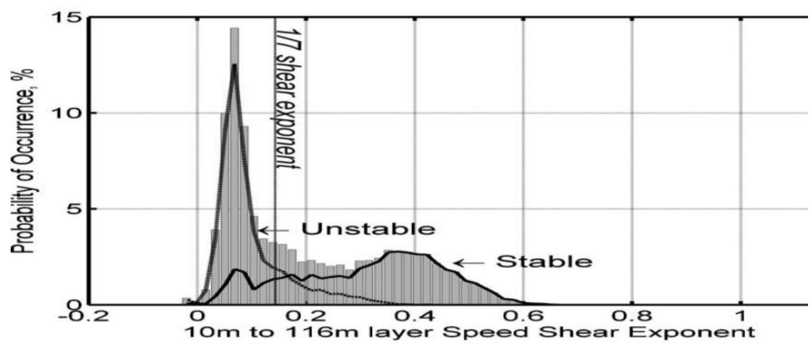
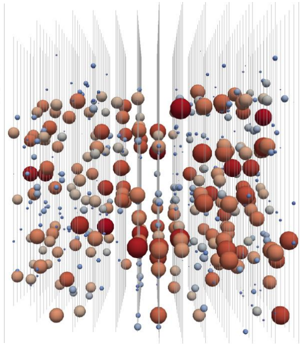
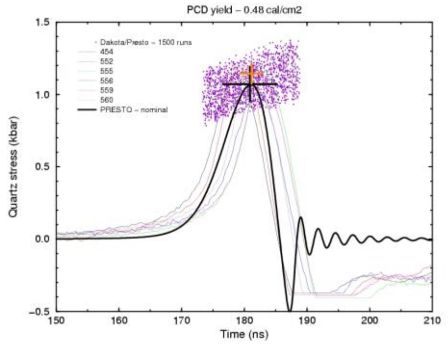
SAND2014-4255 C

Exceptional service in the national interest

Sandia National Laboratories







Dakota Sensitivity Analysis and Uncertainty Quantification, with Examples

Adam Stephens, Laura Swiler

Dakota Clinic at CSDMS May 22, 2014

U.S. DEPARTMENT OF ENERGY

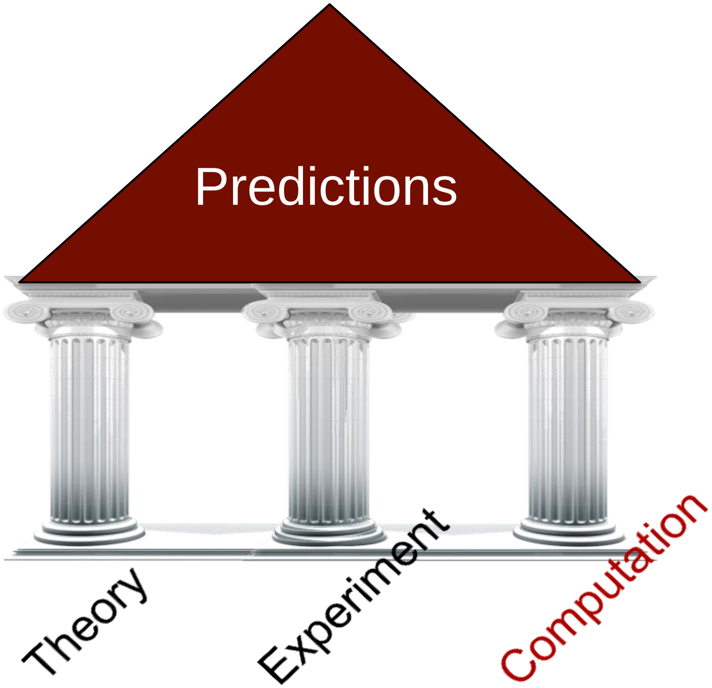
NNSA National Nuclear Security Administration

SAND2014-4255C

Dakota Sensitivity Analysis and Uncertainty Quantification, with Examples

- Dakota overview
  - How Dakota enhances computational models / simulations
  - Dakota project and software
  - Basics of getting started
- Sensitivity Analysis Methods with Examples
  - Parameter studies
  - Global sensitivity analysis
- Uncertainty Quantification Methods with Examples
  - Basic and advanced UQ methods in Dakota
- Model Calibration Methods with Examples
  - Least squares
  - Bayesian

Credible Prediction in Scientific Discovery and Engineering Design



- Predictive computational models, enabled by theory and experiment, can help:
  - Predict, analyze scenarios, including in untestable regimes
  - Assess risk and suitability
  - Design through virtual prototyping
  - Generate or test theories
  - Guide physical experiments
- *Answer what-if? when experiments infeasible…*

For simulation to credibly inform scientific, engineering, and policy decisions we must:

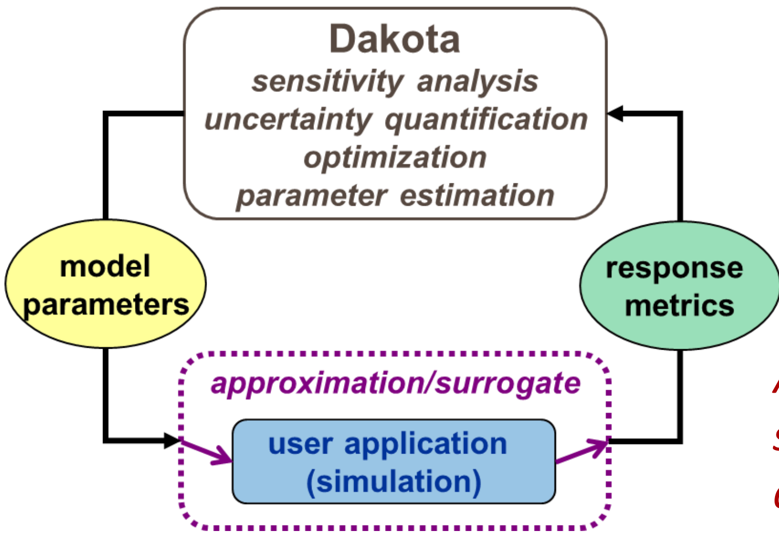
- Ask critical questions of theory, experiments, simulation
- Use software quality and model management best practices
- Manage uncertainties and use tools for UQ, calibration, optimization

