Uncertainty Quantification of the Metal Laser Powder Bed Fusion Additive Manufacturing via the Hypercomplex-based Finite Element Method



Arturo Montoya, Prof.
Matthew Balcer, PhD Candidate
Mauricio Aristizabal, Post Doc.
Juan Rincon-Tabares, PhD Candidate
David Restrepo, Assist. Prof.
Harry Millwater, Prof.

Margie and Bill Klesse College of Engineering and Integrated Design

UTSA®
The University of Texas at San Antonio™



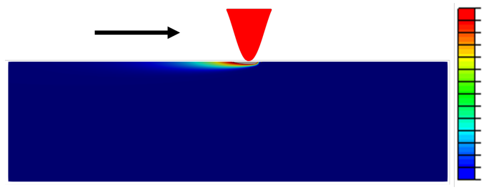
Overview

Long-term objective:

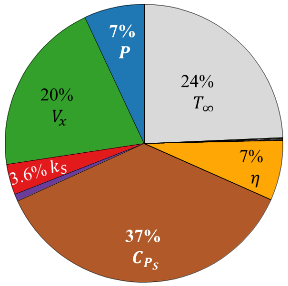
Develop, implement, verify, and validate a new computational methodology to provide sensitivities and uncertainty quantification metrics for metal-based additively manufactured components

What is new with our approach?

Uses hypercomplex algebra combined with traditional finite element methods to compute arbitrary-order high-accuracy derivatives.



- → Arbitrary order, shape, material, and loading parameters available.
- → Linear, nonlinear, or time dependent
- → Step size independent method ensures high accuracy.
- → The traditional real-valued results are still obtained and can be reused.
- → Non-Intrusive – a **postprocessing** code is programmed using hypercomplex algebra
  - → Traditional functions still used, e.g., same shape functions, etc.



Methodology is programmed **based on** a user element (UEL) for the **Abaqus** commercial software.

Partial Derivative Calculation

Finite Differentiation Method (FDM)

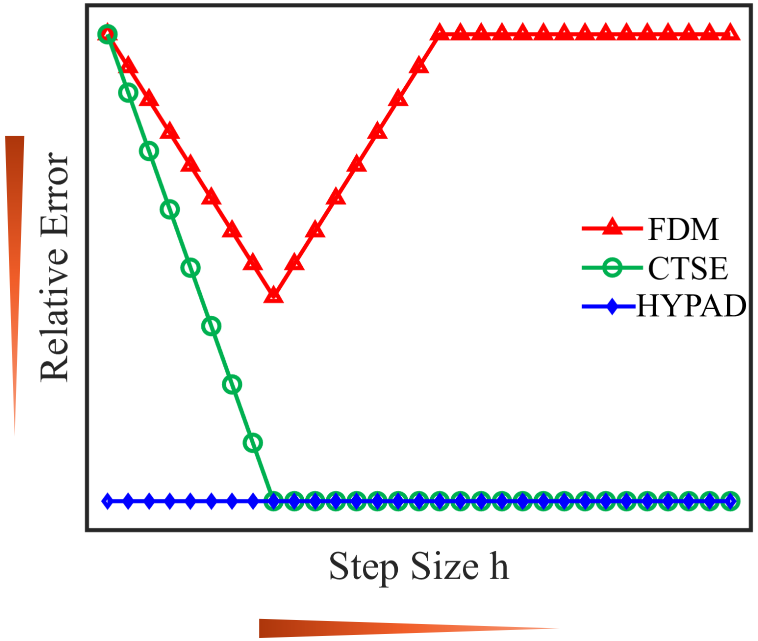
$$\frac{d\mathbf{u}(a_o)}{da} \approx \frac{\mathbf{u}(a_o + h) - \mathbf{u}(a_o)}{h}$$

- Determining h is problematic
- No code modifications

Complex Taylor Series Expansion (CTSE)

$$\frac{d\mathbf{u}(a_o)}{da} \approx \frac{Im\big(\mathbf{u}(a_o + ih)\big)}{h}$$

- h can be "very" small $\sim 10^{-30}$
- Built-in in languages



Set of Hypercomplex Numbers

Complex

$$a^* = a^{Re} + i a^{Im}$$
$$i^2 = -1$$

Dual - OTI

$$a^* = a^{Re} + \epsilon a^{\epsilon}$$
$$\epsilon^2 = 0$$

HYPAD: HYPercomplex Automatic Differentiation

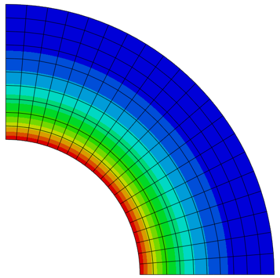
$$\frac{d\mathbf{u}(a_o)}{da} = \frac{Im_{\epsilon}\big(\mathbf{u}(a_o + \epsilon h)\big)}{h}$$

If a is perturbed along multiple imaginary directions high order sensitivities (interactions) are obtained

- h can be unitary
- Exact derivative
- Requires specific algebra packages
- Algebra accounts for composition and chain rule

HYPercomplex Automatic Differentiation (HYPAD)

Mathematical model, $f(\boldsymbol{x})$



Nodal derivatives using HYPAD

$$\frac{\partial f(\boldsymbol{x})}{\partial x_1}, \frac{\partial f(\boldsymbol{x})}{\partial x_2}, \frac{\partial^2 f(\boldsymbol{x})}{\partial x_1^2}, \frac{\partial^2 f(\boldsymbol{x})}{\partial x_1 \partial x_2}, \frac{\partial^3 f(\boldsymbol{x})}{\partial x_1^3}, \ldots$$

Taylor series expansion of nodal outputs

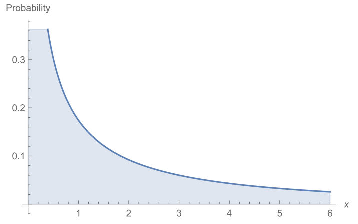
$$f(\boldsymbol{x}) \approx f(\boldsymbol{\mu}_x) + \sum_{i=1}^{m} \frac{\partial f(\boldsymbol{x})}{\partial x_i}(x_i - \mu_{x_i}) + \cdots$$

Uncertainty Quantification

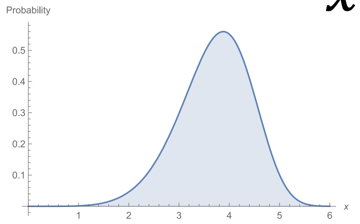
Quantifying Uncertainty in Finite Element Outputs with the Taylor Series

Random Variables

$\boldsymbol{x} = [x_1, x_2, \dots, x_r]$

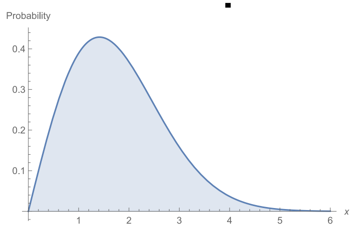


x_1



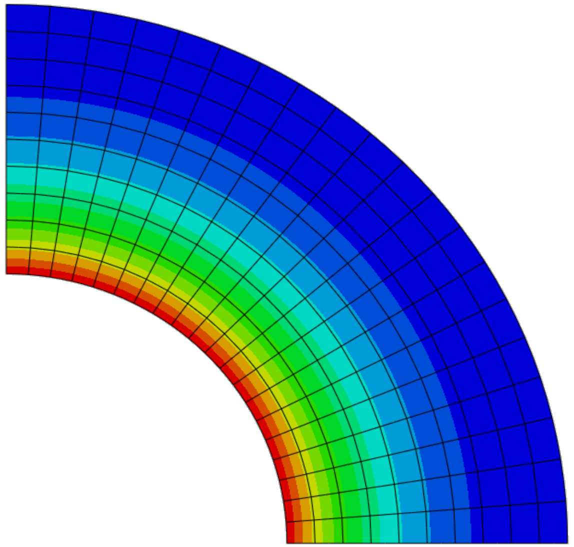
x_2

⋮



x_r

Mathematical Model, $f(\boldsymbol{x})$

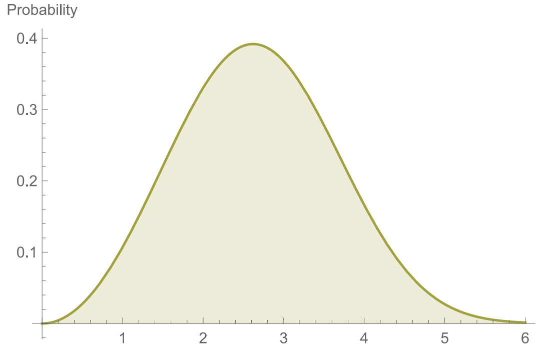


Surrogate Model

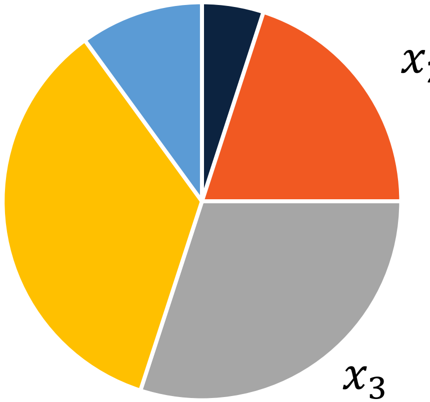
$f(\boldsymbol{x}) \approx Y_n$

(n^{th}-order Taylor series expansion)

Output



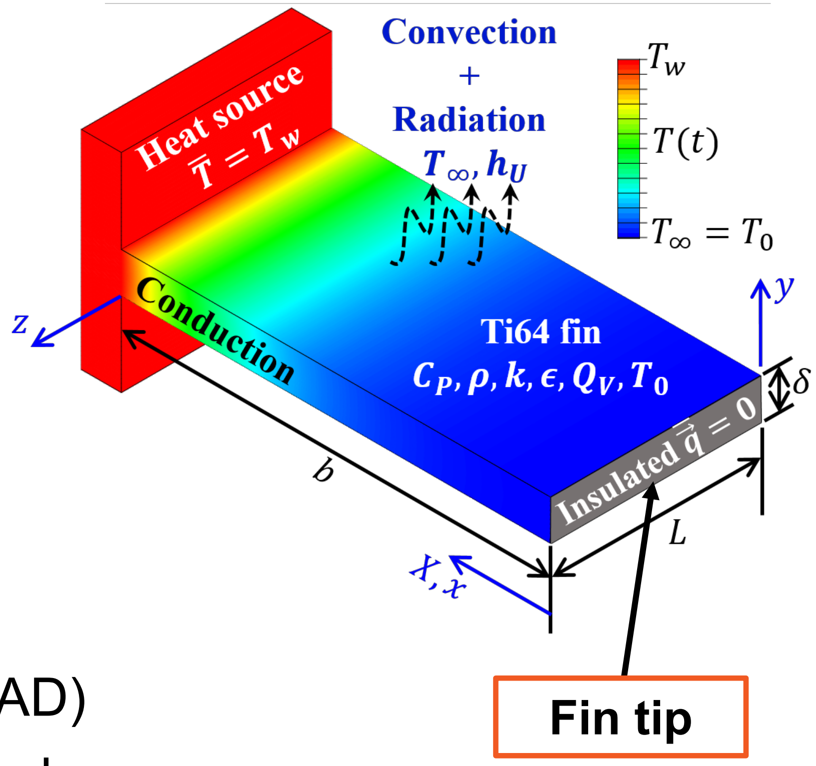
Sobol' Indices



Heated Fin: Verification Problem [1]

Goal: Quantify uncertainty of temperature at tip of fin through time

Variable	Distribution	Mean, μ_x	COV= σ_x/μ_x
Thermal conductivity, k	Log-Normal	$7.1\ \mathrm{W/(m \cdot K)}$	0.20
Specific heat, c_p	Log-Normal	$580\ \mathrm{J/(kg \cdot K)}$	0.20
Density, ρ	Log-Normal	$4430\ \mathrm{kg/m^3}$	0.20
Heat transfer coefficient, h_U	Log-Normal	$114\ \mathrm{W/(m^2 \cdot K)}$	0.20
Ambient temperature, T_∞	Triangular	$283\ \mathrm{K}$	0.01
Heat source temperature, T_w	Uniform	$389\ \mathrm{K}$	0.20
Length of fin, b	Uniform	$51\ \mathrm{mm}$	0.20



- Analytical solution was used for verification [2]
- HYPAD-UQ conducted with a 2D FEM model (using OTI-based HYPAD)
- Compared computational performance against linear regression-based stochastic perturbation finite element method

[1] Balcer, M., Aristizibal, M., Rincon-Tabares, J.-S., Montoya, A., Restrepo, D., & Millwater, H. (2023). HYPAD-UQ: A Derivative-based Uncertainty Quantification Method Using a Hypercomplex Finite Element Method. doi: 10.1115/1.4062459.
[2] Rincon-Tabares, J.-S., Velasquez-Gonzalez, J. C., Ramirez-Tamayo, D., Montoya, A., Millwater, H., & Restrepo, D. (2022). Sensitivity Analysis for Transient Thermal Problems Using the Complex-Variable Finite Element Method. Appl. Sci., 12(5), 2738. doi: 10.3390/app12052738

Hypercomplex-based Taylor Series vs Linear Regression-based Taylor Series

Computational performance of HYPAD-UQ was compared to linear regression

HYPAD-UQ

- Taylor series expansion of $f(\boldsymbol{x})$ about the mean values of $\boldsymbol{x}$

$$f(\boldsymbol{x}) \approx f(\boldsymbol{\mu_x}) + \sum_{i=1}^{m} \frac{\partial f}{\partial x_i}(x_i - \mu_{x_i}) + \frac{1}{2}\sum_{i,j=1}^{m} \frac{\partial^2 f}{\partial x_i \partial x_j}(x_i - \mu_{x_i})(x_j - \mu_{x_j}) + \cdots$$

- Derivatives calculated with HYPAD

Linear Regression-based Stochastic Perturbation Finite Element Method [1]

- Taylor series expansion of $f(\boldsymbol{x})$ (same polynomial basis)

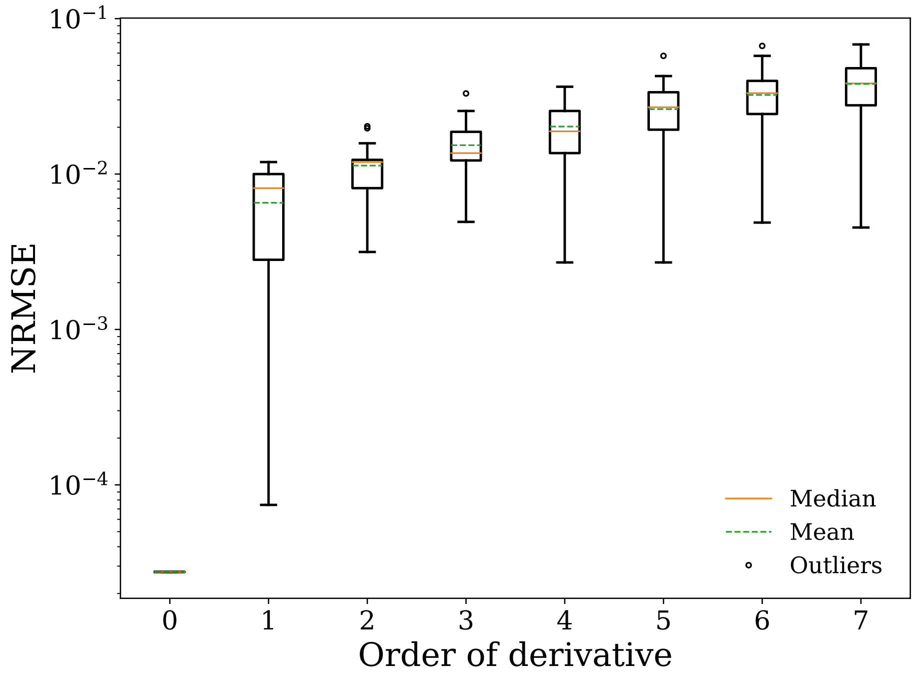
$$f(\boldsymbol{x}) \approx b_0 + \sum_{i=1}^{m} b_i x_i + \sum_{i,j=1}^{m} b_{ij} x_i x_i + \cdots$$

- Samples drawn from $f(\boldsymbol{x})$
- Unknown coefficients, b_i, approximated by Ordinary Least Squares (OLS)

[1] Kaminski, M., 2022, Uncertainty analysis in solid mechanics with uniform and triangular distributions using stochastic perturbation-based finite element method, Finite Elements in Analysis and Design, 200, 3.

