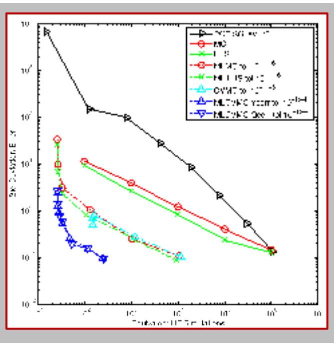
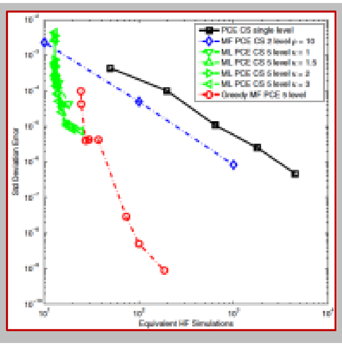
SAND2021-2241C

Exceptional service in the national interest

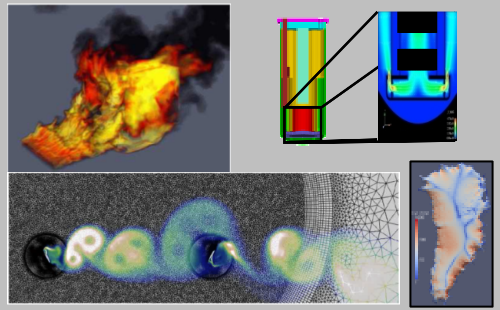
Sandia National Laboratories





DAKOTA

Explore and predict with confidence



Recent Advances in Adaptive Refinement of (Regression-Based) Multifidelity Surrogates for UQ

Michael S. Eldred[1], Alex Gorodetsky[2], Gianluca Geraci[1], John Jakeman[1], Teresa Portone[1]

[1]*Sandia National Laboratories, Albuquerque NM*

[2]*University of Michigan, Ann Arbor*

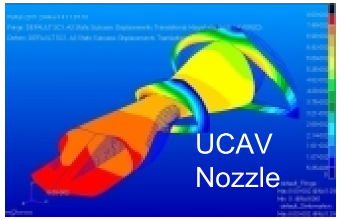
U.S. DEPARTMENT OF ENERGY

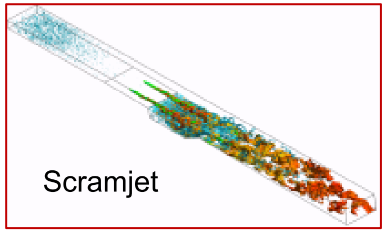
NNSA National Nuclear Security Administration

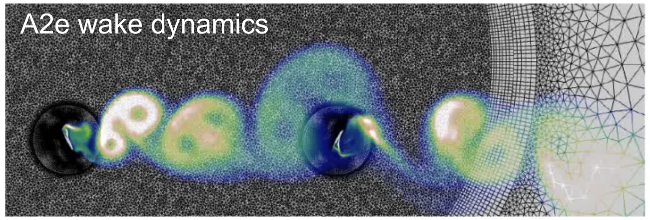
Sandia National Laboratories is a multi-mission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA-0003525.

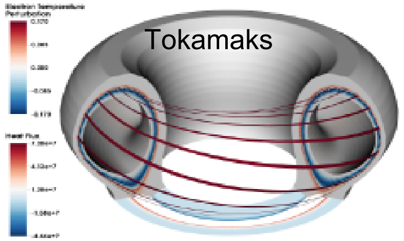
Multifidelity Methods: Sampling UQ, Surrogate UQ, OUU

2018/2019:

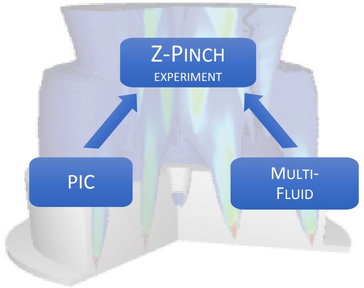


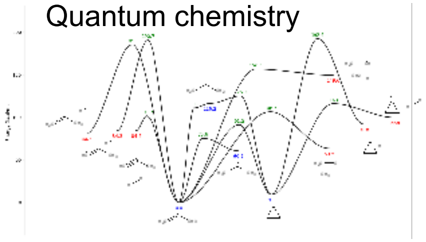


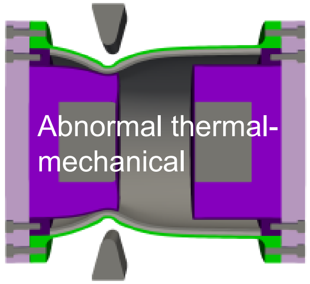


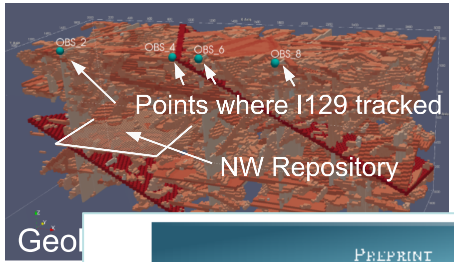


2020/2021:



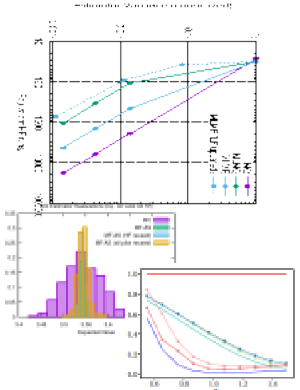






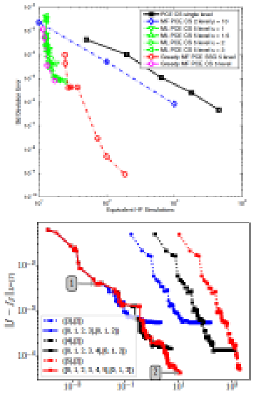
Monte Carlo UQ Methods

- *Production:* optimal resource allocation for multilevel, multifidelity, combined (DARPA EQUiPS x 2, Wind, Cardiovascular)
- *Emerging:* active dimensions ('18 EE LDRD), generalized fmwk for approx control variates (ASC V&V Methods)
- *On the horizon:* control of time avg; full-featured MF UQ (prob. estimation for rare events); model tuning / selection (CIS LDRD, DOE BES)



Surrogate UQ Methods (PCE, SC)