Dakota Supports Simulation Credibility

Provides greater perspective for scientists, engineers, and decision makers—

  - *Enhances understanding of risk by quantifying margins and uncertainties*
  - *Improves products through simulation-based design*
  - *Assesses simulation credibility through verification and validation*

- Enables computer-based experiments analogous to physical experiments
- Manages and analyzes ensembles of simulations:

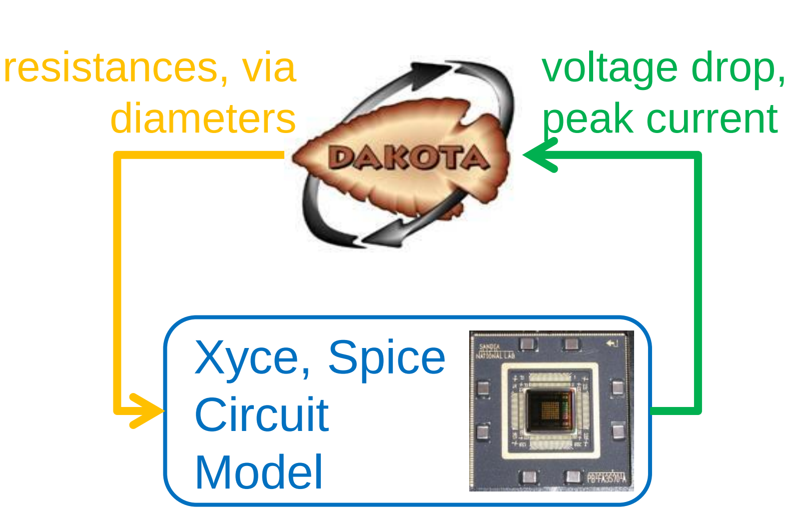


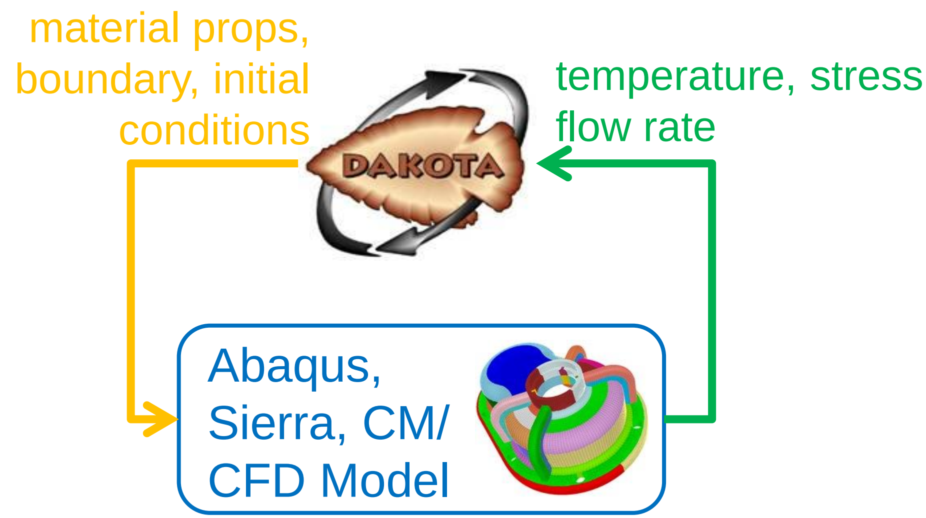
Automate typical "parameter variation" studies with various advanced methods and a generic interface to your simulation.

Advanced Exploration of Simulations

Dakota enriches simulations to address analyst/designer questions:

- Which are crucial factors/parameters, how do they affect key metrics? *(sensitivity)*
- How safe, reliable, robust, or variable is my system? (*UQ)*
- What is the best performing design or control? *(optimization)*
- What models and parameters best match experimental data? *(calibration)*





All based on iterative analysis of a computational model for phenomenon of interest

- Commercial or In-house, loose-coupled/black-box or embedded/tightly integrated…

Dakota History and Resources

- Genesis: 1994—Originally only an optimization tool
- Modern software quality and development practices—continuous integration, nightly cross-platform testing
- Released every May 15 and Nov 15
- Established support process for SNL, Tri-Lab, and beyond

http://dakota.sandia.gov

Algorithm R&D, driven by user needs, deployed in production software

- Extensive website: documentation, training materials, downloads
- Open source LGPL license facilitates external collaboration
- Over 12,000 Downloads

Recent Publications

- Jakeman, J.D. and Narayan, A., "Adaptive Leja sparse grid constructions for stochastic collocation and high-dimensional approximation," *SIAM Journal on Scientific Computing*, submitted.
- Safta, C., Chowdhary, K., Sargsyan, K., Najm, H.N., Debusschere, B.J., Swiler, L.P., and Eldred, M.S., "Probabilistic Methods for Sensitivity Analysis and Calibration in the NASA Challenge Problem," in *AIAA Journal*, submitted.
- Jakeman, J.D. and Wildey, T.M., "Enhancing adaptive sparse grid approximations and improving refinement strategies using adjoint-based a posteriori error estimates," *Journal of Computational Physics*, submitted.
- Bichon, B.J., Eldred, M.S., Mahadevan, S., and McFarland, J.M., "Efficient Global Surrogate Modeling for Reliability-Based Design Optimization," *ASME Journal of Mechanical Design*, Vol. 107, No. 1, Jan. 2013, pp. 011009:1-13.
- Weirs, V.G., Kamm, J.R., Swiler, L.P., Tarantola, S., Ratto, M., Adams, B.M., Rider, W.J., and Eldred, M.S., "Sensitivity analysis techniques applied to a system of hyperbolic conservation laws," *Reliability Engineering and System Safety (RESS)*, Vol. 107, Nov. 2012, pp. 157-170.
- Constantine, P.G., Eldred, M.S., and Phipps, E.T., "Sparse Pseudospectral Approximation Method," *Computer Methods in Applied Mechanics and Engineering*, Volumes 229-232, pp. 1-12, July 2012.
- Eldred, M.S., Swiler, L.P., and Tang, G., "Mixed Aleatory-Epistemic Uncertainty Quantification with Stochastic Expansions and Optimization-Based Interval Estimation," *Reliability Engineering and System Safety (RESS)*, Vol. 96, No. 9, Sept. 2011, pp. 1092-1113.

