Introduction to Big and Deep Data Analysis Methods

Rick Archibald

Oak Ridge National Laboratory

USA

ITER International School Dec 9-13, 2024

Nagoya, Japan

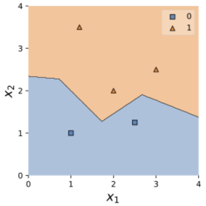
Outline

- Machine Learning Basics
- SAS Federated Learning
- FREDA Machine Learning Grid Generation

Machine Learning Basics

Quick Overview of Machine Learning

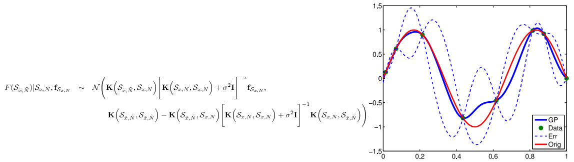
- K-Nearest Neighbors Algorithm (KNN)



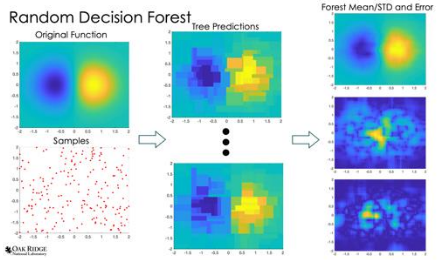
- Support Vector Machines (SVM)

$$K(\mathbf{x}, \mathbf{y}) = e^{-\gamma\|\mathbf{x}-\mathbf{y}\|^2}$$

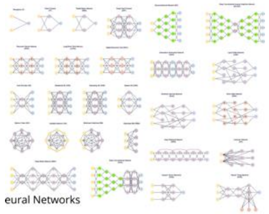
- Gaussian Process Learning (GPL)



- Decision Tree

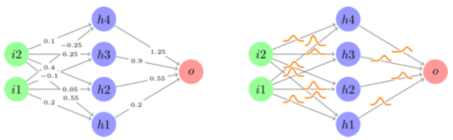


- Deep Neural Networks (DNN)

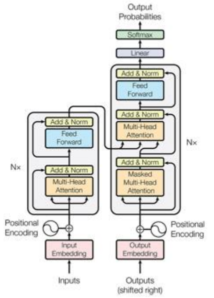


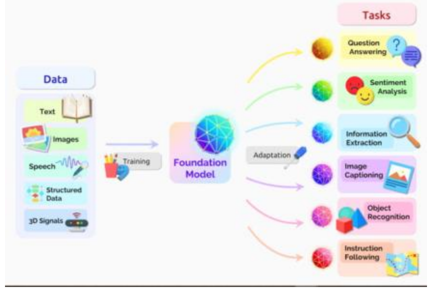
$$\mathbf{Y}_{j+1} = \mathbf{Y}_j + hf(\mathbf{Y}_j, \Theta_j) \xrightarrow[i=1,\cdots,N]{} \dot{\mathbf{Y}}(t) = f(\mathbf{Y}(t), \Theta(t))$$

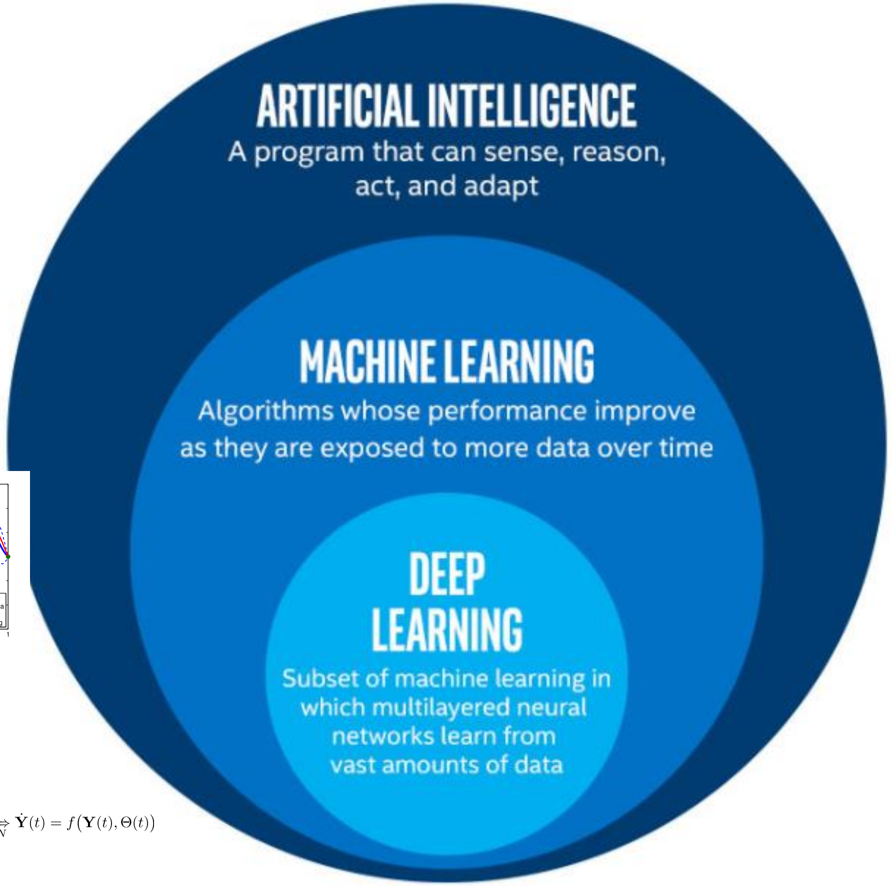
- Bayesian Neural Networks

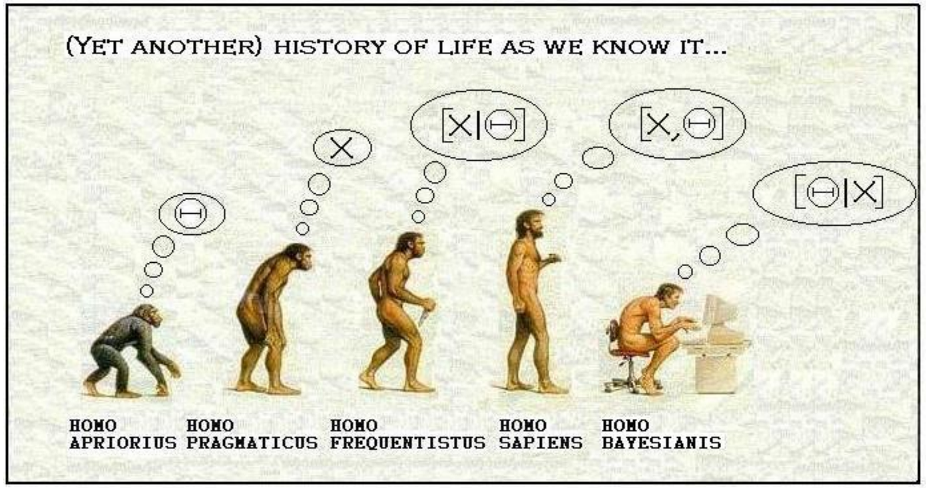


- Foundational Models



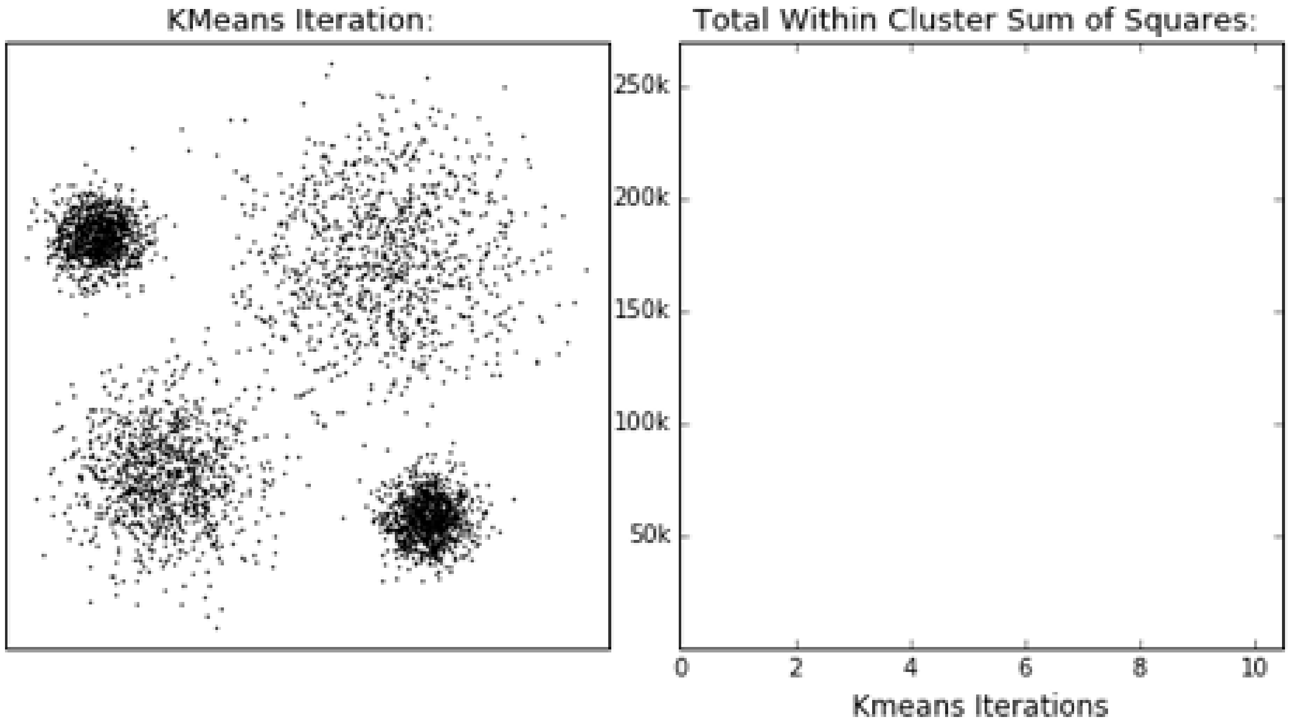


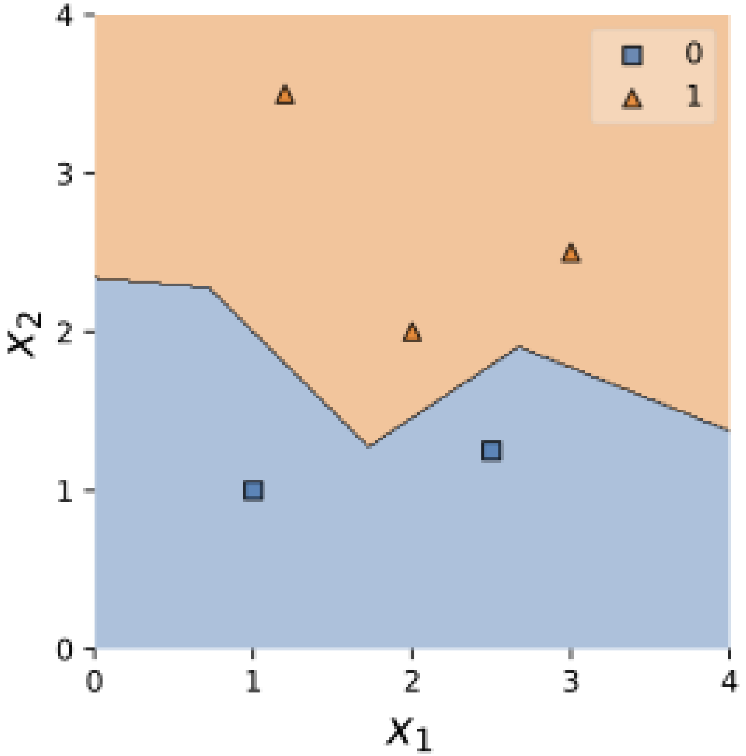




OAK RIDGE National Laboratory

K-Nearest Neighbors Algorithm

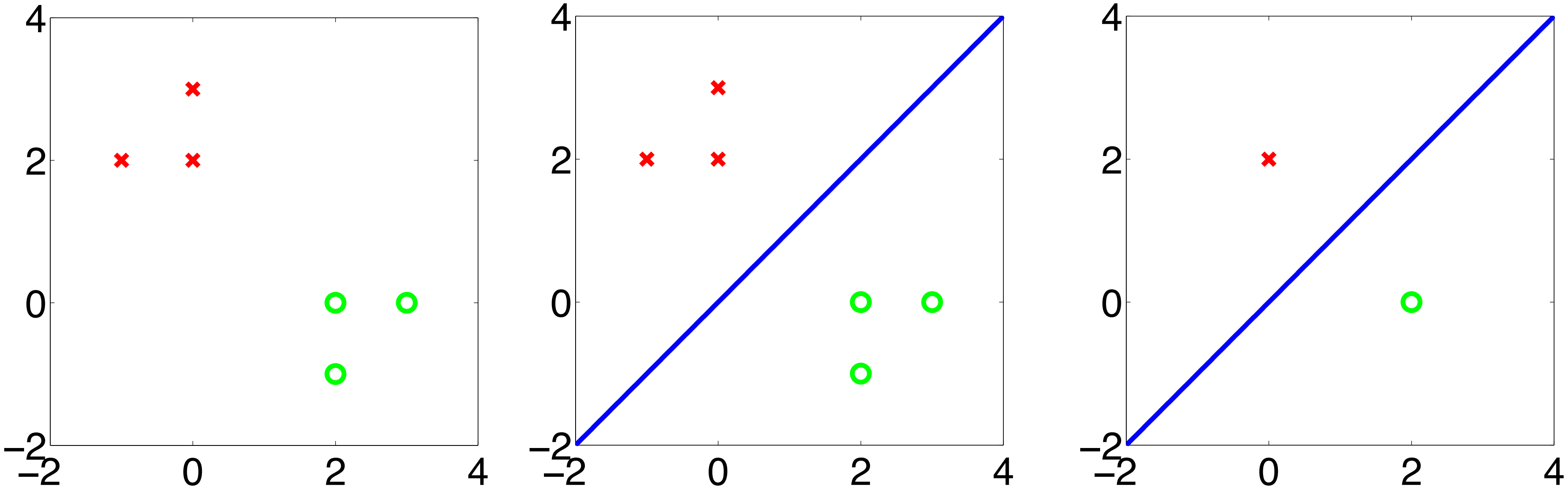




- Minimal or No training!
- More data greater accuracy
- Prediction and storage is computational challenge

Oak Ridge National Laboratory

Support Vector Machines



$$f(\mathbf{x}) = \operatorname{sign}\left(\sum_{i=1}^{N} y_i \alpha_i \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}) + b\right)$$

Support vector function

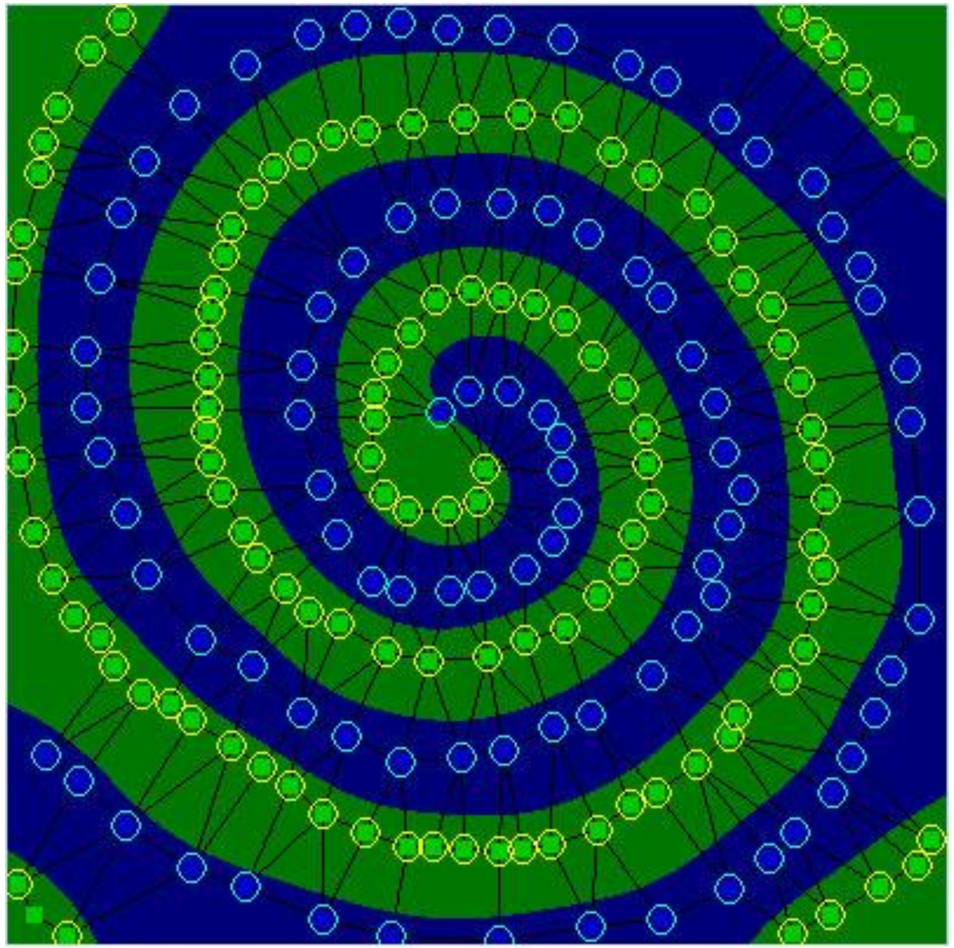
$$\left\{\mathbf{x}_i \in \mathbb{R}^d, y_i \in \{-1, 1\}\right\} \quad i = 1, \ldots, N,$$

Coefficients determine by maximizing margin

$$K(x, y) = \Phi(x) \cdot \Phi(y)$$

Kernel Trick makes computations fast

Support Vector Machines



Gaussian Kernel capable of classifying complicated domains

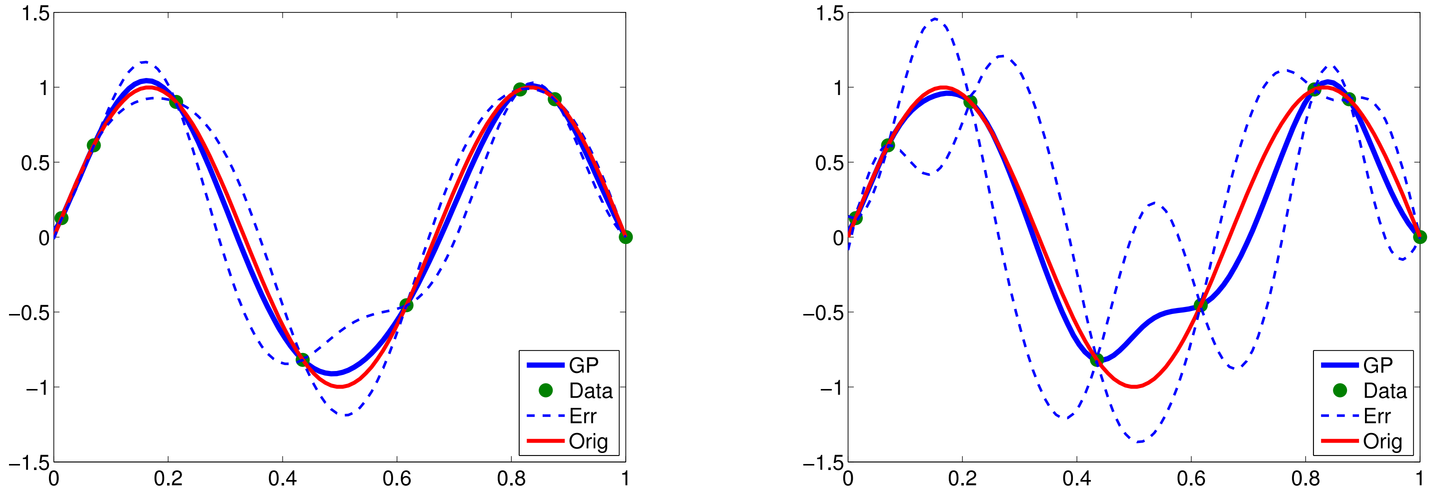
$$K(\mathbf{x}, \mathbf{y}) = e^{-\gamma \|\mathbf{x} \cdot \mathbf{y}\|^2}$$

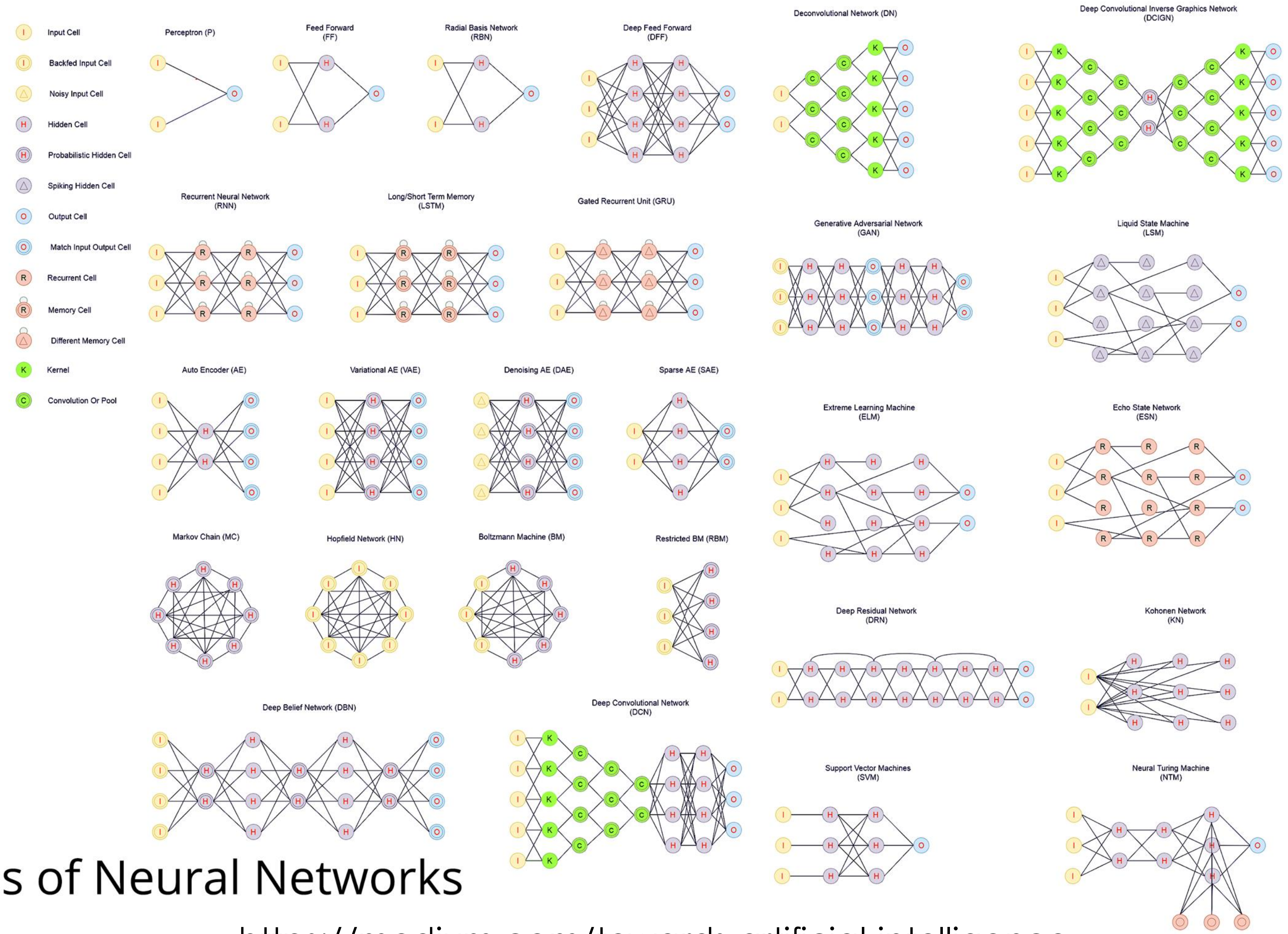
Gaussian Process Learning

Given the sets $\mathcal{S}_{x,N} := \{x_1, \ldots, x_N\}$, $\mathcal{S}_{f,N} := \{f(x_1), \ldots, f(x_N)\}$, and $\mathcal{S}_{\tilde{x},\tilde{N}} := \{\tilde{x}_1, \ldots, \tilde{x}_{\tilde{N}}\}$

$$F(\mathcal{S}_{\tilde{x},\tilde{N}})|\mathcal{S}_{x,N}, \mathbf{f}_{\mathcal{S}_{x,N}} \sim \mathcal{N}\Bigg(\mathbf{K}\Big(\mathcal{S}_{\tilde{x},\tilde{N}}, \mathcal{S}_{x,N}\Big)\Big[\mathbf{K}\Big(\mathcal{S}_{x,N}, \mathcal{S}_{x,N}\Big) + \sigma^2\mathbf{I}\Big]^{-1}\mathbf{f}_{\mathcal{S}_{x,N}},$$
$$\mathbf{K}\Big(\mathcal{S}_{\tilde{x},\tilde{N}}, \mathcal{S}_{\tilde{x},\tilde{N}}\Big) - \mathbf{K}\Big(\mathcal{S}_{\tilde{x},\tilde{N}}, \mathcal{S}_{x,N}\Big)\Big[\mathbf{K}\Big(\mathcal{S}_{x,N}, \mathcal{S}_{x,N}\Big) + \sigma^2\mathbf{I}\Big]^{-1}\mathbf{K}\Big(\mathcal{S}_{x,N}, \mathcal{S}_{\tilde{x},\tilde{N}}\Big)\Bigg),$$

where $K(\cdot,\cdot)$ is the covariance matrix, i.e $k_{i,j}(\alpha) = e^{-\frac{\alpha}{2}\|x_i - x_j\|^2}$.