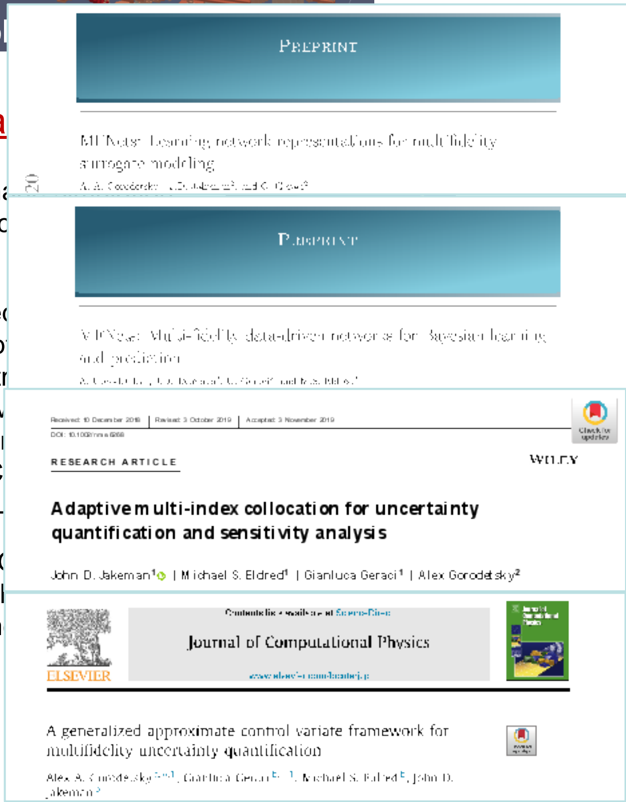
- *Production (v6.10):* ML PCE w/ projection & regression; ML SC w/ nodal/hierarchical interp; greedy ML adaptation (DARPA SEQUOIA)
- *Emerging:* multi-index stochastic collocation; multilevel function train; multiphysics/multiscale integration (ASC V&V Methods)
- *On the horizon:* new surrogates (GP, ROM, deep NN) with error mgmt ('19 EE LDRD, DOE BES); learning latent var relationships, unification of surrogate + sampling approaches (CIS LDRD)



Optimiza

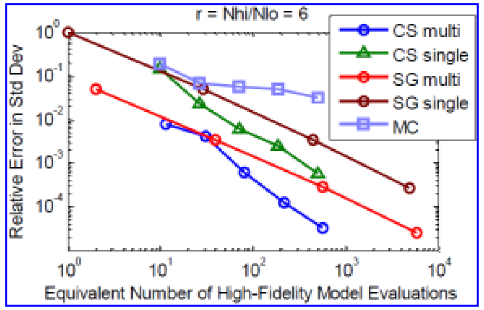
- *Production:* mana and/or stochastic
- *Emerging:*
  Derivative-based
  - Multigrid op
  - Recursive tr extend TRM

  Derivative-free
  - SNOWPAC oriented ML
- *On the horizon:* approaches: mul Optimal experim



Non-hierarchical methods, alternative surrogates, goal-orientation, subspace methods for dissimilar models

Formulations for Multilevel Surrogates: PCE / SC / FTT



Starting point (2012): prescribed ML/MF resolutions w/ adaptivity

$$\hat{f_{hi}}(\xi) = \sum_{j=1}^{N_{lo}} f_{lo}(\xi_j) L_j(\xi) + \sum_{j=1}^{N_{hi}} \Delta f(\xi_j) L_j(\xi)$$

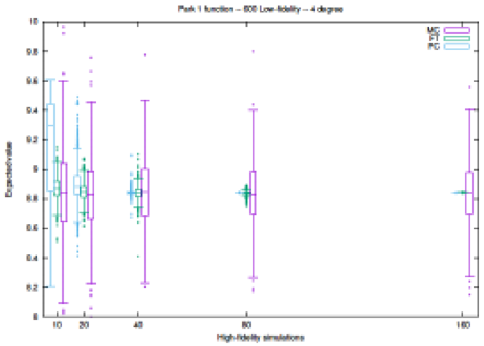
$$N_{lo} >> N_{hi}$$

Rate estimation

1. Optimal resource allocation: parameterize estimator variance → optimal N_l

   Global κ and $\gamma > 0$

$$Var[\hat{Y}_l] = \frac{Var[Y_l]}{\gamma N^{\kappa}} \quad \rightarrow \quad N_l = \sqrt[\kappa]{\frac{2}{\epsilon^2 \gamma} \sum_{q=0}^{L} \sqrt[\kappa+1]{Var[Y_q] C_q^{\kappa}}} \sqrt[\kappa+1]{\frac{Var[Y_l]}{C_l}}$$



E., G. Geraci, J.D. Jakeman, "Multilevel Monte Carlo Hybrids Exploiting Multidelity Modeling and Sparse Polynomial Chaos Estimation," SIAM UQ 2016, Lausanne.

Main challenge: abrupt transitions in sparse / low rank recovery

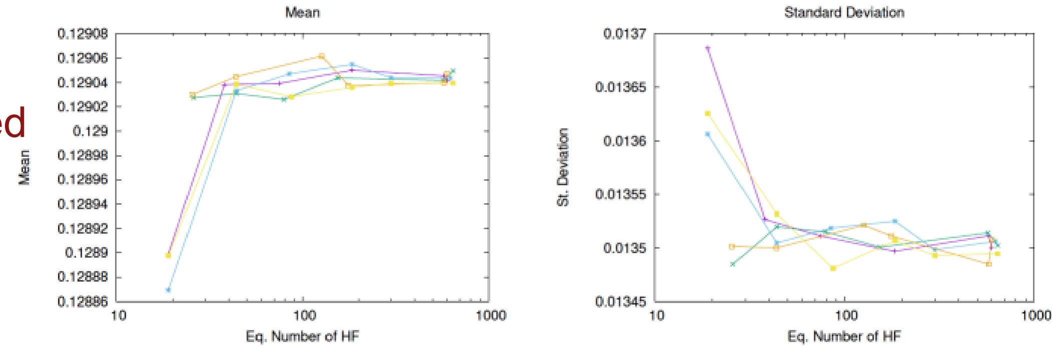
Recovery theory

2. Iterative sample resolution based on observed sparsity (CS: RIP), rank (FTT), et al. (OLS: BLUE)

$$N_l \geq s_l \, log^3(s_l) \; L_l \, log(C_l)$$

Jakeman, Narayan, and Zhou, 2016

$N_{low} = 600$, degree=4



Challenge (CS): compressible fns → s_l feedback not well controlled

$$N_l \sim O(p\, r^2\, d) \text{ per level}$$

Need (FTT): embed within (greedy) adaptive refinement scheme

Greedy

3. Greedy Multilevel refinement

   ML competition with multiple level candidate generators

   Main challenges (projection): scalable refinement schemes, preservation of precision in fine-grained increments

   Main challenges (regression): mutual coherence, over-fitting

Multilevel / Multi-index PCE / HSC: greedy competition across models

Model problem results

Steady state diffusion

$$-\frac{d}{dx}\left[a(x,\xi)\frac{du}{dx}(x,\xi)\right] = 10, \quad (x,\xi) \in (0,1) \times I_{\xi}$$