HYPAD Derivative Accuracy and CPU Time

Normalized Root Mean Square Error (NRMSE)



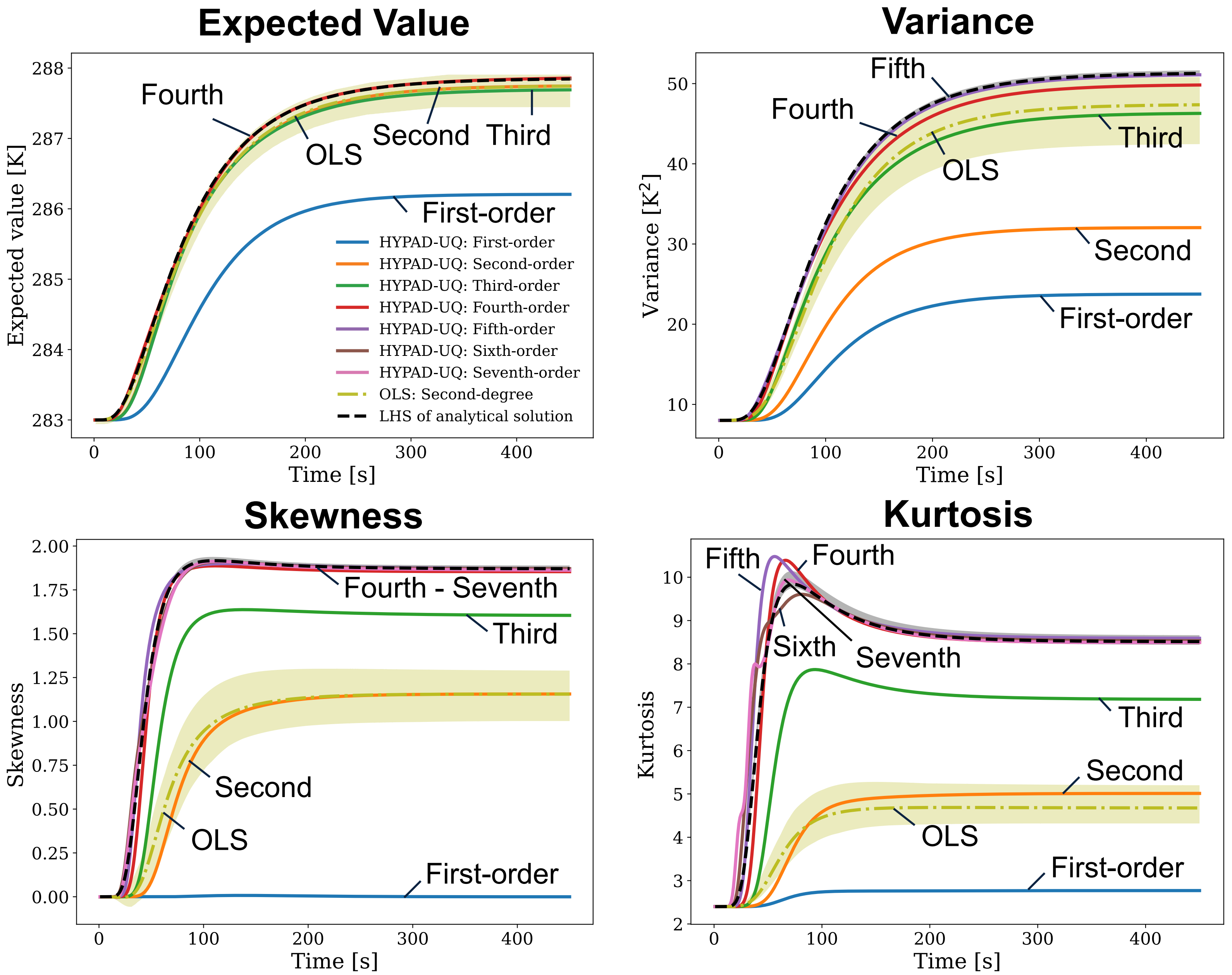
- Derivatives calculated using OTI Algebra [1]
- Each run computes all 1^{st}- through n^{th}-order partial derivatives
- **NRMSE** measured using derivatives of the analytic solution
- Error increases with order of derivative

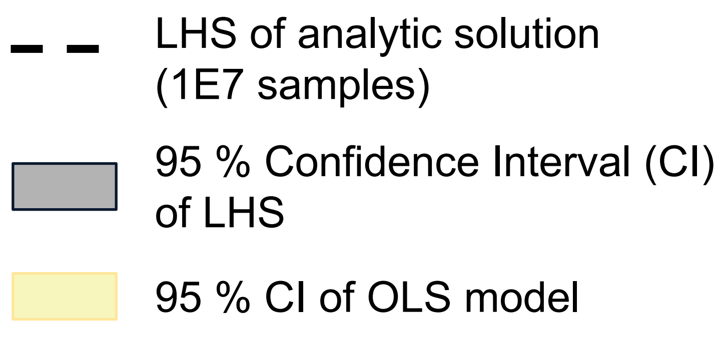
$$\mathrm{NRMSE} = \frac{\sqrt{\dfrac{\sum_{i=1}^{N}\left(\phi_{\mathrm{approx}}^{(i)} - \phi_{\mathrm{analytic}}^{(i)}\right)^2}{N}}}{\max\left(\phi_{\mathrm{analytic}}\right)}$$

Derivative order, n	First	Second	Third	Fourth	Fifth	Sixth	Seventh
Total computed derivatives	7	35	119	329	791	1715	3431
CPU time relative to a single real analysis	2.60	5.00	10.4	22.1	64.7	133.5	205.5

[1] Aristizabal Cano, M., (2020). Order truncated imaginary algebra for computation of multivariable high-order derivatives in finite element analysis, PhD thesis, Universidad EAFIT.
[2] Balcer, M., Aristizibal, M., Rincon-Tabares, J.-S., Montoya, A., Restrepo, D., & Millwater, H. (2023). HYPAD-UQ: A Derivative-based Uncertainty Quantification Method Using a Hypercomplex Finite Element Method. doi: 10.1115/1.4062459.

Central Moments

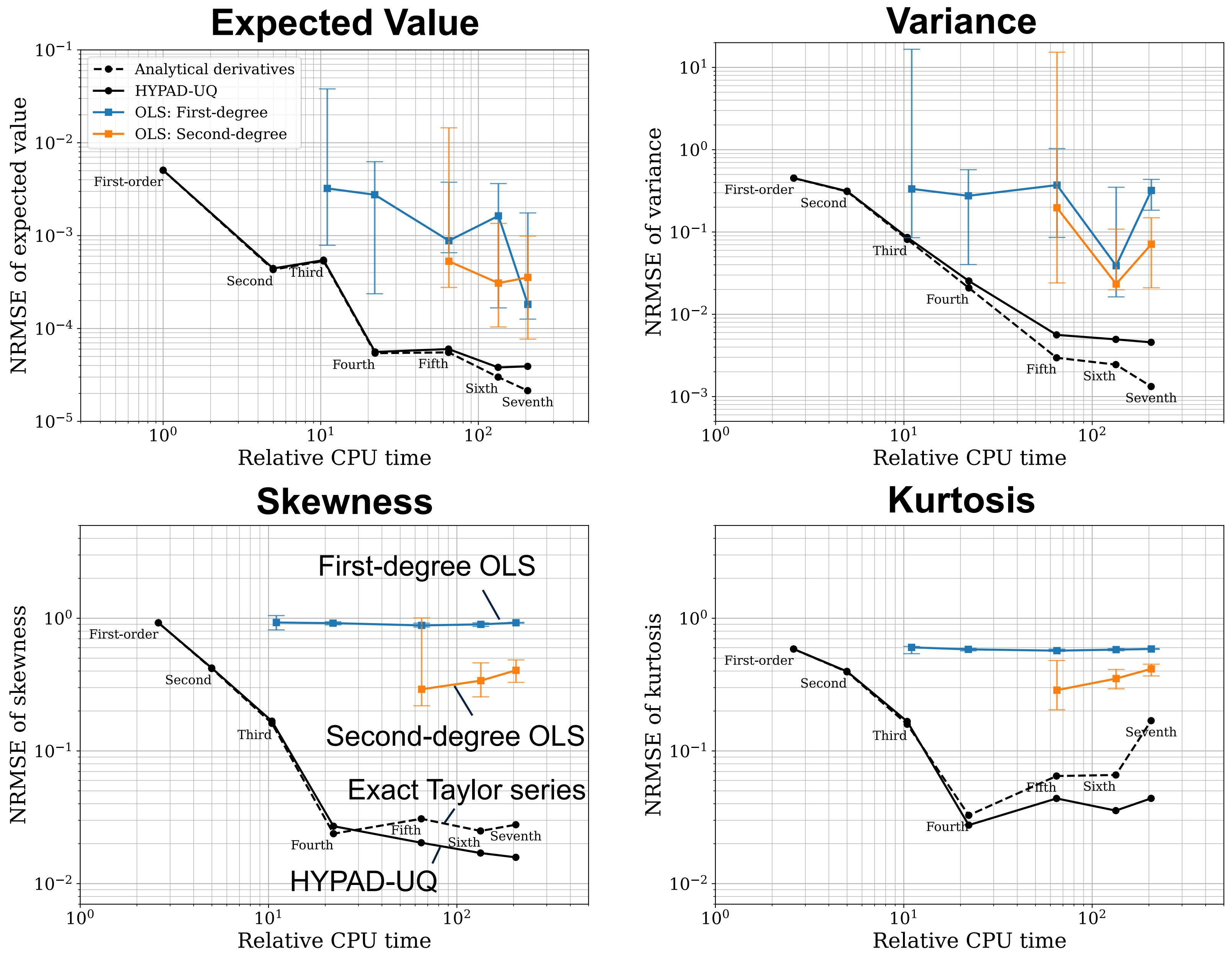




HYPAD-UQ is compared to:

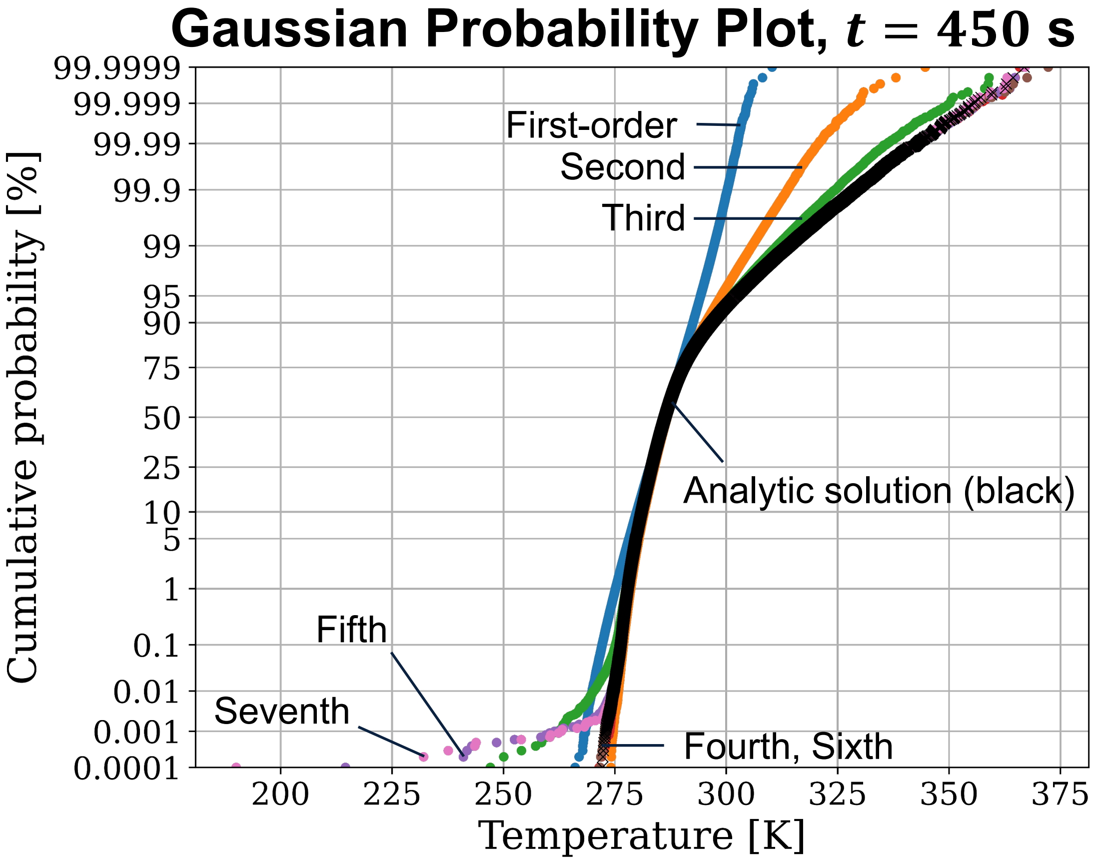
- LHS of analytical solution (1E7 samples)
- 2^{nd}-degree OLS regression, trained with 206 samples

Error of Central Moments



- HYPAD-UQ moments converge to lower errors than OLS within the same CPU time
- Higher-order expansions can increase accuracy
  - Higher-order expansions do not guarantee monotonic convergence