http://dakota.sandia.gov/publications.html

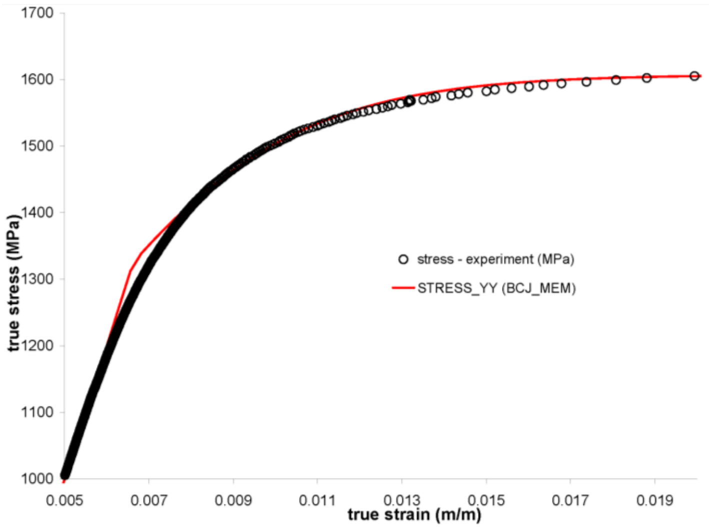
Broad Science and Engineering Needs Drive Dakota Development

Many simulation areas: mechanics, structures, shock, fluids, electrical, radiation, bio, chemistry, climate, infrastructure, etc, for applications in varied disciplines—

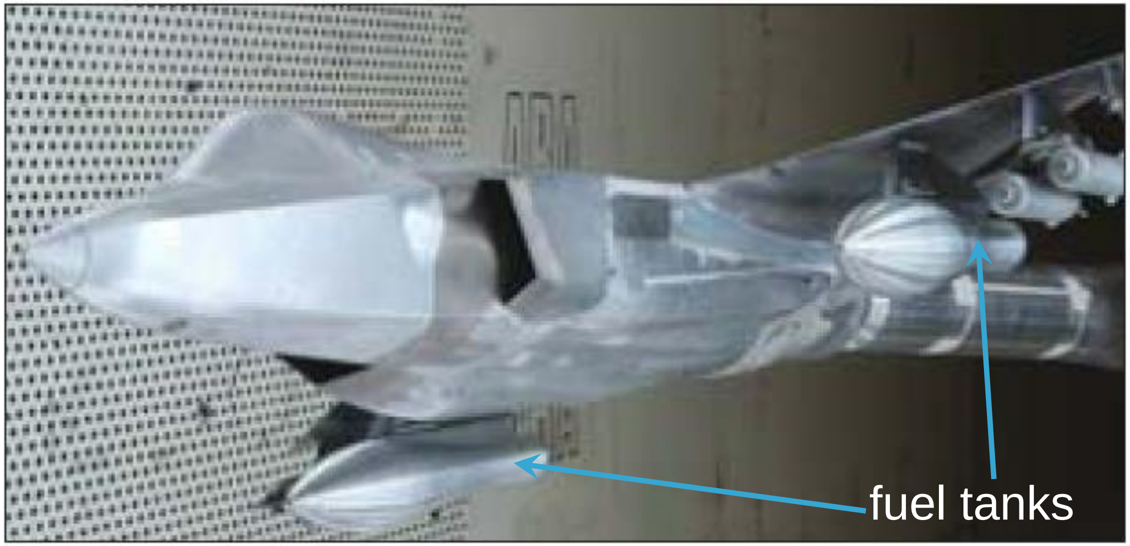
- Alternative energy:
  - Wind turbine and farm uncertainty
  - Hydropower optimization
- Nuclear energy and safety
  - NASA launch safety
  - Nuclear reactor analysis
- Climate:
  - Ice sheet model calibration
  - UQ for community climate models
- DoD applications
  - Shock physics
  - Aeroheating

Optimization and Calibration

- ***Goal-oriented: find best performing design, scenario, or model agreement***
  - Identify system designs with maximal performance
  - Determine operational settings to achieve goals
  - Minimize cost over system designs/operational settings
  - Identify best/worst case scenarios
  - Calibration: determine parameter values that maximize agreement between simulation and experiment



Calibrate parameters to match experimental stress observations



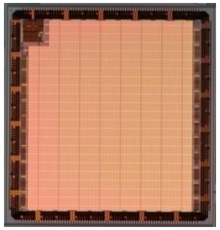
- Lockheed Martin CFD code to model F-35 performance
- Find fuel tank shape with constraints to minimize drag, yaw while remaining sufficiently safe and strong

Sensitivity Analysis

- ***Which are the most influential parameters?***
- Understand code output variations as input factors vary to identify most important variables and their interactions
  - Identify key model characteristics/trends, robustness
  - Focus resources for data gathering, model/code development, characterizing uncertainties
  - Screening: reduce variables further UQ or optimization analysis
  - Construct surrogate models from sim data
- Dakota SA formalizes and generalizes one-off parameter variation / sensitivity studies you're likely already doing
- Provides richer global sensitivity analysis methods

	Vdd Metrics	
	node max	node avg
METAL1	0.96	0.82
METAL2	0.11	0.04
METAL3	0.10	0.05
METAL4	0.80	0.81
METAL5	0.86	0.91
VIA1	0.71	0.66
VIA2	0.80	0.76
VIA3	0.57	0.60
VIA4	0.91	0.94
CONTACT	0.21	0.13
polyc	0.04	0.05