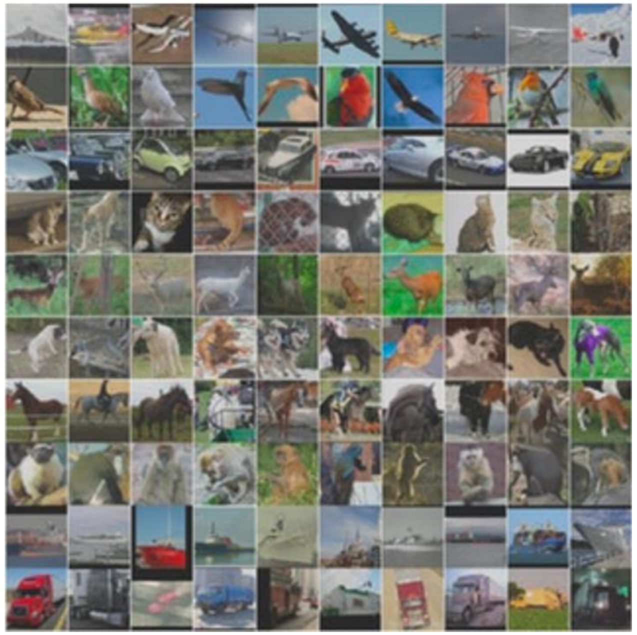


Main Types of Neural Networks

https://medium.com/towards-artificial-intelligence

Deep Neural Networks

$$\mathbf{Y}_{j+1} = \mathbf{Y}_j + hf\left(\mathbf{Y}_j, \Theta_j\right) \xRightarrow[i=1,\cdots,N]{} \dot{\mathbf{Y}}(t) = f\left(\mathbf{Y}(t), \Theta(t)\right)$$



Left: ImageNET Database

B. Chang, L Meng, E. Holtham, E. Haber, LR, D Begert *Reversible Architectures for Arbitrarily Deep ResNNs*. in review, arXiv, 2017.

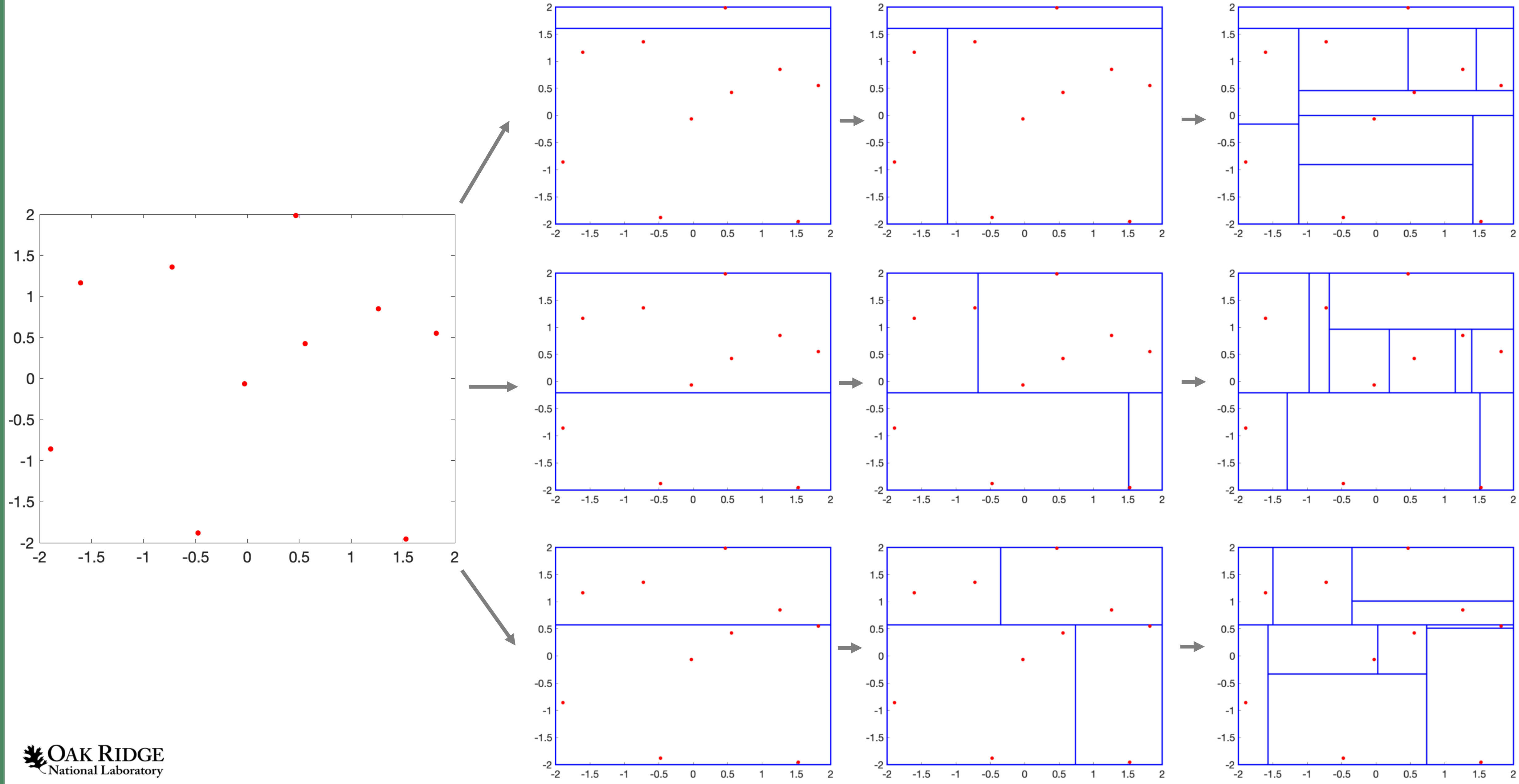
A. Mahendran, A Vedaldi *Understanding deep image representations by inverting them*. CVPR, 2015.

Google	Intel	Chinese UHK	Nvidia	Facebook	Xtract *
2013	2014	2015	2015	2016	2017
70.2%	72.8%	73.15%	74.10%	74.33%	85.5%

Stability Requirement: Network is forward stable when it does not amplify perturbations of the input features due to, for example, noise or adversarial attacks.

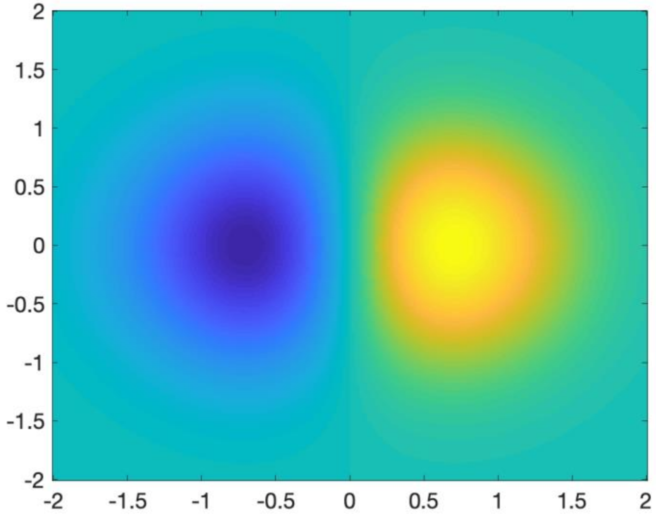
* Current record is 88.61% by **EfficientNet-L2-475** (in review arxiv.org/pdf/2010.01412v2.pdf) Google Research on Sharpness Aware Minimization

Random Decision Forest

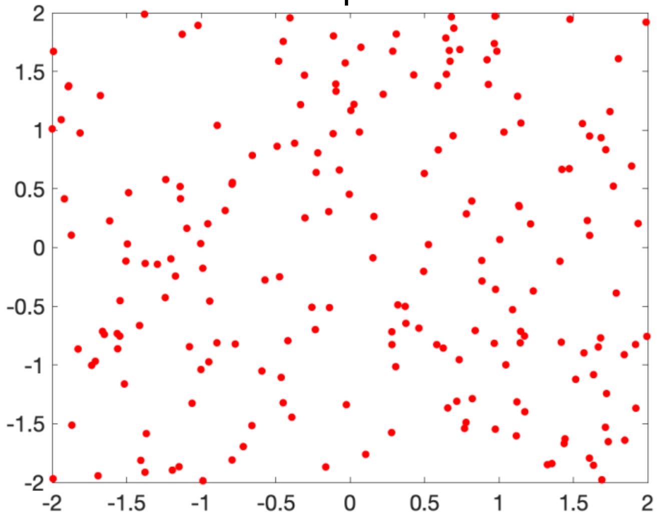


Random Decision Forest

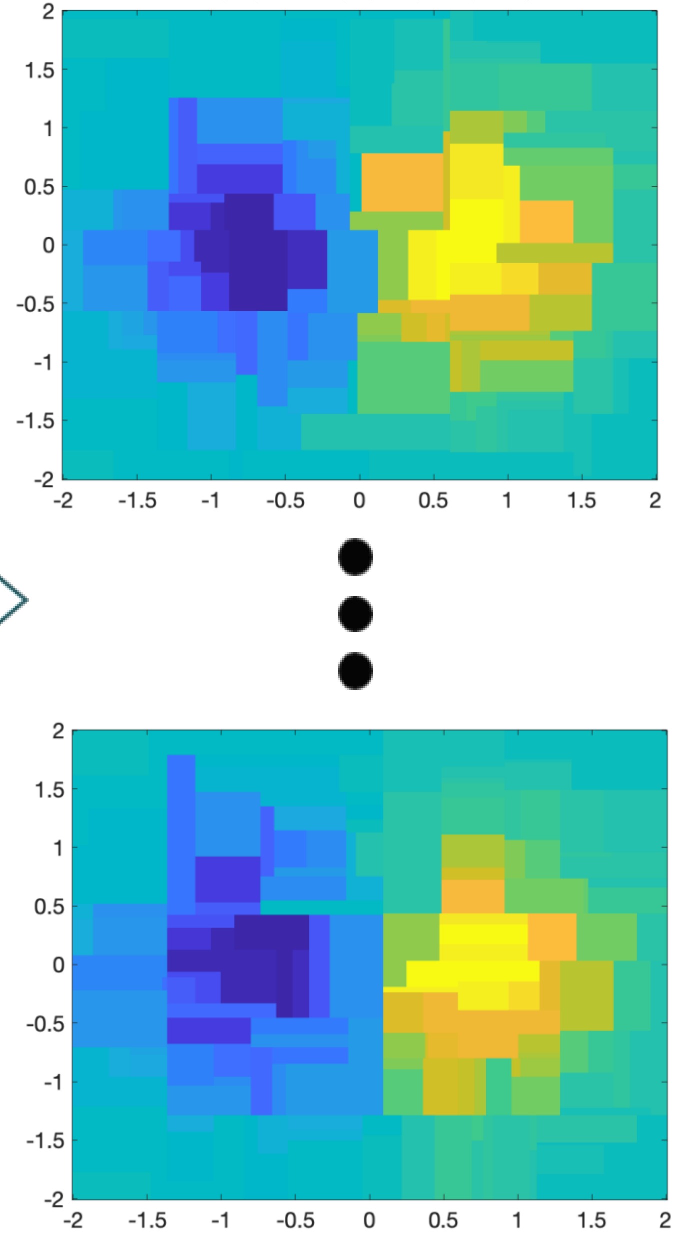
Original Function



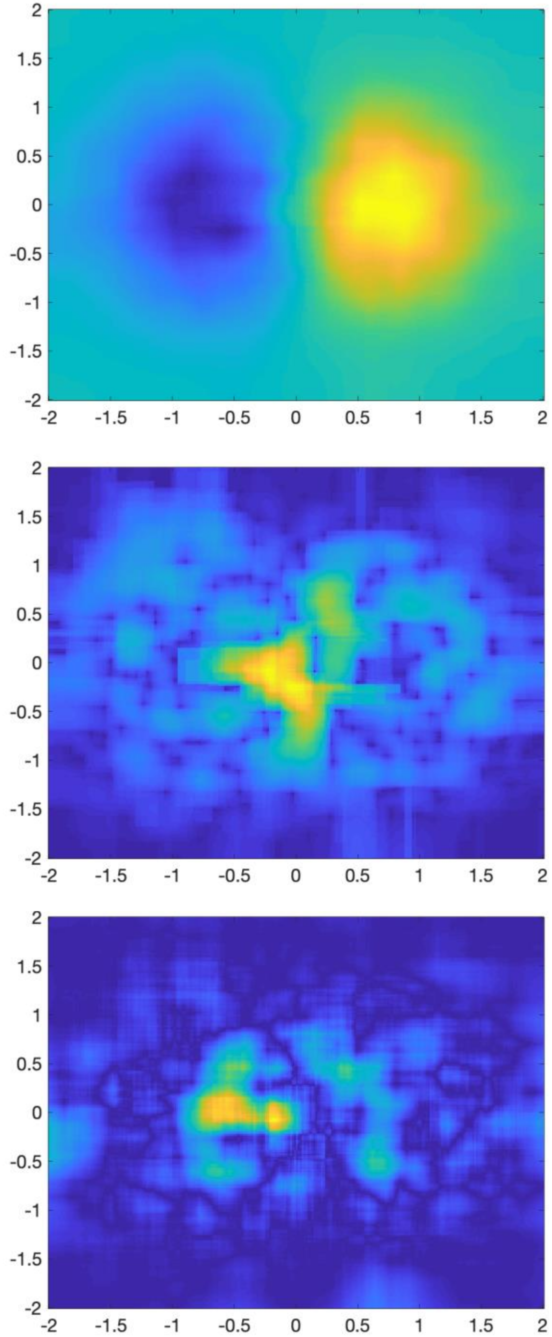
Samples



Tree Predictions

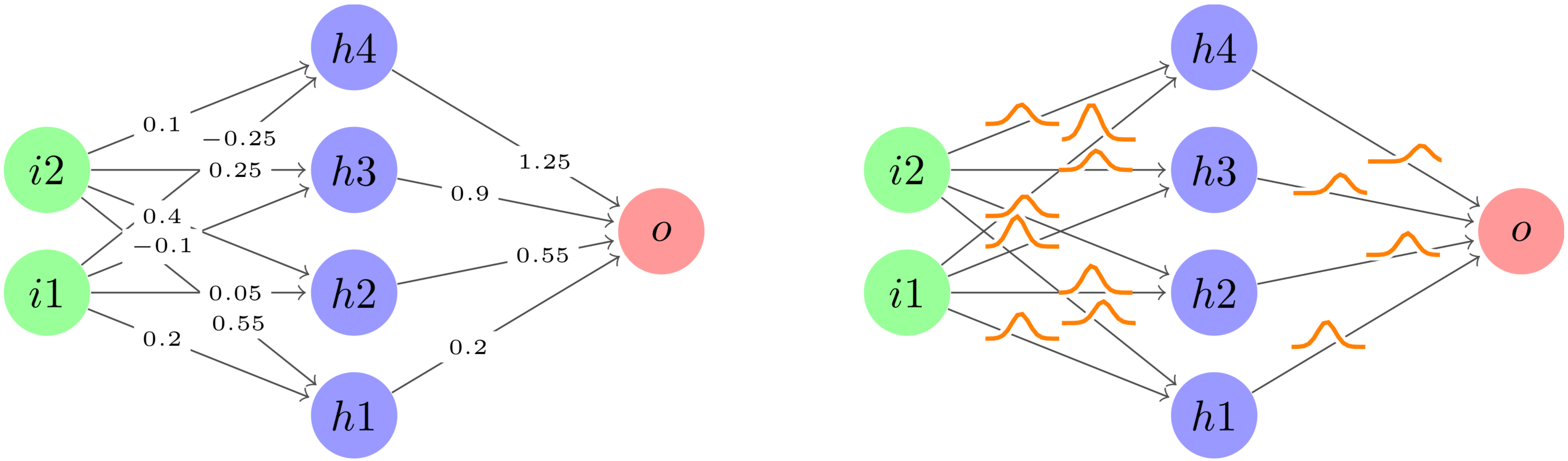


Forest Mean/STD and Error



Oak Ridge
National Laboratory

Bayesian Neural Networks



Unlike deterministic Neural Networks(left) that have a fixed value of their parameters, Bayesian Neural Networks(right) has a distributions linking nodes.

Stochastic Neural Network

How to account for uncertainty and control machine learning training.

Stochastic model equation

Activation function

Noise component

$$X_t = X_0 + \int_0^t F(X_s, \theta_s) ds + \int_0^t \sigma_s dW_s$$

Control process

Control terms

$$dX_t = F(X_t, \theta_t) dt + \sigma_t dW_t, \qquad 0 \leq t \leq T$$

Cost function

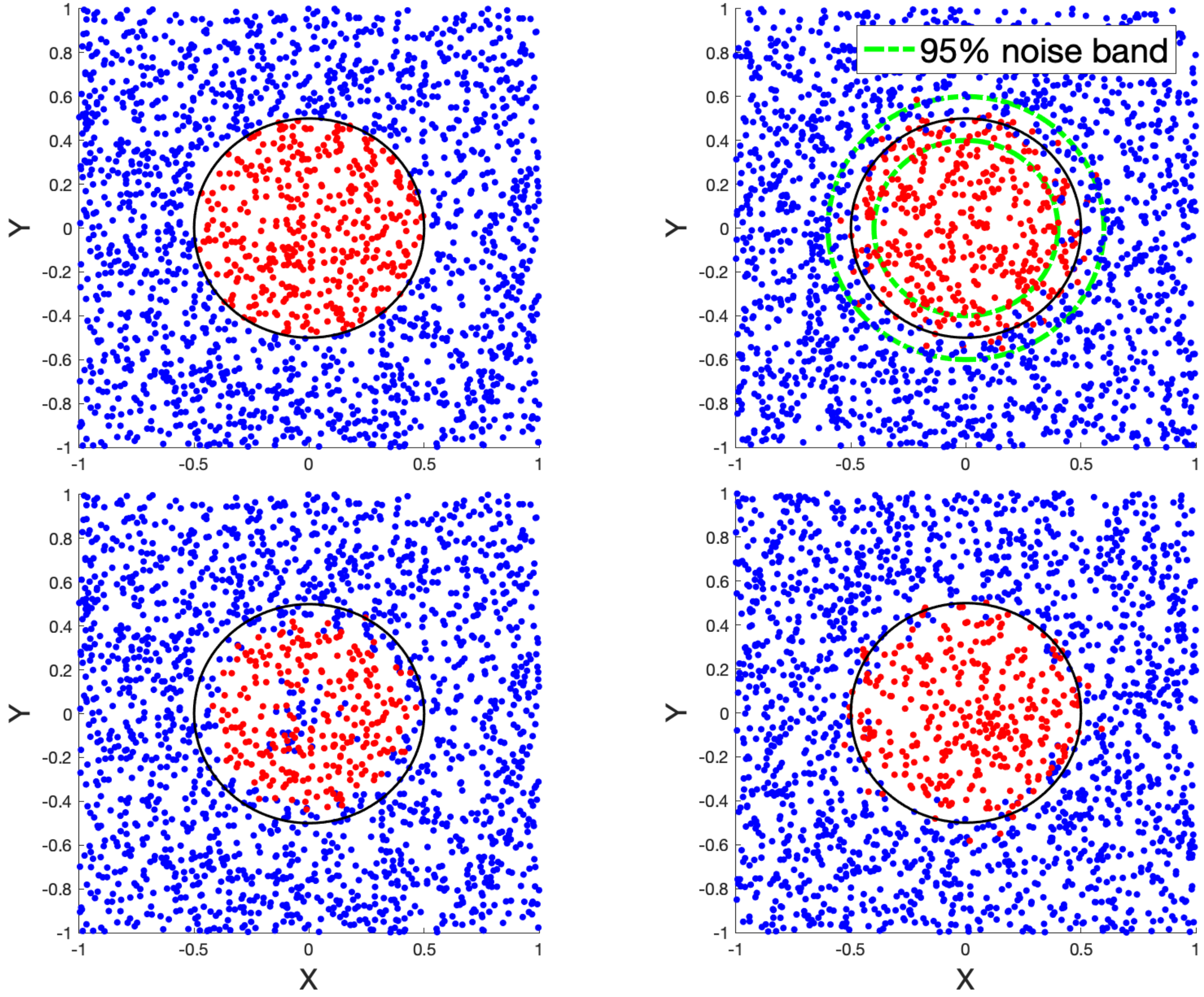
Measured data

$$J(u) := \mathbb{E}[\Phi(X_T, \Gamma)]$$

Challenge: Adaptation, robustness, and speed.