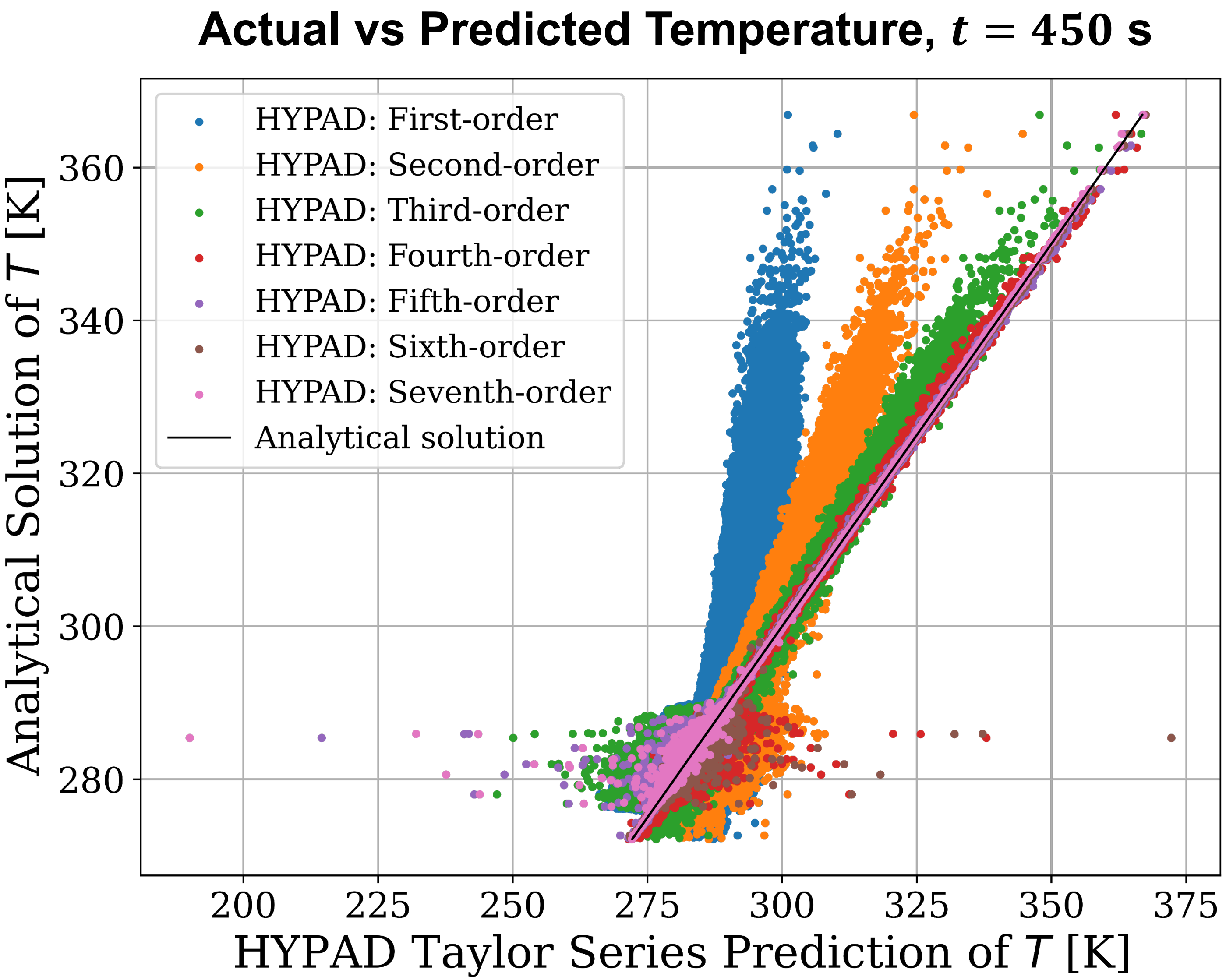
Cumulative Distribution at Steady-State



- HYPAD-UQ accurate near mean of temperature
- Higher-order HYPAD-UQ Taylor series expansions can *diverge* near the *tails* of distribution
  - Odd-ordered Taylor series diverge near low probabilities

[1] Balcer, M., Aristizibal, M., Rincon-Tabares, J.-S., Montoya, A., Restrepo, D., & Millwater, H. (2023). HYPAD-UQ: A Derivative-based Uncertainty Quantification Method Using a Hypercomplex Finite Element Method. doi: 10.1115/1.4062459.

HYPAD-based Taylor Series Prediction vs Actual Temperature

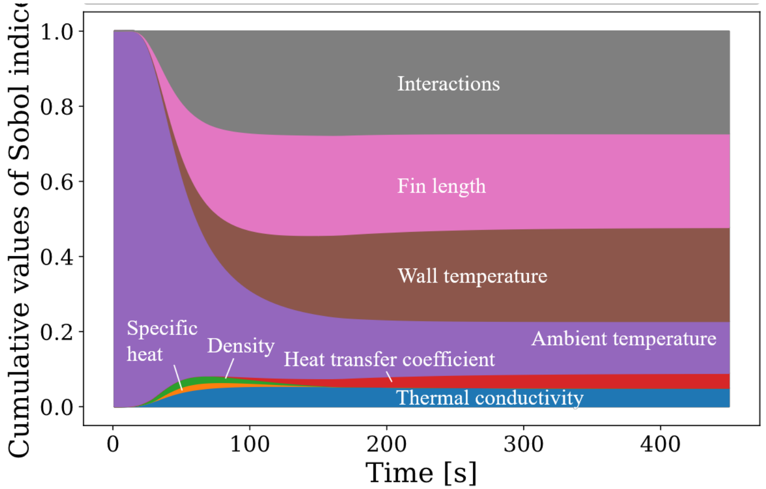


- 1E6 evaluations
- Taylor series converges to analytical solution for most of the random variable domain
- Certain combinations of random variables lead to large error in *higher-order* Taylor series expansions

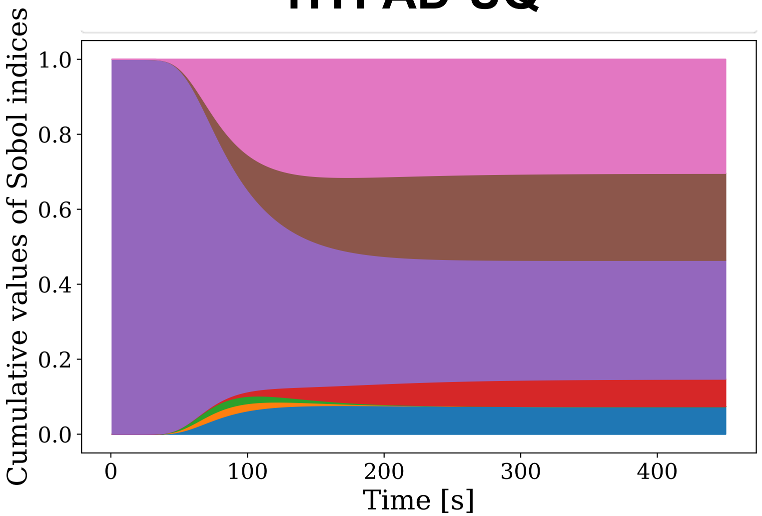
[1] Balcer, M., Aristizibal, M., Rincon-Tabares, J.-S., Montoya, A., Restrepo, D., & Millwater, H. (2023). HYPAD-UQ: A Derivative-based Uncertainty Quantification Method Using a Hypercomplex Finite Element Method. doi: 10.1115/1.4062459.

Sobol' Indices

LHS of Analytic Solution (7×10^7 samples)

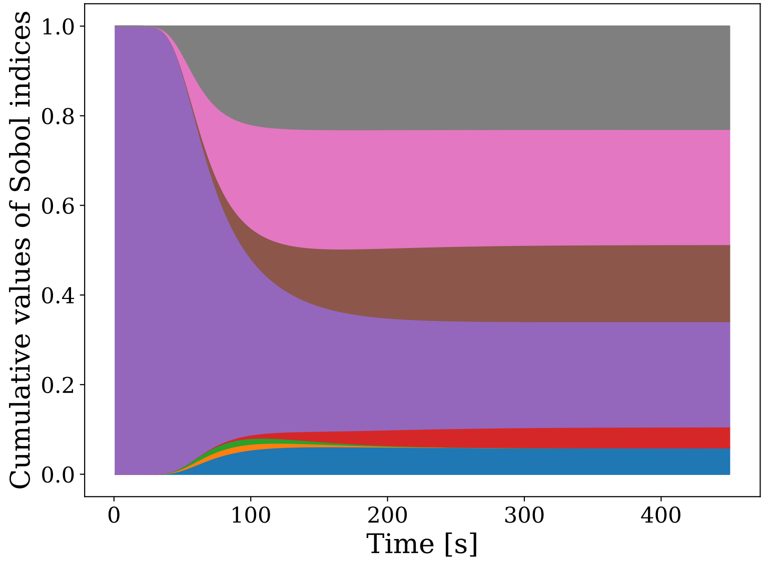


First-order HYPAD-UQ

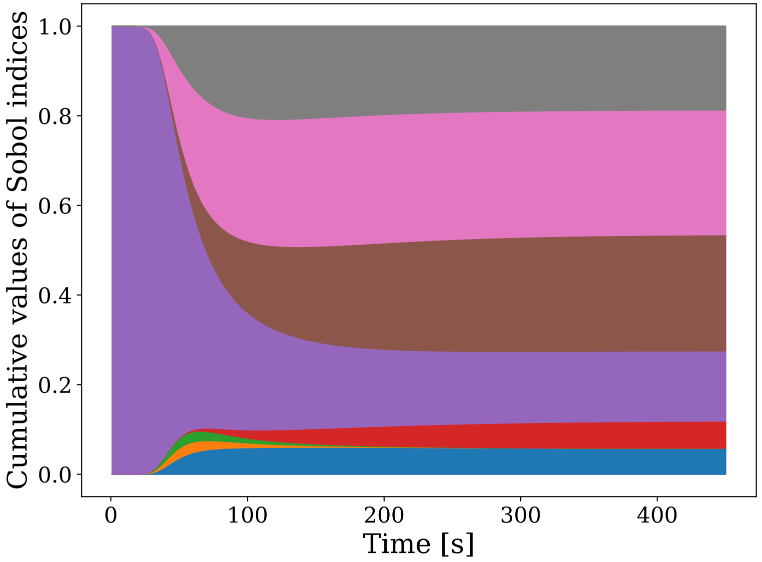


- Max of 28% of the total variance is due to *interactions*
- First-order HYPAD-UQ correctly identifies important variables
- Second-order HYPAD-UQ captures most of the interaction effect

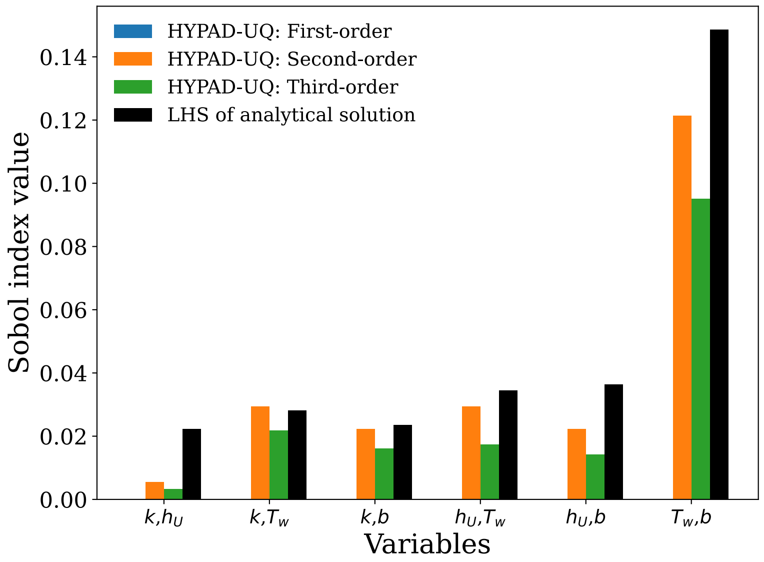
Second-order HYPAD-UQ



Third-order HYPAD-UQ



Interaction Effects at Steady-State



AM Application: Physics Involved

Thermal Profile

Transient heat transfer:

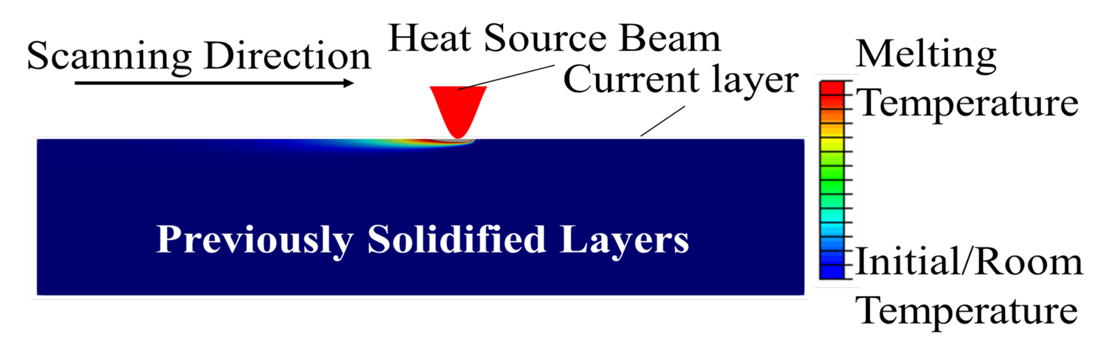
- **Fourier equation**
- **Convection/Radiation boundary conditions**
- **Moving heat source boundary condition**
- **Temperature-dependent properties**

Thermal history →

Thermomechanical

- T-dependent properties
- Residual thermal strain
- Thermoelasticity
- Thermoplasticity

