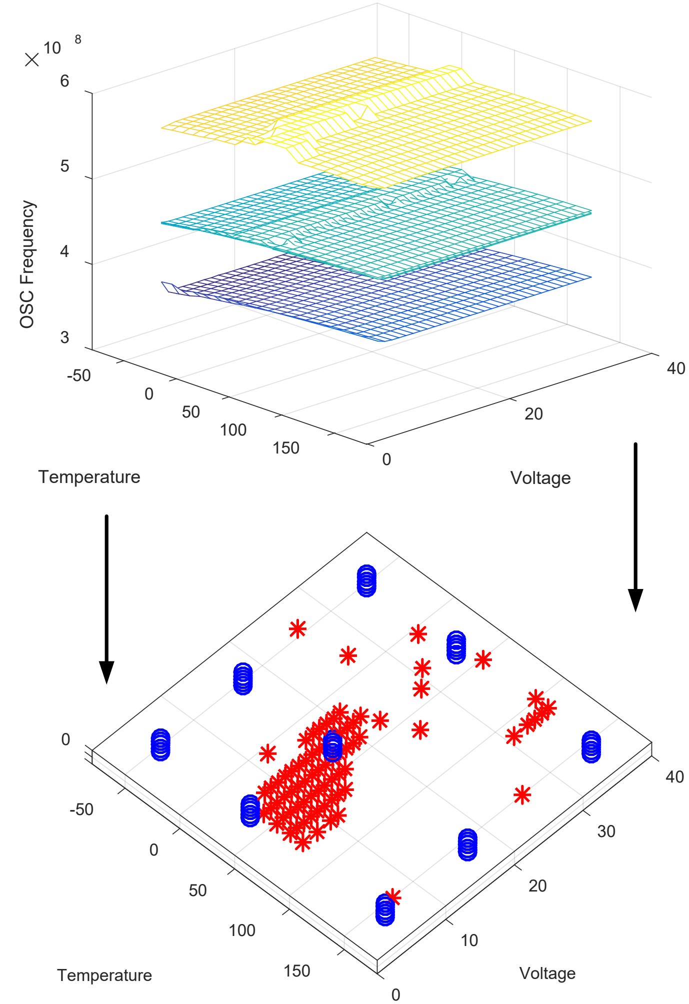
Example: OSC Frequency (1)

- Exploration process (2-D projection)



Example: OSC Frequency (2)

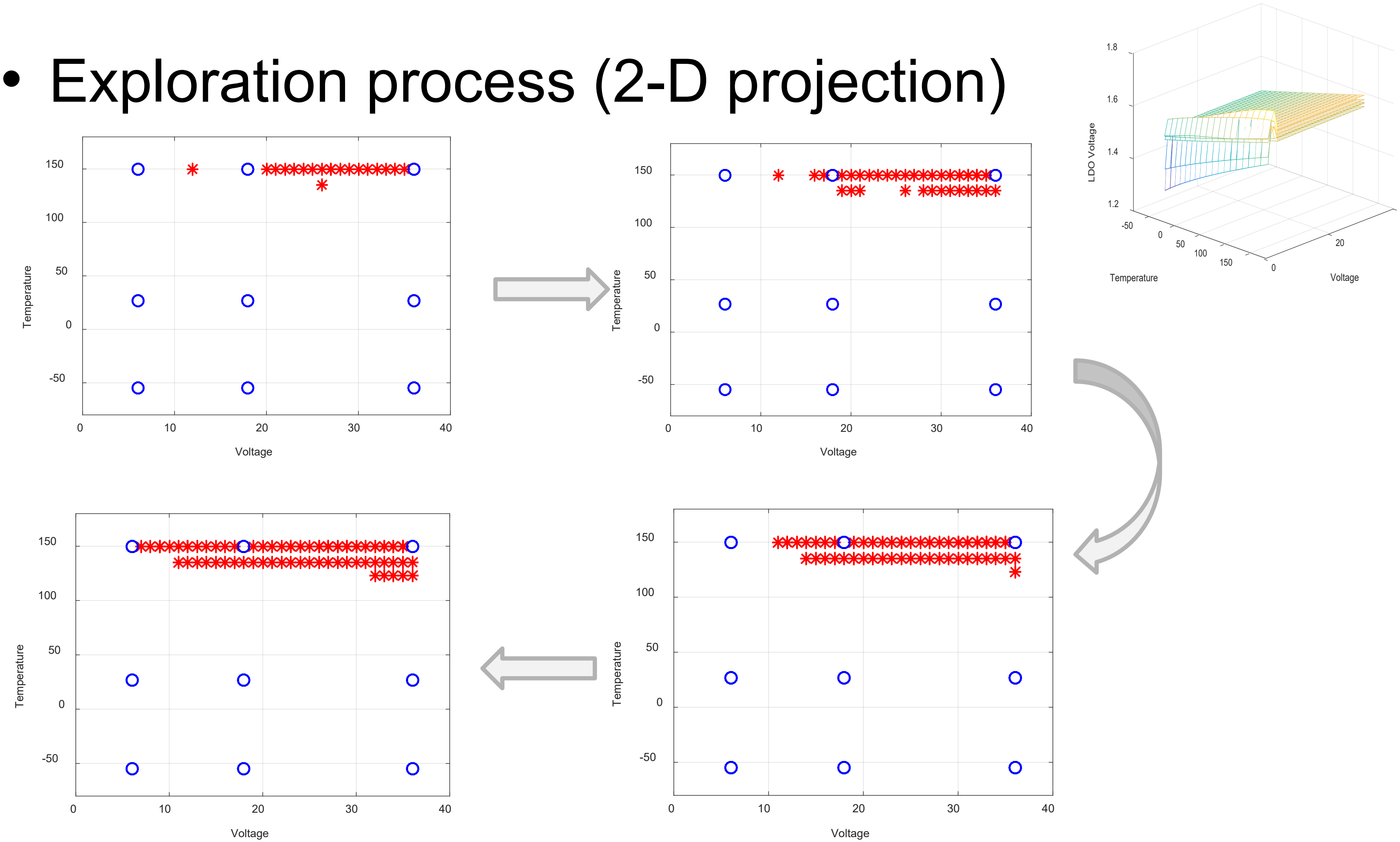
- Successfully capture the worst case
  - Only needs **76** simulations
  - Uniform discretization needs **605** simulations
  - Saves **87%** simulations