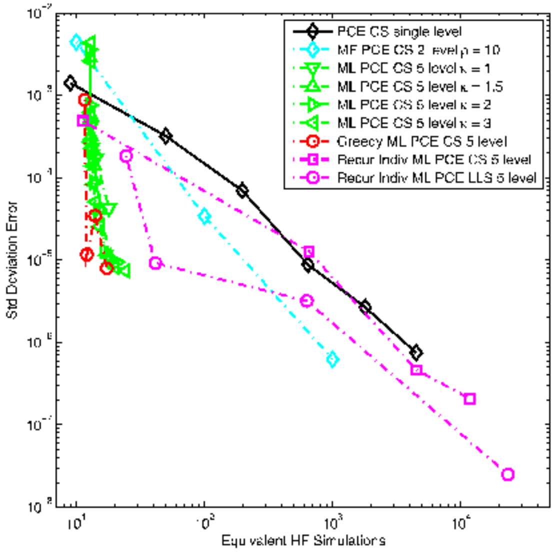
$$u(0,\xi) = 0, \quad u(1,\xi) = 0.$$

Advection diffusion

$$\frac{du}{dt}(x_1,t,Z) + a\frac{du}{dx}(x_1,t,Z) - \frac{d}{dx}\left[k(x_1,Z)\frac{du}{dx}(x_1,t,Z)\right] = g(x_1,t,Z) \qquad (x_1,Z) \in (0,1)\times\Gamma$$

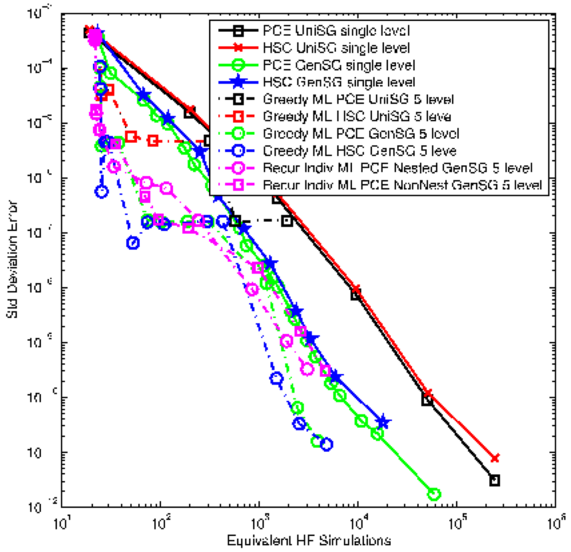
$$u(0,t,Z) = 0 \quad u(1,t,Z) = 0 \quad u(x_1,0,Z) = 0,$$

Greedy ML PCE: compr. sensing, OLS
- Uniform refinement candidates
- Individual / integrated competition
- Distinct / recursive discrepancy



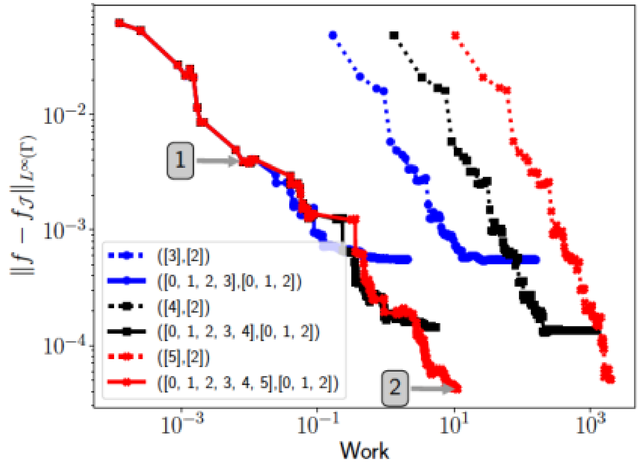
Conv Tol	N_1	N_2	N_3	N_4	N_5
1.e-1	198	9	9	9	9
1.e-2	644	198	9	9	9
1.e-3	1802	644	9	9	9
1.e-4	4505	1802	50	9	9

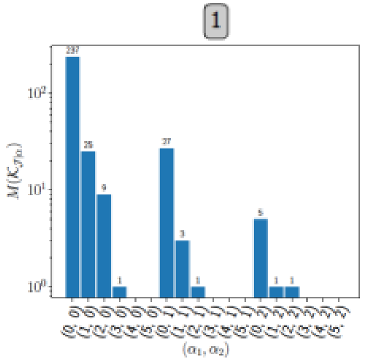
Greedy ML PCE / HSC: sparse grids
- Uniform / generalized candidates
- Individual / integrated competition
- Distinct / recursive discrepancy

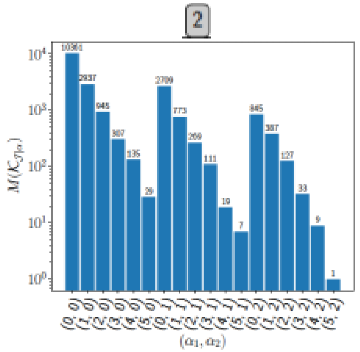


Conv Tol	N_1	N_2	N_3	N_4	N_5
1.e-2	43	23	19	19	19
1.e-4	211	83	19	19	19
1.e-6	391	271	156	19	19
1.e-8	1359	743	327	59	19
1.e-10	3535	2311	1039	391	19
1.e-12	10319	5783	2783	1343	43
1.e-14	26655	14991	8063	3703	1535

Greedy multi-index PCE: sparse grids with generalized refinement







Jakeman, E., Geraci, Gorodetsky, "Adaptive Multi-index Collocation for Uncertainty Quantification and Sensitivity Analysis," IJNME, Aug. 2020.