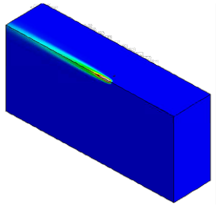
3D Sequential Model



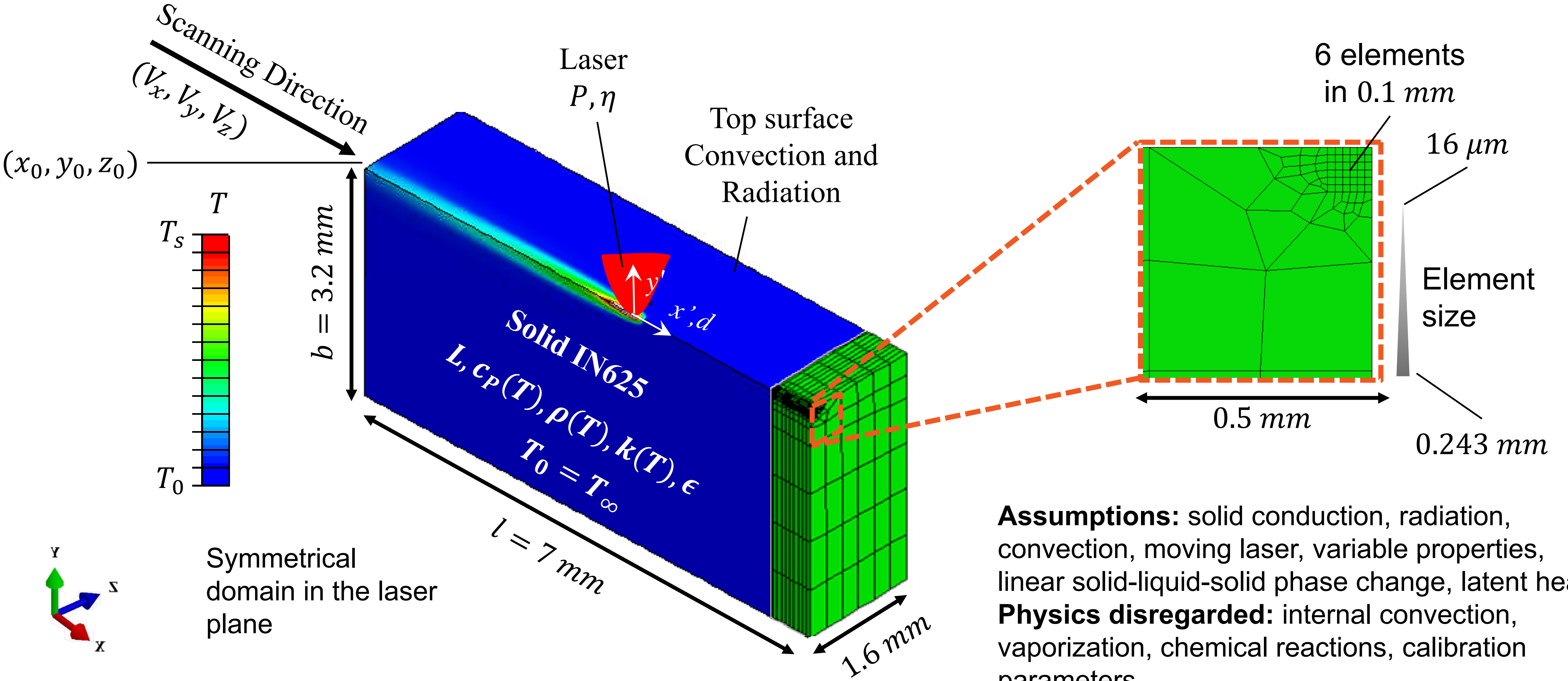
Outputs

Thermal Residual Stresses, Final Track Shape, **Thermal History**

AM Application: Bare Plate Single Track 3D Model



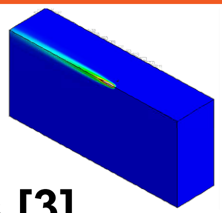
Goal: Quantify uncertainty in mean surface temperature



Assumptions: solid conduction, radiation, convection, moving laser, variable properties, linear solid-liquid-solid phase change, latent heat

Physics disregarded: internal convection, vaporization, chemical reactions, calibration parameters

AM Application: Random Variable Distribution Parameters

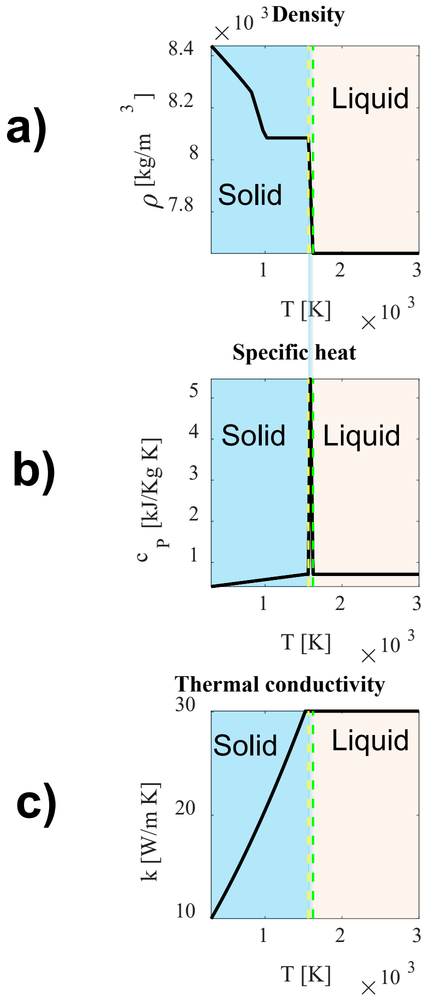


All variables are normally distributed

Type	Physics	Parameter	Mean, μ_x [1]	COV= σ_x/μ_x (%)
Constant	Laser	Radius, r_x	$0.1\ mm$	5.0
		Depth, r_y	$0.1\ mm$	5.0
		Absorption, η	0.43	2.5
		Power, P	$195\ W$	2.5 [2]
		Initial location, x_0	$-2\ mm$	1.5
		Initial location, y_0	$0\ mm$	$Std = 1.5e-4$
		Scanning speed, V_x	$800\ mm/s$	1.5 [2]
	Build Chamber Conditions	Chamber temperature, T_∞	$303K$	1.5
		Convection, h_{conv}	$18\ W/mK$	5.0
		Emissivity, ϵ	0.4	3.0
	Initial Condition	Temperature, T_0	$303\ K$	1.5
	Phase Change	Energy, ΔH_{LS}	$290\ kJ/kg\ K$	3.0
		Solidus temperature, T_S	$1563\ K$	0.5
		Liquidus temperature, T_L	$1623\ K$	0.5
Temperature -dependent	Material Properties	Density, ρ_s	Figure (a)	3.0
		Specific heat, c_{P_s}	Figure (b)	3.0 [2]
		Thermal conductivity, k_s	Figure (c)	3.0 [2]
Mesh Dependent	Geometry	Solid layers length, l	$14\ mm$	0.5 [2]
		Solid layers thickness, b	$3.2\ mm$	0.5 [2]

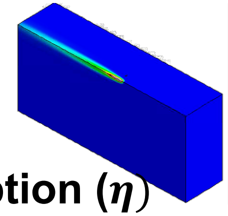
* Values were assumed

INC625 Properties [3]

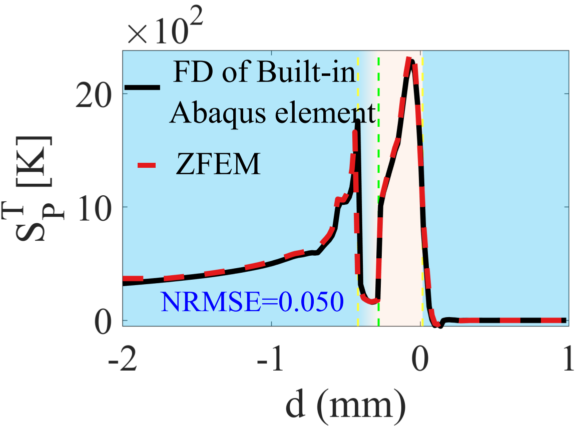


[1] Heigel, J.C.; Lane, B.M.; Levine, L.E. In Situ Measurements of Melt-Pool Length and Cooling Rate During 3D Builds of the Metal AM-Bench Artifacts. *Integr. Mater. Manuf. Innov.* **2020**, *9*, 31–53, doi:10.1007/s40192-020-00170-8.
[2] Moges, T.; Witherell, P.; Ameta, G. On characterizing uncertainty sources in laser powder bed fusion additive manufacturing models. In Proceedings of the ASME International Mechanical Engineering Congress and Exposition, Proceedings (IMECE); American Society of Mechanical Engineers (ASME): Salt Lake City, UT, USA IMECE2019-11727, 2019; Vol. 2A-2019.
[3] AFRL Additive Manufacturing (AM) Modeling Challenge Series; 2019;

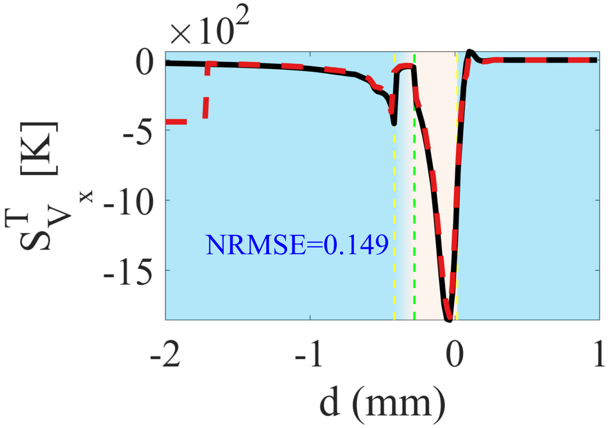
AM Application: First-order Sensitivities of Temperature, $S_\theta^T = \frac{\partial T}{\partial \theta}$



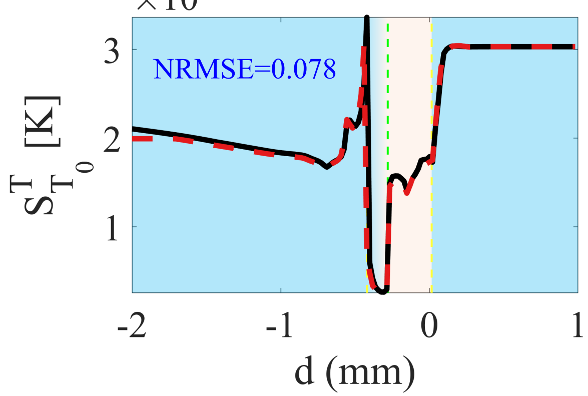
Laser Power (P)



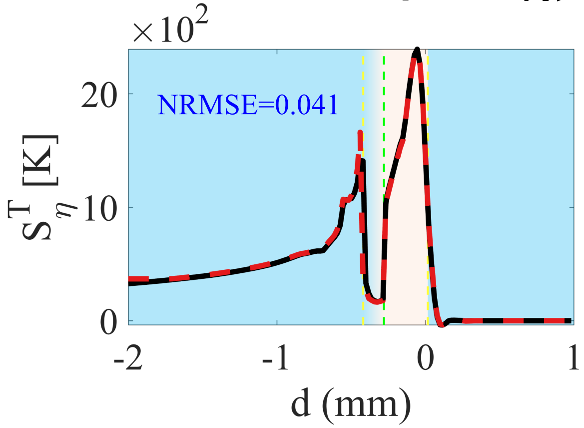
Laser speed (V_X)



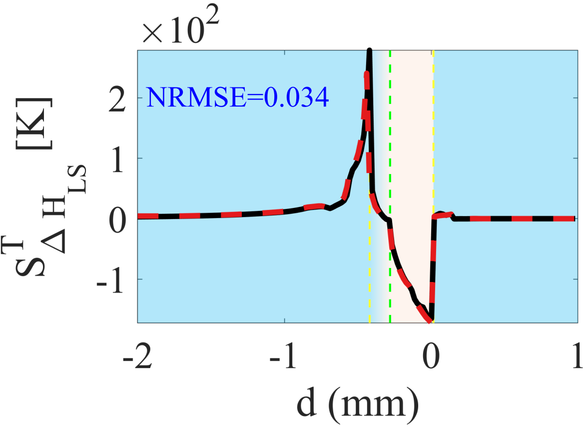
Initial Temperature (T_0)



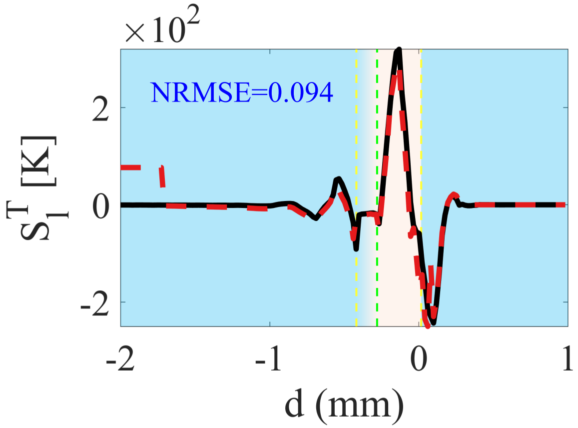
Laser absorption (η)



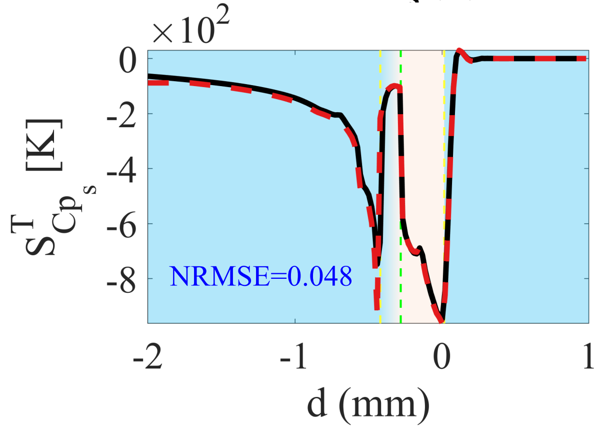
Latent heat (L)



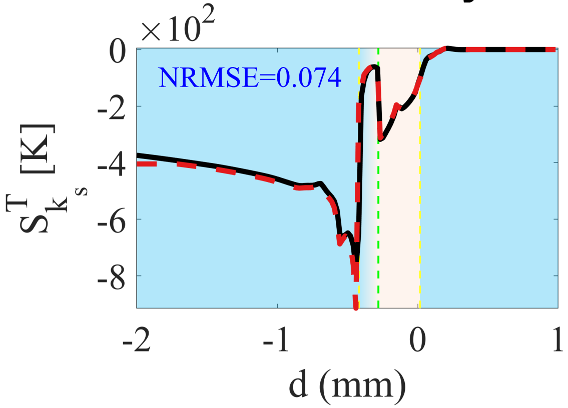
Track length (l)



Specific heat solid ($C_{P_s}(T)$)