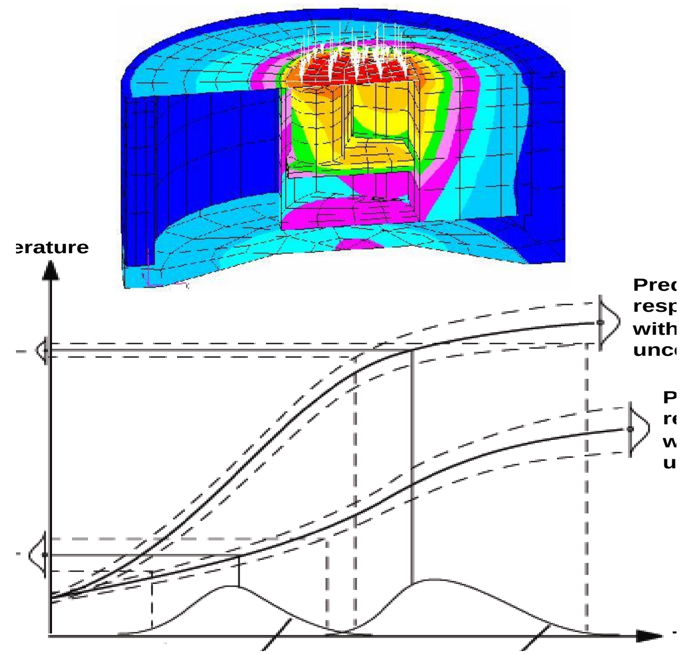
correlation coefficients



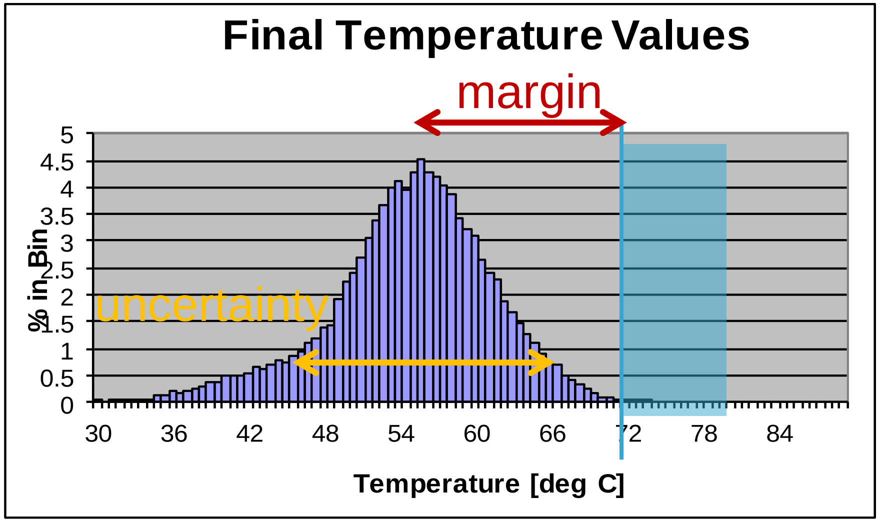
Dakota + Xyce SA for CMOS7 ViArray performance during photocurrent event

Uncertainty Quantification

- ***Assess effect of input parameter uncertainty on model outputs***
  - Determine mean or median performance of a system
  - Assess variability in model response
  - Find probability of reaching failure/success criteria (reliability)
  - Assess range/intervals of possible outcomes
- UQ simulation ensembles also used for validation with experimental data

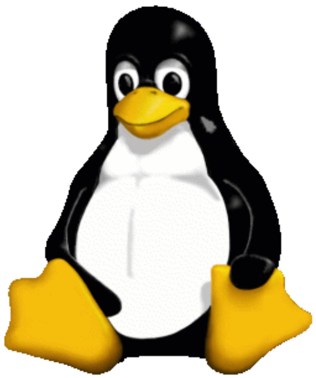


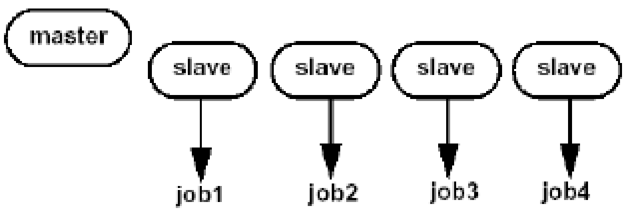
- Device subject to heating, e.g., modeled with heat transfer code
- Uncertainty in composition/ environment (thermal conductivity, density, boundary)
- Make risk-informed decisions for strong link / weak link thermal race



Simulation management and Parallelism

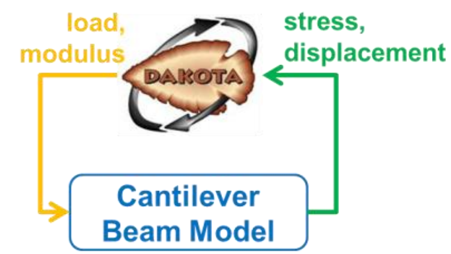
- Runs in most commonly-used computing environments
  - Desktop: Mac, Linux, Windows
  - HPC: Linux clusters, IBM Blue Gene/P and /Q, IBM AIX
- Exploits available concurrency at multiple levels. E.g.
  - Multiprocessor simulations
  - Multiple simulations per response
  - Samples in a parameter study
  - Optimizations from multiple starting points
- File management features, including
  - Work directories to partition analysis files
  - Template directories to share files common to all analyses