Surrogate approaches: Greedy multilevel refinement

$$\hat{Q}_L = \hat{Q}_0 + \sum_{l=1}^{L} \hat{\Delta}_l \quad \text{for} \quad \Delta_l \stackrel{\text{def}}{=} Q_l - Q_{l-1}$$

Compete refinement candidates across model levels: max induced change / cost

- 1 or more refinement candidates per model level within an integrated pool
- Score: relative/absolute impact on level/combined QoI statistics (default: relative + combined roll-up)
  - norm of change in response covariance (default) or level mappings (goal-oriented: $z/p/\beta/\beta^*$)
  - normalize by cost of candidate ($\Delta N_l\ C_l$)
- Greedy selection of best candidate, which then generates new candidates at selected model level

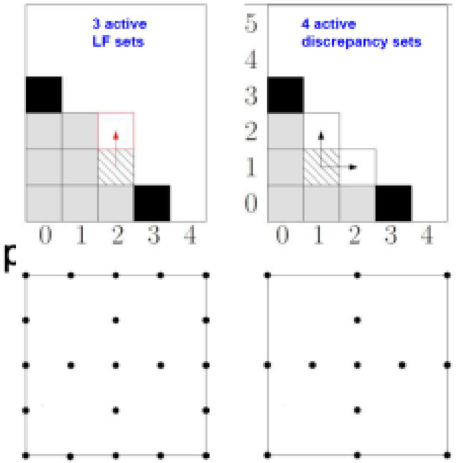
Level candidate generators:

- *Uniform refinement:* 1 exp order / grid level candidate per model level
  - Tensor / sparse grids: projection PCE, nodal/hierarchical SC
  - Regression PCE: least squares / compressed sensing
  - FTT: regression, cross-approximation

Cross-validation, candidate saturation logic

- *Anisotropic refinement:* 1 exp order / grid level candidate per model level
  - Tensor / sparse grids
- *Index-set refinement:* many candidates per level
  - Generalized sparse grids: projection PCE, nodal/hierarch SC
  - Regression PCE
- *Adapted basis frontier:* selectable numberType equation here. of frontier advancements p (avoids scaling issues with index-set refinement)
  - Regression PCE

    (Jakeman, E., Sargsyan, "Enhancing ℓ1-minimization estimates of polynomial chaos expansions using basis selection," *J. Comp. Phys.*, Vol. 289, May 2015.)



Production Usage of ML/MF Surrogate Methods

Extracting performance in practical applications

Regression-based methods are attractive for their flexibility and fault-tolerance, but tight ML/MF convergence is challenging

- Performance
  - *Avoid over-fitting using cross-validation*
    - Candidate basis orders / noise tols in PCE with CS
    - Candidate ranks / basis orders in FTT
  - Manage *discrepancy scale* relative to regression solver / noise tolerances (not an issue with explicit)
- Usability / convenience
  - Relax sample pairing, simplify import / legacy reuse, recover from level corruption ($l < L$)
    - Recursive emulation for discrepancies

} Greedy adaptation incl. saturation (***TO DO***)

Function Train

The rank $r = (1, r_1, \cdots, r_{d-1}, 1)$ function train representation of f is

$$f_r(x) = \sum_{i_0=1}^{r_0} \sum_{i_1=0}^{r_1} \cdots \sum_{i_d=1}^{r_d} f_1^{i_0 i_1}(x_1) f_2^{i_1 i_2}(x_2) \cdots f_d^{i_{d-1} i_d}(x_d)$$

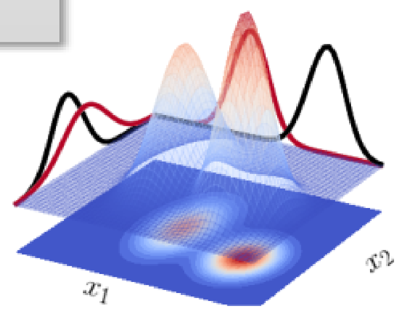
Which is a sum of the tensor product of univariate functions

It is often more intuitive to write the function train as a set of matrix valued products

$$f_r(x) = F_1(x_1) F_2(x_2) \cdots F_d(x_d)$$

Where the so called cores are given by

$$F_k = \begin{bmatrix} f_k^{11}(x_k) & \cdots & f_k^{1 r_k}(x_k) \\ \vdots & \ddots & \vdots \\ f_k^{r_{k-1} 1}(x_k) & \cdots & f_k^{r_{k-1} r_k}(x_k) \end{bmatrix}$$



Marginals are examples of f_k^{ij}

In 3D the rank 2 function train representation of f is

$$f_r(x) = \begin{bmatrix} f_1^{11}(x_1) & f_1^{12}(x_1) \end{bmatrix} \begin{bmatrix} f_2^{11}(x_2) & f_2^{12}(x_2) \\ f_2^{22}(x_2) & f_2^{22}(x_2) \end{bmatrix} \begin{bmatrix} f_3^{11}(x_3) \\ f_3^{21}(x_3) \end{bmatrix}$$

In 4D the rank $r = (1,7,5,3,1)$ function train (FT) representation of f is

$$f_r(x) = \begin{bmatrix} f_1^{11}(x_1) & \cdots & f_1^{17}(x_1) \end{bmatrix} \begin{bmatrix} f_2^{11}(x_2) & \cdots & f_2^{15}(x_2) \\ \vdots & \ddots & \vdots \\ f_2^{71}(x_2) & \cdots & f_2^{75}(x_2) \end{bmatrix} \begin{bmatrix} f_3^{11}(x_3) & \cdots & f_3^{13}(x_3) \\ \vdots & \ddots & \vdots \\ f_3^{51}(x_3) & \cdots & f_3^{53}(x_3) \end{bmatrix} \begin{bmatrix} f_4^{11}(x_4) \\ \vdots \\ f_4^{31}(x_4) \end{bmatrix}$$

Function Train

The rank $r = (1, r_1, \cdots, r_{d-1}, 1)$ function train representation of f is

$$f_r(x) = \sum_{i_0=1}^{r_0} \sum_{i_1=0}^{r_1} \cdots \sum_{i_d=1}^{r_d} f_1^{i_0 i_1}(x_1) f_2^{i_1 i_2}(x_2) \cdots f_d^{i_{d-1} i_d}(x_d)$$

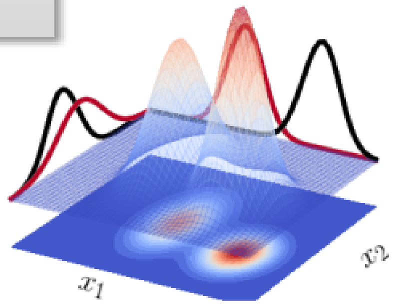
Which is a sum of the tensor product of univariate functions

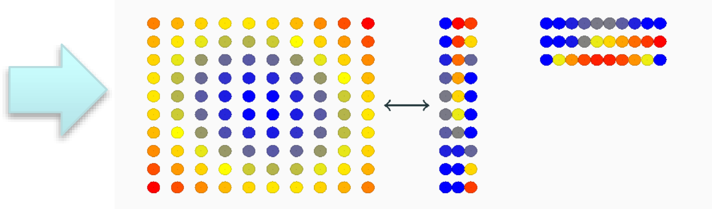
It is often more intuitive to write the function train as a set of matrix valued products

$$f_r(x) = F_1(x_1) F_2(x_2) \cdots F_d(x_d)$$

Where the so called cores are given by

$$F_k = \begin{bmatrix} f_k^{11}(x_k) & \cdots & f_k^{1 r_k}(x_k) \\ \vdots & \ddots & \vdots \\ f_k^{r_{k-1} 1}(x_k) & \cdots & f_k^{r_{k-1} r_k}(x_k) \end{bmatrix}$$





Marginals are examples of f_k^{ij}

Existing literature assume identical basis expansions of each univariate function of a particular dimension, e.g.