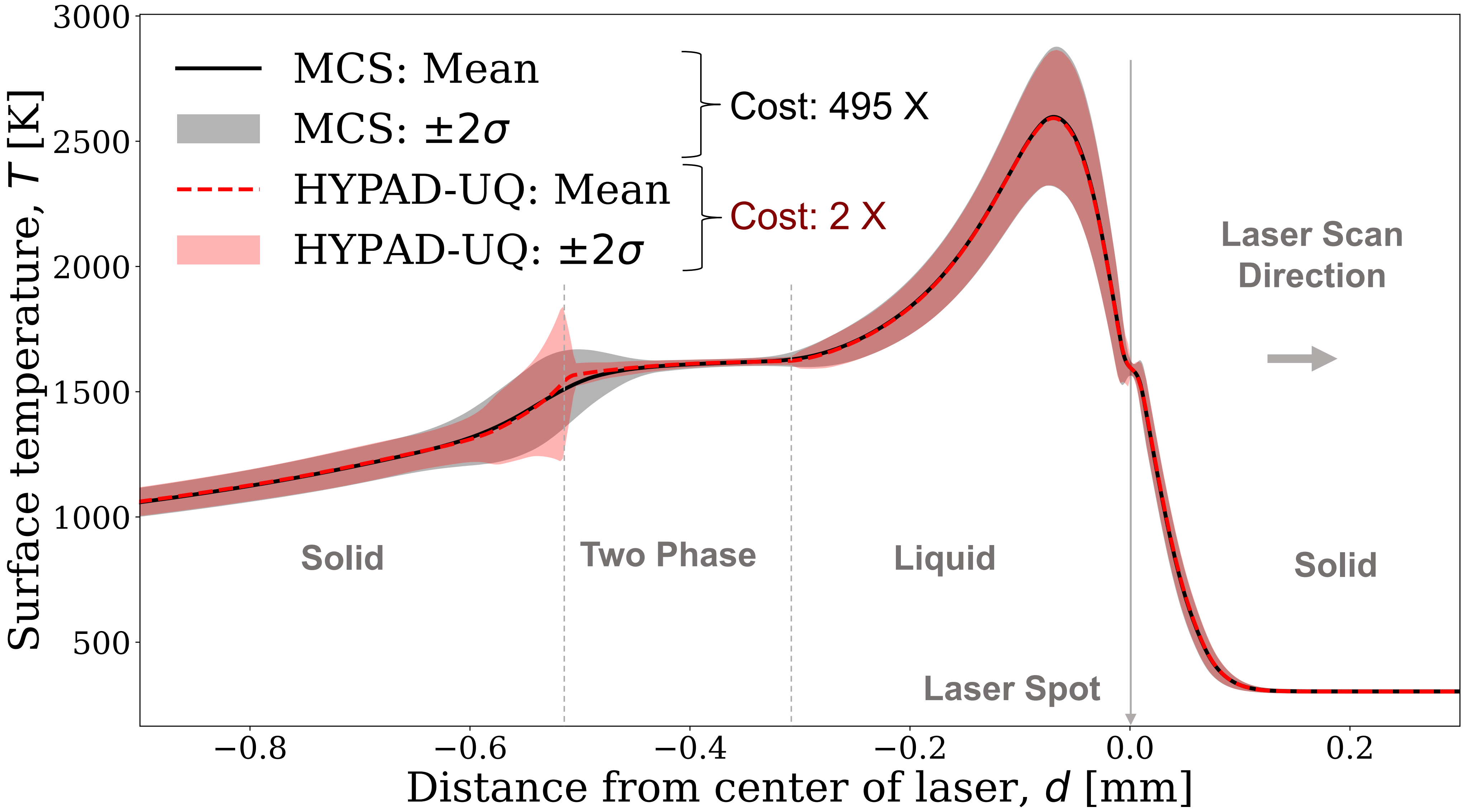
Thermal conductivity



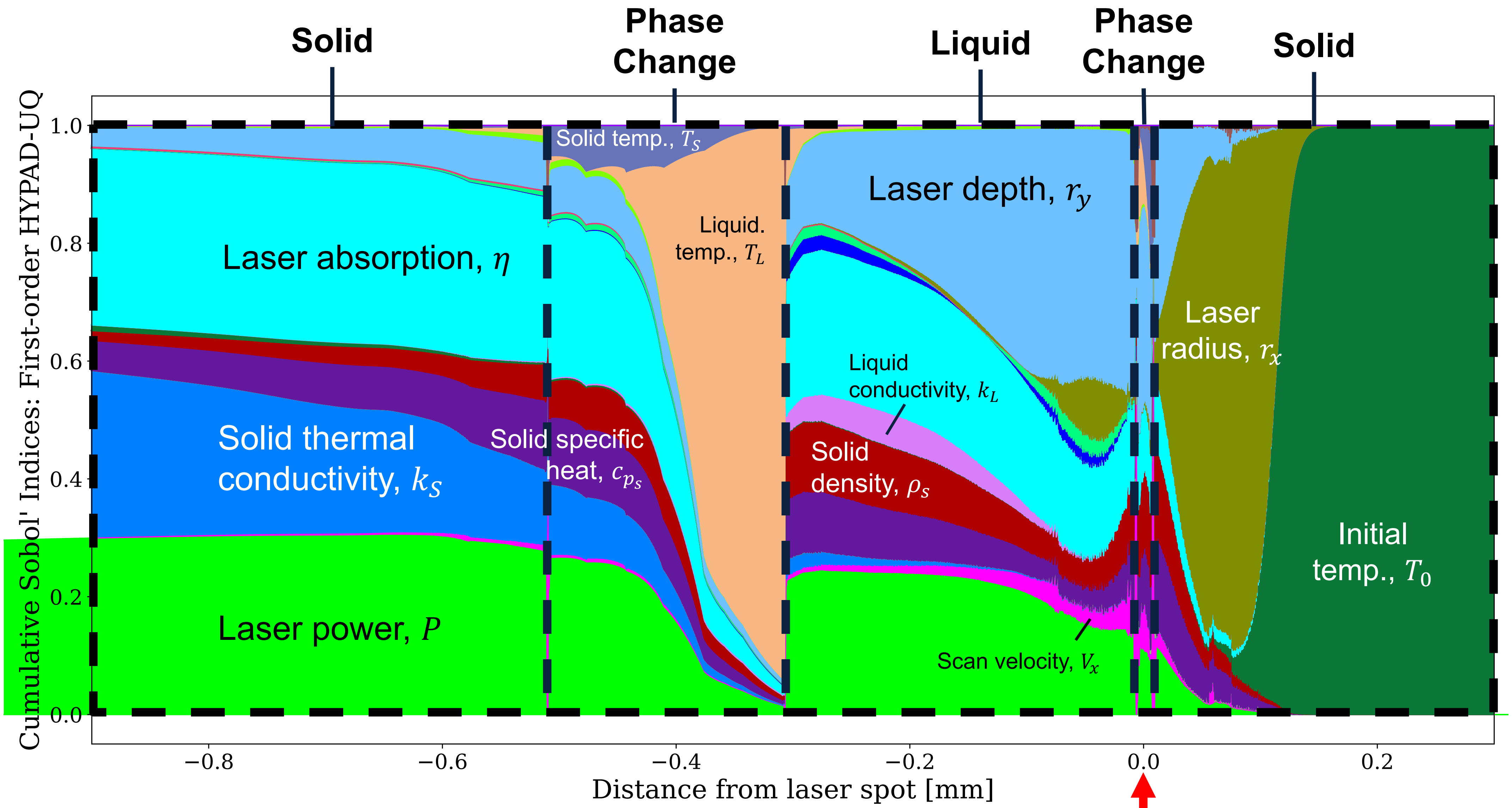
Liquid phase   Solid phase

First-order HYPAD CPU Time = ~2x a single real analysis

AM Application: Uncertainty in Mean Surface Temperature



Sobol' Indices: First-order HYPAD-UQ

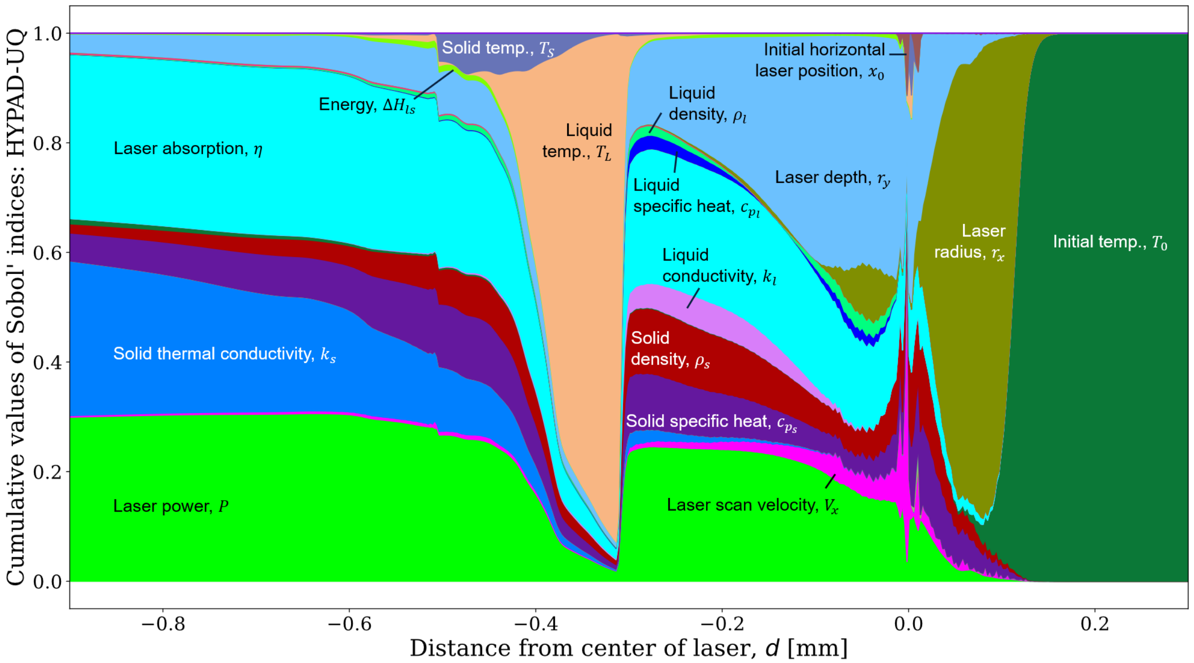


Laser Scan Direction →

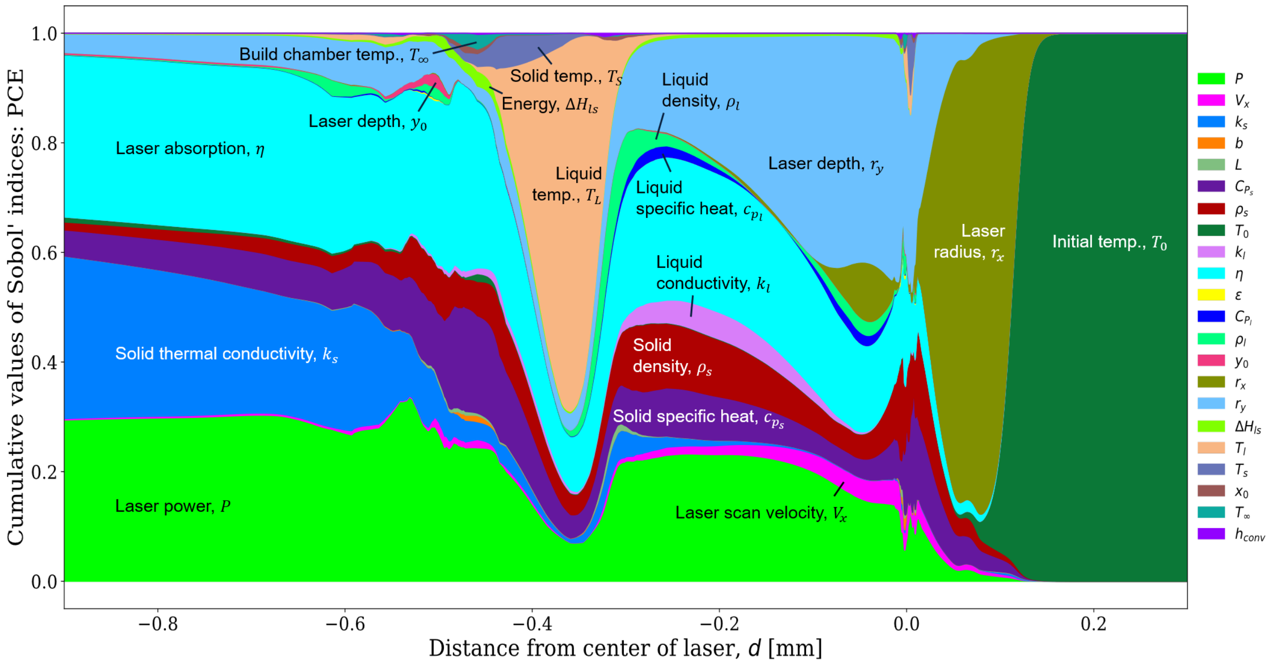
Laser Spot

Sobol' Indices: First-order HYPAD-UQ vs First-degree PCE

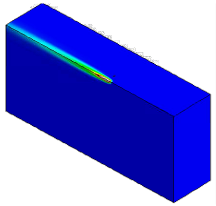
First-order HYPAD-UQ (CPU Time = 3 X)



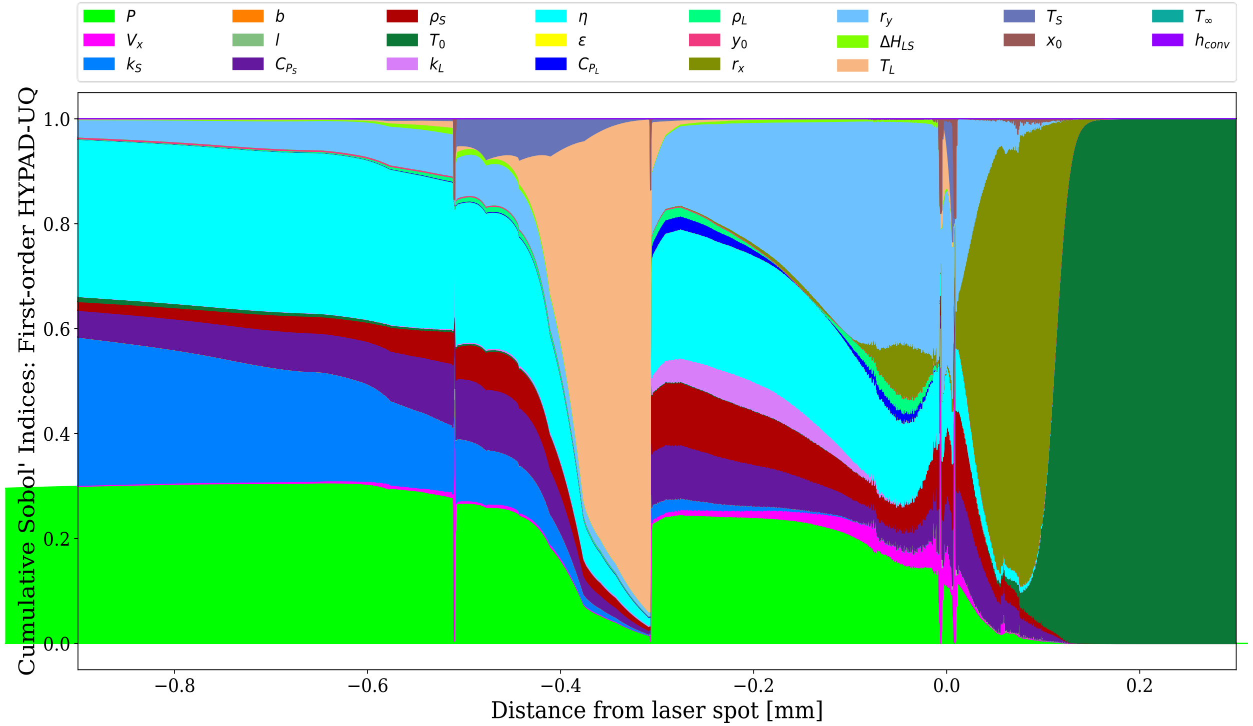
First-degree Polynomial Chaos Expansion (PCE), 100 training points from MCS design