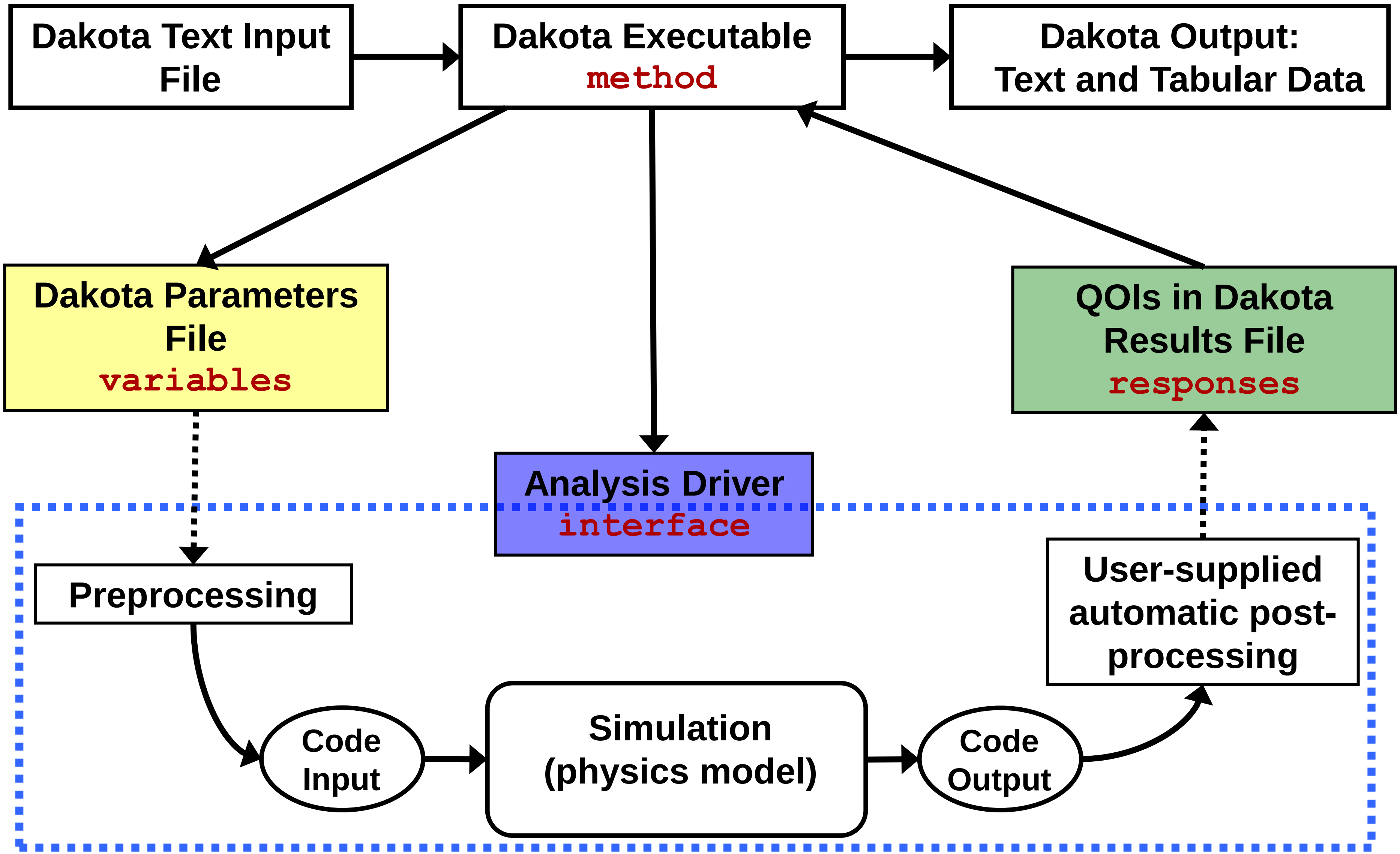
Steps to Get Started with Dakota

1. Define analysis goals; understand how Dakota helps, learn about and select from possible methods
2. Access Dakota and understand help resources
3. Automated workflow: create a workflow so Dakota can communicate with your simulation
   - Parameters to model, responses from model to Dakota
   - Typically requires programming (Python, Perl, Shell, Matlab, C, C++, Java, Fortran, …)
   - Workflow reusable; crosscuts Dakota analysis types

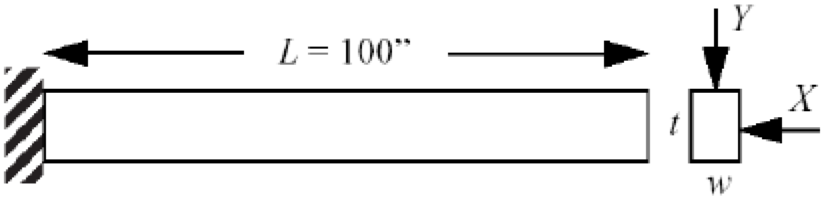


4. Dakota input file: Using your favorite text editor, configure Dakota to exercise the workflow to meet your goals
   - Tailor variables, methods, responses to analysis goals
   - Syntax documented in Reference Manual
5. Run Dakota: command-line; text input / output

Dakota Execution and Information Flow



Cantilever Beam Application Example



Constants

L: length (inches)

D_0: max displacement

Parameters (Variables)

w: width (in.)

t: thickness (in.)

R: yield stress (lb./in^2)

E: Young's modulus (lb./in^2)

X: horizontal load (lb.)

Y: vertical load (lb.)

QOIs (Responses)