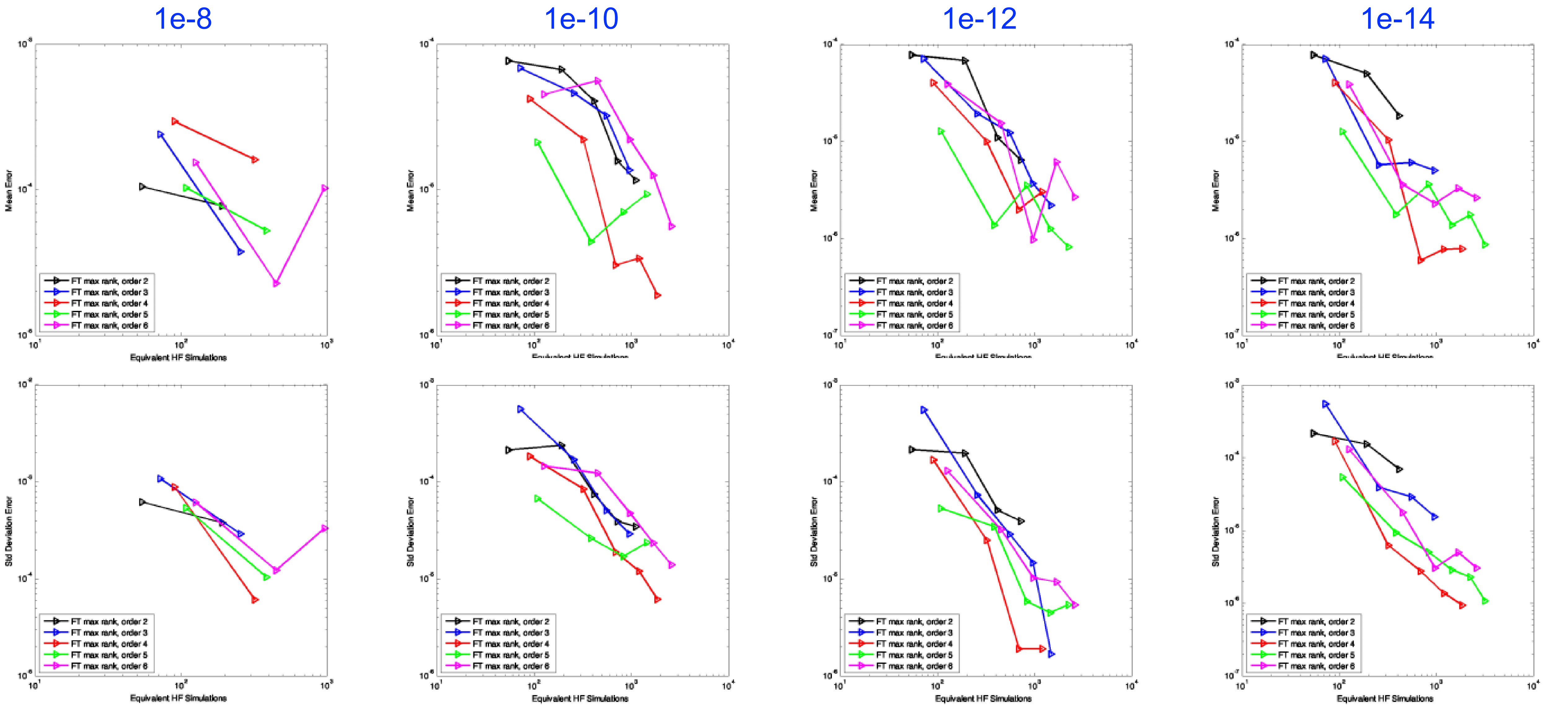
$$f_k^i(x_k) = \sum_{j=0}^{p} \alpha_{kj}^i \phi_{kj}(x_k) = \Phi_k(x_k) \alpha_k^i \text{ (matrix form)}$$

This simplification converts the problem of learning a low rank function to determining the low rank tensor of the coefficients

$$f(x) \approx f_r(x) = \left(\sum_{i=1}^{r} \alpha_1^i \circ \cdots \circ \alpha_d^i \right) \Phi_1(x_1) \otimes \cdots \otimes \phi_d(x_d)$$

Single-fidelity studies: tuning FT hyper-parameters (solver/rounding tolerances)

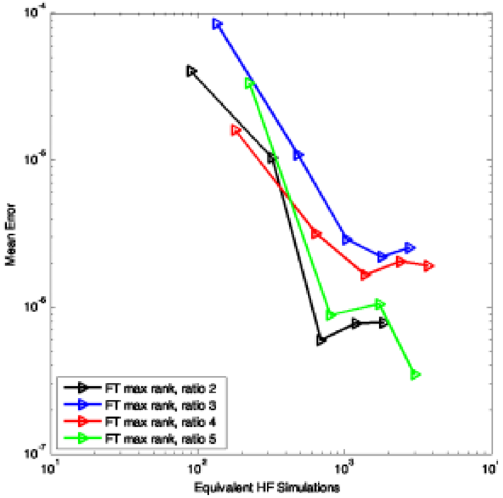
Sandia National Laboratories

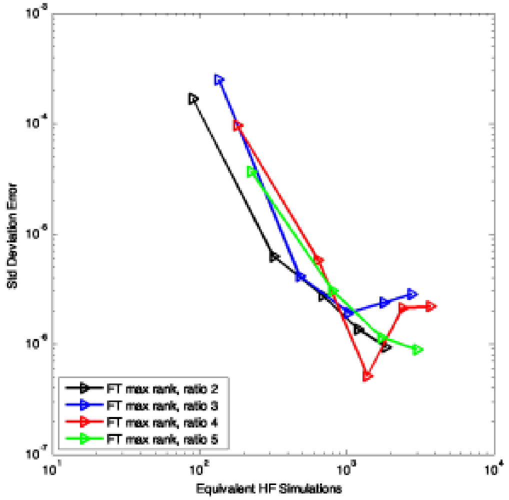


Significant benefit to tighter regression tolerances (w/ solver expense usually subordinate to sample expense in DOE apps.)