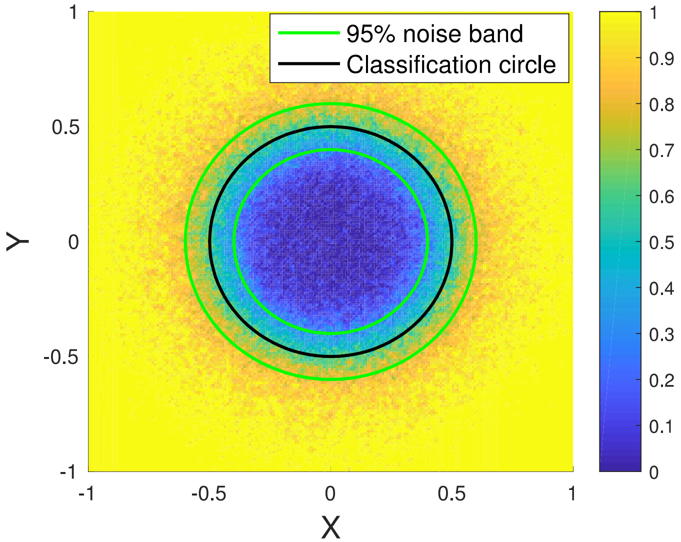
Bayesian vs Stochastic Neural Networks

Deterministic Sample | Uncertainty in Data



BNN | SNN-MP

- Robust Training
- Improved accuracy in prediction and uncertainty
- Adaptive Control

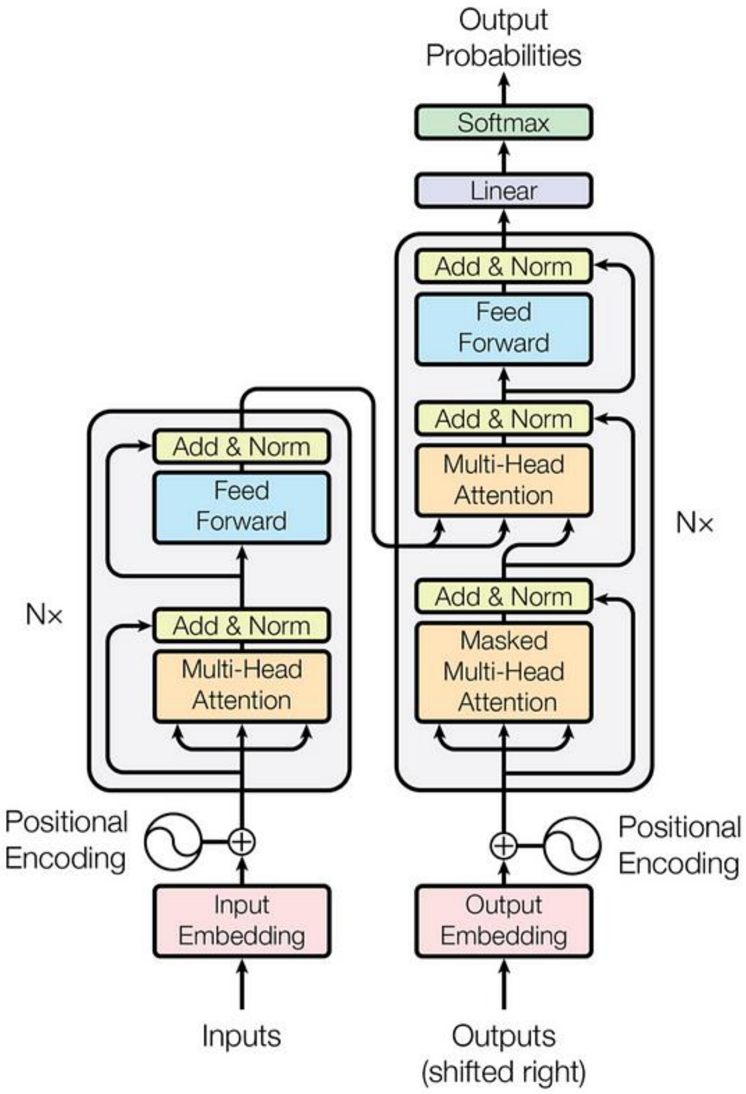


SNN-MP Weighted Prediction

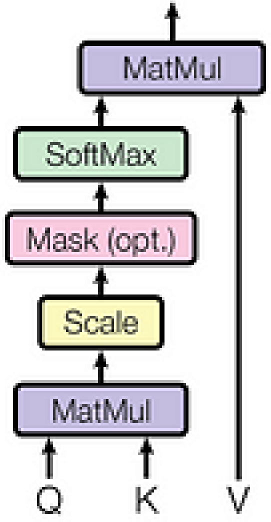
Archibald, Boa, Cao, & Zhang, 'Uncertainty Quantification in Deep Learning through Stochastic Maximum Principle', Submitted, 2020

Generative Pre-trained Transformer (GPT) - 4

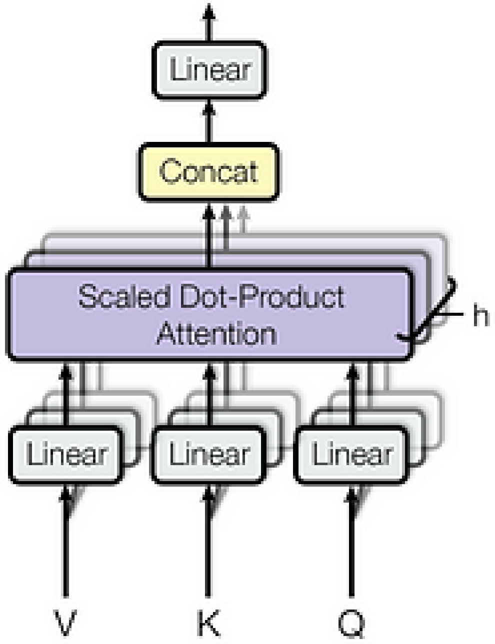
From ‘Attention Is All You Need’ by Vaswani et al. doi.org/10.48550/arXiv.1706.03762



Scaled Dot-Product Attention



Multi-Head Attention



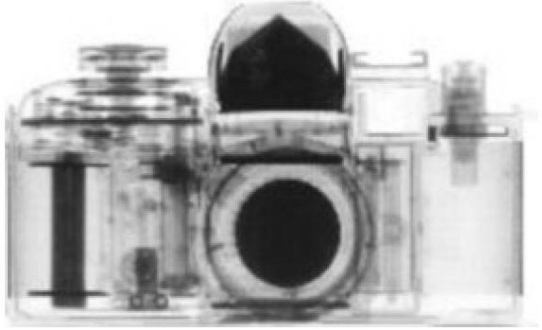
English-to-German and English-to-French newstest2014 tests

Model	BLEU		Training Cost (FLOPs)	
	EN-DE	EN-FR	EN-DE	EN-FR
ByteNet [18]	23.75			
Deep-Att + PosUnk [39]		39.2		$1.0 \cdot 10^{20}$
GNMT + RL [38]	24.6	39.92	$2.3 \cdot 10^{19}$	$1.4 \cdot 10^{20}$
ConvS2S [9]	25.16	40.46	$9.6 \cdot 10^{18}$	$1.5 \cdot 10^{20}$
MoE [32]	26.03	40.56	$2.0 \cdot 10^{19}$	$1.2 \cdot 10^{20}$
Deep-Att + PosUnk Ensemble [39]		40.4		$8.0 \cdot 10^{20}$
GNMT + RL Ensemble [38]	26.30	41.16	$1.8 \cdot 10^{20}$	$1.1 \cdot 10^{21}$
ConvS2S Ensemble [9]	26.36	**41.29**	$7.7 \cdot 10^{19}$	$1.2 \cdot 10^{21}$
Transformer (base model)	27.3	38.1	$\mathbf{3.3 \cdot 10^{18}}$	
Transformer (big)	**28.4**	**41.8**	$2.3 \cdot 10^{19}$	

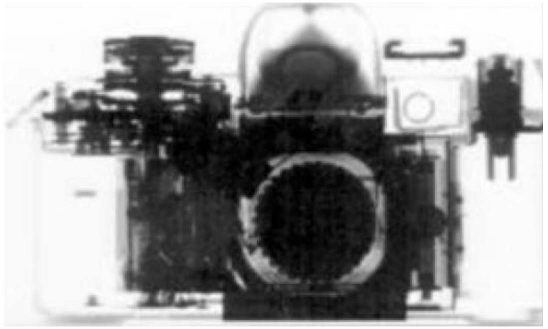
Experimental Science at DOE Facilities

$$\mathbf{d} = F(S, R) = S_{\{\Phi\}}(\mathbf{Q}, \omega) * R(\mathbf{Q}, \omega)$$

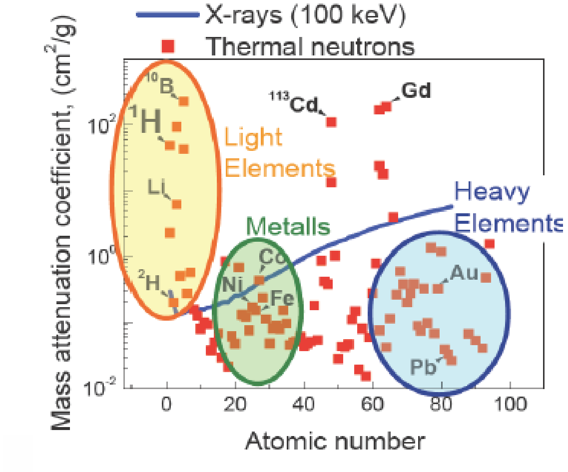
Why Neutrons, X-Rays or Electrons?



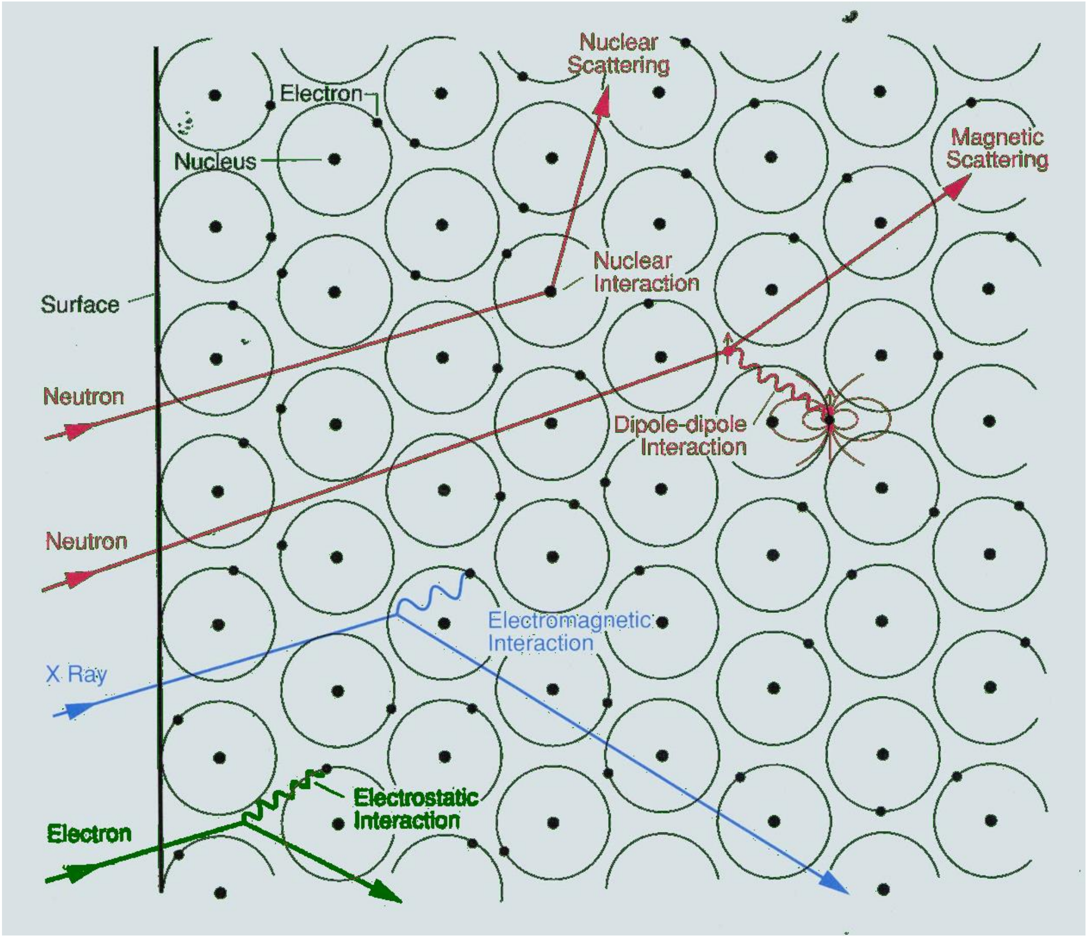
Neutrons



X-Rays

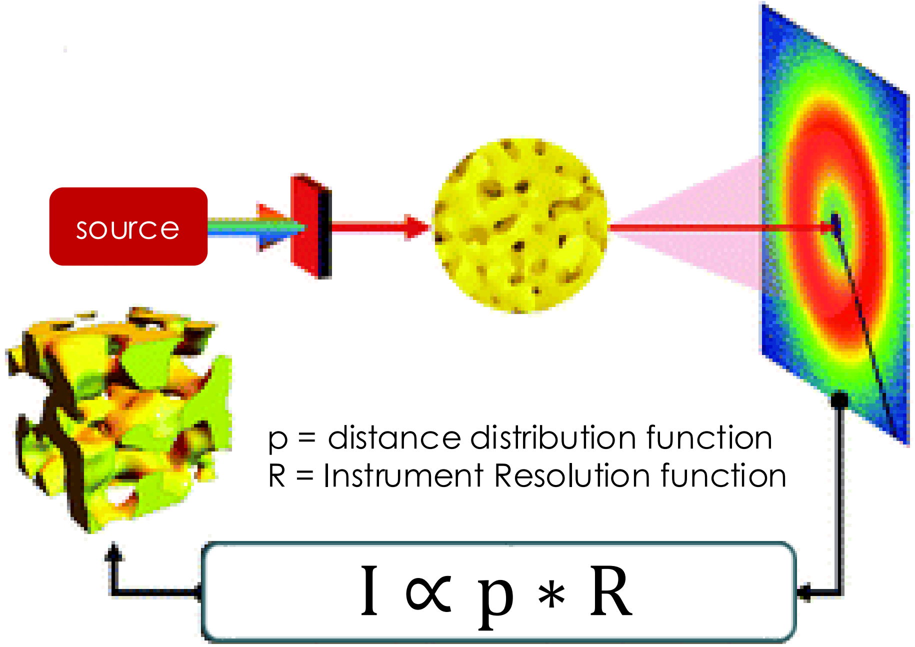


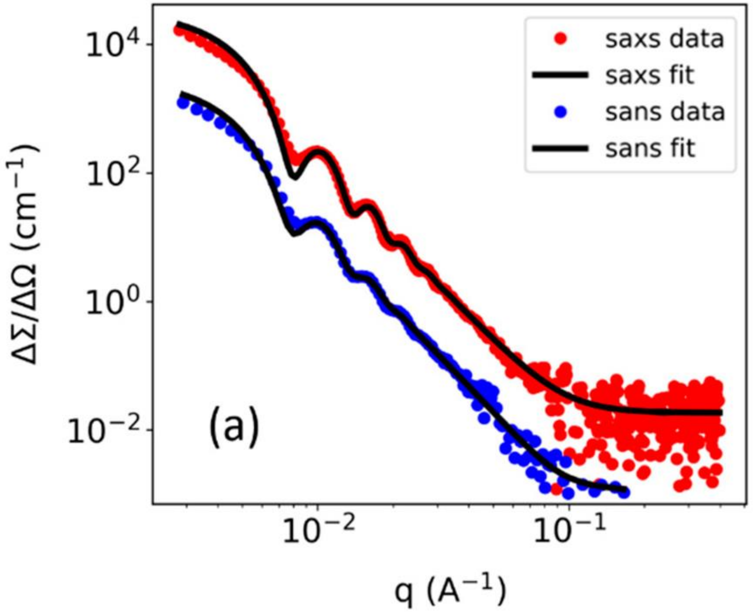
- Penetrate metals without absorbing
- Highly sensitive to water and hydrocarbons
- High contrast to light elements
- Sensitivity to magnetism
- Measure dynamics and structure

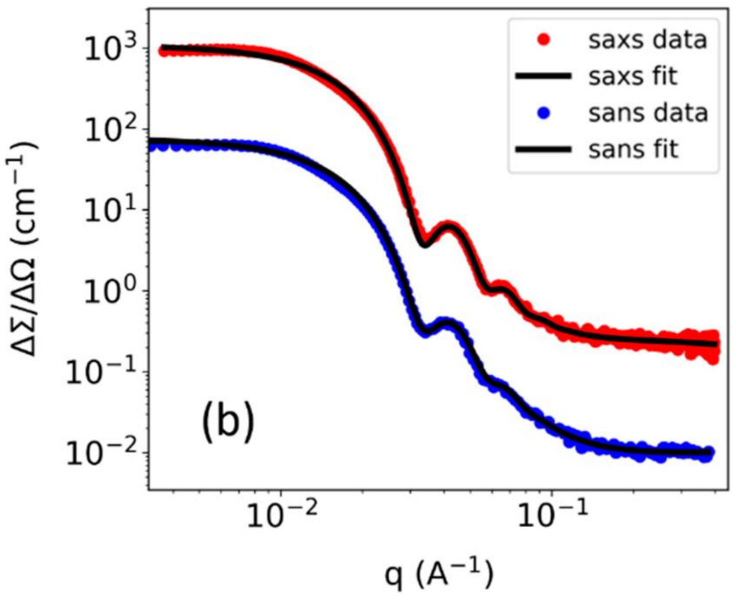


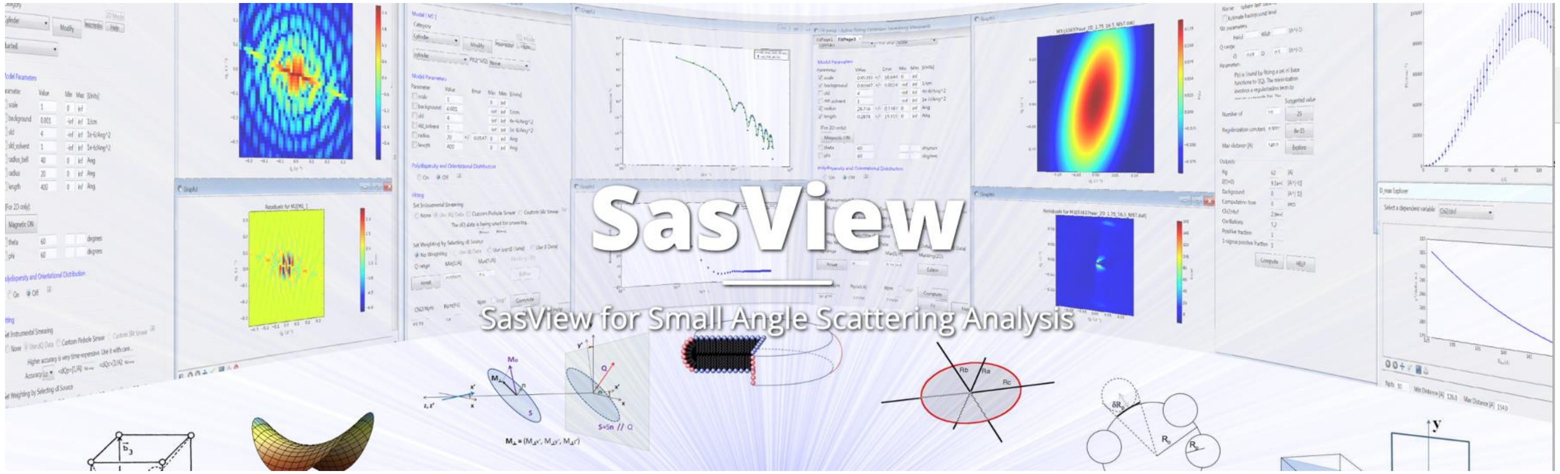
R. Pynn, ‘Neutron Scattering’, LANL

Small Angle Scattering (SAS)









