Active machine learning driven wind tunnel experiments: Realizing the benefits of automation at the UF-BLWT

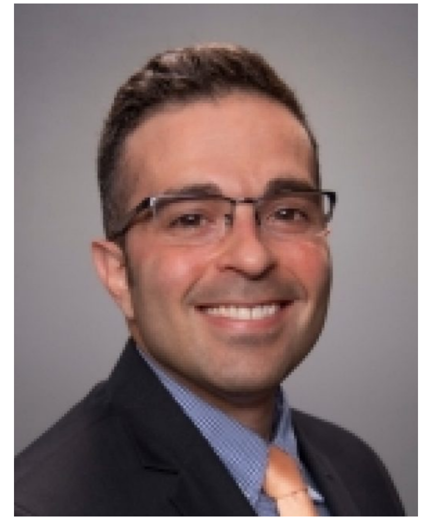
Michael D. Shields
Associate Professor
Dept. of Civil & Systems Engineering
Dept. of Materials Science and Engineering
Johns Hopkins University

Acknowledgements

Mohit Chauhan

Mariel Ojeda Tuz

Kurtis Gurley



Ryan Catarelli

NHERI – SimCenter Tools

SimCenter provides a suite of computational tools for the hazards community

QUOFEM

Quantified Uncertainty with Optimization for the Finite Element Method

WE UQ

Wind Engineering with UQ for uncertain response of buildings to wind loads

EE UQ

Earthquake Engineering with UQ for uncertain response of buildings to seismic loads

Hydro UQ

Building response to water loading – tsunami and storm surge events

PBE

Performance Based Earthquake Engineering computations for individual buildings

R2D

Regional Resilience Determination for regional hazards modeling

UQpy

SURG Shields Uncertainty Research Group

UF UNIVERSITY of FLORIDA

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

UQpy

A collection of Python modules used for uncertainty quantification and propagation

- Includes commonly applied methods and new developments
- Serves as a UQ toolbox and a Python development environment

- Developed collaboratively by members of SURG
  - Author: Michael D. Shields,
  - Contributors: Dimitris Giovanis, Audrey Olivier, Aakash Bangalore Satish, Lohit Vandanapu, Mohit Chauhan, Katiana Kontolati, Dimitris Loukrezis, Ketson R.M. dos Santos

- Version control through git (requires Python 3)
  - Version 3.1.4 available for download/installation via GitHub (https://github.com/SURGroup/UQpy)

- Available on the Python Package Index (PyPI) and Conda (`pip install UQpy`)

Journal of Computational Science 47 (2020) 101204

Contents lists available at ScienceDirect

Journal of Computational Science

journal homepage: www.elsevier.com/locate/jocs

ELSEVIER

Check for updates

UQpy: A general purpose Python package and development environment for uncertainty quantification

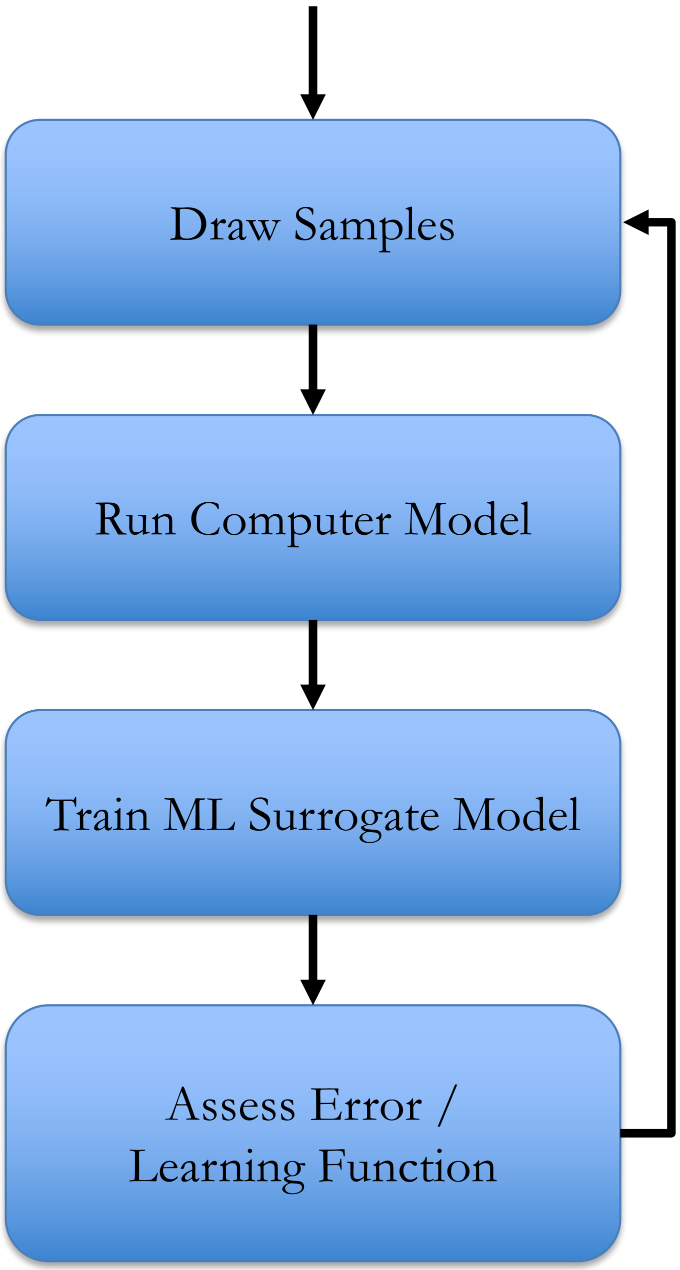
Audrey Olivier, Dimitris G. Giovanis, B.S. Aakash, Mohit Chauhan, Lohit Vandanapu, Michael D. Shields *

SURG Shields Uncertainty Research Group

UF UNIVERSITY of FLORIDA

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

Simple Active Learning Framework



Active Learning for UQ

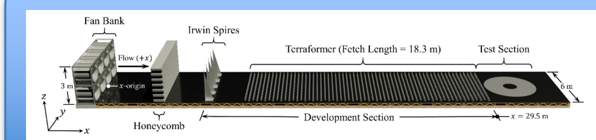
This framework is nothing new:

- Jones et al. (1998) *Efficient Global Optimization of Expensive Black-Box Functions,* Journal of Global Optimization
  Developed the **Expected Improvement Function** – A learning function for global optimization
- Bichon et al. (2008). *Efficient Global Reliability Analysis for Nonlinear Implicit Performance Functions,* AIAA Journal
  Developed the Efficient Global Reliability Analysis (EGRA) method based on the
  Expected Feasibility Function
- Echard et al. (2011). *AK-MCS: An active learning reliability method combining Kriging and Monte Carlo Simulation,* Structural Safety
  Developed the Adaptive Kriging with Monte Carlo Simulation (AK-MCS) based on
  the **U Learning Function** for reliability analysis
- Lam. *"Sequential adaptive designs in computer experiments for response surface model fit."* PhD diss., The Ohio State University, 2008.
  Developed the **Expected Improvement for Global Fit** function to adaptive construct accurate surrogates.