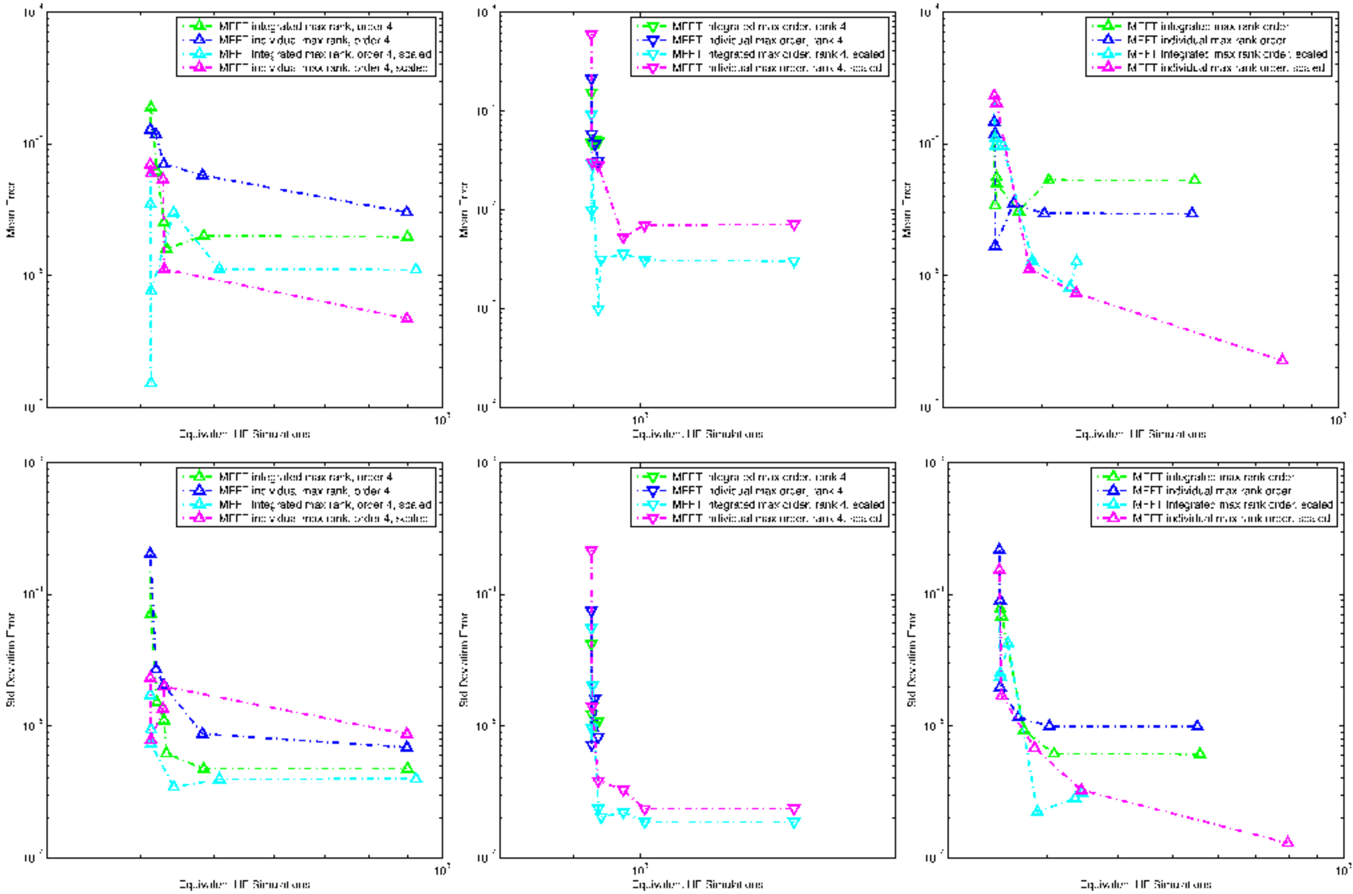
Single-fidelity tuning studies:
FT hyper-parameters (over-sample ratio) & data processing (response scaling)

Effect of over-sampling regression size by 2x, 3x, 4x, 5x
- Little to no benefit for over-sample ratios > 2





Scaling (solver convergence for small discrepancies)
- Can gain ~ an order of magnitude reduction in final error

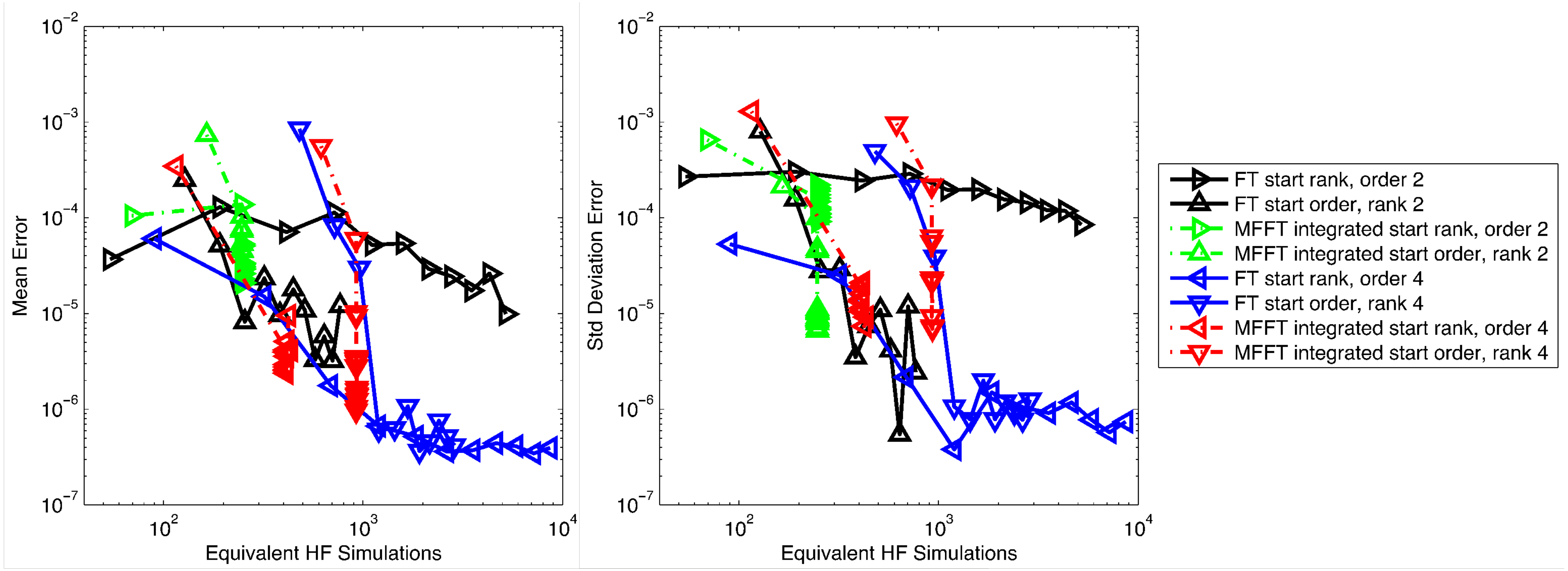


Greedy ML/MF FT regression (tight tols, cr=2, scaling, *NO CROSS-VALIDATION*)