SAS Model for distance distribution function

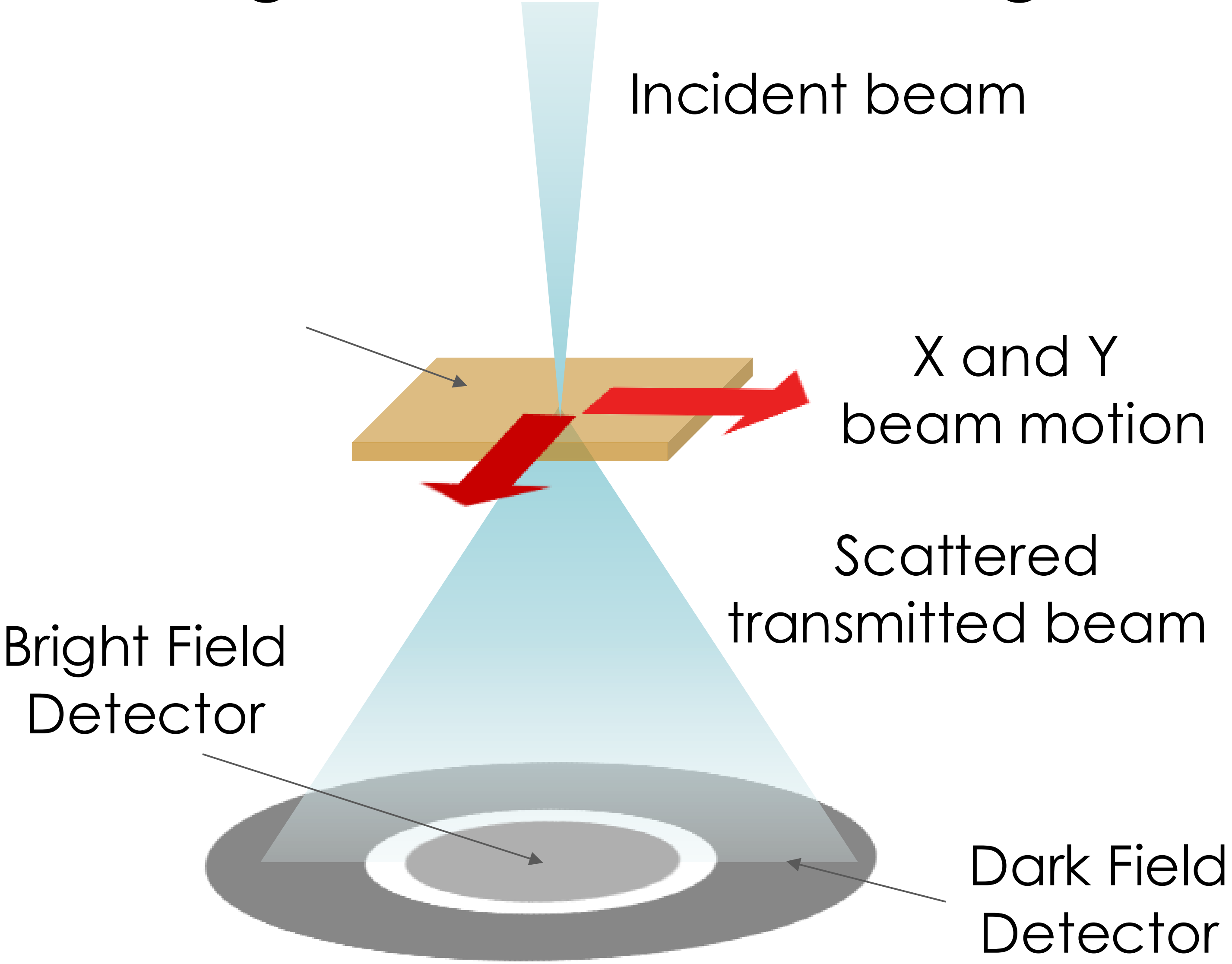
sphere

Spheres with uniform scattering length density

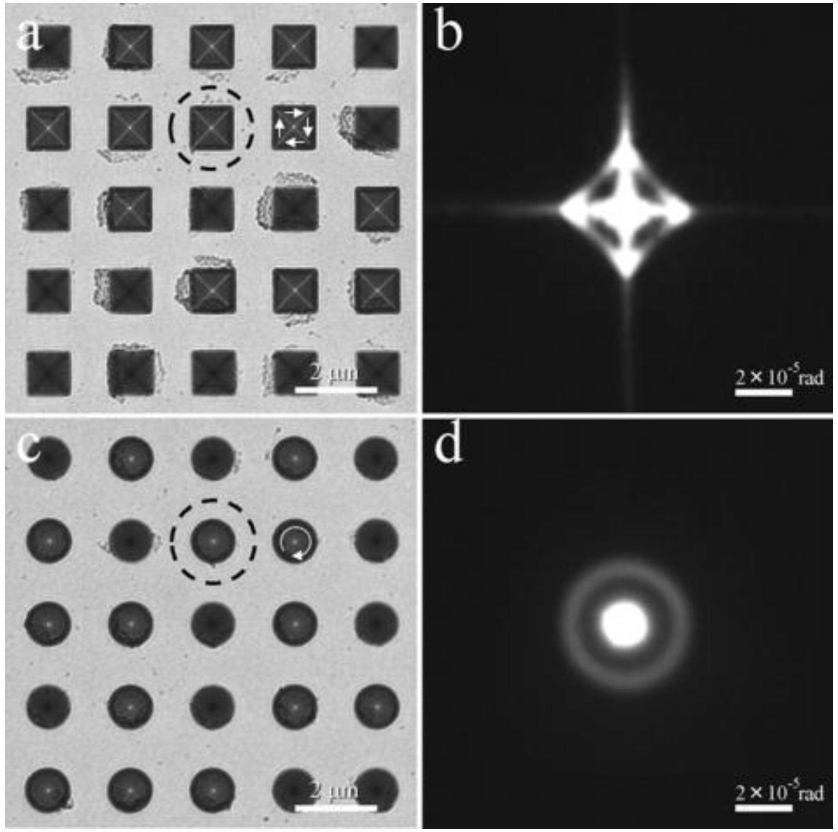
Parameter	Description	Units	Default value
scale	Scale factor or Volume fraction	None	1
background	Source background	cm^{-1}	0.001
sld	Layer scattering length density	$10^{-6}Å^{-2}$	1
sld_solvent	Solvent scattering length density	$10^{-6}Å^{-2}$	6
radius	Sphere radius	Å	50

$$I(q) = \frac{\text{scale}}{V} \cdot \left[3V(\Delta\rho) \cdot \frac{\sin(qr) - qr\cos(qr)}{(qr)^3} \right]^2 + \text{background}$$

Small Angle Electron Scattering



Scanning Transmission Electron Microscopy



(a) Fresnel micrograph of Permalloy squares and (c) disks. Small angle Lorentz deflection data of the single square (b) and disk (d).

AIP Advances. doi:10.1063/1.3701703

DOE Landscape Experimental Facility Landscape

Electron Sources

X-Ray Light Sources

SLAC NATIONAL ACCELERATOR LABORATORY

Six SAS Instruments

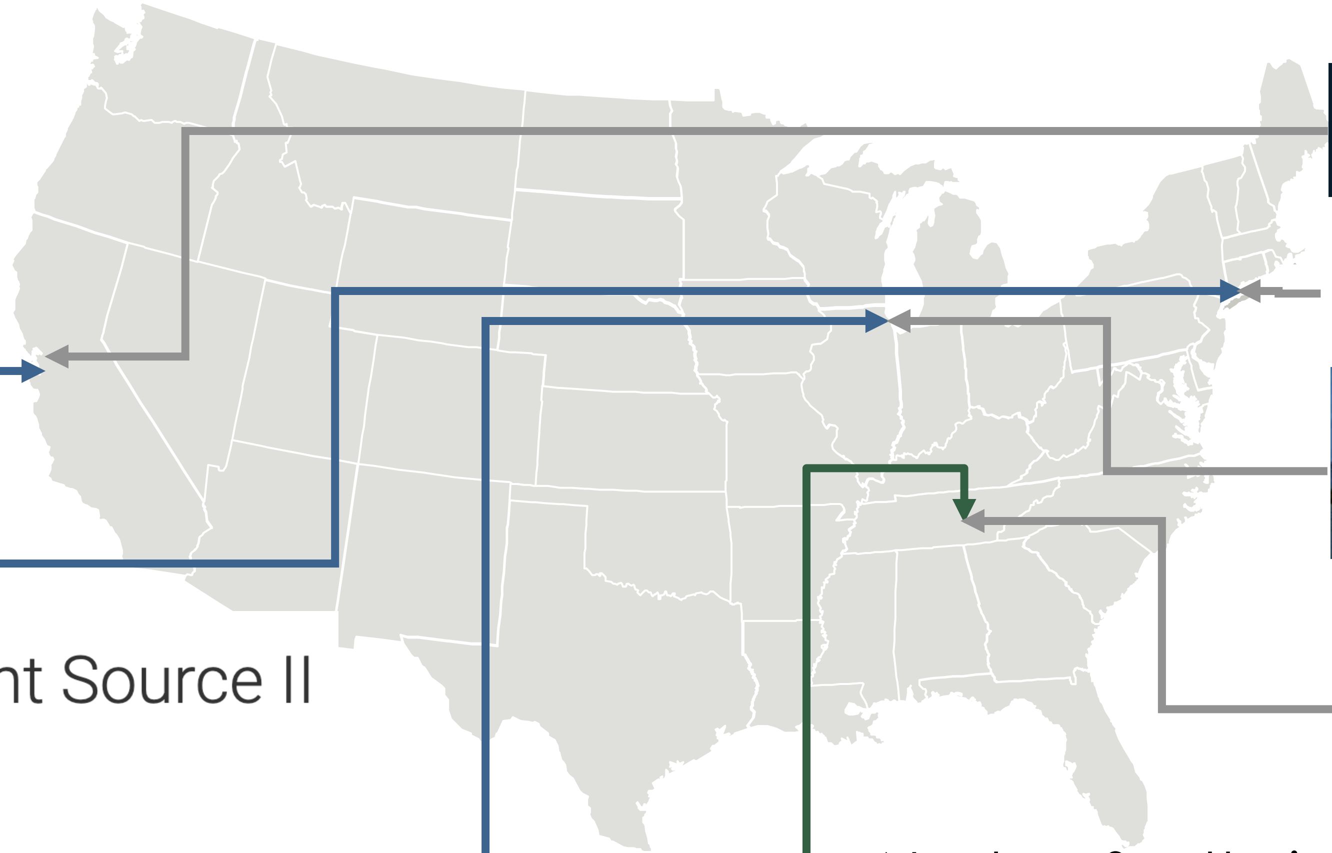
Brookhaven National Laboratory

National Synchrotron Light Source II

Five SAS Instruments

Advanced Photon Source

Twelve SAS Instruments



MOLECULAR FOUNDRY

Center for Functional Nanomaterials
Brookhaven National Laboratory

THE CENTER FOR NANOSCALE MATERIALS

Center for Nanophase Materials Sciences
AT OAK RIDGE NATIONAL LABORATORY

Neutron Scattering Facilities

HFIR

Two SAS Instruments

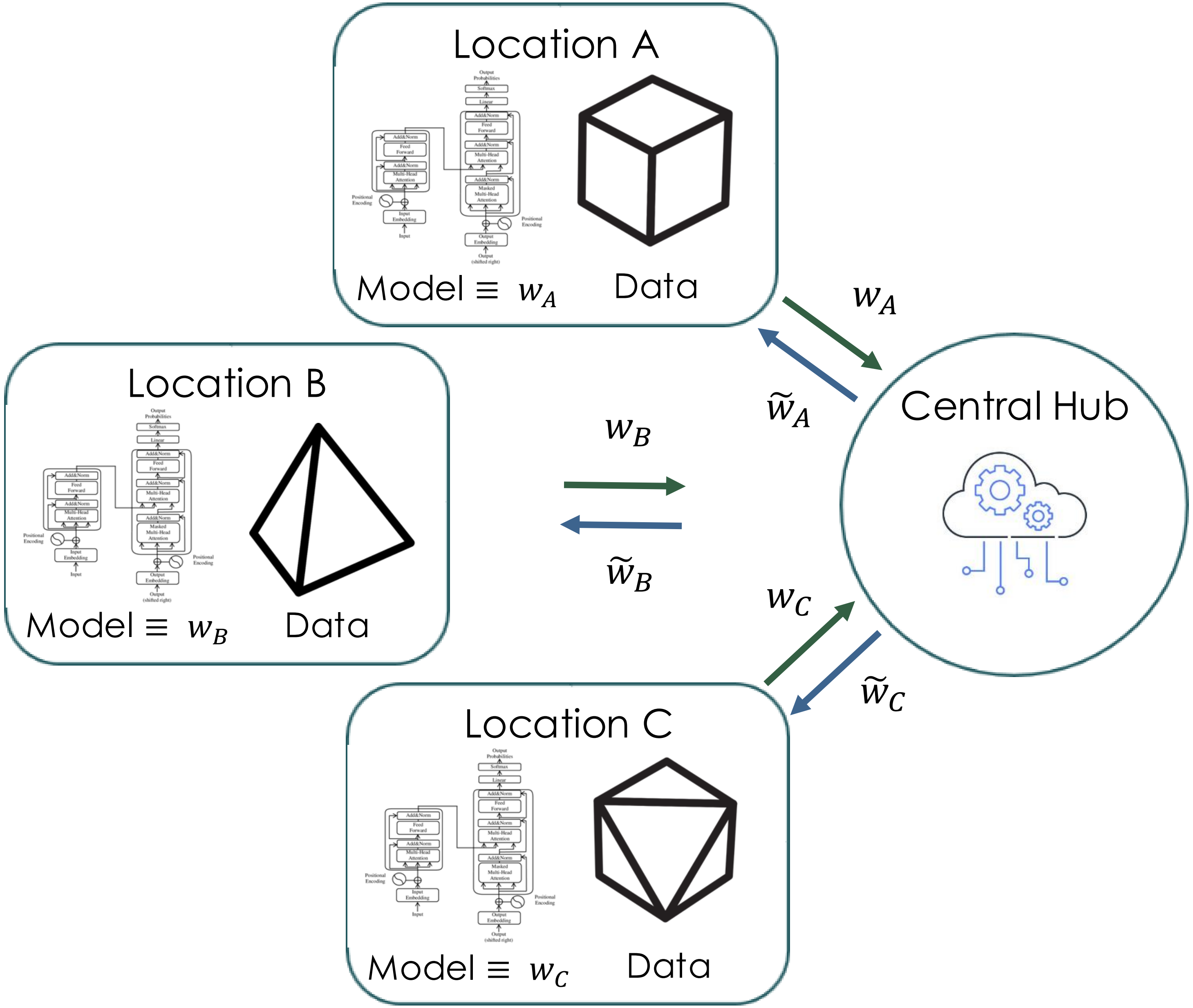
SNS SPALLATION NEUTRON SOURCE

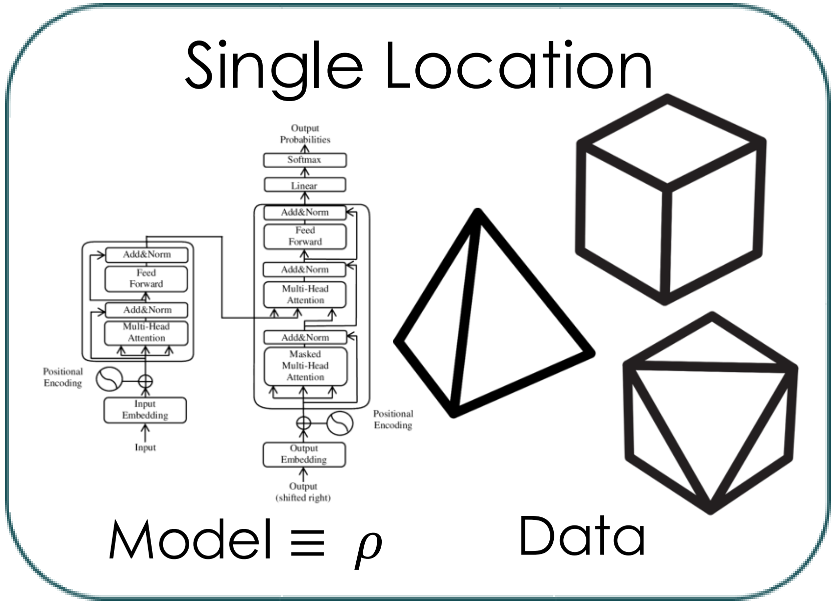
One SAS Instruments

OAK RIDGE National Laboratory

FedAvg Update

$$\tilde{w} = \frac{1}{N}\sum_{j=1}^{J} N_j\, w_j;$$





Gold Standard Federated Learning Comparison

Accelerated Update

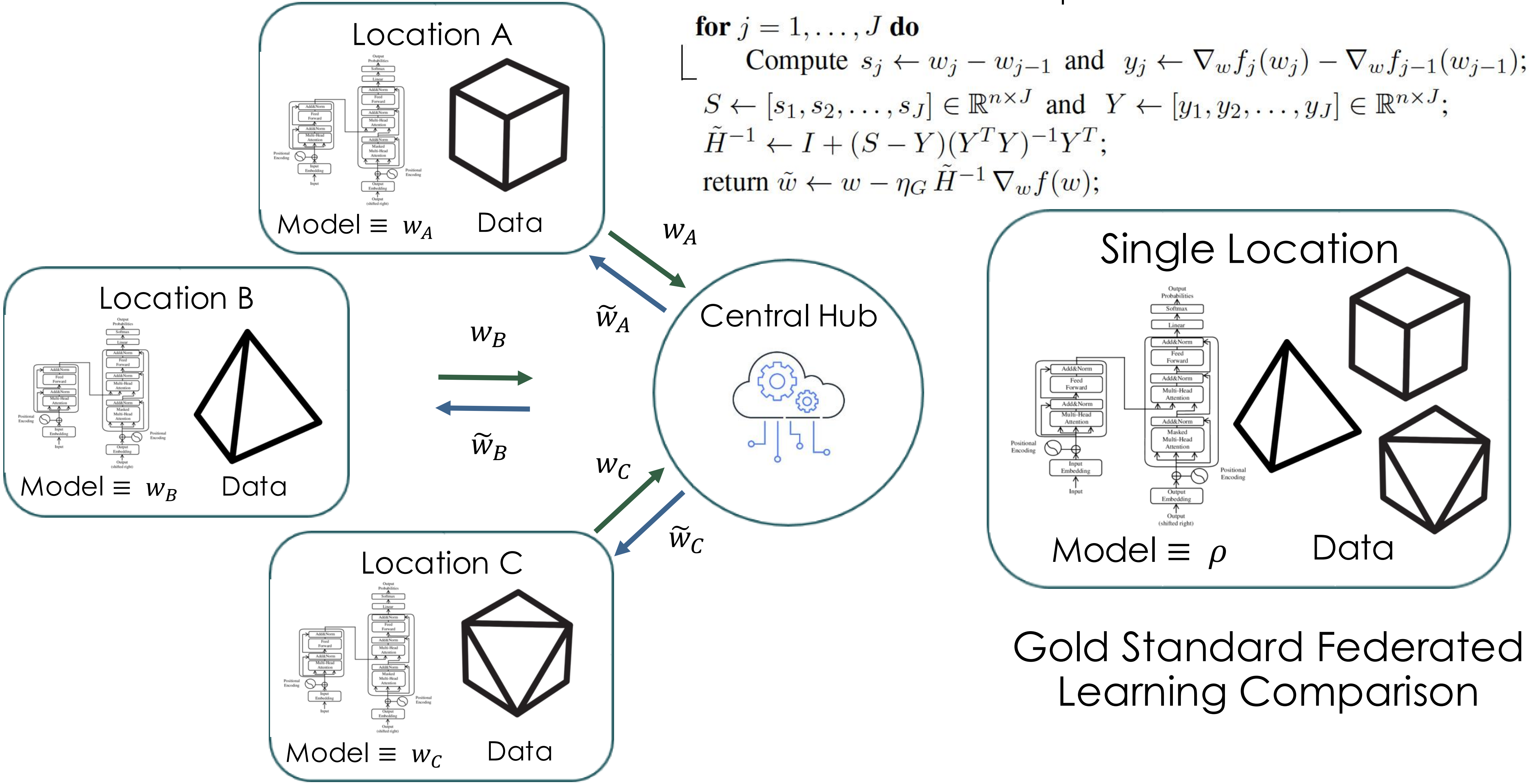
for $j = 1, \ldots, J$ **do**

  Compute $s_j \leftarrow w_j - w_{j-1}$ and $y_j \leftarrow \nabla_w f_j(w_j) - \nabla_w f_{j-1}(w_{j-1})$;

$S \leftarrow [s_1, s_2, \ldots, s_J] \in \mathbb{R}^{n \times J}$ and $Y \leftarrow [y_1, y_2, \ldots, y_J] \in \mathbb{R}^{n \times J}$;

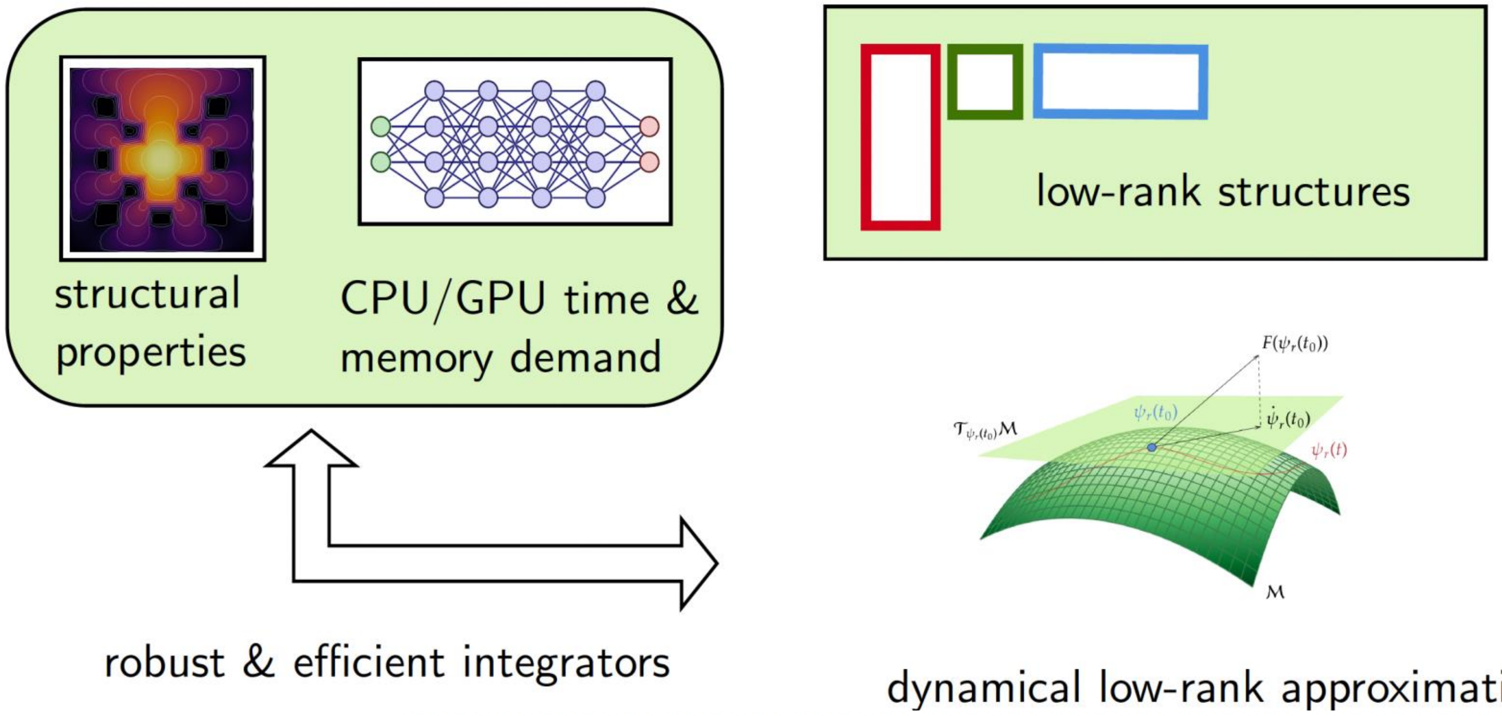
$\tilde{H}^{-1} \leftarrow I + (S - Y)(Y^T Y)^{-1} Y^T$;

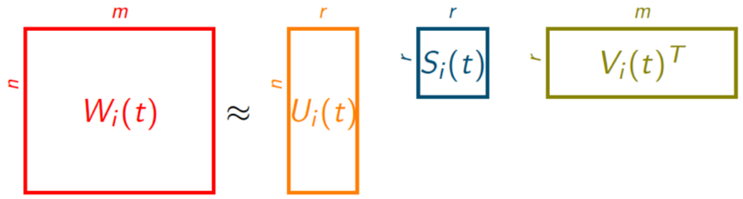
return $\tilde{w} \leftarrow w - \eta_G \tilde{H}^{-1} \nabla_w f(w)$;