⋮

Simple Active Learning Framework

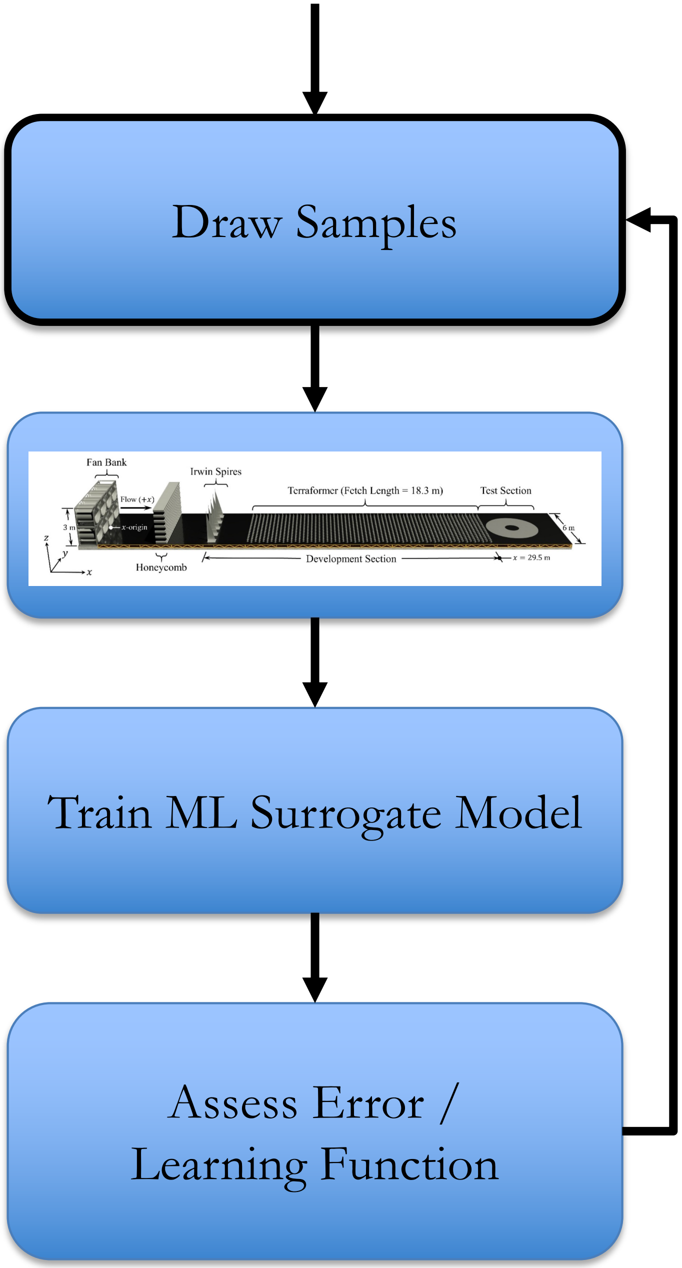
Draw Samples



Train ML Surrogate Model

Assess Error /
Learning Function

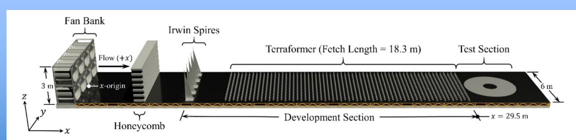
Simple Active Learning Framework



Experiments must be parameterized

Simple Active Learning Framework

Draw Samples



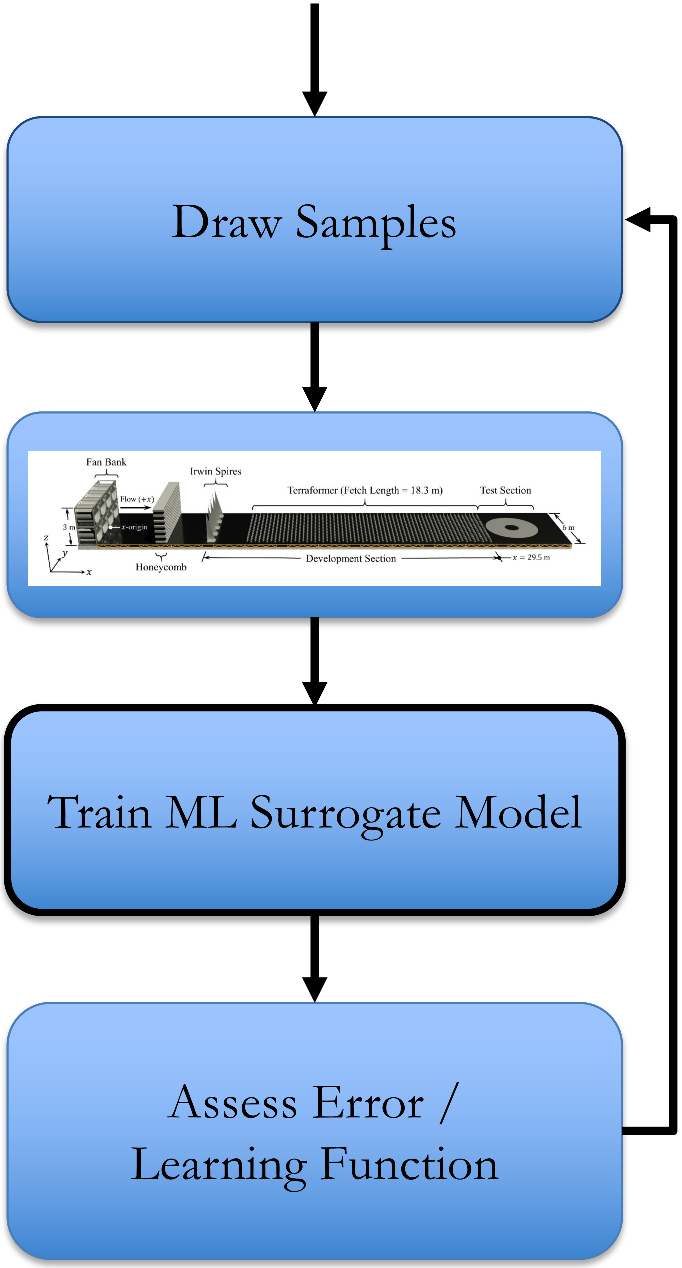
Train ML Surrogate Model

Assess Error / Learning Function

Experiments must be parameterized

Testing apparatus, data collection, and data processing must be automated

Simple Active Learning Framework



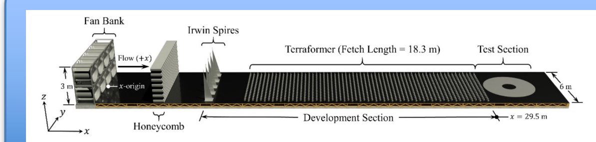
Experiments must be parameterized