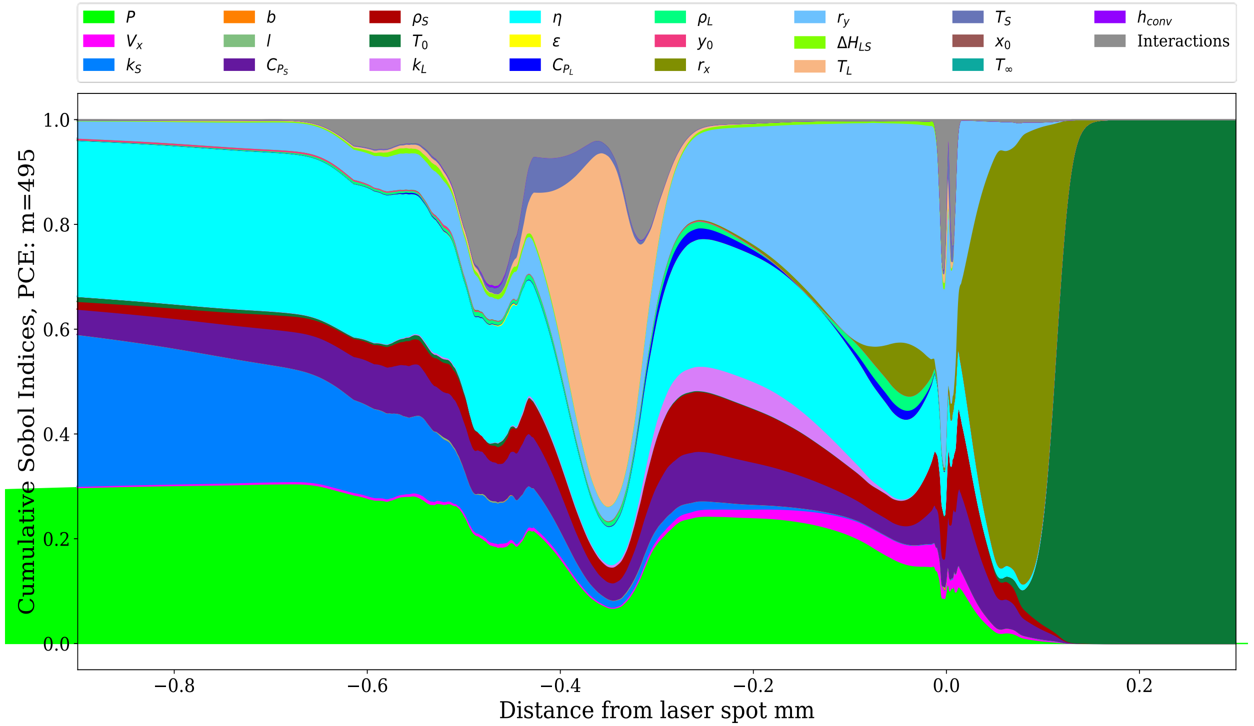
Sobol' Indices: First-order HYPAD-UQ vs Second-degree PCE



First-order HYPAD-UQ (CPU time = 3 X)



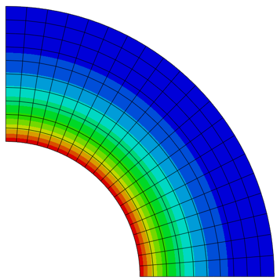
Second-degree Polynomial Chaos Expansion (PCE), 495 training points from MCS design



Summary

- Higher-order partial derivatives were calculated with HYPAD in finite elements
  - Significantly faster than finite difference with no step size issues
- HYPAD sensitivities were used to construct Taylor series expansions for UQ (HYPAD-UQ)
- HYPAD-UQ was conducted on:
  - Transient linear thermal analysis of a fin
  - Non-linear thermal analysis of an AM PBF process

Mathematical model, $f(\boldsymbol{x})$



Nodal derivatives with HYPAD

$$\frac{\partial f(\boldsymbol{x})}{\partial x_1}, \frac{\partial f(\boldsymbol{x})}{\partial x_2}, \frac{\partial^2 f(\boldsymbol{x})}{\partial x_1^2}, \frac{\partial^2 f(\boldsymbol{x})}{\partial x_1 \partial x_2}, \frac{\partial^3 f(\boldsymbol{x})}{\partial x_1^3}, \ldots$$

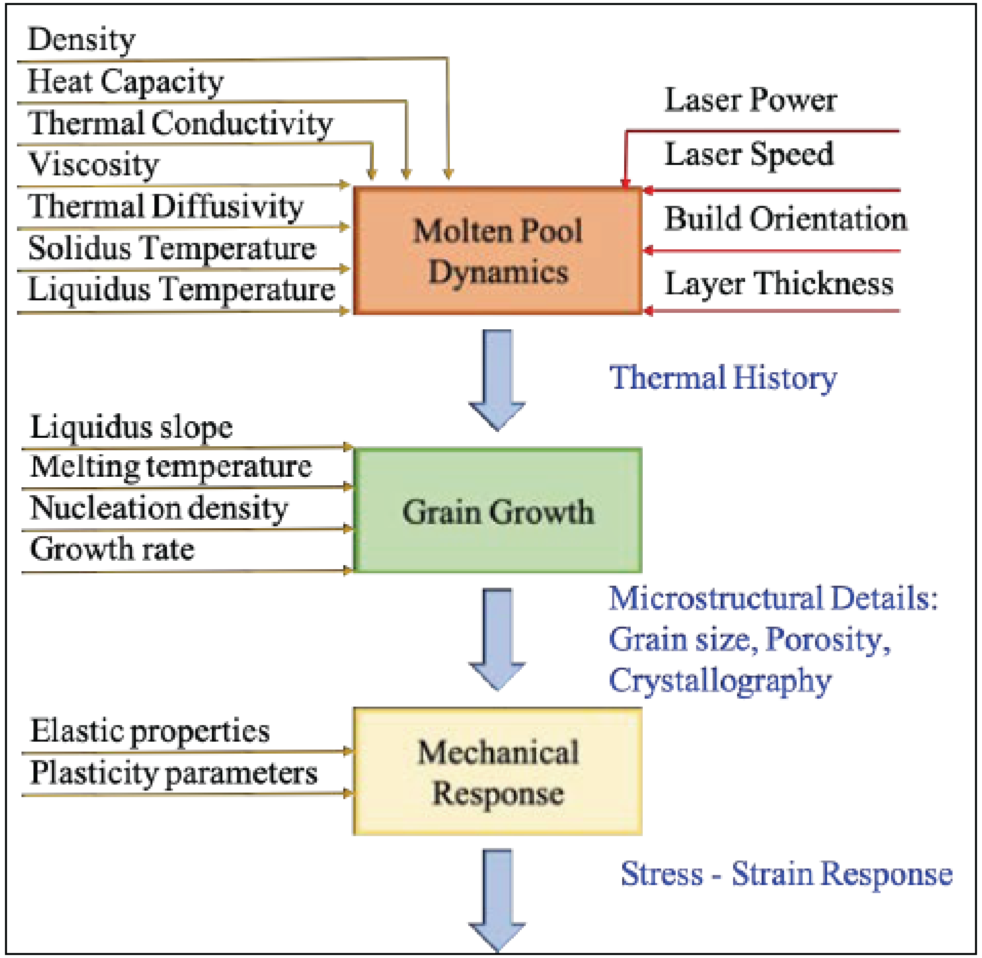
Taylor series expansion of nodal outputs

$$f(\boldsymbol{x}) \approx f(\boldsymbol{\mu}_x) + \sum_{i=1}^{m} \frac{\partial f(\boldsymbol{x})}{\partial x_i}\left(x_i - \mu_{x_i}\right) + \cdots$$

Uncertainty Quantification

Future Work

- The current development will allow the investigation of the uncertainty propagation starting from the process parameters, to the material microstructure and the bulk mechanical properties of the fabricated parts.



- Acknowledgements:
    - **Department of Energy CONNECT Consortium**
    - **Army Research Office under grant W911NF2010315. Dr. Michael Bakas Program Manager.**
    - **National Nuclear Security Administration under grant DE-NA0003948. Dr. David Canty Program Manager.**

Learn how to compute derivatives with HYPAD!

Questions ?

Backup

HYPAD Libraries

MultiZ [1]

- Multicomplex and multidual algebra support
  - Type declarations
  - Operation overloading ($+, -, \times, \div$)
  - Mathematical operation support (sine, cosine, exponential, log, sqrt, and power)
  - Arbitrary-order of hypercomplex numbers available
- Can be used with FEA simulation and other codes for sensitivity analysis
- Fortran and Python languages supported

OTI Library [2]

- Order Truncated Imaginary (OTI) algebra support
- Can be used with FEA simulation and other codes for sensitivity analysis
- Python, C, and Fortran versions developed

[1] Aguirre-Mesa, A. M., Garcia, M. J., and Millwater, H. (2020). Multiz: A library for computation of high-order derivatives using multicomplex or multidual numbers. *ACM Trans. Math. Softw.*, 46(3).
[2] Aristizabal Cano, M., (2020). Order truncated imaginary algebra for computation of multivariable high-order derivatives in finite element analysis, PhD thesis, Universidad EAFIT.

HYPAD-UQ Method Overview

Advantages

- HYPAD computes accurate Taylor series expansions
- Higher-order expansions can yield accurate results for large variation in random variables
- Works with any distribution of random variables
- Change in standard deviation or distribution is trivial to recalculate (mean stays the same)
- Computationally efficient compared to finite difference, stochastic perturbation finite element method, and random sampling

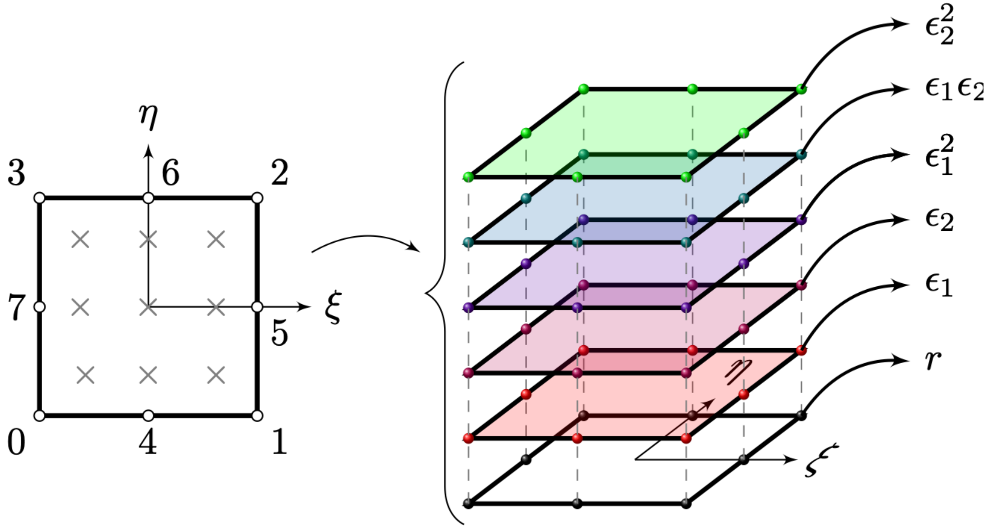
Limitations

- Potentially many terms in the Taylor series expansion
- Increase in order of expansion does not guarantee monotonic increase in accuracy
- HYPAD is intrusive – requires source code alterations
    - Once implemented, the code can be reused to compute sensitivities evaluated at any parameter

Hypercomplex Finite Element Method

- Real-valued variables are "uplifted" to hypercomplex variables
- External library used to "overload" elemental algebraic operations with hypercomplex algebra
  - Hypercomplex numbers can be expressed in matrix form to allow real-only linear algebra operations (avoids use of external library, but inefficient)
- Additional degrees of freedom to nodes for each imaginary direction

Degrees of freedom in an OTI element for truncation order of $n = 2$ and $r = 2$ variables



[*] Aristizabal Cano, M., (2020). Order truncated imaginary algebra for computation of multivariable high-order derivatives in finite element analysis, PhD thesis, Universidad EAFIT.

Block Forward Substitution to Solve Hypercomplex System of Equations

Full OTI system of equations for $n = 2$ and $r = 2$ variables

$$\mathbf{K}^*\mathbf{u}^* = \mathbf{f}^* \rightarrow \begin{bmatrix} \mathbf{K}_R & 0 & 0 & 0 & 0 & 0 \\ \mathbf{K}_{\epsilon_1} & \mathbf{K}_R & 0 & 0 & 0 & 0 \\ \mathbf{K}_{\epsilon_2} & 0 & \mathbf{K}_R & 0 & 0 & 0 \\ \mathbf{K}_{\epsilon_1^2} & \mathbf{K}_{\epsilon_1} & 0 & \mathbf{K}_R & 0 & 0 \\ \mathbf{K}_{\epsilon_1\epsilon_2} & \mathbf{K}_{\epsilon_2} & \mathbf{K}_{\epsilon_1} & 0 & \mathbf{K}_R & 0 \\ \mathbf{K}_{\epsilon_2^2} & 0 & \mathbf{K}_{\epsilon_2} & 0 & 0 & \mathbf{K}_R \end{bmatrix} \begin{Bmatrix} \mathbf{u}_R \\ \mathbf{u}_{\epsilon_1} \\ \mathbf{u}_{\epsilon_2} \\ \mathbf{u}_{\epsilon_1^2} \\ \mathbf{u}_{\epsilon_1\epsilon_2} \\ \mathbf{u}_{\epsilon_2^2} \end{Bmatrix} = \begin{Bmatrix} \mathbf{f}_R \\ \mathbf{f}_{\epsilon_1} \\ \mathbf{f}_{\epsilon_2} \\ \mathbf{f}_{\epsilon_1^2} \\ \mathbf{f}_{\epsilon_1\epsilon_2} \\ \mathbf{f}_{\epsilon_2^2} \end{Bmatrix}$$

Solve real-only system

$$\mathbf{K}_R\mathbf{u}_R = \mathbf{f}_R$$

Solve first-order system

$$\mathbf{K}_R\mathbf{u}_{\epsilon_1} = \mathbf{f}_{\epsilon_1} - \mathbf{K}_{\epsilon_1}\mathbf{u}_R$$
$$\mathbf{K}_R\mathbf{u}_{\epsilon_2} = \mathbf{f}_{\epsilon_2} - \mathbf{K}_{\epsilon_2}\mathbf{u}_R$$

Solve second-order system

$$\mathbf{K}_R\mathbf{u}_{\epsilon_1^2} = \mathbf{f}_{\epsilon_1^2} - \mathbf{K}_{\epsilon_1}\mathbf{u}_{\epsilon_1} - \mathbf{K}_{\epsilon_1^2}\mathbf{u}_R$$
$$\mathbf{K}_R\mathbf{u}_{\epsilon_1\epsilon_2} = \mathbf{f}_{\epsilon_1\epsilon_2} - \mathbf{K}_{\epsilon_1}\mathbf{u}_{\epsilon_2} - \mathbf{K}_{\epsilon_2}\mathbf{u}_{\epsilon_1} - \mathbf{K}_{\epsilon_2^2}\mathbf{u}_R$$
$$\mathbf{K}_R\mathbf{u}_{\epsilon_2^2} = \mathbf{f}_{\epsilon_2^2} - \mathbf{K}_{\epsilon_2}\mathbf{u}_{\epsilon_2} - \mathbf{K}_{\epsilon_2^2}\mathbf{u}_R$$

Summary of HYPAD

Advantages

- Simplicity – No new formulation of equations; same shape functions, integration schemes, time-integration algorithms, etc.
- Robust - No step size considerations (use very small step size or dual variables).
- Comprehensive - Once “hypercomplexified”, derivatives with respect to ANY parameter available. Selection made from the input file.
- Scalable – Mixed and higher order derivatives available.
- Intrinsic support (1^{st} order only) - No additional libraries required for first order derivatives using complex variables.