Gold Standard Federated Learning Comparison

Dynamical Low-Rank Approximation for Neural Networks





$$
\begin{aligned}
\dot{S}_k &= -U_k^\top \nabla_{W_k} \mathcal{L}(W_k(t)) V_k \ , \\
\dot{U}_k &= -(I - U_k U_k^\top) \nabla_{W_k} \mathcal{L}(W_k(t)) V_k S_k^{-1} \ , \\
\dot{V}_k &= -(I - V_k V_k^\top) \nabla_{W_k} \mathcal{L}(W_k(t))^\top U_k S_k^{-\top} \ .
\end{aligned}
$$

$$f(x; W) = \{\sigma_\ell \circ A_\ell \circ \cdots \circ \sigma_1 \circ A_1\}(x)$$

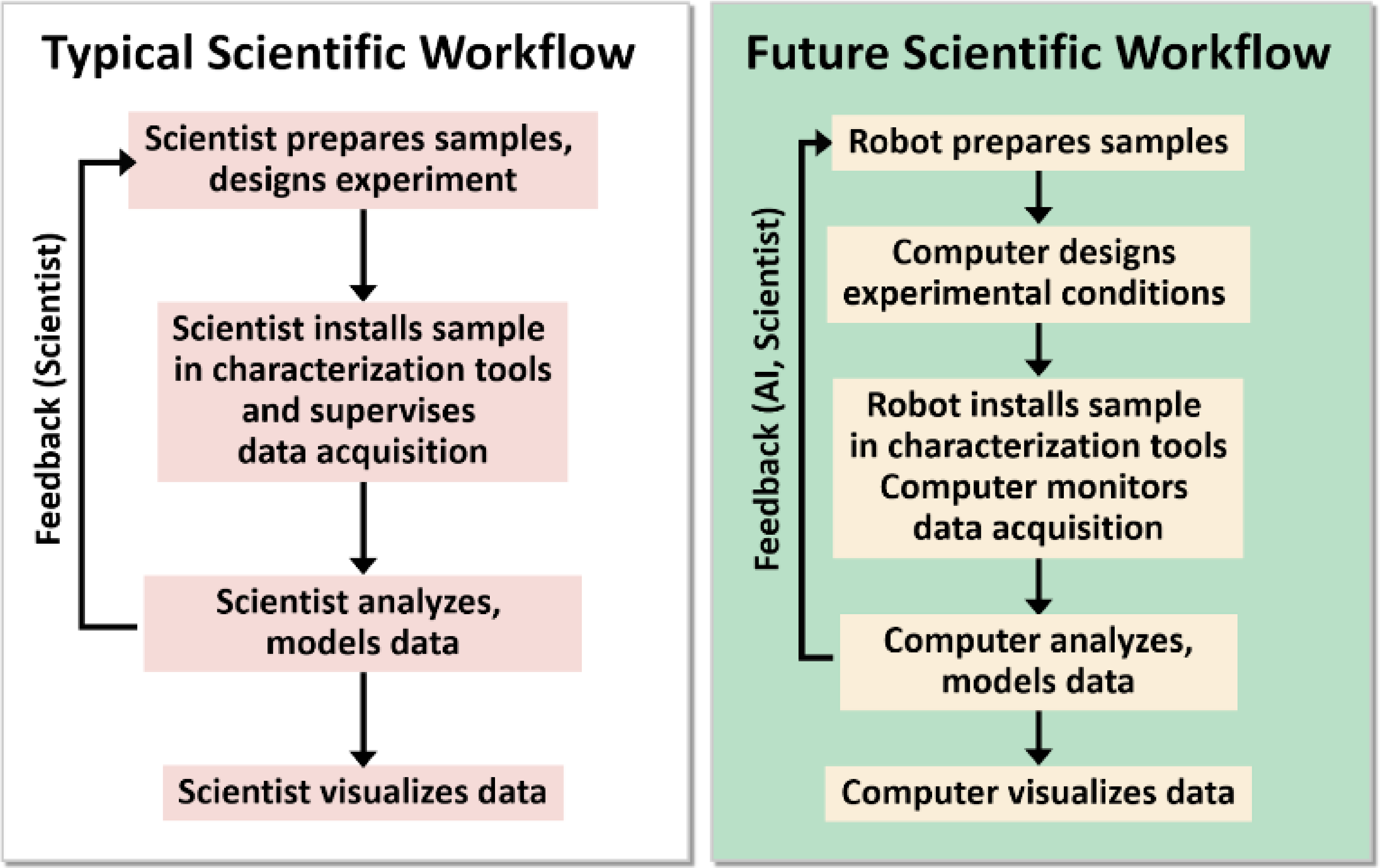
$$A_i(x) = W_i x$$

$$\dot{W}_i(t) = -\nabla_{W_i} \mathcal{L}(f(x; W(t)), y) \qquad \forall i = 1, \dots, \ell$$

Oak Ridge National Laboratory

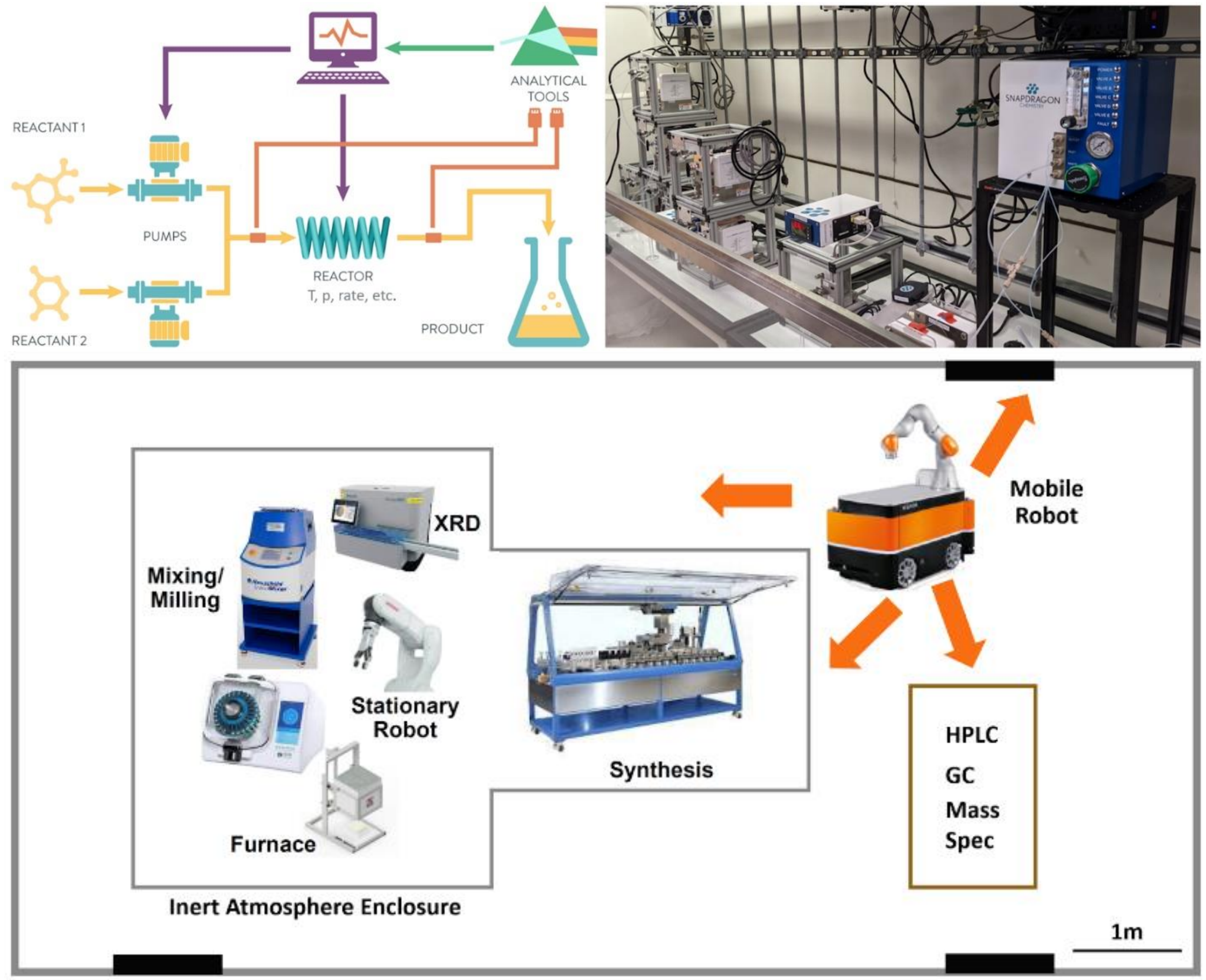
*Research and code from Steffen Schotthöfer at ORNL

Federated Learning + INTERSECT + Laboratory of the Future

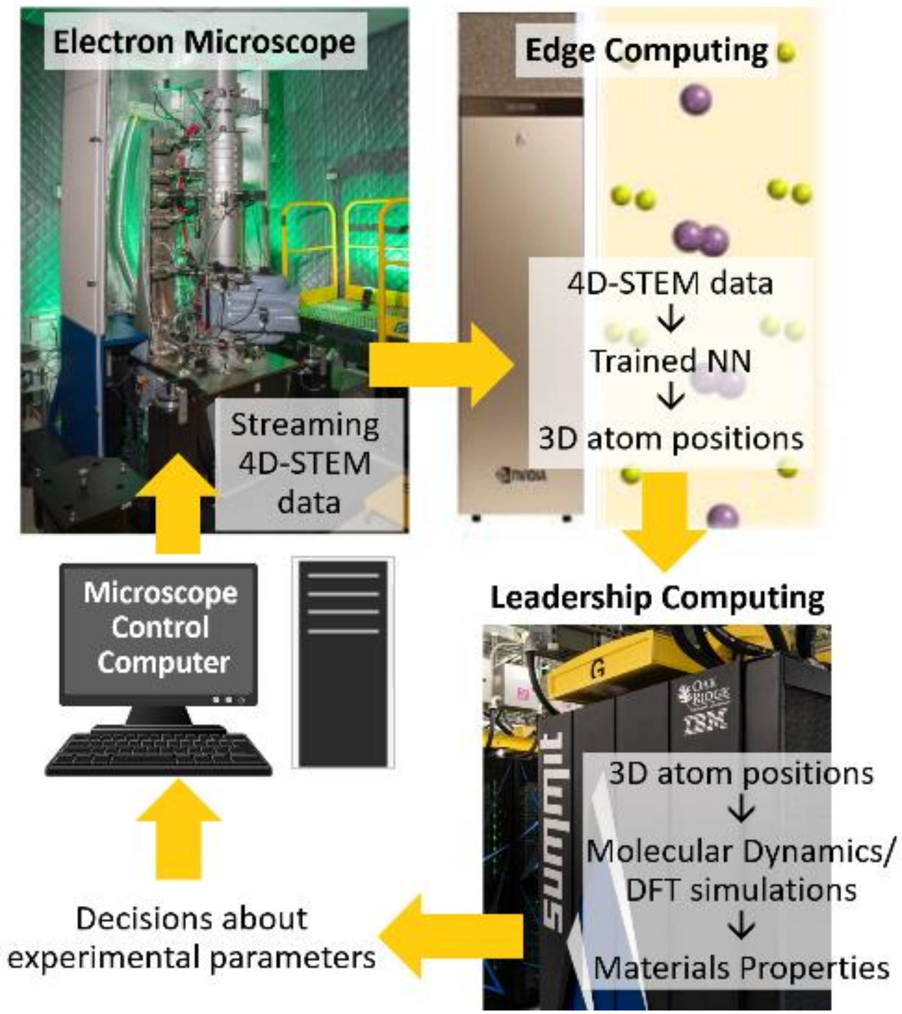


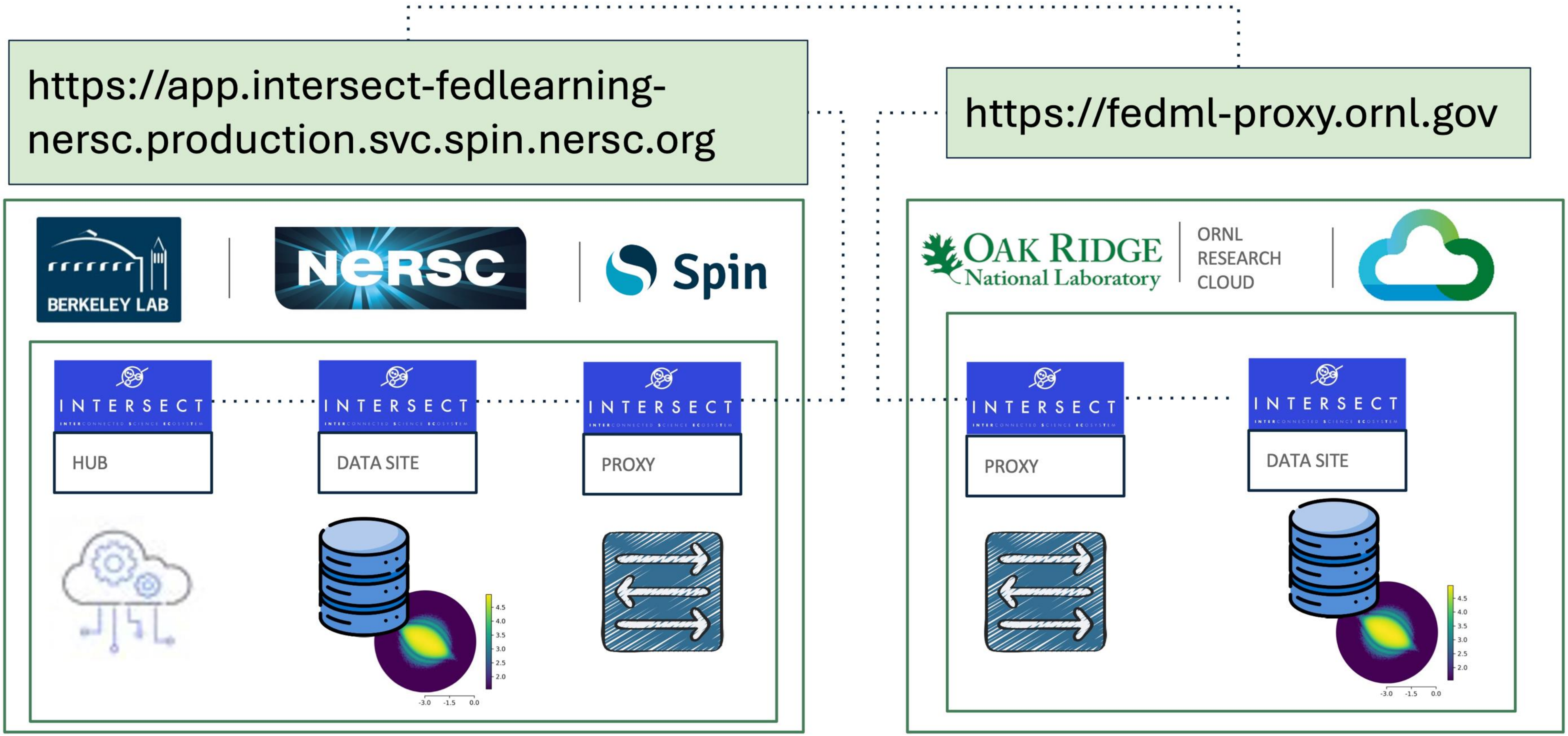
Federated Learning + INTERSECT + Laboratory of the Future