Testing apparatus, data collection, and data processing must be automated

Various flavors of ML models are readily available

Simple Active Learning Framework

Draw Samples



Train ML Surrogate Model

Assess Error / Learning Function

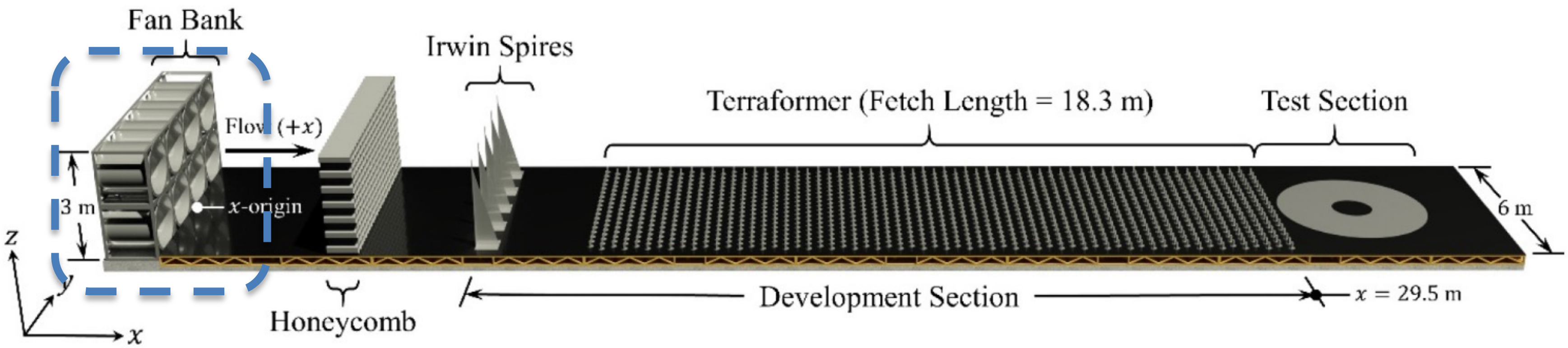
Experiments must be parameterized

Testing apparatus, data collection, and data processing must be automated

Various flavors of ML models are readily available

Define the objective of the study
What are we trying to learn/discover?

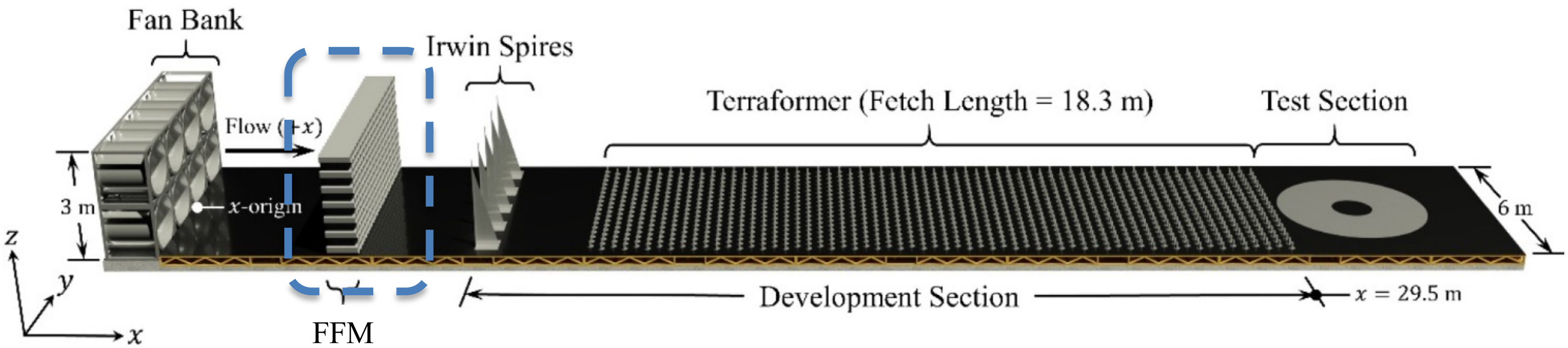
Parameterized Experiments



Some Available Parameters in the UF BLWT

- Vaneaxial Fan Bank:
  - 8 fans, independent RMP
  - Potentially time varying RMP