A: area

Sc: stress constraint

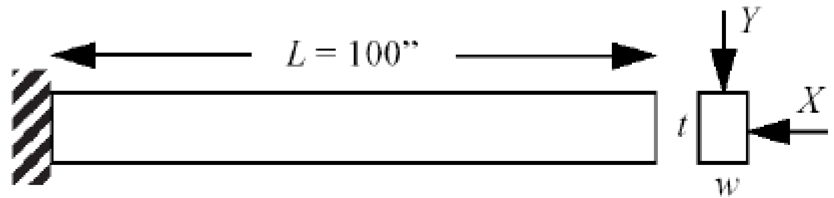
Dc: displacement constraint

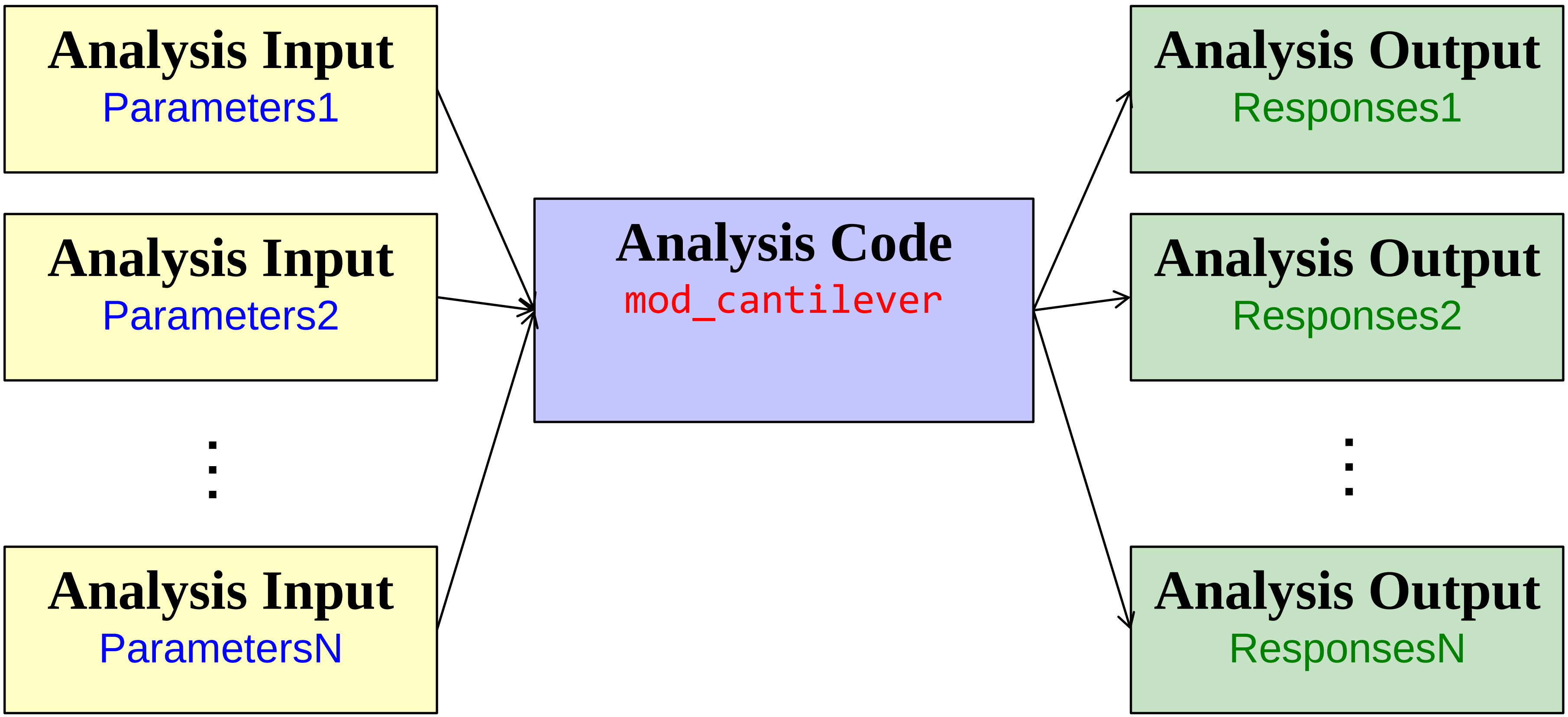
$$A = wt$$ *(surrogate for weight)*

$$Sc = \frac{600}{wt^2}Y + \frac{600}{w^2 t}X - R$$

$$Dc = \frac{4L^3}{Ewt}\sqrt{\left(\frac{Y}{t^2}\right)^2 + \left(\frac{X}{w^2}\right)^2} - D_o$$

Dakota Concurrent Interaction with the Cantilever Beam Analysis Code

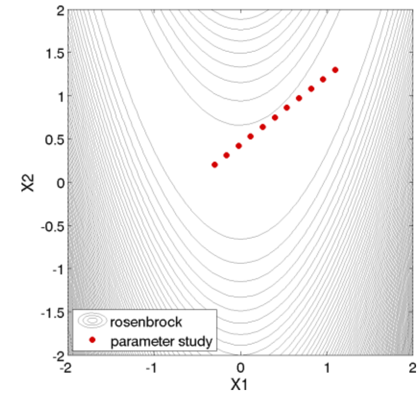


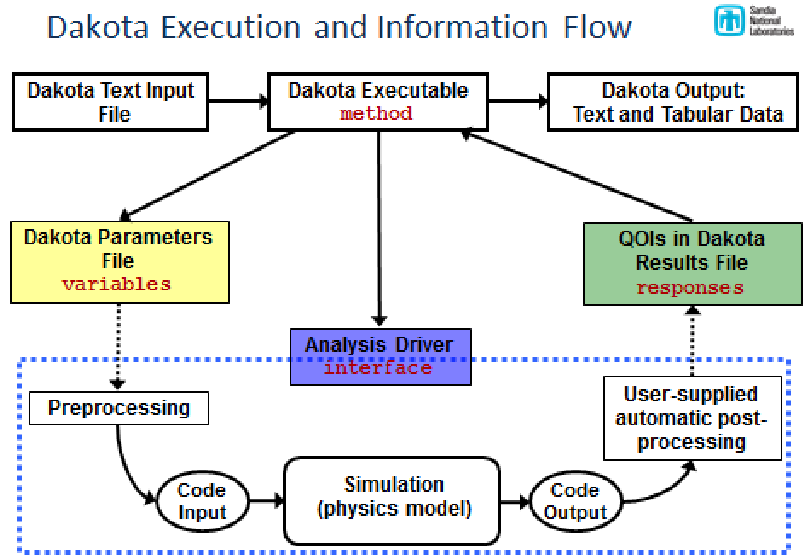


Sample Dakota Input File: Vector Parameter Study

```
strategy
  single_method
  graphics, tabular_graphics_data

method
  vector_parameter_study
  num_steps = 10
  final_point  4.0 4.0 40000. 29.E+6 500. 1000.

variables
  continuous_design = 2
    initial_point  1.0          1.0
    descriptors    'beam_width' 'beam_thickness'
  continuous_state = 4
    initial_state  40000. 29.E+6 500. 1000.
    descriptors    'R'    'E'    'X'   'Y'

interface
  direct
  analysis_driver = 'mod_cantilever'

responses
  num_objective_functions = 3
  descriptors = 'area' 'stress' 'displacement'
  no_gradients
  no_hessians
```