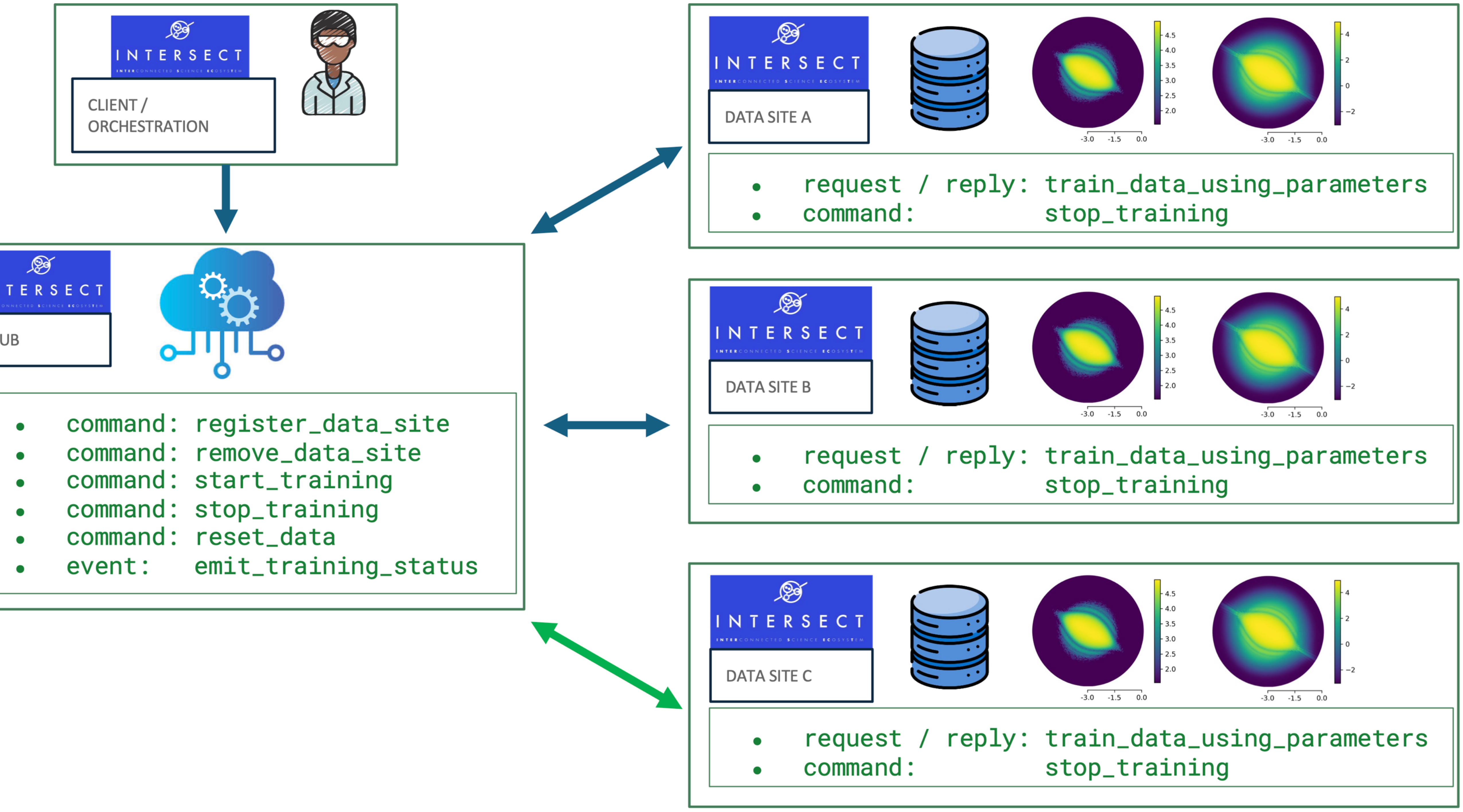
Autonomous Chemistry



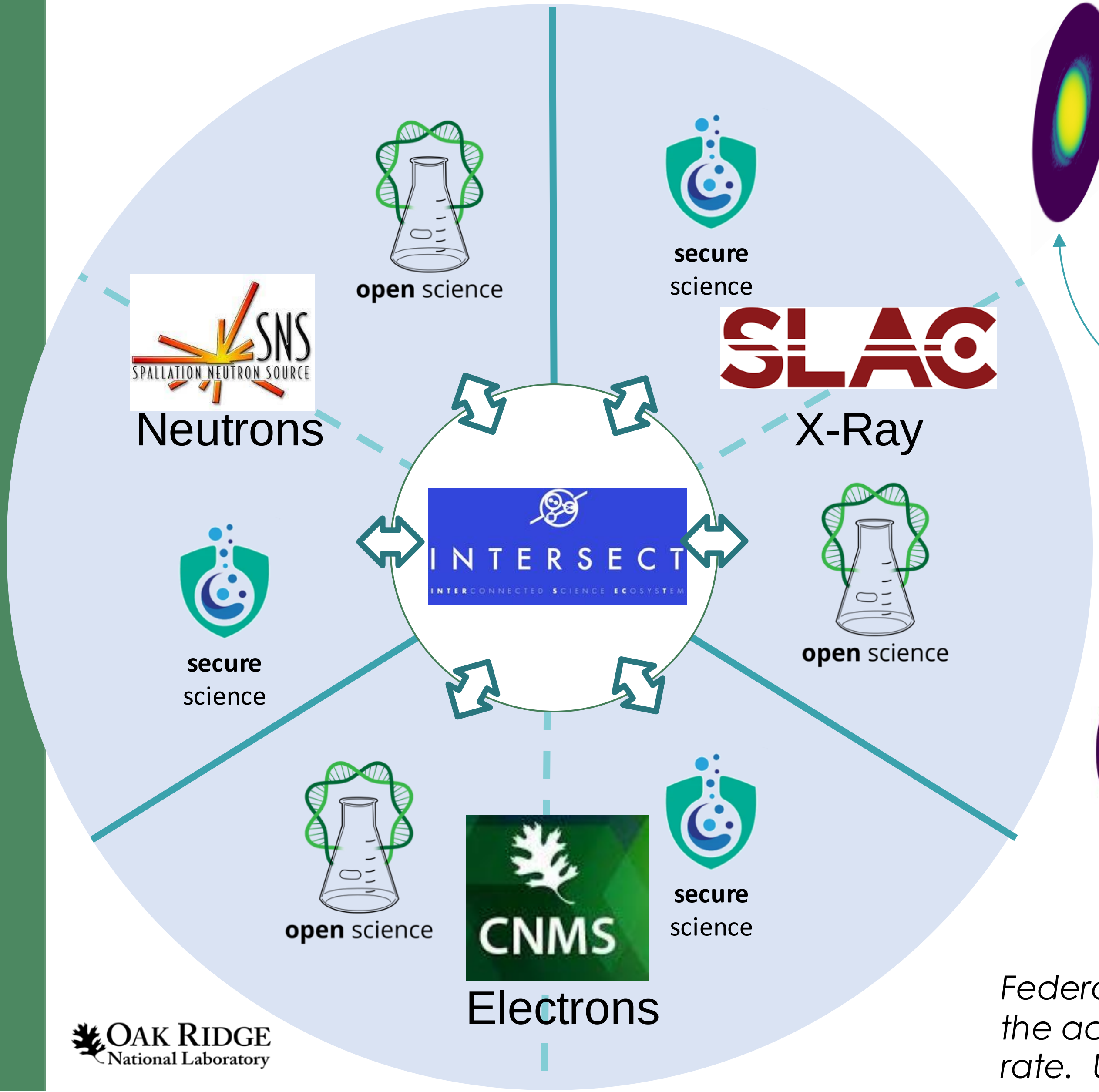
Autonomous Spectroscopy

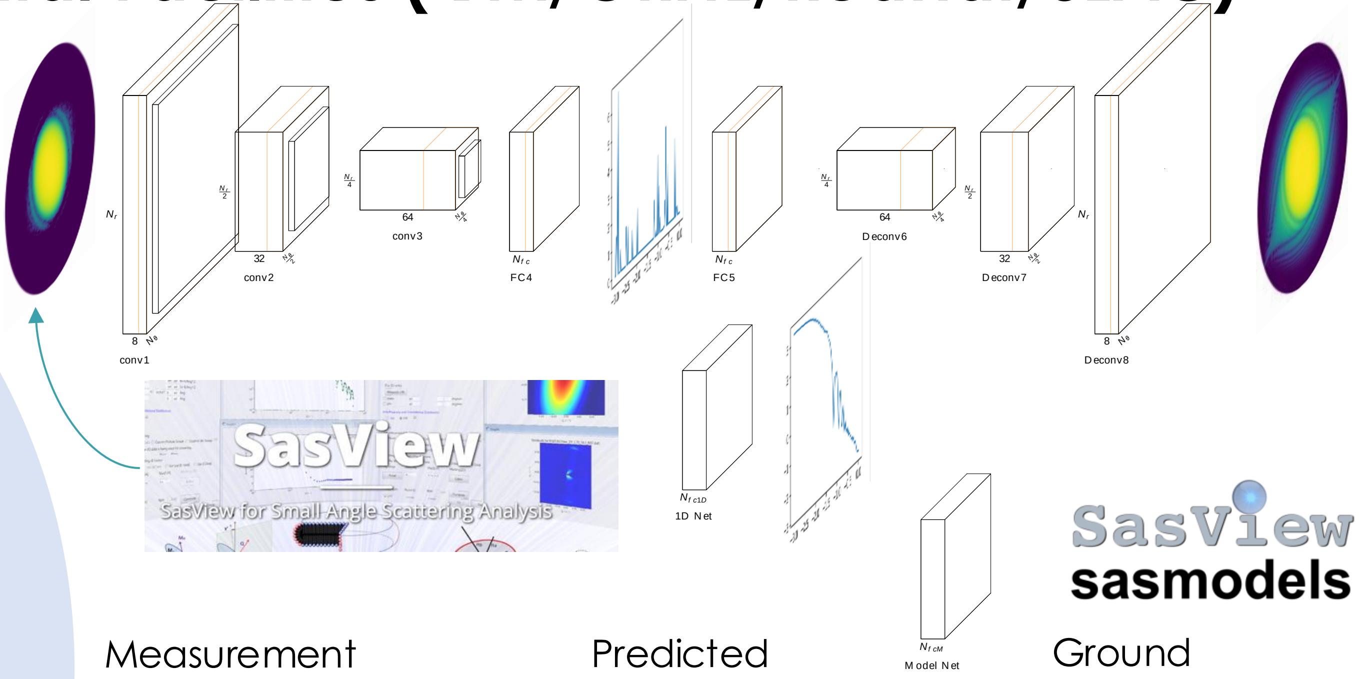


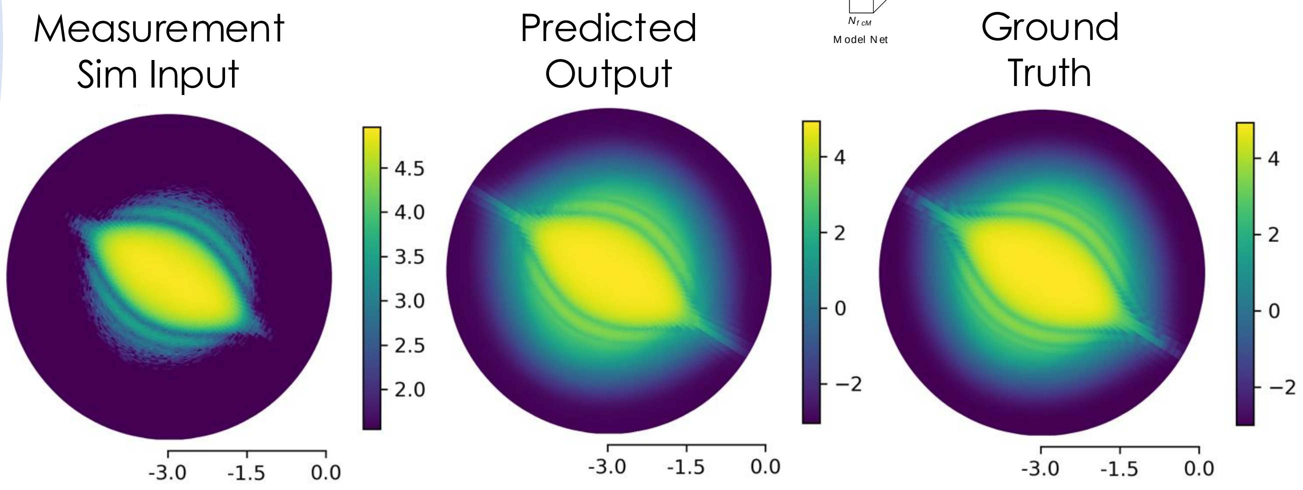

https://app.intersect-fedlearning-nersc.production.svc.spin.nersc.org
https://fedml-proxy.ornl.gov
BERKELEY LAB
NERSC
Spin
OAK RIDGE National Laboratory
ORNL RESEARCH CLOUD
INTERSECT
HUB
INTERSECT
DATA SITE
INTERSECT
PROXY
INTERSECT
PROXY
INTERSECT
DATA SITE


INTERSECT
INTERCONNECTED SCIENCE ECOSYSTEM
CLIENT / ORCHESTRATION
TERSECT
UB
• command: register_data_site
• command: remove_data_site
• command: start_training
• command: stop_training
• command: reset_data
• event: emit_training_status
INTERSECT
DATA SITE A
• request / reply: train_data_using_parameters
• command: stop_training
INTERSECT
DATA SITE B
• request / reply: train_data_using_parameters
• command: stop_training
INTERSECT
DATA SITE C
• request / reply: train_data_using_parameters
• command: stop_training

Federated Learning for Experimental Facilities (IBM/ORNL/RedHat/SLAC)

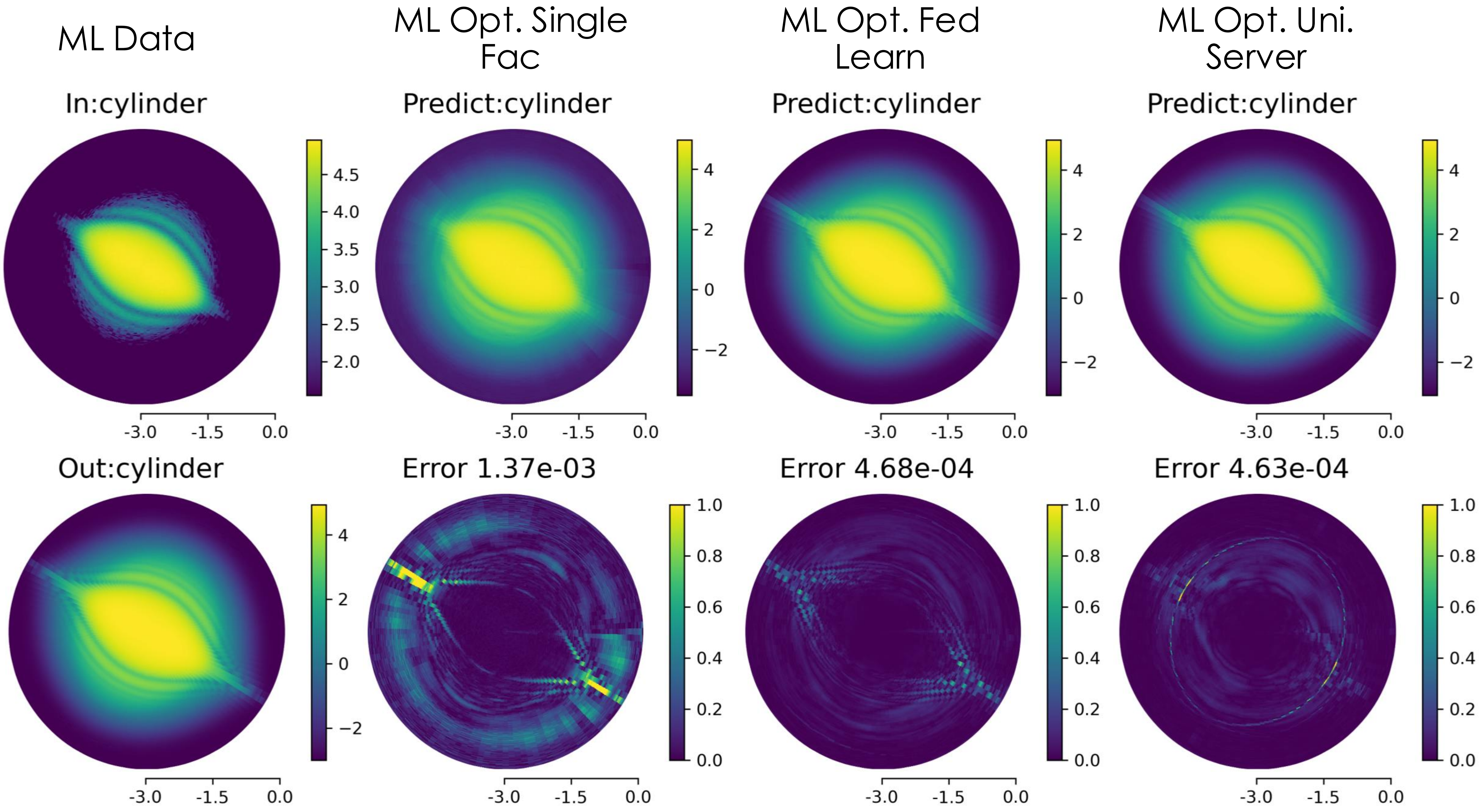


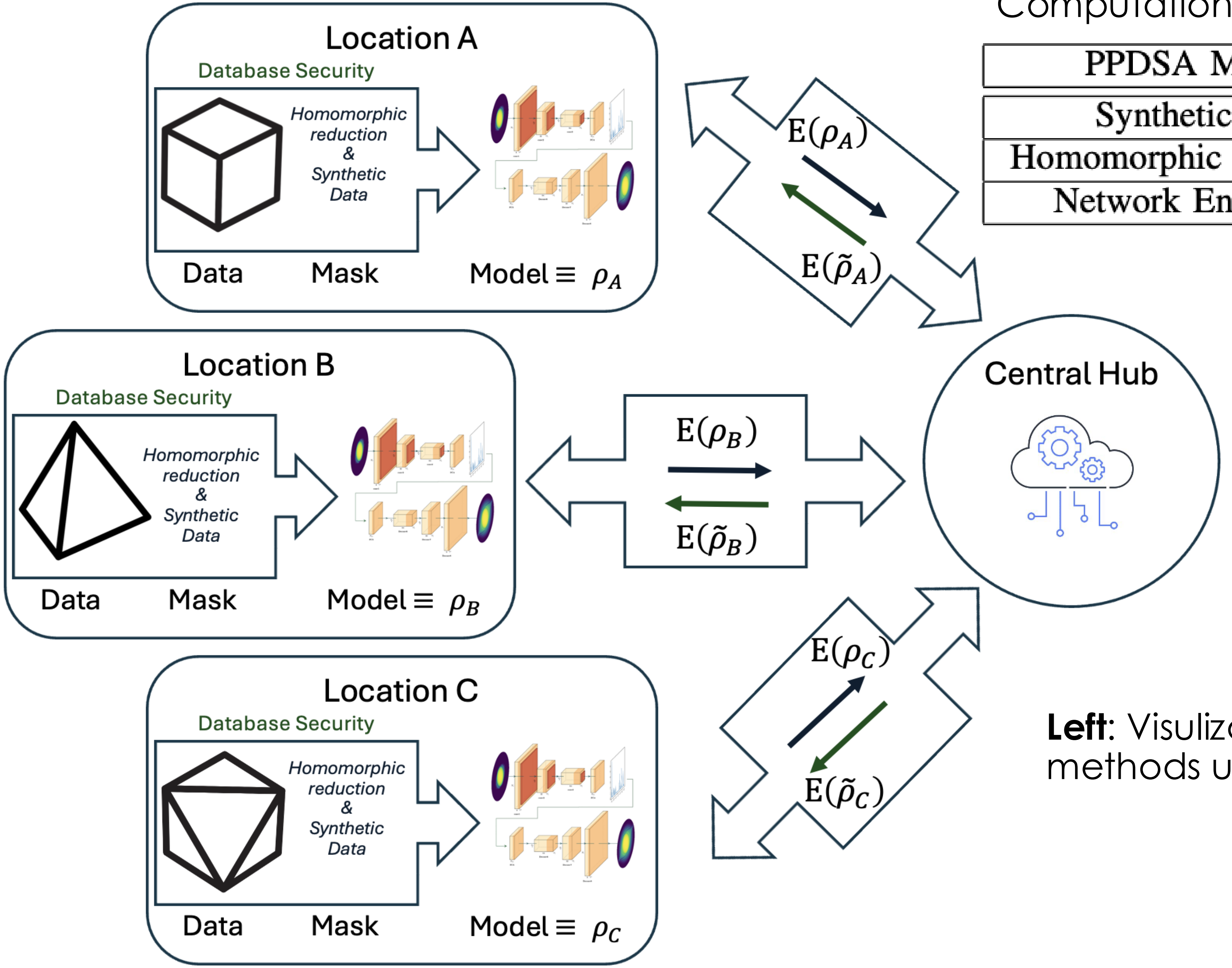




Federated Learning across different facilities and security domains increase the accuracy in reconstruction, model classification, and decompression rate. Using ***SasView*** *we build-in community based physical knowledge.*

OAK RIDGE National Laboratory


ML Data
ML Opt. Single Fac
ML Opt. Fed Learn
ML Opt. Uni. Server
In:cylinder
Predict:cylinder
Predict:cylinder
Predict:cylinder
Out:cylinder
Error 1.37e-03
Error 4.68e-04
Error 4.63e-04
-3.0
-1.5
0.0

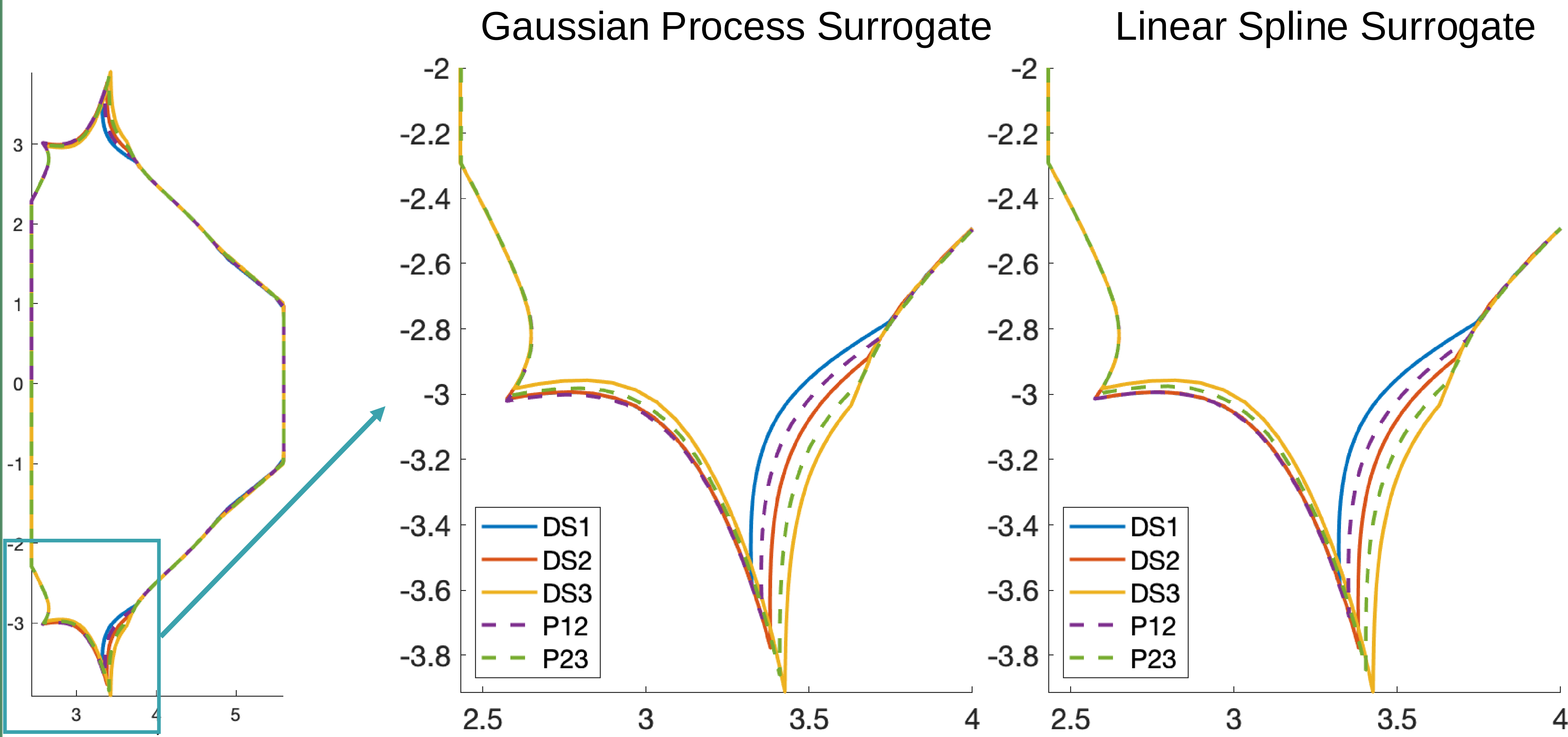


Computational Cost in Training and Prediction

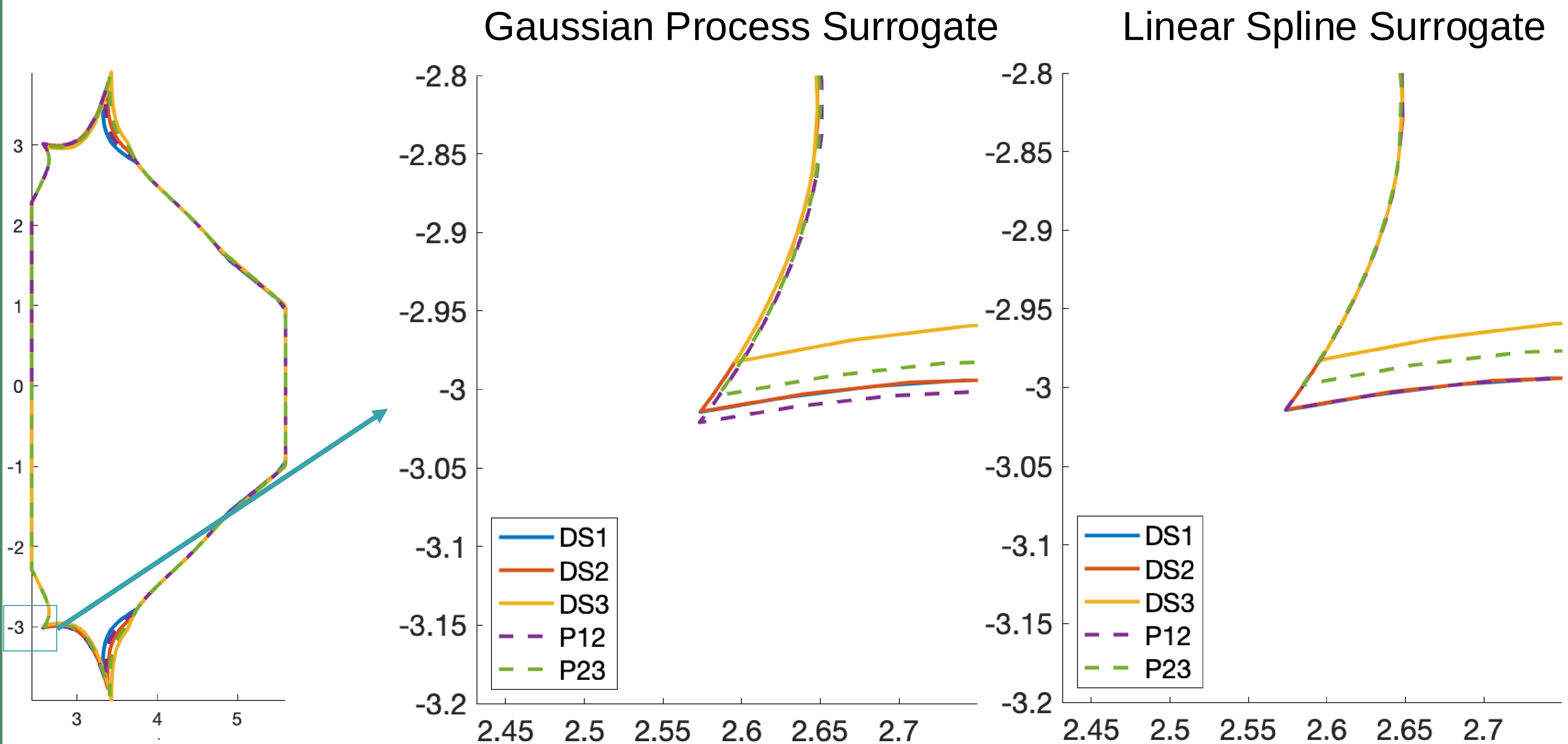
PPDSA Method	Training	Prediction
Synthetic Data	$1.2 \pm 3\%$	–
Homomorphic Computing	$2.8 \pm 0.2\%$	$226 \pm 9\%$
Network Encryption	$1.2 \pm 0.1\%$	–