Compute time/ resource saving

Example: LDO Voltage (1)

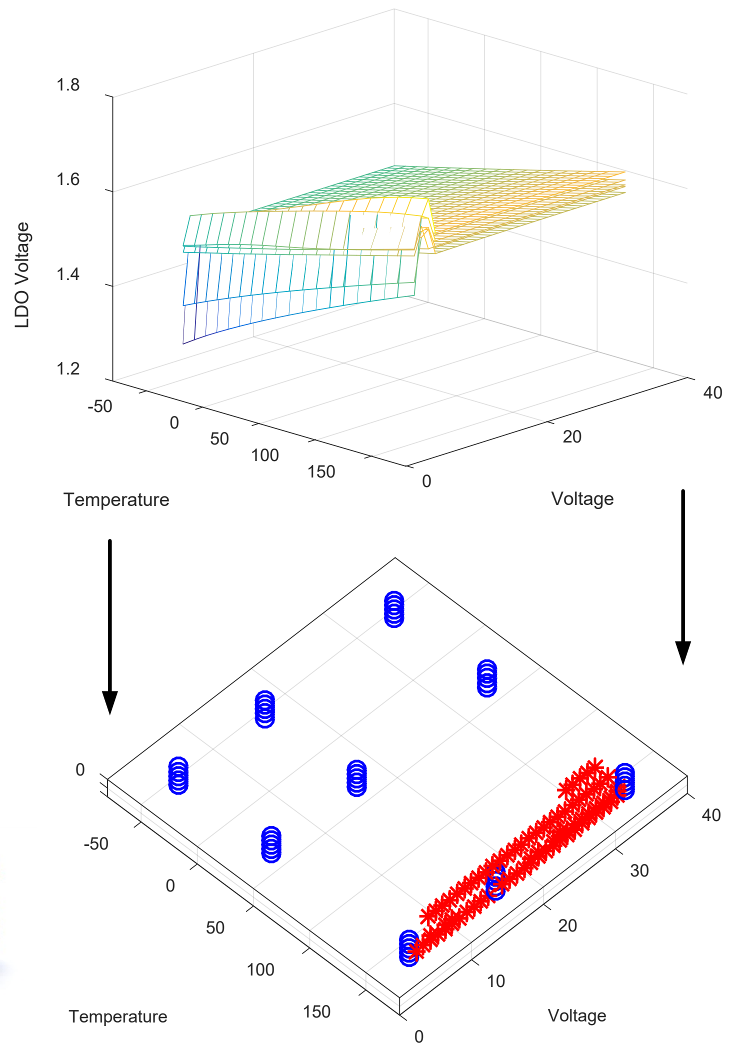
- Exploration process (2-D projection)



Example: LDO Voltage (2)

- Successfully capture the worst case
  - Only needs **139** simulations
  - Uniform discretization needs **1280** simulations
  - Saves **89%** simulations

Compute time/ resource saving



Conclusion

SVM based PVT space coverage → Reliability evaluation

Active learning based worst case exploration → Smarter PVT configuration selection

Capability of reducing the need of simulation

Stress the design for more robust verification

References

- H. Lin, and P. Li. "Circuit performance classification with active learning guided sampling for support vector machines," in *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 34.9, pp. 1467-1480, 2015.
- V. Vapnik. *Statistical Learning Theory*. Wiley, New York, 1998.
- C. M. Bishop. *Neural networks for pattern recognition*. Oxford University Press, 1995.
- M. Fernández-Delgado, *et al*. "Do we need hundreds of classifiers to solve real world classification problems." *Journal of Machine Learning.* Res 15.1 (2014): 3133-3181.
- C. Chang and C. Lin, LIBSVM : a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1--27:27, 2011. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

Thanks!

- Questions?