Define Flow / Algorithm





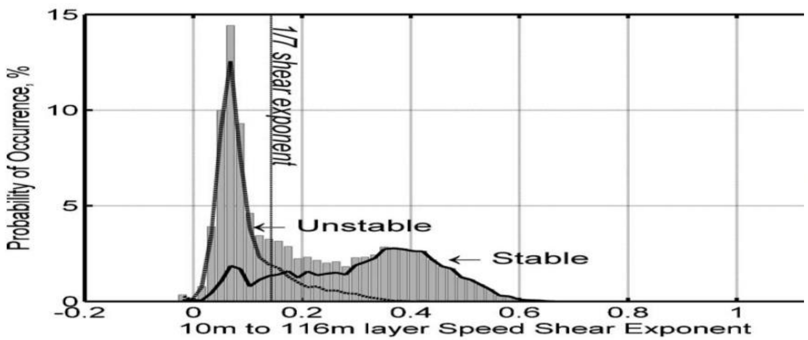
Define Problem / Mapping

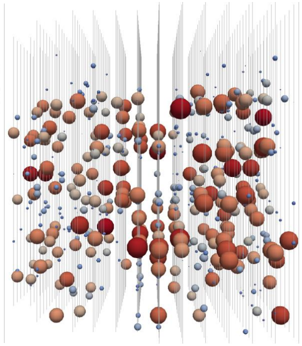
Getting Started and Getting Help

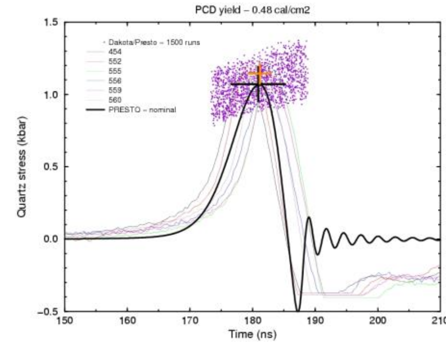
- Supported platforms: Linux/Unix, Mac OS X, Windows
- Dakota web page: http://dakota.sandia.gov
  - Extensive documentation (user, reference, developer)
  - Support mailing lists / archives
  - Software downloads: official releases and stable development version (freely available worldwide via GNU LGPL)
- User's Manual, Chapter 2: Tutorial with example input files
- Support:
  - dakota-users@software.sandia.gov
    (Dakota team and user community)

Exceptional service in the national interest

Sandia National Laboratories







Dakota Sensitivity Analysis and Uncertainty Quantification, with Examples

Adam Stephens, Laura Swiler

Dakota Clinic at CSDMS May 22, 2014

U.S. DEPARTMENT OF ENERGY NNSA National Nuclear Security Administration

SAND2014-4255C

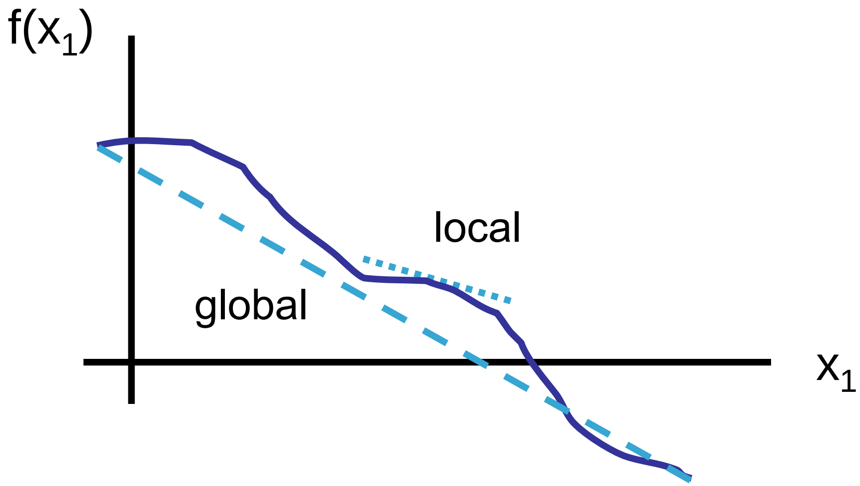
Dakota Sensitivity Analysis (SA)

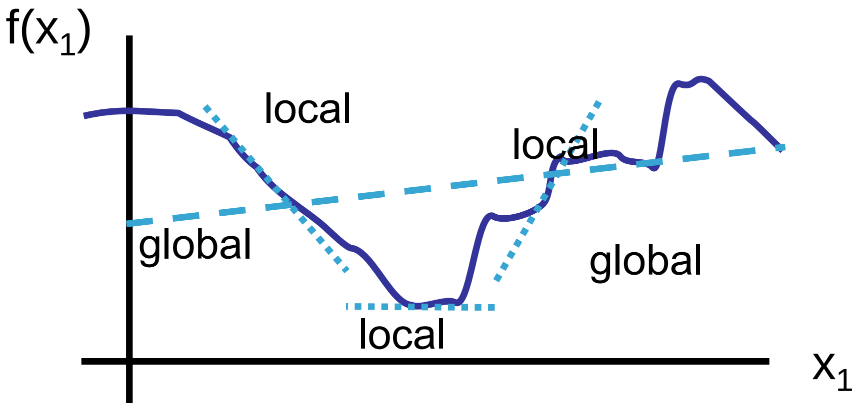
- SA goals and examples
- Global SA approaches and metrics available in Dakota
- Select Dakota examples for parameter studies and global SA

Why Perform Sensitivity Analysis?

- What? Understand code output variations as input factors vary
- Why? Identify most important variables and their interactions
  - Identify key model characteristics: smoothness, nonlinear trends, robustness
  - Provide a focus for resources
    - Data gathering and model development
    - Code development
    - Uncertainty characterization
  - Screening: Identity the most important variables, down-select for further UQ or optimization analysis
  - Can have the side effect of identifying code and model issues
  - Data can be used to construct surrogate models
- Dakota SA formalizes and generalizes one-off sensitivity studies you're likely already doing
- Provides richer global sensitivity analysis methods

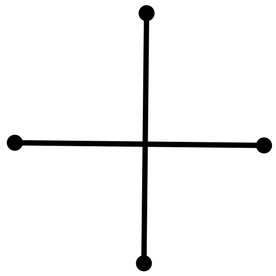
Sensitivity Analysis: Influence of Inputs on Outputs





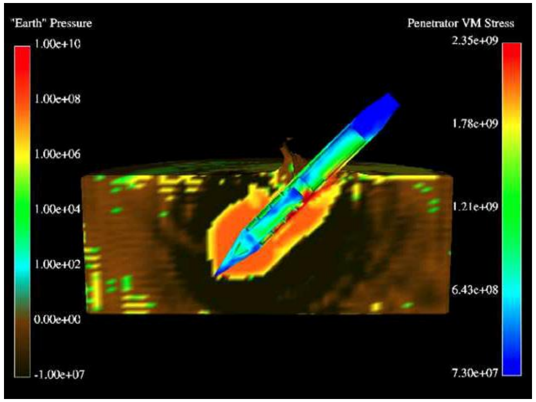
Assess variations in f(x1) due to (small or large) perturbations in x1.