Left: Visulization of the privacy preservation methods used in our study

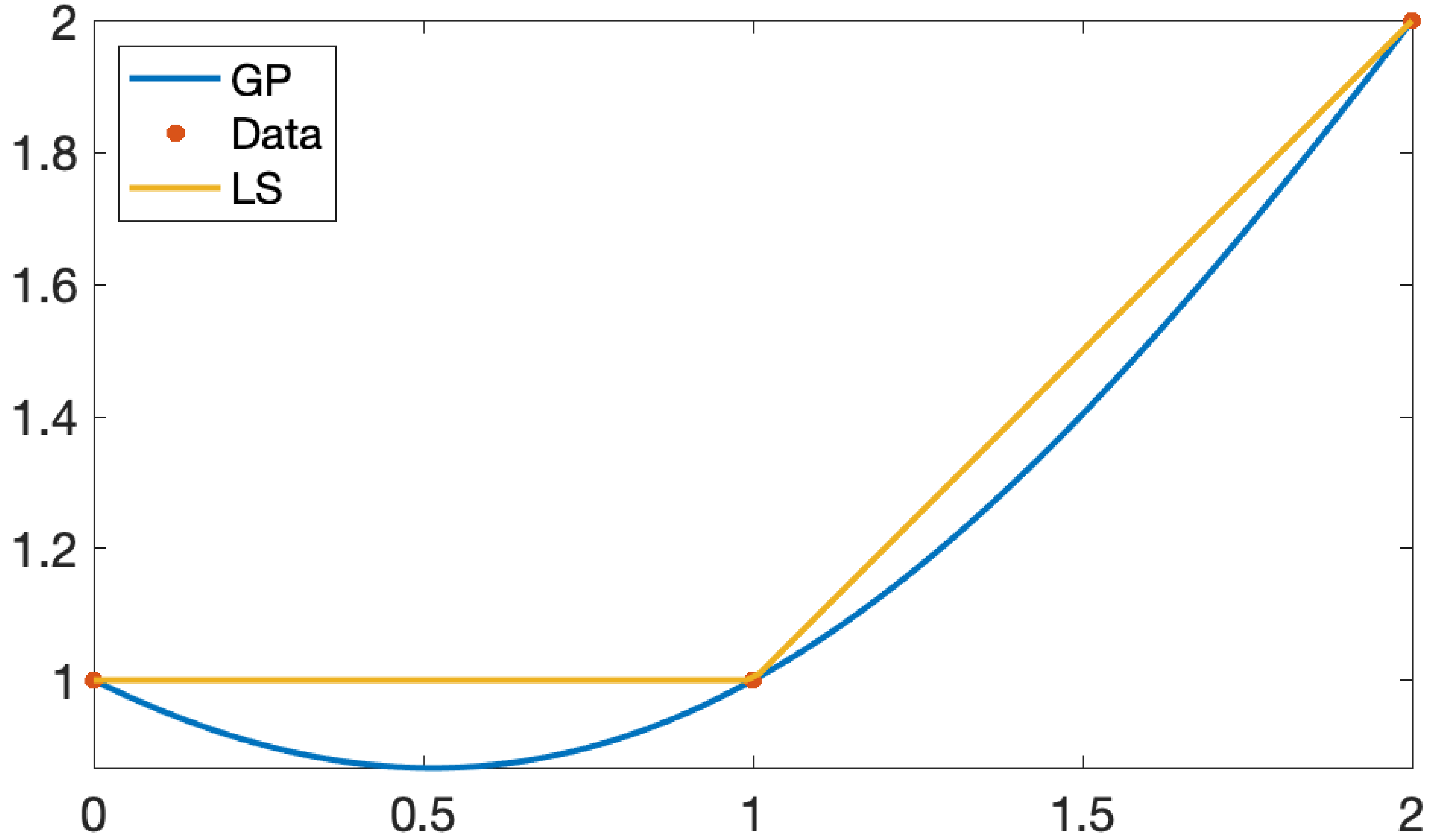
ML Grid Generation Fusion REactor Design and Assessment (FREDA) and Scrape-Off Layer Plasma Simulation(SOLPS)



ML Grid Generation Fusion REactor Design and Assessment (FREDA) and Scrape-Off Layer Plasma Simulation(SOLPS)



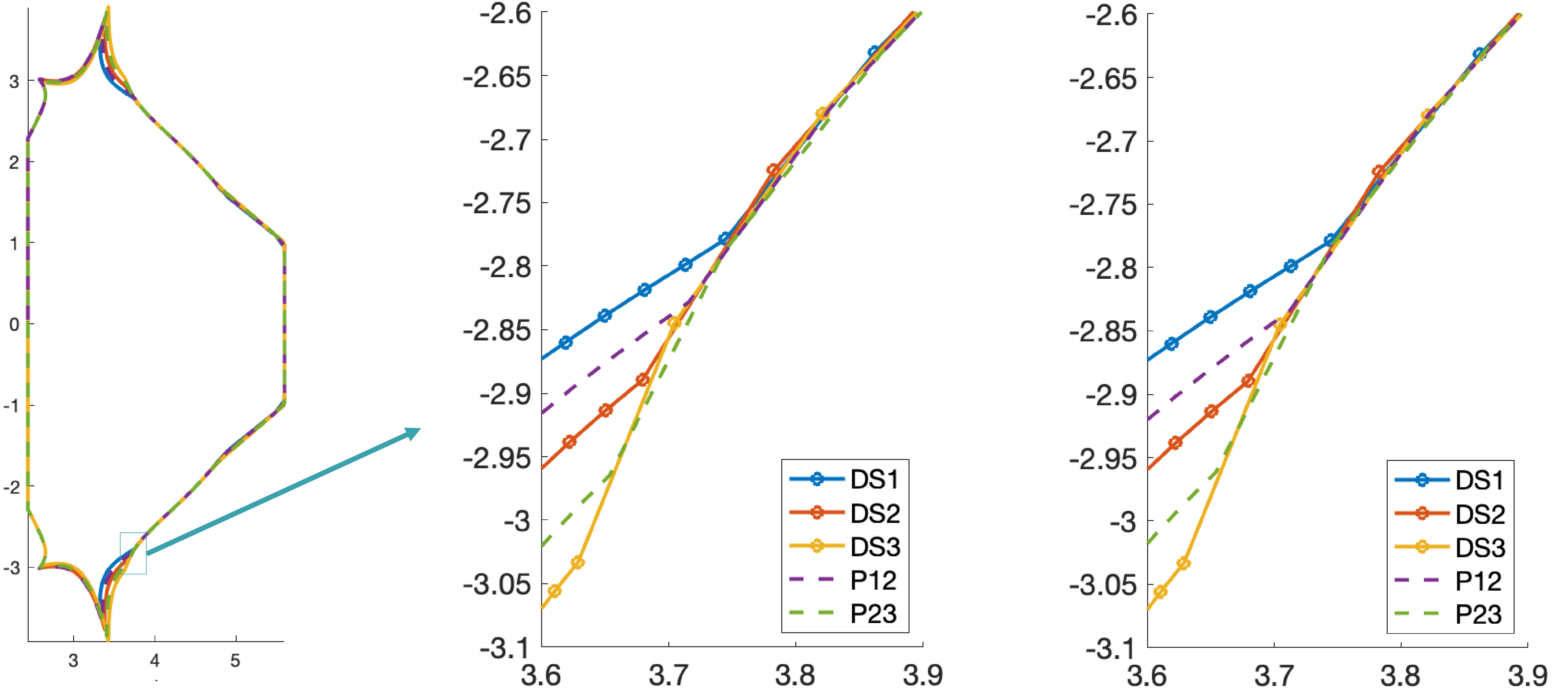
Gaussian Process and Linear Spline Low Resolution



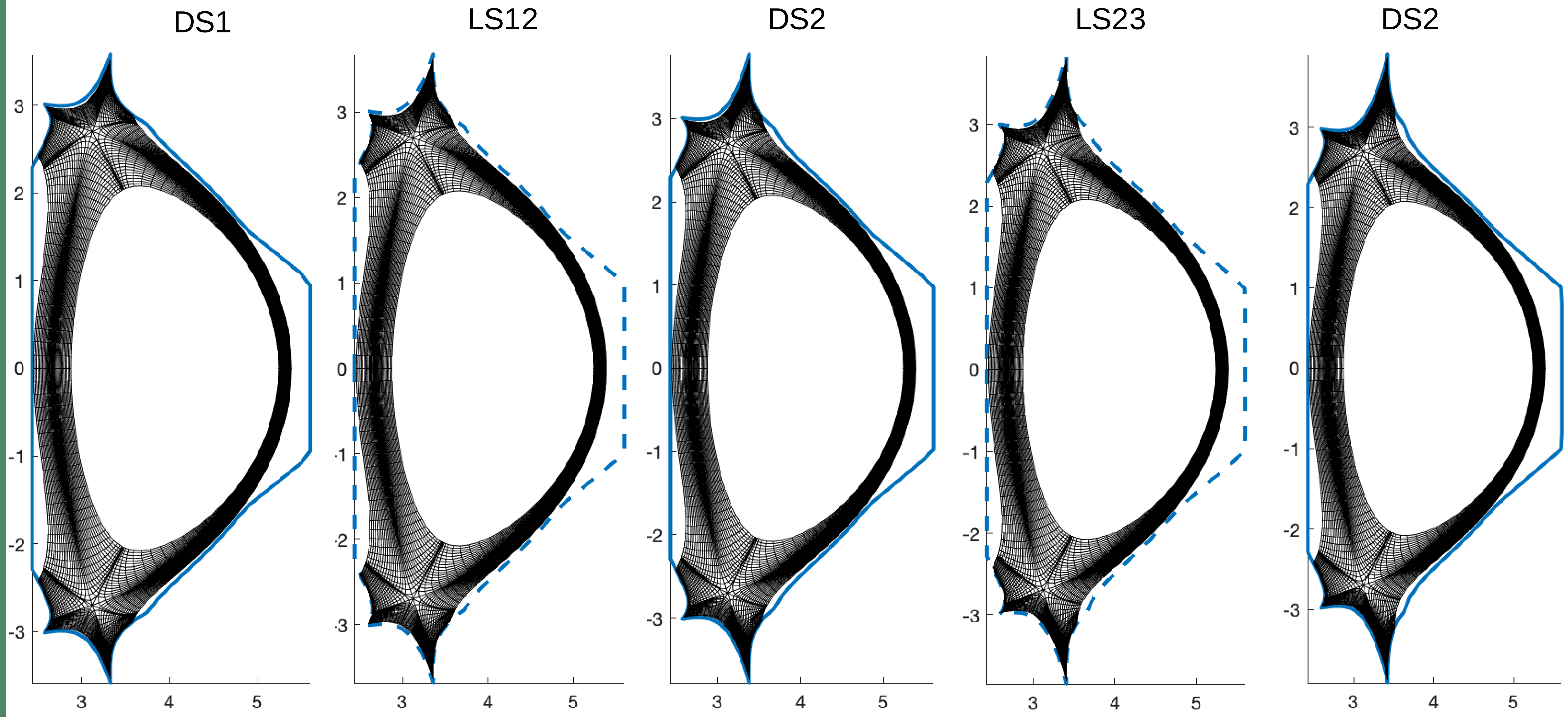
ML Grid Generation Fusion REactor Design and Assessment (FREDA) and Scrape-Off Layer Plasma Simulation(SOLPS)

Gaussian Process Surrogate

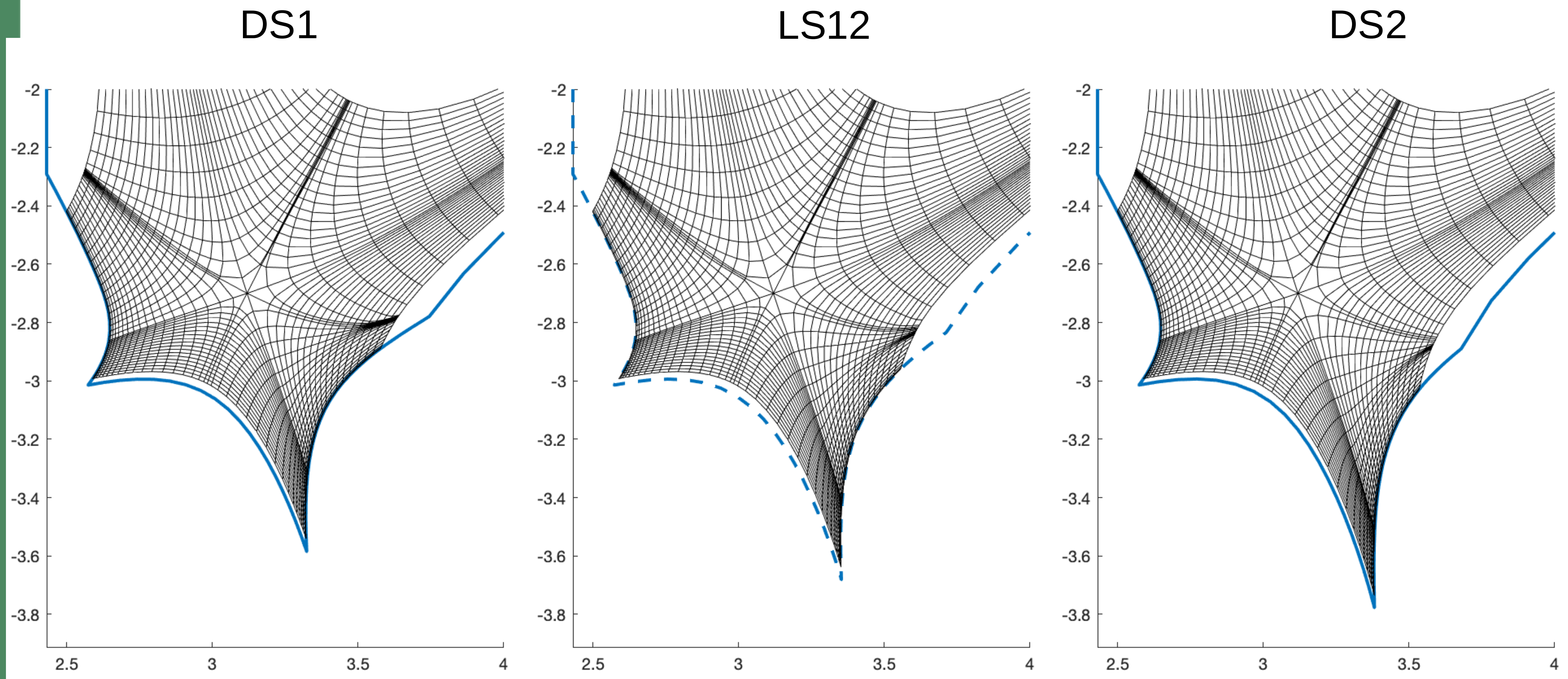
Linear Spline Surrogate



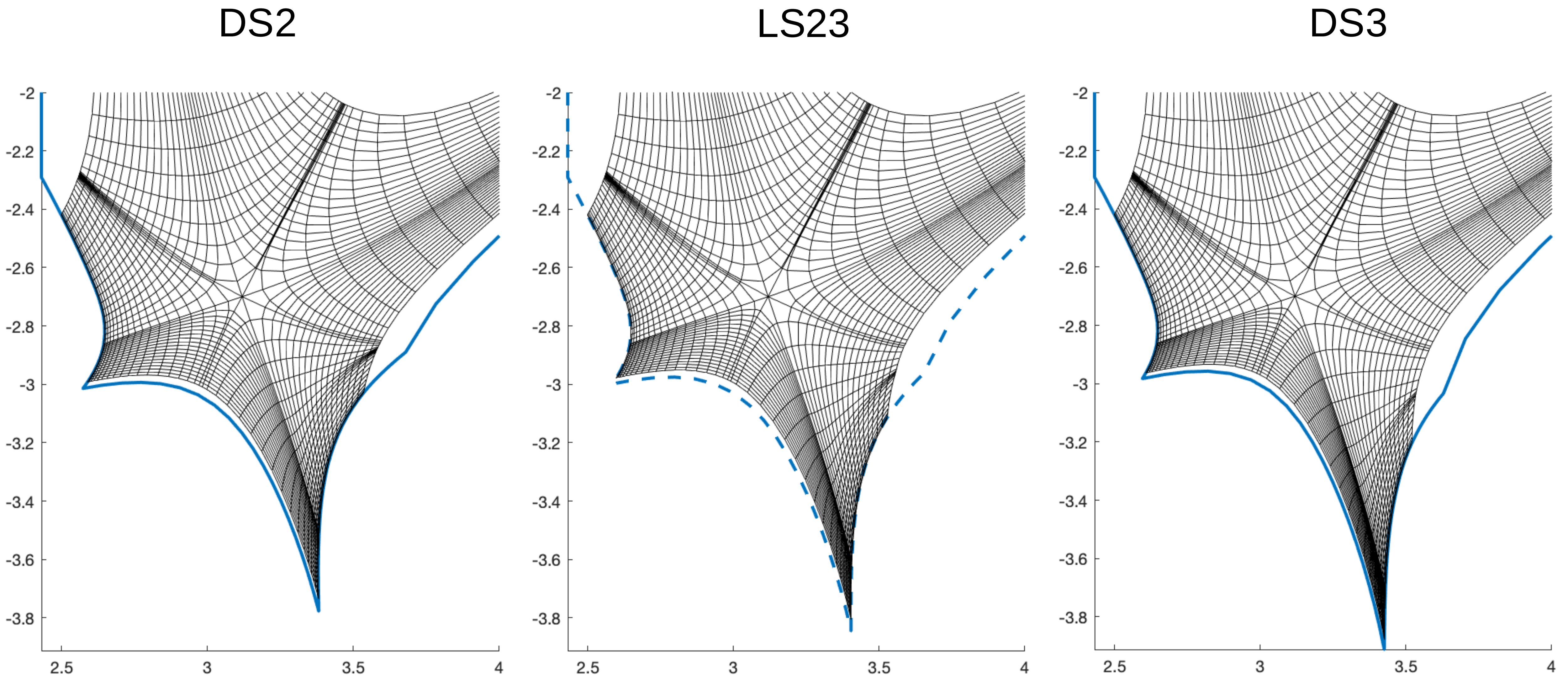
ML Grid Generation Fusion REactor Design and Assessment (FREDA) and Scrape-Off Layer Plasma Simulation(SOLPS)