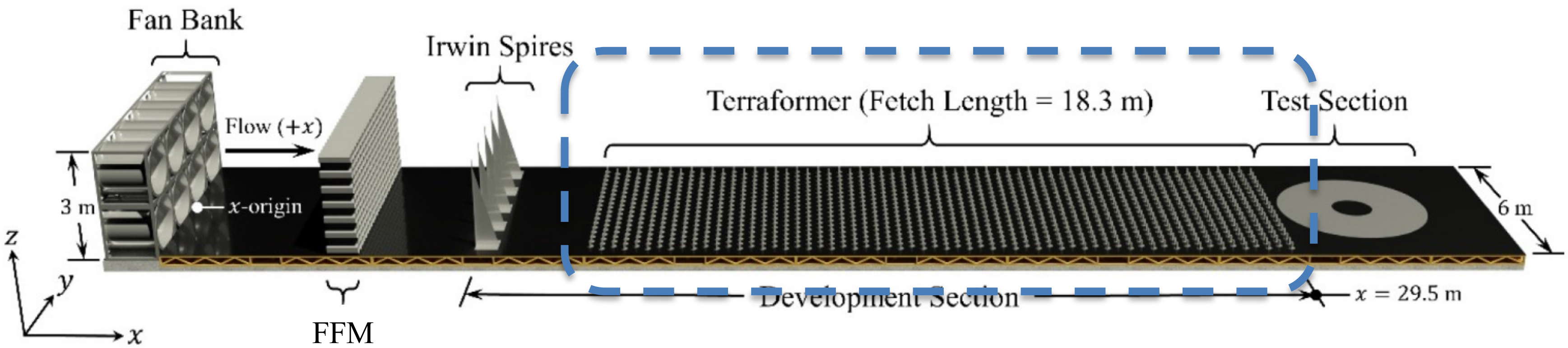
Parameterized Experiments



Some Available Parameters in the UF BLWT

- Vaneaxial Fan Bank:
    - 8 fans, independent RMP
    - Potentially time varying RMP
- Flow Field Modulator (FFM)
    - 319 fans, independent RMP
    - Potentially time varying RMP

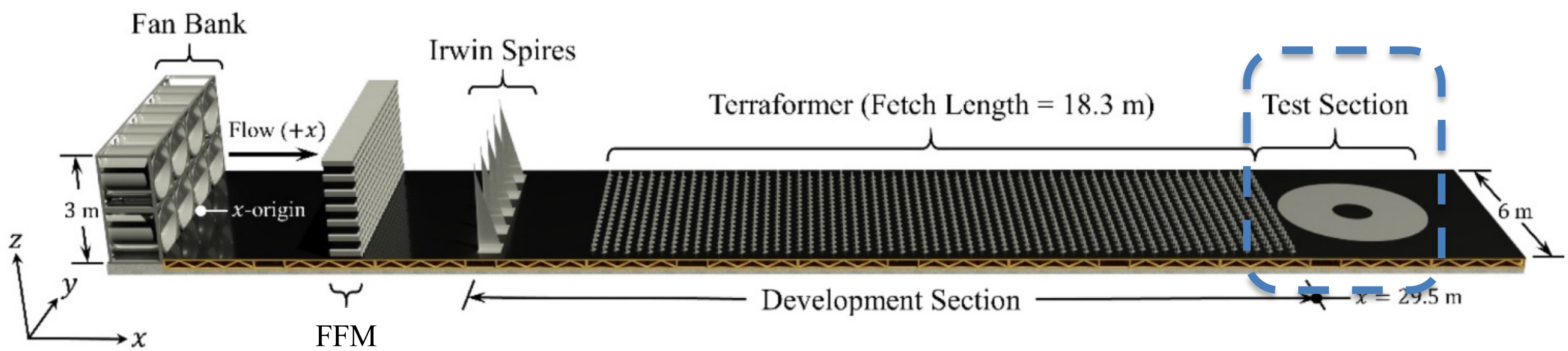
Parameterized Experiments



Some Available Parameters in the UF BLWT

- Vaneaxial Fan Bank:
  - 8 fans, independent RMP
  - Potentially time varying RMP
- Flow Field Modulator (FFM)
  - 319 fans, independent RMP
  - Potentially time varying RMP
- Terraformer
  - 1118 individual roughness elements
  - Each with controlled height and width

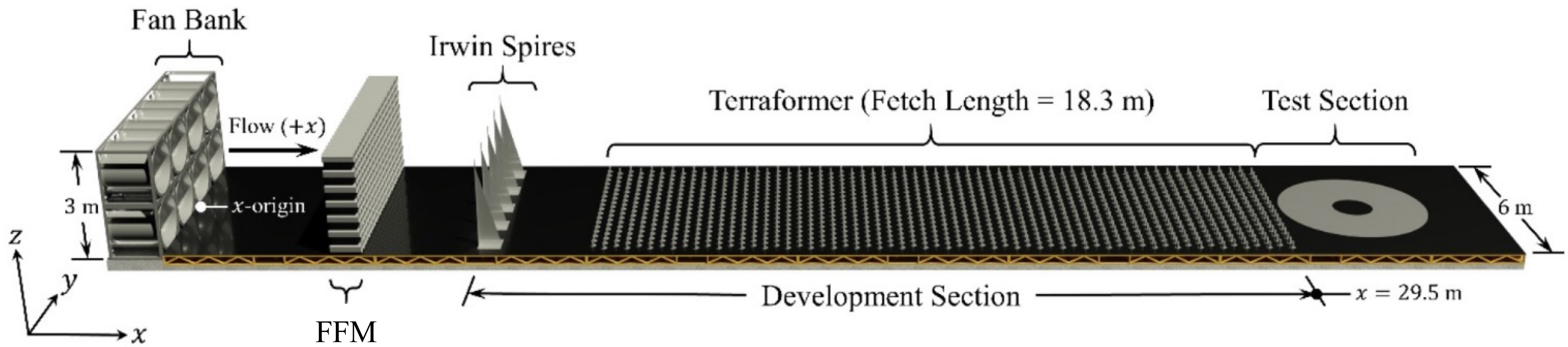
Parameterized Experiments



Some Available Parameters in the UF BLWT

- Vaneaxial Fan Bank:
  - 8 fans, independent RMP
  - Potentially time varying RMP
- Flow Field Modulator (FFM)
  - 319 fans, independent RMP
  - Potentially time varying RMP
- Terraformer
  - 1118 individual roughness elements
  - Each with controlled height and width
- Test Section
  - Rotational degrees of freedom