Sandia National Laboratories

Most direct uniform refinement candidates for single- and multifidelity:

- advance rank (basis order fixed at 2 or 4) || advance basis order (rank fixed at 2 or 4)

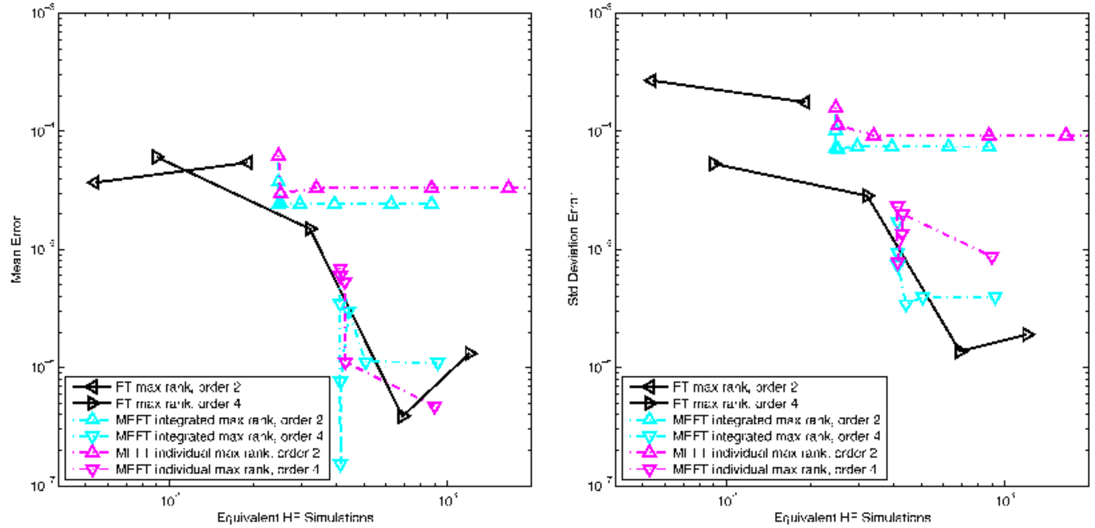


- Despite noise, some basic trends are evident
  - Rank / Order 2 curves (black, green) stall quickly; Rank / Order 4 curves (blue, red) fare better → auto-tuning
  - MFFT rates (green, red) are generally better than FT rates (black, blue)
- Stalls and rebounds indicate issues to try to address → best cases only reduce error ~3 orders of magnitude

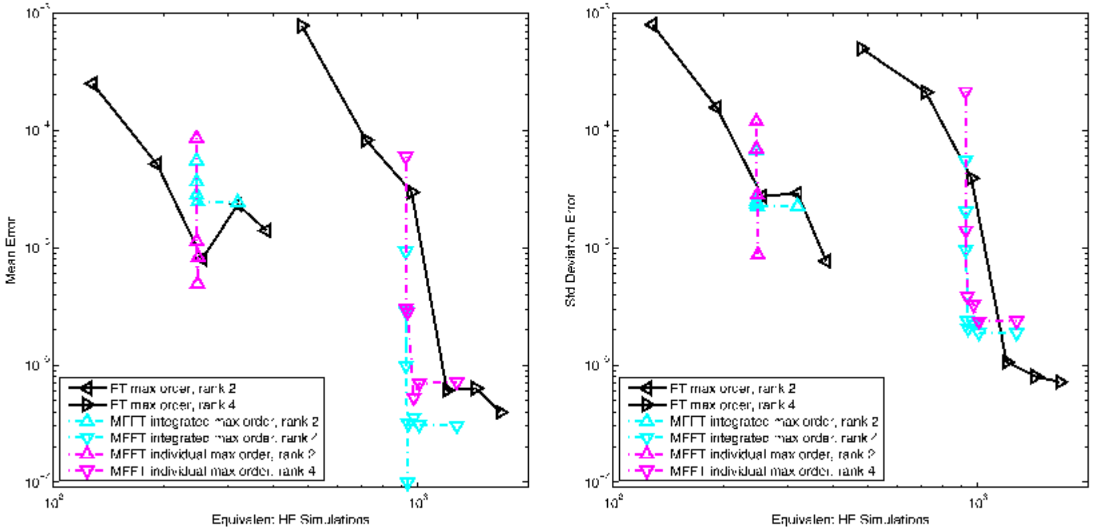
Greedy ML/MF FT regression (*WITH EMBEDDED CROSS-VALIDATION*)