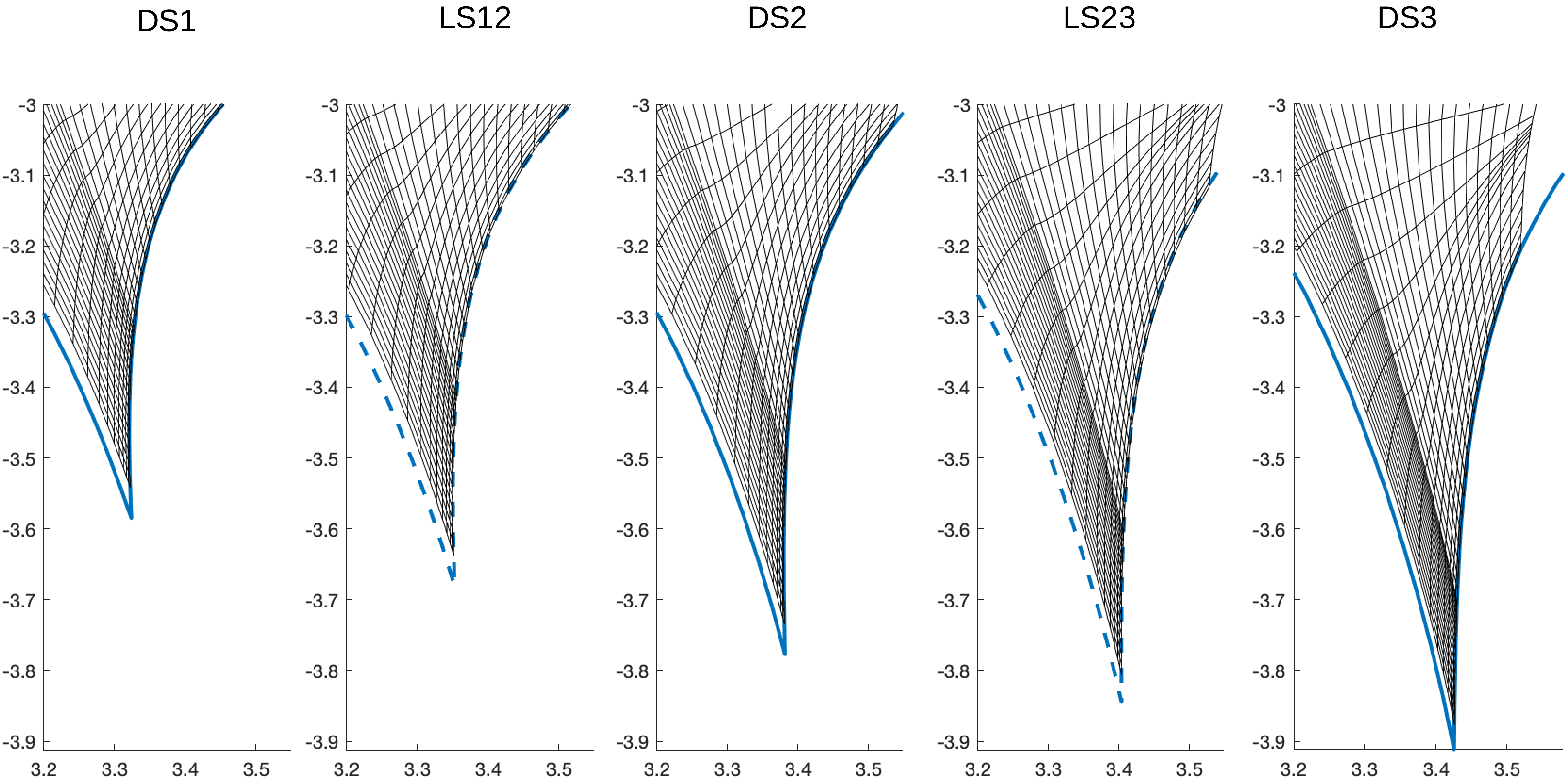
ML Grid Generation



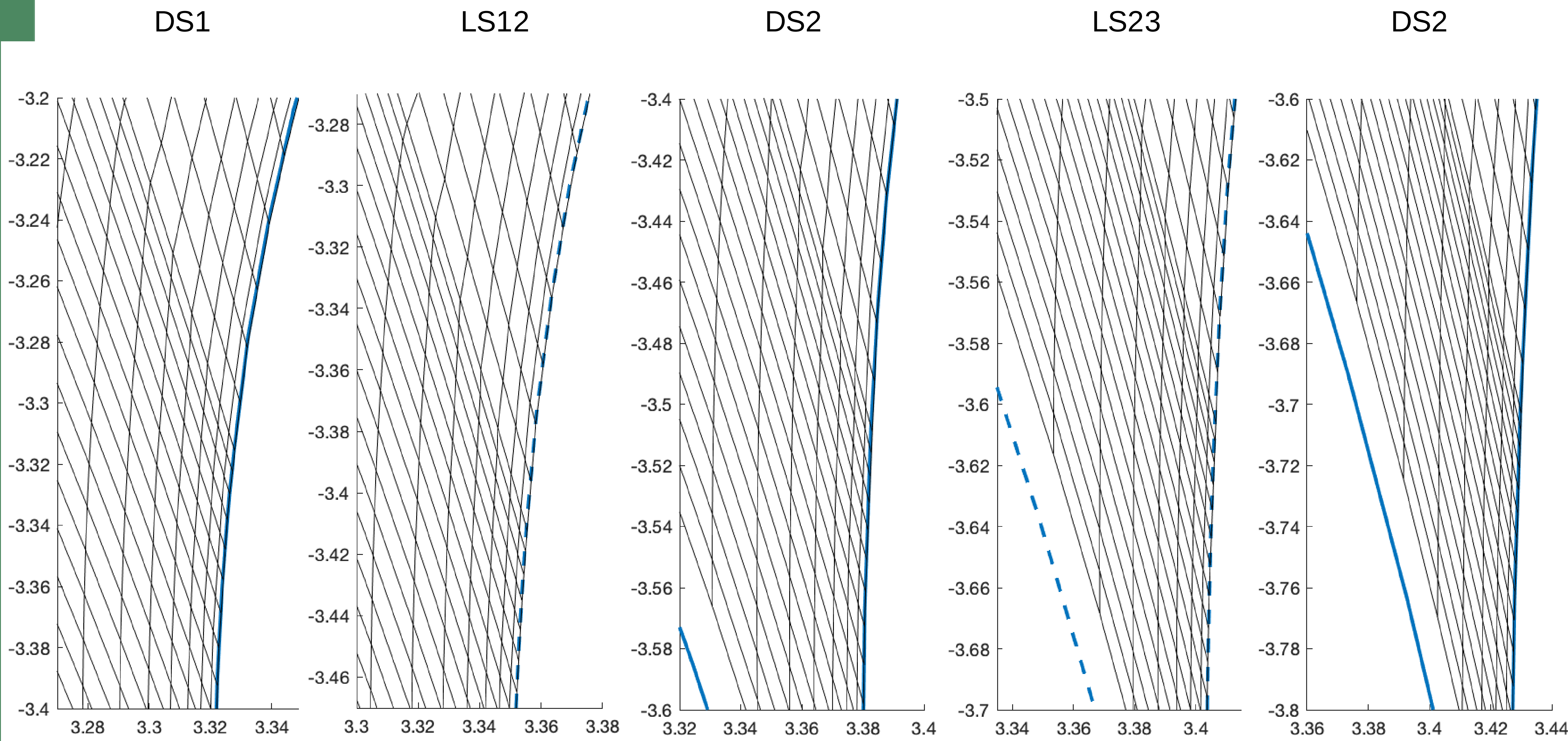
ML Grid Generation



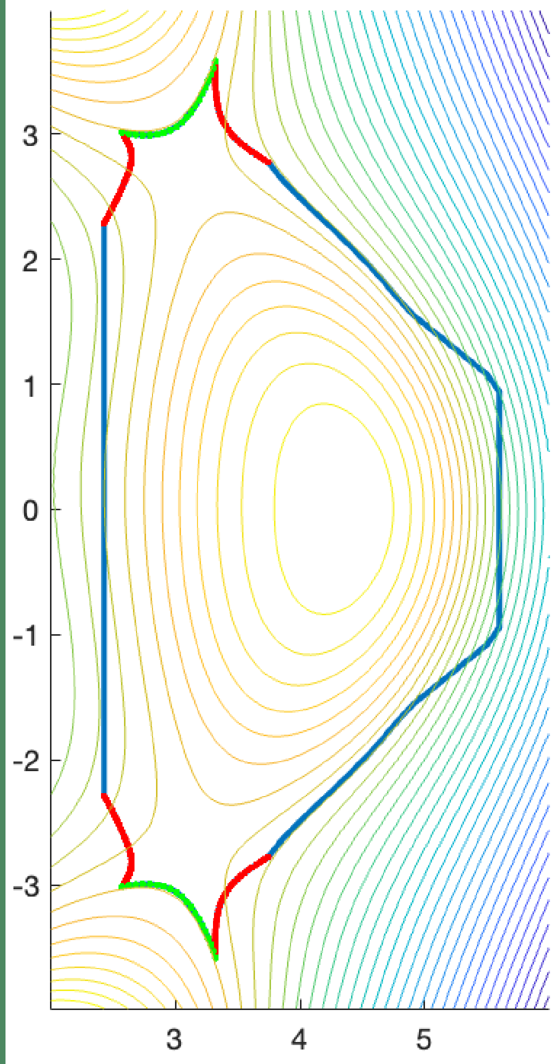
ML Grid Generation



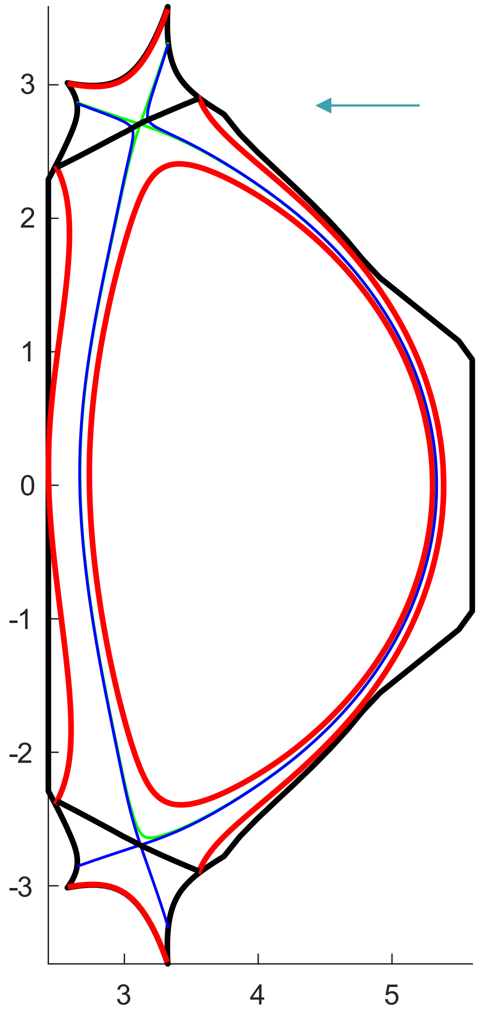
ML Grid Generation



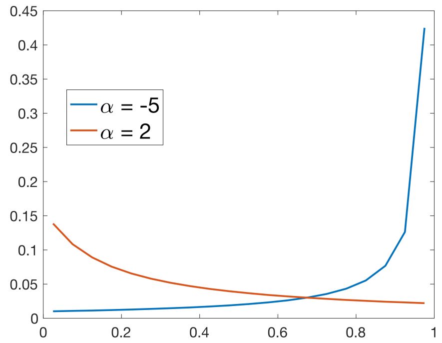
Functional Representation of Grids – Initialize

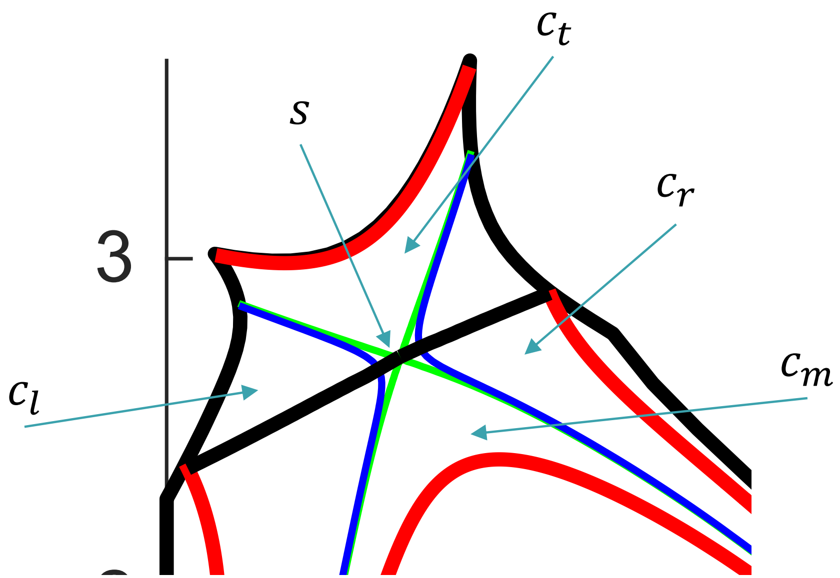


Given boundary geometry, psi field, and diverter plate locations



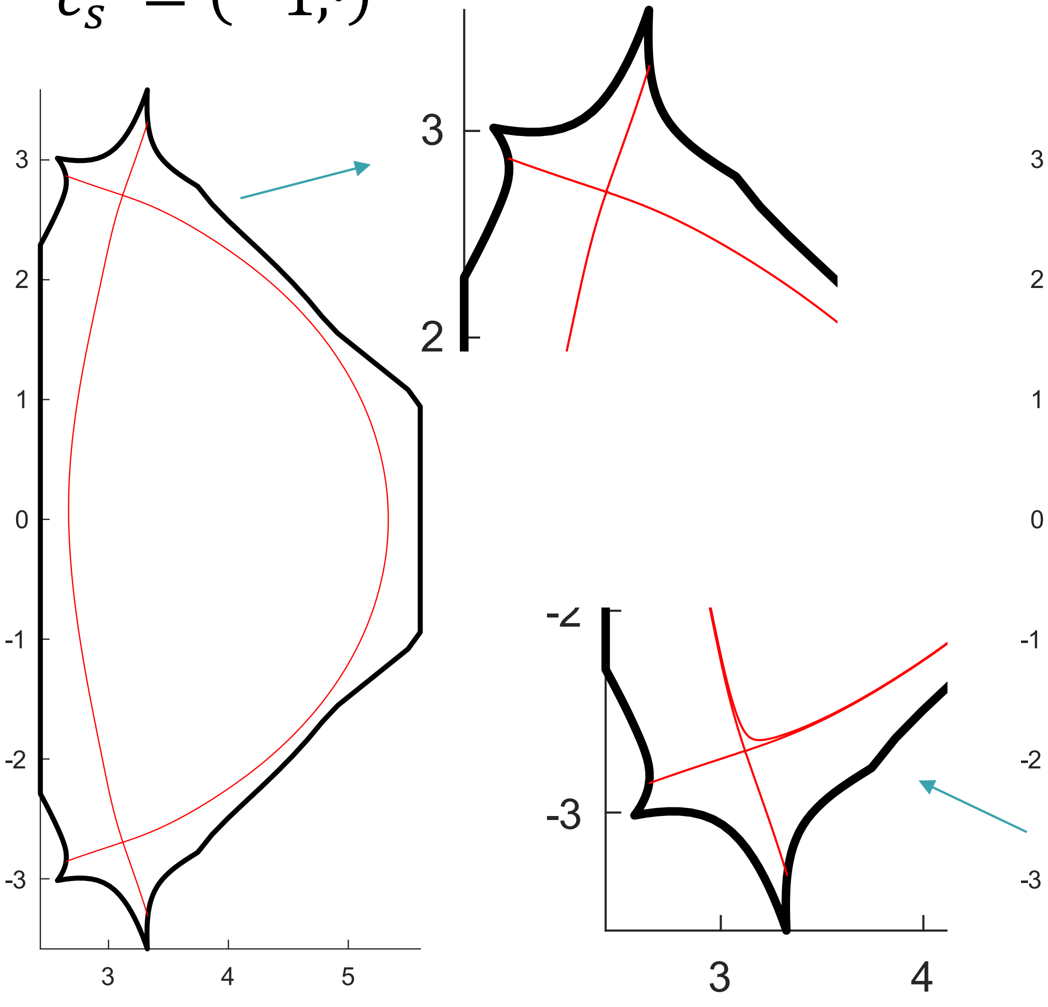
Initialize grid structure and determine ranges for variables $\{c_l, c_r, c_t, c_b, c_m, \tilde{c}_s\}$, where $c. = (c, N, \alpha)$ and $\tilde{c}_s = (N, \alpha)$. PDF is $f(x) = \frac{e^{\alpha x}-1}{e^{\alpha}-1}$



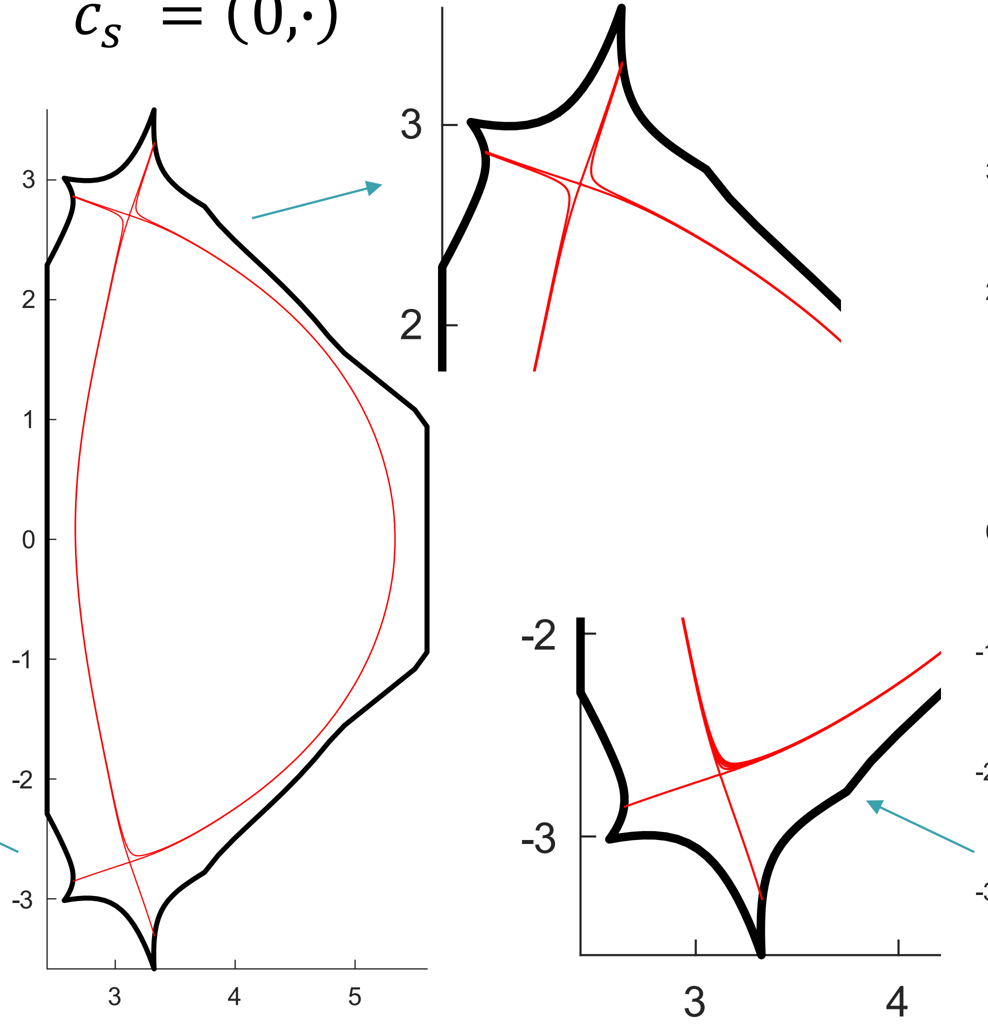


Functional Representation of Grids – $\tilde{c}_s$

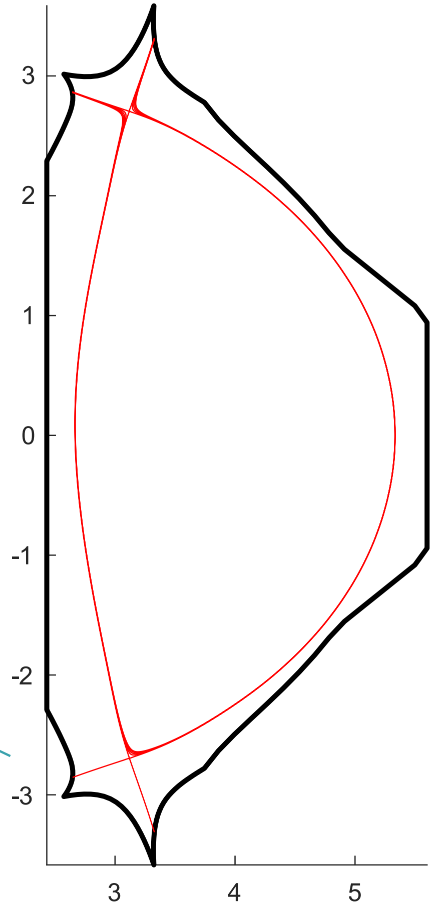
$\tilde{c}_s = (-1,\cdot)$



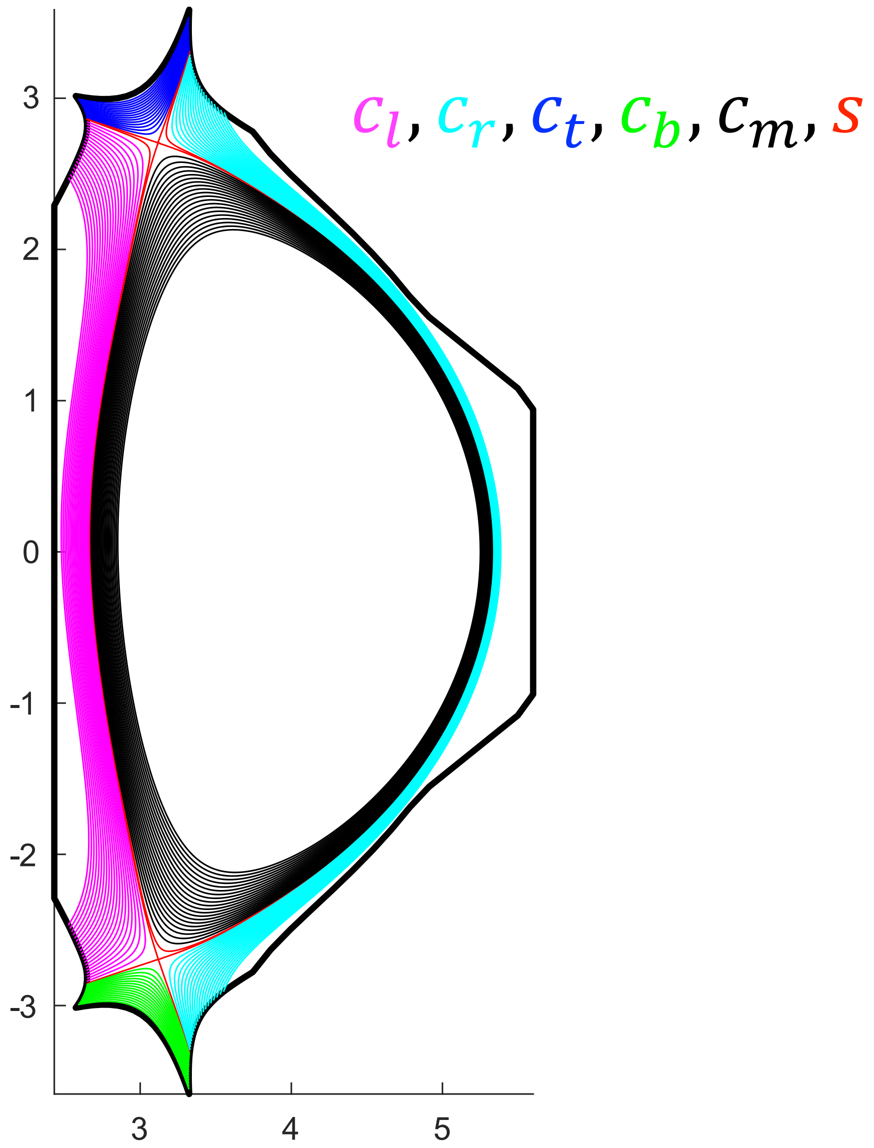
$\tilde{c}_s = (0,\cdot)$

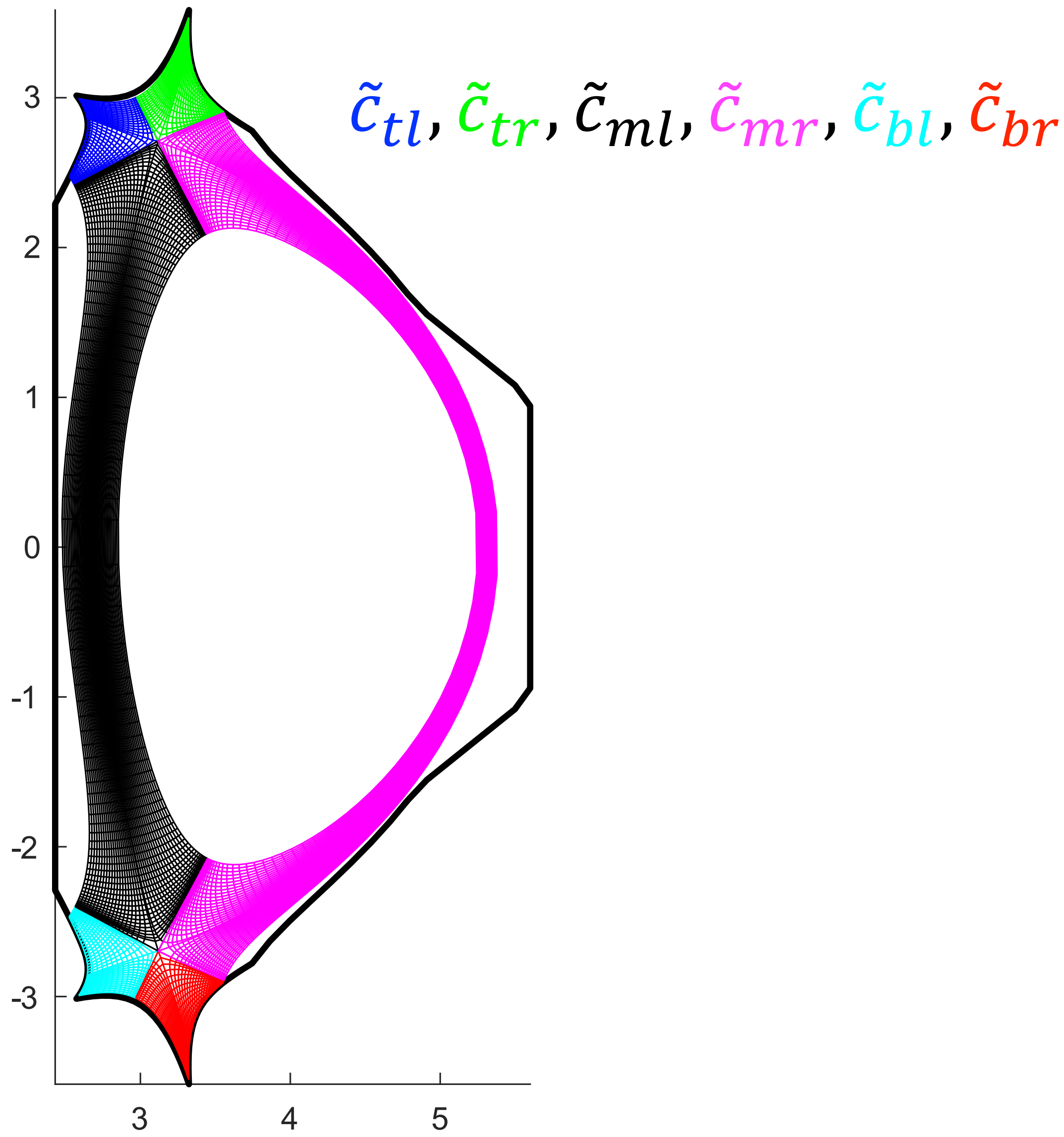


$\tilde{c}_s = (3,0)$



Functional Representation of Grids





Thanks!
Questions???

Acknowledgments
DOE Funding from ASCR & BES
ORNL Funding from LDRD