Parameterized Experiments

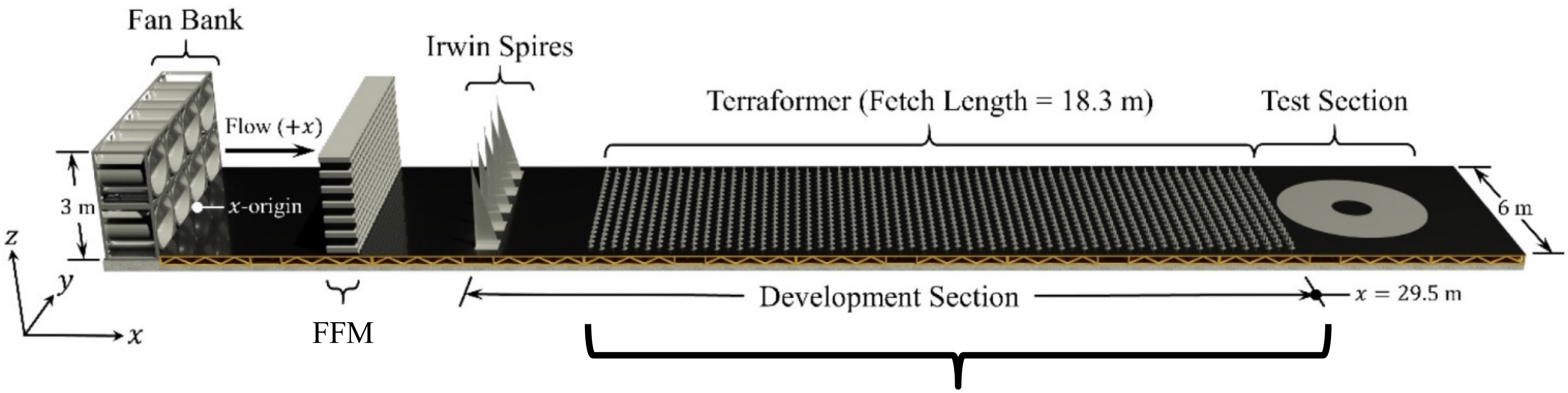


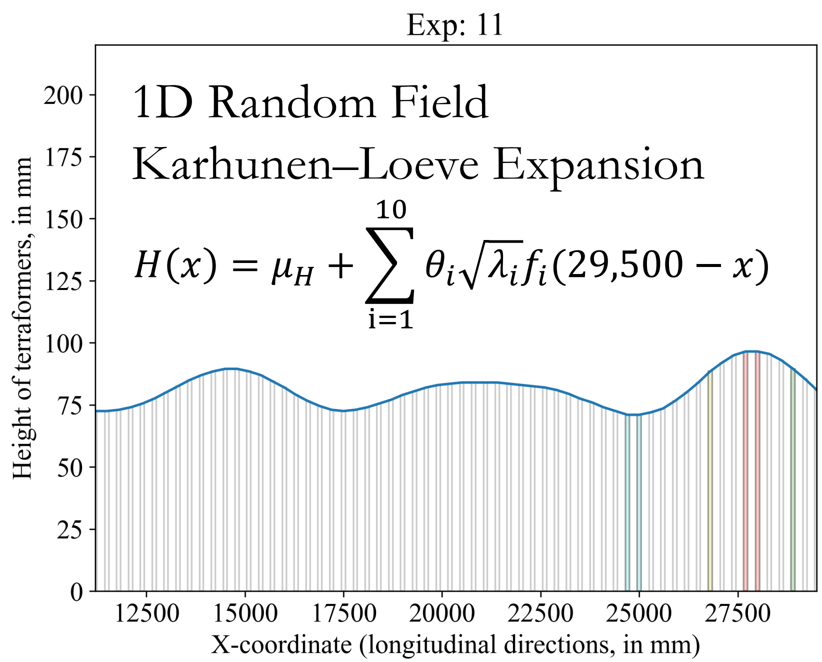
Some Available Parameters in the UF BLWT

- Vaneaxial Fan Bank:
  - 8 fans, independent RMP
  - Potentially time varying RMP
- Flow Field Modulator (FFM)
  - 319 fans, independent RMP
  - Potentially time varying RMP
- Terraformer
  - 1118 individual roughness elements
  - Each with controlled height and width
- Test Section
  - Rotational degrees of freedom

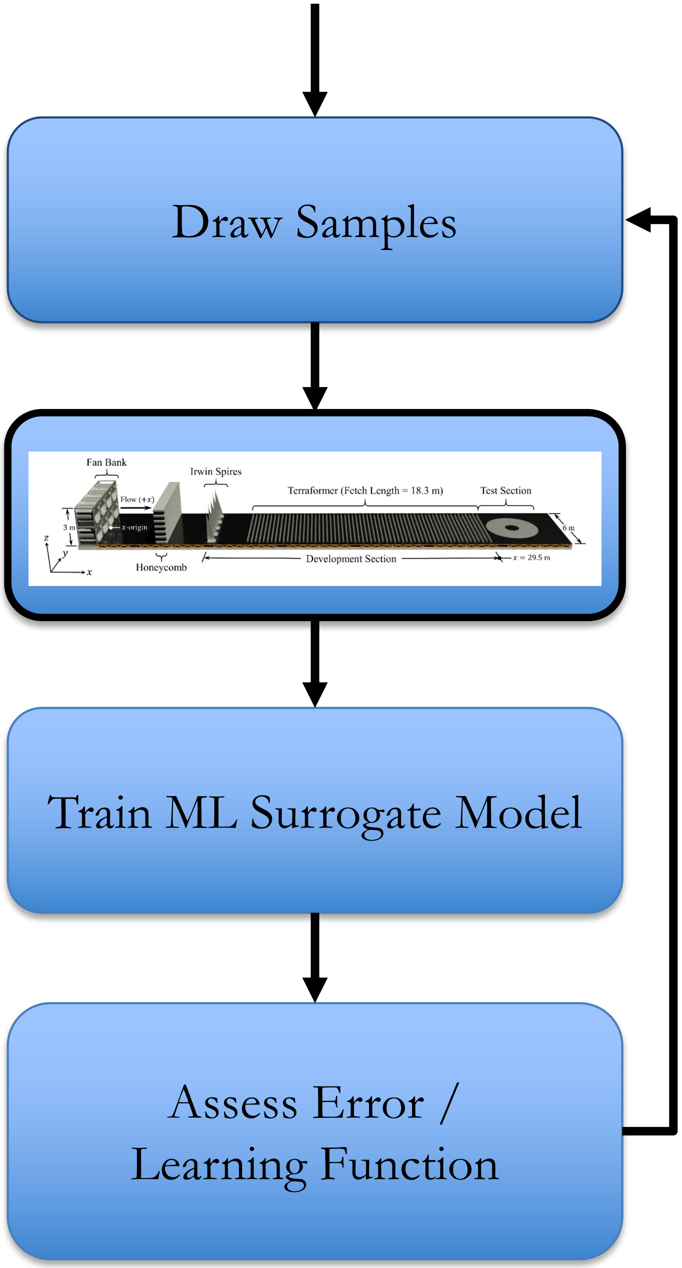
The number of different ways the UF BLWT can be configured is enormous