Disadvantages

- Intrusive – requires source code modification.
- Library support (mixed and higher order) - libraries required to support hypercomplex operations for mixed and higher order derivatives.
- Efficiency - Increased run time.

Taylor Series Expansions of Central Moments

Taylor series expansion of the r^{th} central moment

$$\mu_r\left(f\left(\boldsymbol{x}\right)\right) \approx \mu_r\left(Y_n\right) = E\left[\left(Y_n - E\left[Y_n\right]\right)^r\right]$$

can be computed with algebraically for any distribution of random variables, $\boldsymbol{x}$

Expected Value

$$E[Y_0] = Y(\mu_{\boldsymbol{x}})$$

$$E[Y_1] = E[Y_0]$$

$$E[Y_2] = E[Y_1] + \sum_{i=1}^{m} \frac{1}{2}\mathcal{D}_{ii}^{(2)}\mu_{2i}$$

where, $\mathcal{D}_{ij\ldots}^{(n)} = \dfrac{\partial^n f\left(\boldsymbol{x}\right)}{\partial x_i \partial x_j \ldots}$

$$\mu_{ri} = \mu_r(x_i)$$

Variance

$$\mu_2\left(Y_1\right) = \sum_{i=1}^{m}\left(\mathcal{D}_i^{(1)}\right)^2 \mu_{2i}$$

$$\mu_2\left(Y_2\right) = \mu_2\left(Y_1\right) + \sum_{i=1}^{m}\{\frac{1}{4}\left(\mathcal{D}_{ii}^{(2)}\right)^2 \mu_{4i} - \frac{1}{4}\left(\mathcal{D}_{ii}^{(2)}\right)^2 \mu_{2i}^2 + \mathcal{D}_i^{(1)}\mathcal{D}_{ii}^{(2)}\mu_{3i}\} + \sum_{i<j}^{m}\left(\mathcal{D}_{ij}^{(2)}\right)^2 \mu_{2i}\mu_{2j}$$

Sobol' Indices (Global Sensitivity Analysis)

1. Decompose function into High Dimensional Model Representation (HDMR)

$$f(\boldsymbol{x}) = f_0 + \sum_i f_i(x_i) + \sum_{i<j} f_{ij}(x_i, x_j) + \ldots + f_{12\ldots m}(x_1, x_2, \ldots, x_m)$$

$\boldsymbol{x}$ are independent random variables

$$f_0 = E[f(\boldsymbol{x})]$$
$$f_i = E[f(\boldsymbol{x})|x_i] - f_0$$
$$f_{ij} = E[f(\boldsymbol{x})|x_i, x_j] - f_i - f_j - f_0$$

2. Take variance of HDMR function

$$V = \sum_i V_i + \sum_{i<j} V_{ij} + \ldots + V_{12\ldots m}$$

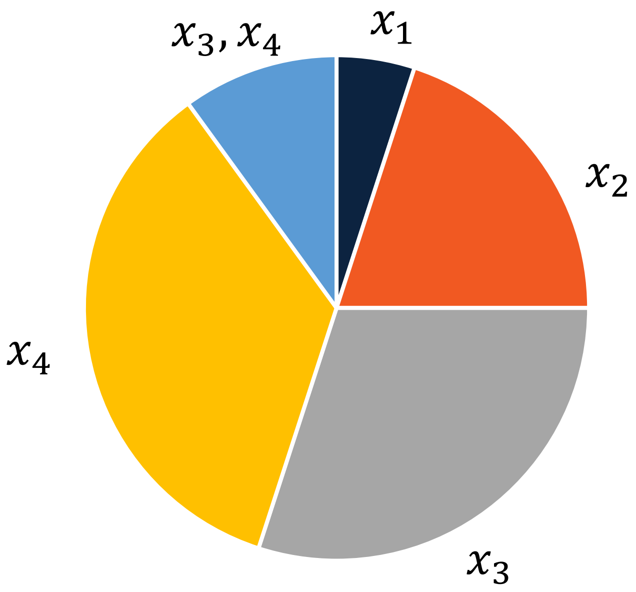
3. Divide by total variance

$$1 = \sum_i S_i + \sum_{i<j} S_{ij} + \ldots + S_{12\ldots m}$$

Main Effects $S_i = V_i/V$

Interaction Effects $S_{ij} = V_{ij}/V$

$S_{ij\ldots m} = V_{ij\ldots m}/V$



Sobol' indices sum to 100% of the total variance

Taylor Series Expansions of Sobol' Indices

Substitute $f(\boldsymbol{x}) = Y_n(\boldsymbol{x})$ (n'th-order Taylor series expansion)

Main Effects $S_i = \dfrac{V_i}{V}$

First-order $V_i\left[Y_1\right] = \left(\mathcal{D}_i^{(1)}\right)^2 \mu_{2i}$

Second-order $V_i\left[Y_2\right] = V_i\left[Y_1\right] + \frac{1}{4}\left(\mathcal{D}_{ii}^{(2)}\right)^2 \mu_{4i} - \frac{1}{4}\left(\mathcal{D}_{ii}^{(2)}\right)^2 \mu_{2i}^2 + \mathcal{D}_i^{(1)}\mathcal{D}_{ii}^{(2)}\mu_{3i}$

Interaction Effects $S_{ij} = \dfrac{V_{ij}}{V}$

Second-order $V_{ij}\left[Y_2\right] = \left(\mathcal{D}_{ij}^{(2)}\right)^2 \mu_{2i}\mu_{2j}$

Iterative Construction of a Sparse Taylor Series Expansion

An increase in:

- Number of random variables, r
- Order of expansion, n

Leads to an increase in:

- Number of partial derivatives, d
- Computational time to compute the complete n'th-order Taylor series
- Unnecessary derivative computations
    - Some terms in the expansion will not significantly contribute to increasing the accuracy in the Taylor series estimation of the output variance

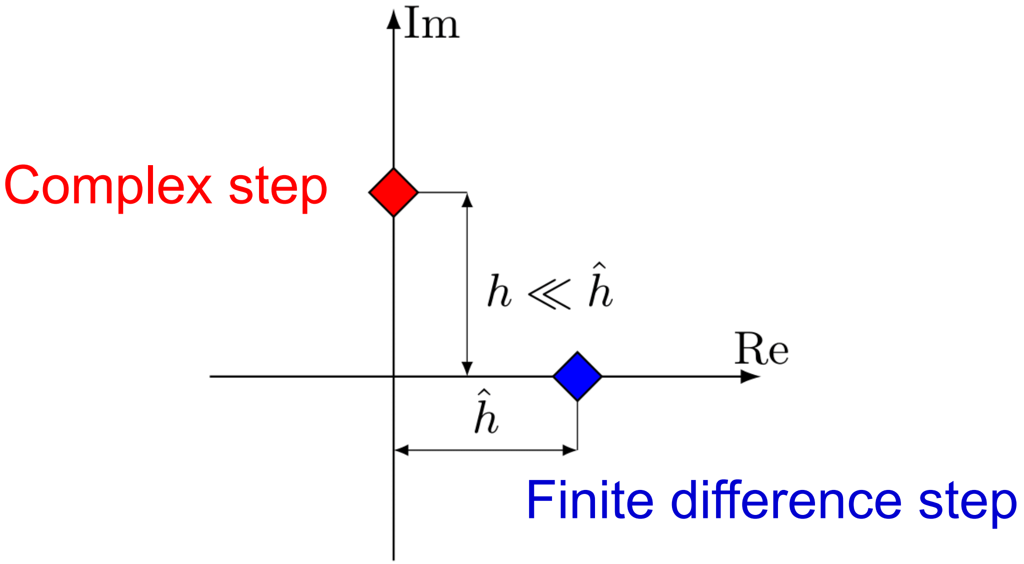
Sparse Taylor series expansion

1. Compute first-order Taylor series expansion
    - Sobol' indices to identify unimportant variables (screening)
2. Compute second-order derivatives of important variables

Partial Derivative Calculation using Hypercomplex Algebra

Complex-step Method for First-order Derivatives

- Perturb variable of interest along the imaginary axis
- Imaginary axis can be represented by a:
    - Complex number, $i^2 = 1$
    - Dual number, $\epsilon^2 = 0$
- The step size can be made arbitrarily small to neglect truncation error



HYPercomplex Automatic Differentiation (HYPAD) for Higher-order Derivatives

1. Variables are perturbed along multiple imaginary directions using hypercomplex numbers
    - Multicomplex numbers generalizes imaginary numbers to any number of directions
    - Multidual numbers generalizes dual numbers to any number of directions
    - Order Truncated Imaginary (OTI) numbers efficiently compute all derivatives in Taylor series expansion in a single analysis
2. The function is evaluated using hypercomplex algebra
3. Derivatives are extracted from the imaginary parts of the output

Hypercomplex Differentiation Implementation in Source Code

Setup

- Initialize hypercomplex library (for algebraic operation overloading)
- Define variables of interest as hypercomplex
- Define functions that use these variables as hypercomplex
- If variable/function is an array, change syntax to match hypercomplex library
- Write code to extract real and non-real parts (derivatives) of output

Running the code

- Add a non-real step to variable(s) of interest
- Run code
- Real part of output = output evaluation
- First non-real part = first derivative
- Second non-real part = second derivative, etc.

Multidual Code Conversion Example

Example

$$f\left(\boldsymbol{x}\right) = e^{x_2}\sin\left(x_1 x_2\right)$$

$$\boldsymbol{x} = [x_1, x_2] = [2, 3]$$

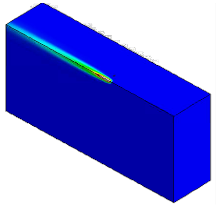
Real Code

```
program main
  implicit none
  ! declare variables
  real*8 x(2) ! input: real vector
  real*8 f    ! output: real number
  ! assign input
  x(1) = 2.0d0
  x(2) = 3.0d0
  ! calculate output
  f = exp(x(2))*sin(x(1)*x(2))
end program
```

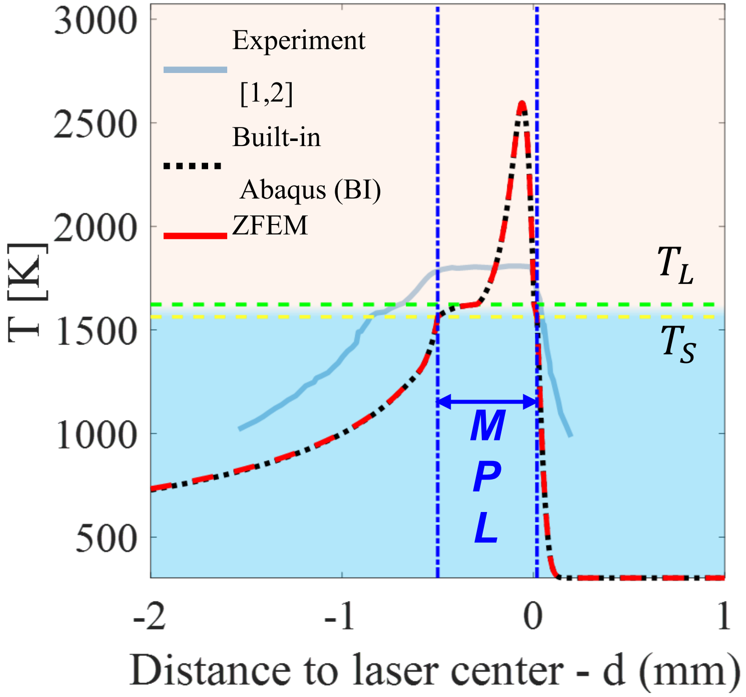
Multidual Code

```
program main
  use multiz          ! use MultiZ library
  implicit none
  ! declare variables
  type(mduvec) x      ! input: multidual vector
  type(mdual) f       ! output: multidual number
  integer n           ! size of multidual numbers for allocation
  ! derivatives
  real*8 d1, d2, d11, d12, d22, d111, d112, d122, d222
  n = 6               ! for 6 non-real steps
  ! allocate multidual vector
  call mallocate(x, n, 2)
  ! assign input with non-real steps
  call mset(x, 1, 2.0d0 + eps(1) + eps(2) + eps(3))
  call mset(x, 2, 3.0d0 + eps(4) + eps(5) + eps(6))
  ! calculate output
  f = exp(mget(x,2))*sin(mget(x,1)*mget(x,2))
  ! extract sensitivities
  d1   = aimag(f,1)         ! $\partial f(\boldsymbol{x})/\partial x_1$
  d2   = aimag(f,4)         ! $\partial f(\boldsymbol{x})/\partial x_2$
  d11  = aimag(f,[1,2])     ! $\partial^2 f(\boldsymbol{x})/\partial x_1^2$
  d12  = aimag(f,[1,4])     ! $\partial^2 f(\boldsymbol{x})/\partial x_1 \partial x_2$
  d22  = aimag(f,[4,5])     ! $\partial^2 f(\boldsymbol{x})/\partial x_2^2$
  d111 = aimag(f,[1,2,3])   ! $\partial^3 f(\boldsymbol{x})/\partial x_1^3$
  d112 = aimag(f,[1,2,4])   ! $\partial^3 f(\boldsymbol{x})/\partial x_1^2 \partial x_2$
  d122 = aimag(f,[1,4,5])   ! $\partial^3 f(\boldsymbol{x})/\partial x_1 \partial x_2^2$
  d222 = aimag(f,[4,5,6])   ! $\partial^3 f(\boldsymbol{x})/\partial x_2^3$
end program
```