Sandia National Laboratories

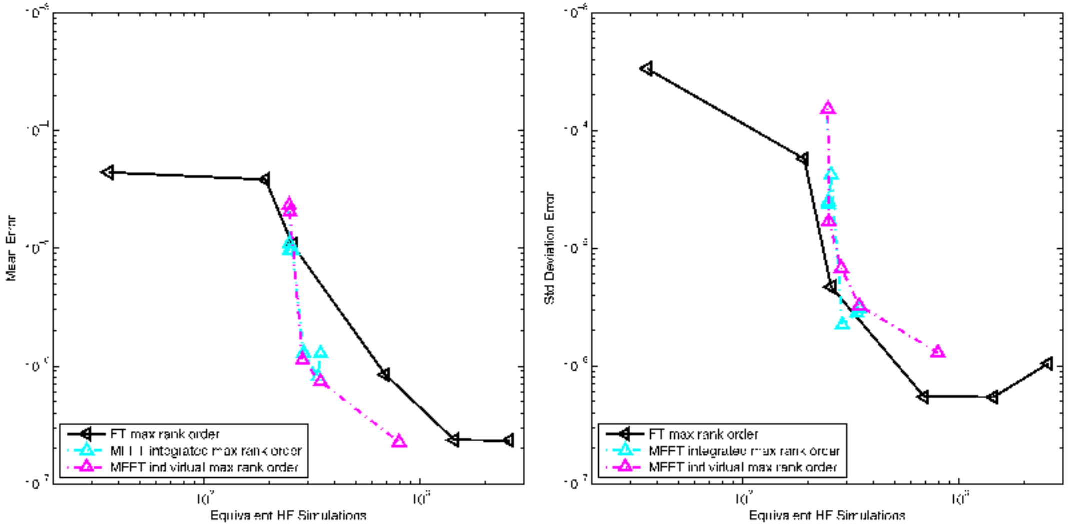
Max rank candidate generation



Max order candidate generation



Max rank & order candidate generation

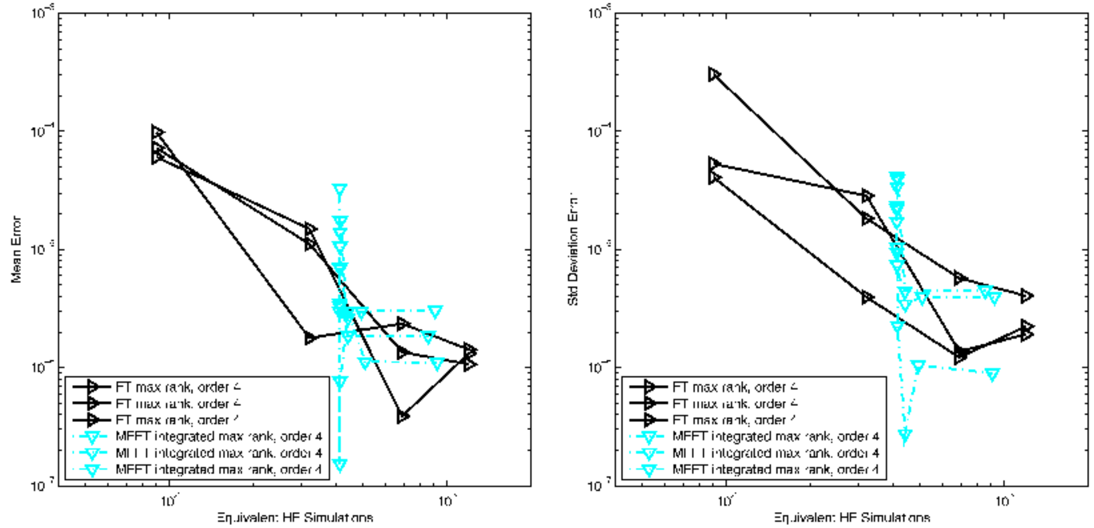


Integrated / indiv. are comparable, sr2 / so2 less reliable → down-select to integrated / sr4 / so4 cases and run ensembles

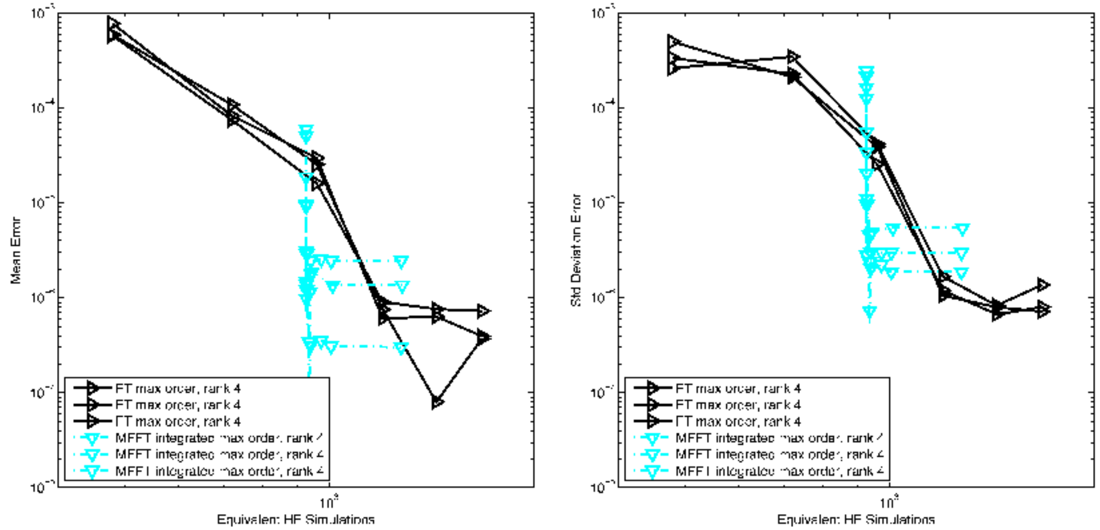
Greedy ML/MF FT regression (*ENSEMBLES FOR DOWN-SELECTED CONFIGS*)

Sandia National Laboratories

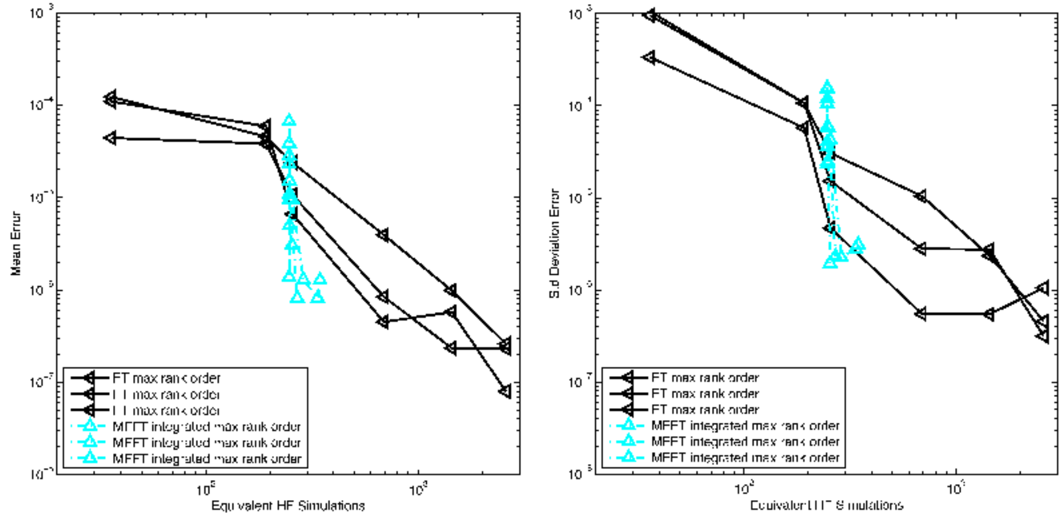
Max rank candidate generation



Max order candidate generation



Max rank & order candidate generation



Superior MFFT convergence rates but capitalization on this remains impeded, stagnations delayed?

Summary Observations

Maintaining accuracy in adaptive ML/MF is challenging in regression contexts (MF PCE with CS,LLS; MF FT regression)

- Overfitting – degrades convergence in statistics, can lead to spurious candidate selections
- Recursive emulation amplifies this by forcing dependent surrogates to accommodate high-order behavior at lower levels

Mitigations

- Tuning: solver/rounding/noise tolerances, over-sample ratio, response scaling for consistent level/fidelity treatment
- Embedded cross-validation

Observations

- Mitigations help with ~1-2 orders of magnitude in error reduction, but believe more should be attainable
- Saturation logic is a double-edged sword – mitigates over-fitting but also caps accuracy per level

Next steps

- All at once (AAO) regression → binding levels together can penalize overfitting at lower levels
- Alternative FTT solution approaches ?
- Related topic (USNCCM): automated hyper-parameter model tuning for MLMF