Terraformer Parameterized





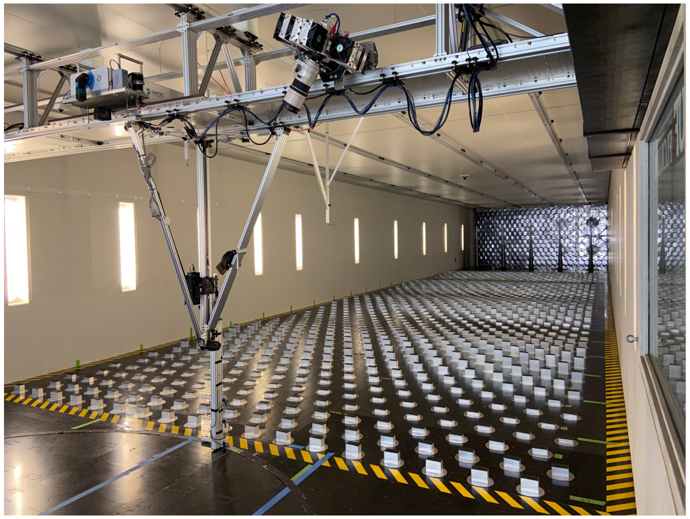
Simple Active Learning Framework

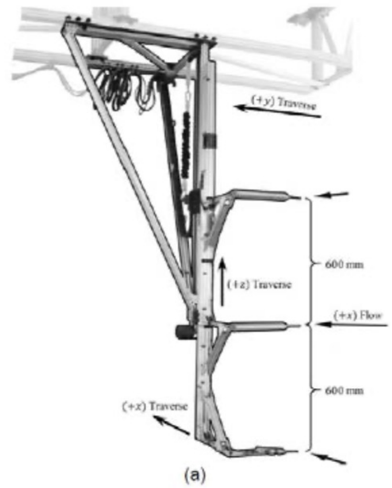


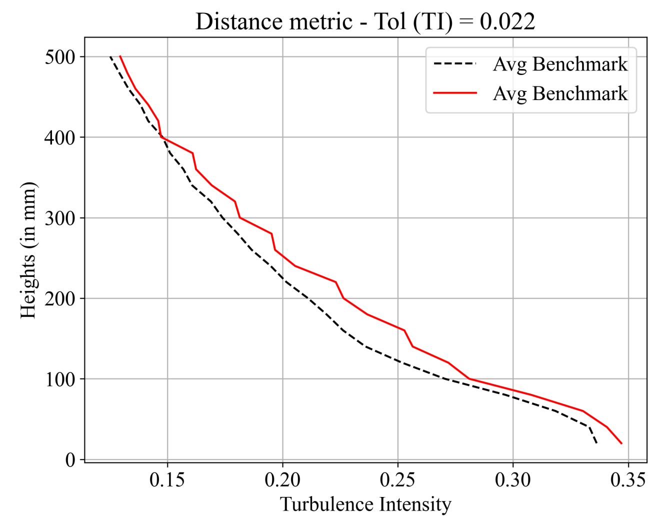
Experiments must be parameterized

Testing apparatus, data collection, and data processing must be automated

Automation







Set Experimental Parameters (Terraformer heights, Fan RMPs, etc.) → Initialize Gantry and Data Collection Scripts → Collect Data → Postprocess Data

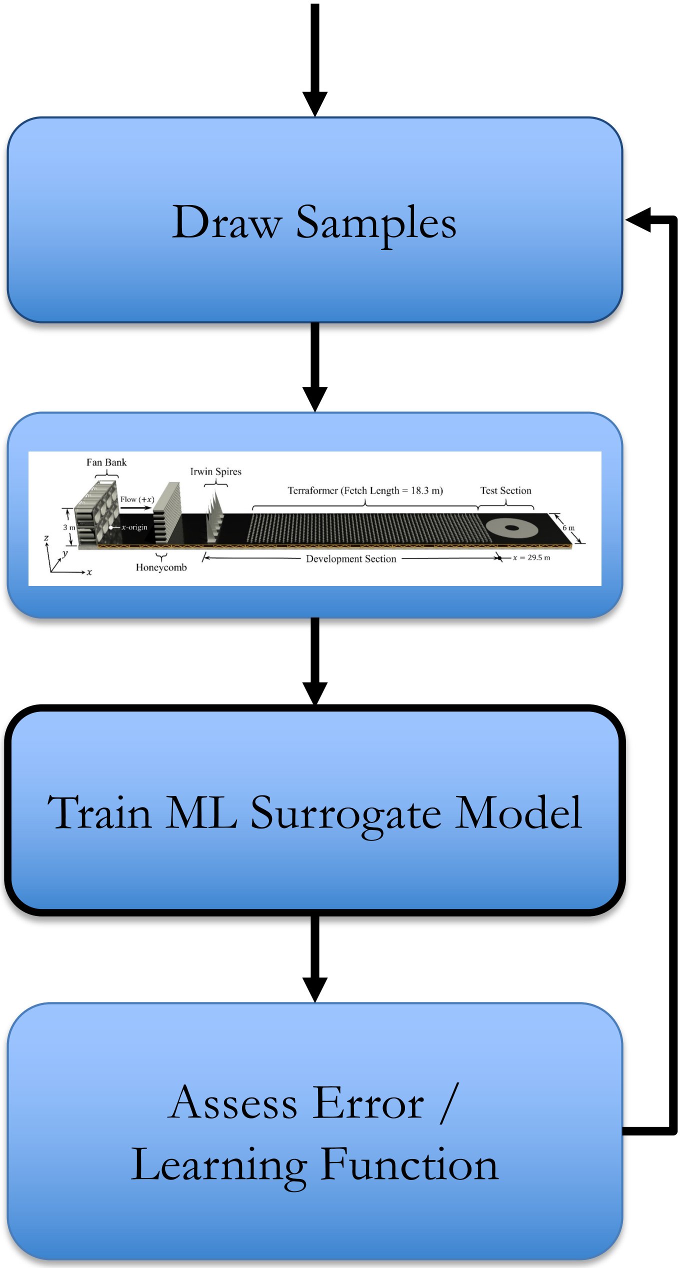
python™ MATLAB®

SURG Shields Uncertainty Research Group

UF UNIVERSITY of FLORIDA

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

Simple Active Learning Framework



Experiments must be parameterized

Testing apparatus, data collection, and data processing must be automated

Various flavors of ML models are readily available

Machine Learning Models

We use Gaussian Process Regression

Many other flavors of ML models exist

- Artificial Neural Networks
  - Deep or Shallow Neural Networks
  - Convolutional Neural Networks
  - Recursive Neural Networks
  - Physics Informed Neural Networks
- Polynomial Chaos Expansions
- Support Vector Regression