AM Application: Real Value of Mean Surface Temperature



Mean Surface Temperature Profile



Mean surface temperature profile

$$NRMSE(BI, ZFEM) = 1.92e-3$$

Melt Pool Length (MPL)

$$MPL_{exp} = 0.782\ mm$$

$$e_{ZFEM}(MPL) = 33.7\%$$

Melt Pool Depth (MPD)

$$MPD_{exp} = 0.091\ mm$$

$$e_{ZFEM}(MPD) = 38.6\%$$

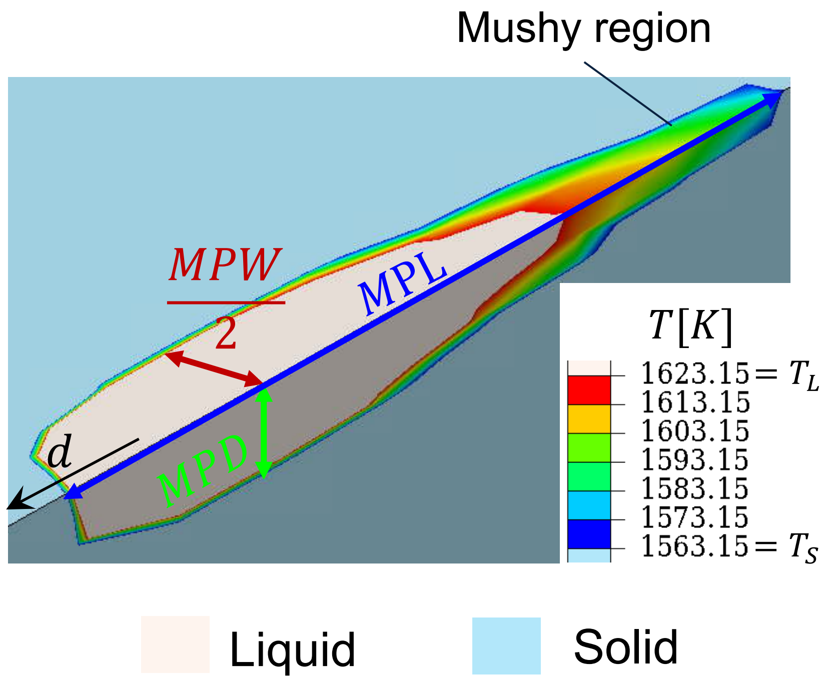
Melt Pool Width (MPW)

$$MPW_{exp} = 0.133\ mm$$

$$e_{ZFEM}(MPW) = 16.8\%$$

Model underpredicts dimensions

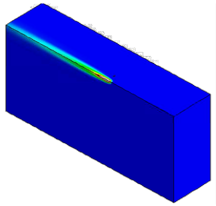
Melt Pool Section



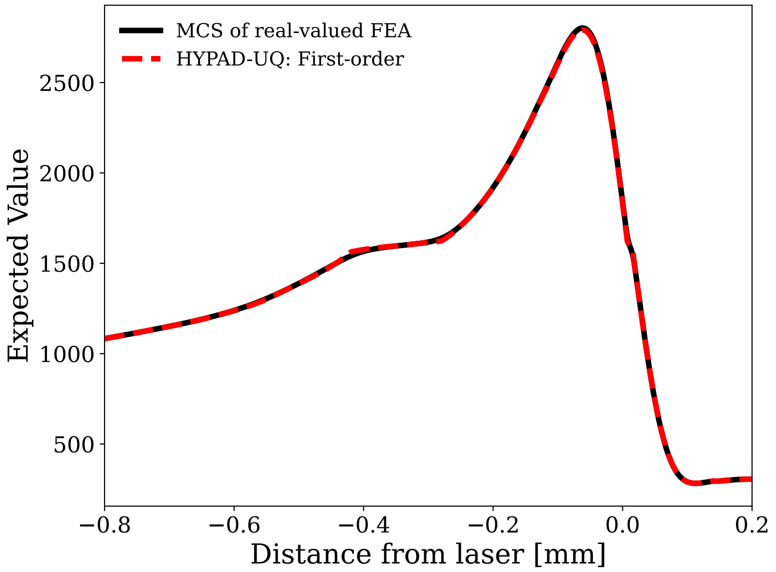
Simplifications of this model limit the precision compared to the experiments. However, the trend is in agreement.

1. Kollmannsberger, S., Carraturo, M., Reali, A., & Auricchio, F. (2019). Accurate Prediction of Melt Pool Shapes in Laser Powder Bed Fusion by the Non-Linear Temperature Equation Including Phase Changes. *Integrating Materials and Manufacturing Innovation*, *8*(2), 167–177. https://doi.org/10.1007/s40192-019-00132-9
2. Heigel, J. C., Lane, B. M., & Levine, L. E. (2020). In Situ Measurements of Melt-Pool Length and Cooling Rate During 3D Builds of the Metal AM-Bench Artifacts. Integrating Materials and Manufacturing Innovation, 9(1), 31–53. https://doi.org/10.1007/s40192-020-00170-8
3. K.-M. Hong, C. M. Grohol, and Y. C. Shin, "Comparative Assessment of Physics-Based Computational Models on the NIST Benchmark Study of Molten Pool Dimensions and Microstructure for Selective Laser Melting of Inconel 625," *Integr Mater Manuf Innov*, vol. 10, no. 1, pp. 58–71, Mar. 2021, doi: 10.1007/s40192-021-00201-y.

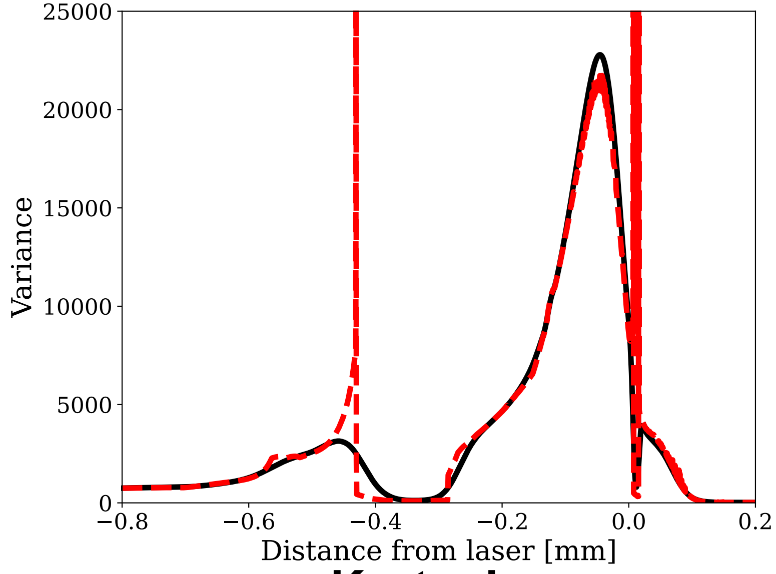
AM Application: Central Moments of Mean Surface Temperature



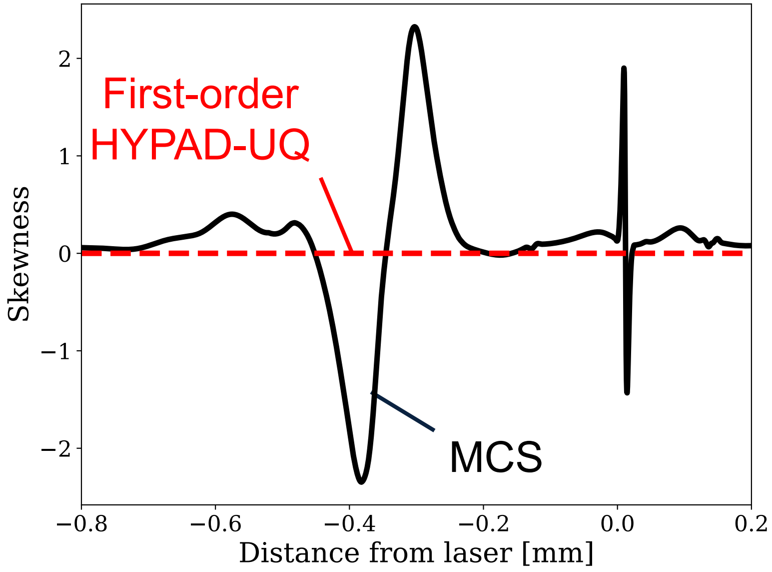
Expected Value



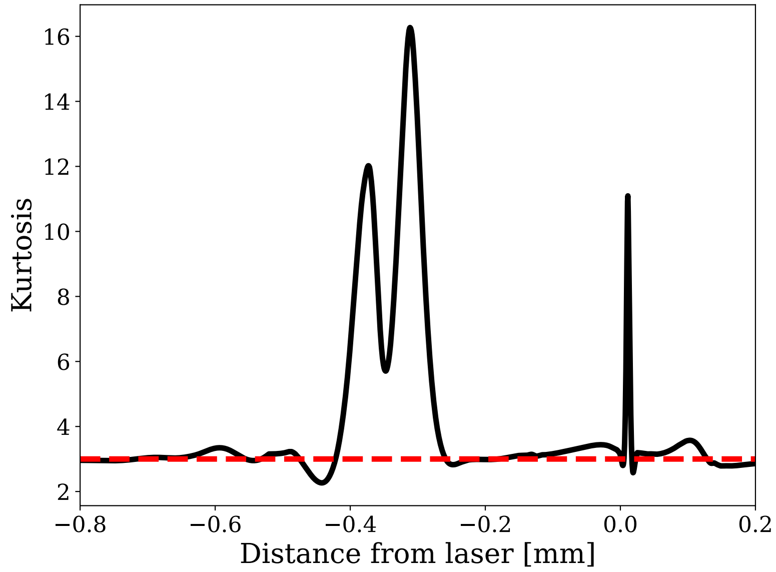
Variance



Skewness



Kurtosis

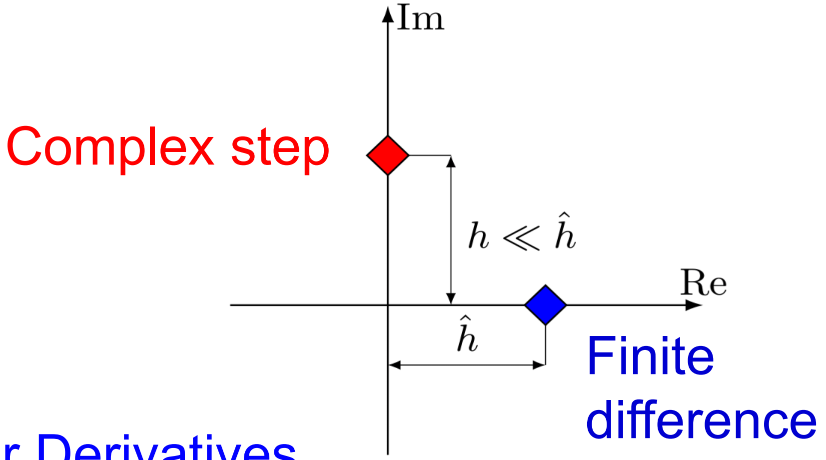


MCS of ABAQUS built-in simulation (356 samples)

1st-order HYPAD-UQ

- Higher-order Taylor series expansion needed to capture non-Gaussian skewness and kurtosis

Partial Derivative Calculation using Hypercomplex Algebra



Complex-Step Differentiation Method

- Perturb variable of interest along the imaginary axis
- Imaginary axis can be represented by a complex number, $i^2 = 1$
- Machine precision derivatives

HYPercomplex Automatic Differentiation (HYPAD) for Higher-order Derivatives

1. Variables are perturbed along multiple imaginary directions using hypercomplex numbers
   - Multicomplex numbers generalizes imaginary numbers to any number of directions
   - Multidual numbers generalizes dual numbers to any number of directions
   - Order Truncated Imaginary (OTI) numbers efficiently compute all derivatives in Taylor series expansion in a single analysis
2. The function is evaluated using hypercomplex algebra
3. Derivatives are extracted from the imaginary parts of the output

Postprocess to Compute HYPAD Derivatives

- n'th-order derivatives computed from the residual of the converged finite element solution

HYPAD-UQ Overview

HYPercomplex Automatic Differentiation (HYPAD)

- Accurate *arbitrary*-order partial derivatives
- Straight-forward implementation for any order of derivative
- Implemented in Finite Element Method (FEM)

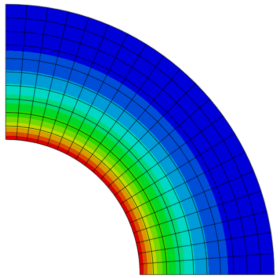
Taylor series expansion of finite element outputs

- Taylor series constructed from HYPAD sensitivities

Uncertainty Quantification (UQ) with Taylor series

- Taylor series is a surrogate model used to approximate:
  - Probability distributions
  - Central moments
  - Sobol' indices (global sensitivity analysis)

Mathematical model, $f(\boldsymbol{x})$



Nodal derivatives with HYPAD

$$\frac{\partial f(\boldsymbol{x})}{\partial x_1}, \frac{\partial f(\boldsymbol{x})}{\partial x_2}, \frac{\partial^2 f(\boldsymbol{x})}{\partial x_1^2}, \frac{\partial^2 f(\boldsymbol{x})}{\partial x_1 \partial x_2}, \frac{\partial^3 f(\boldsymbol{x})}{\partial x_1^3}, \ldots$$

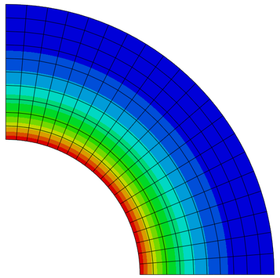
Taylor series expansions of nodal outputs

$$f(\boldsymbol{x}) \approx f(\boldsymbol{\mu_x}) + \sum_{i=1}^{m} \frac{\partial f}{\partial x_i}(x_i - \mu_{x_i}) + \cdots$$

Uncertainty Quantification

Uncertainty Quantification using HYPAD (HYPAD-UQ)

Mathematical model, $f(\boldsymbol{x})$



↓

Nodal derivatives using HYPAD

$$\frac{\partial f(\boldsymbol{x})}{\partial x_1}, \frac{\partial f(\boldsymbol{x})}{\partial x_2}, \frac{\partial^2 f(\boldsymbol{x})}{\partial x_1^2}, \frac{\partial^2 f(\boldsymbol{x})}{\partial x_1 \partial x_2}, \frac{\partial^3 f(\boldsymbol{x})}{\partial x_1^3}, \ldots$$

↓

Taylor series expansion of nodal outputs

$$f(\boldsymbol{x}) \approx f(\boldsymbol{\mu}_x) + \sum_{i=1}^{m} \frac{\partial f(\boldsymbol{x})}{\partial x_i}\left(x_i - \mu_{x_i}\right) + \cdots$$

↓

Uncertainty Quantification