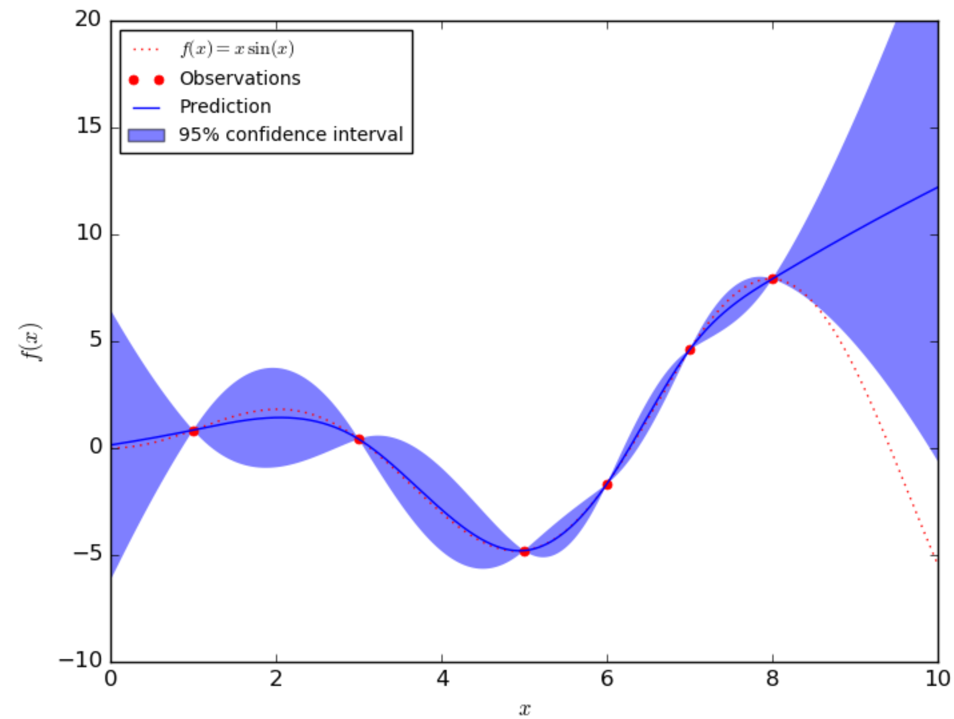
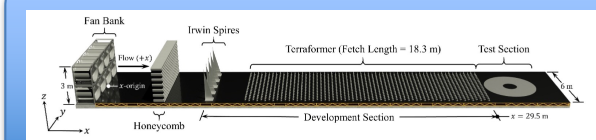


Image from scikit-learn documentation

Simple Active Learning Framework

Draw Samples



Train ML Surrogate Model

Assess Error /
Learning Function

Experiments must be parameterized

Testing apparatus, data collection, and data processing must be automated

Various flavors of ML models are readily available

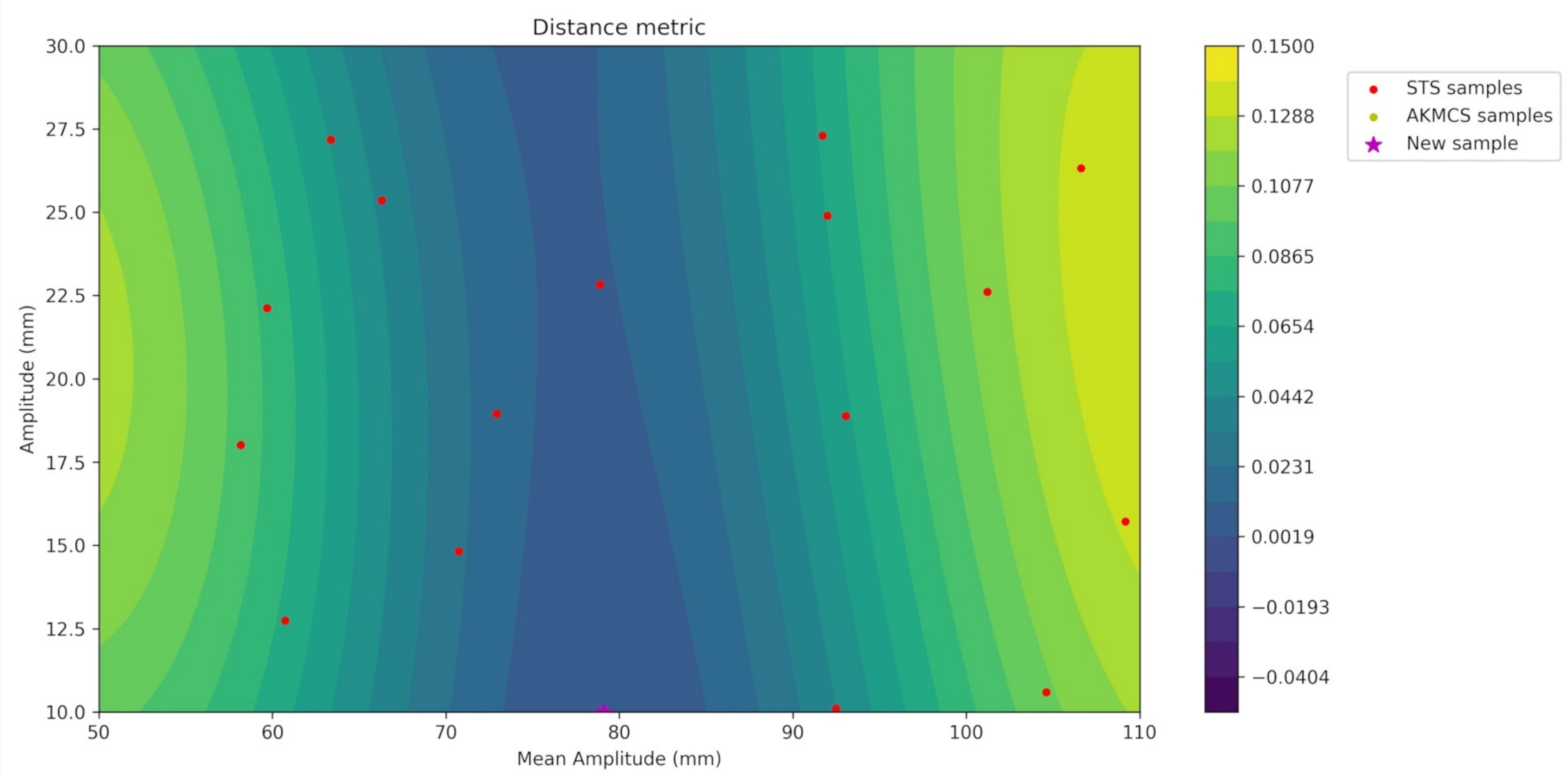
Define the objective of the study
What are we trying to learn/discover?

Learning Functions

Learning Function	Expression	Objective
Noisy U-function	$$U_n(\theta) = \frac{\mu_g(\theta)}{\sqrt{\sigma_g^2(\theta) - \sigma_\epsilon^2(\theta)}}$$	**Goal:** Conduct experiments along the surface separating the 2nd order equivalent and non-equivalent regions. **How:** Conduct experiments that have the highest probability of incorrectly predicting the sign of the performance function.
Noisy EIGF (Expected Improvement for Global Fit)	$$E[I_n(\theta)] = \left(\mu_g(\theta) - g(\theta^*)\right)^2 + \sigma_g^2(\theta) - \sigma_\epsilon^2(\theta)$$	**Goal:** Conduct experiments that globally best approximate the performance function. **How:** Conduct experiments that have both high prediction uncertainty and large difference from nearby experiments
MUSIC	$$♫(\theta_j^{(i)}) = \left(\mu_{A^{(i)}}\left(\theta_j^{(i)}\right) - \mu_{A^{(i)}}(\theta^*)\right)^2 + \sigma_{A^{(i)}}^2\left(\theta_j^{(i)}\right)$$	**Goal:** Conduct experiments that allow efficient computation of sensitivity indices. **How:** Conduct experiments that have both high prediction uncertainty and large differences from nearby experiments in conditional GP

Results

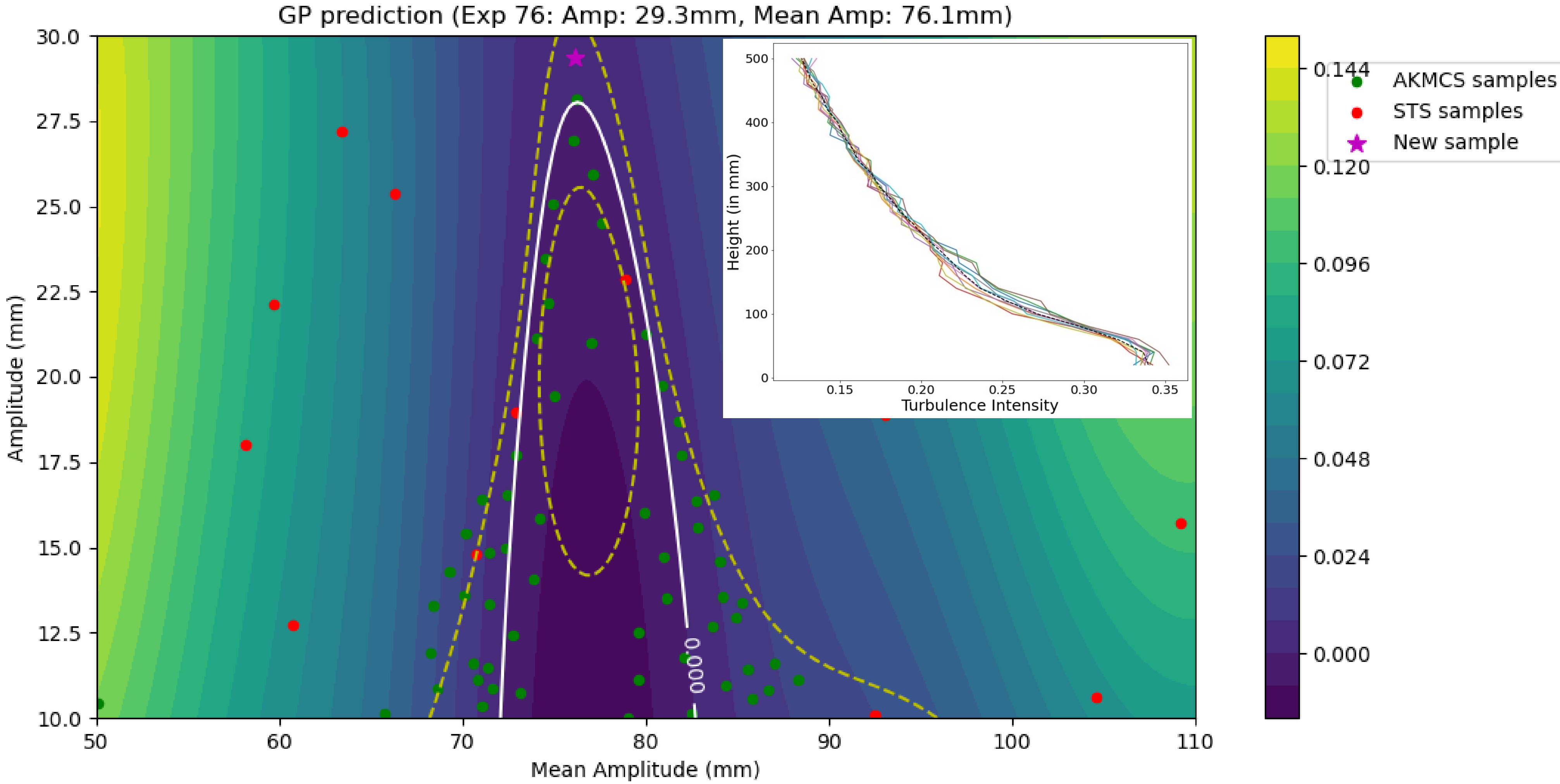
Over 1000 experiments conducted in the past 2 years!



900+ unique terraformer configurations

Results

Much of this data will be published to DesignSafe



Thank You!