- Local sensitivities
  - Partial derivatives at a specific point in input space.
  - Given a specific x1, what is the slope at that point?
  - Can be estimated with finite differences
- **Global sensitivities**
  - Found via sampling and regression.
  - What is the general trend of the function over all values of x1?
  - Typically consider inputs uniformly over their whole range



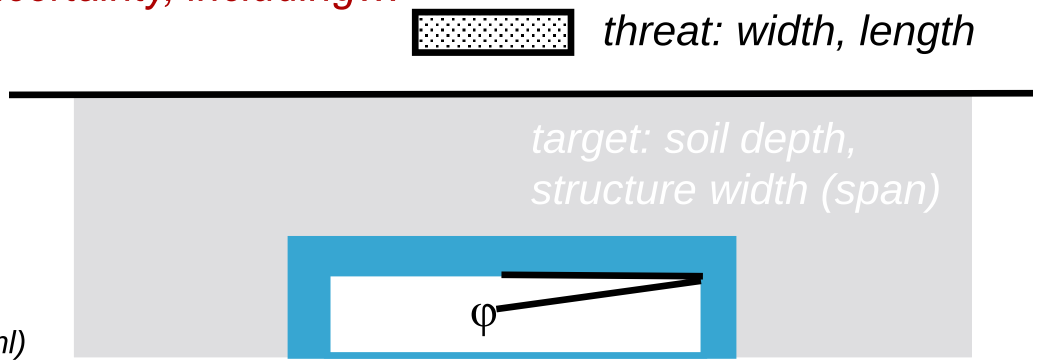
many already do basic SA; perturb from nominal, see effect

Global Sensitivity Analysis Example: Earth Penetrator

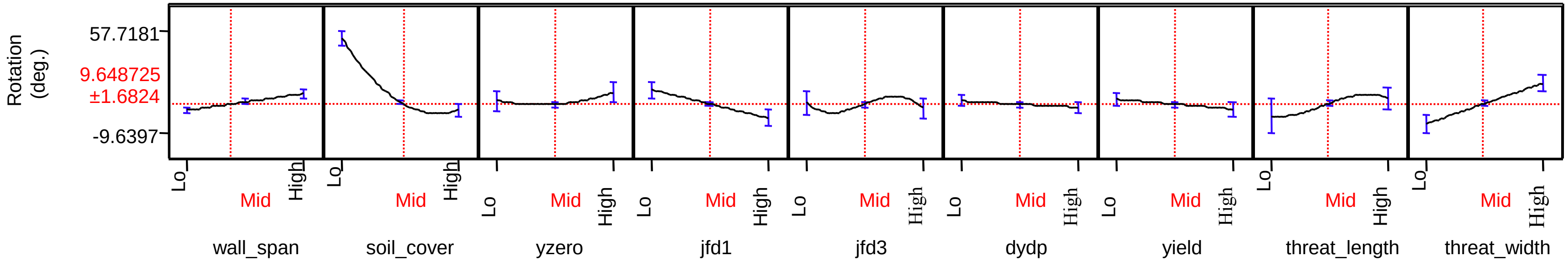


Notional model for illustration purposes only (http://www.sandia.gov/ASC/library/fullsize/penetrator.html)

12 parameters describing target & threat uncertainty, including…

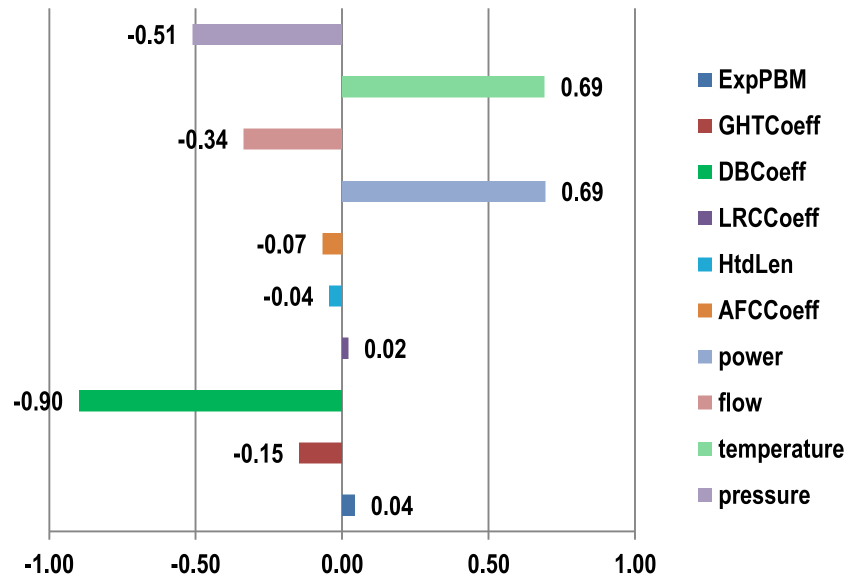


- Underground target with external threat: assess sensitivity in target response to target construction and threat characteristics
- Response: angular rotation (ϕ) of target roof at mid-span
- Analysis: CTH Eulerian shock physics code; JMP stats
- Revealed most sensitive input parameters and nonlinear relationships

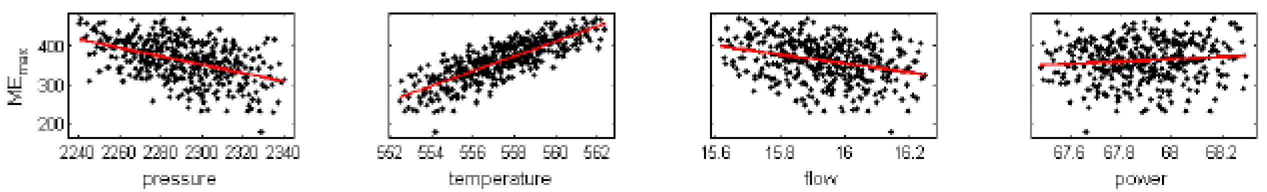


Global SA Example:
Nuclear Reactor Thermal-Hydraulics Model

parameter influence on number of boiling sites

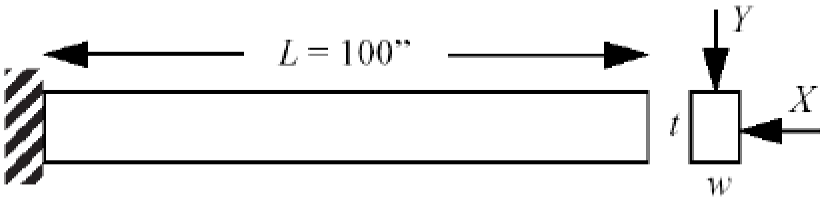


- Assess parameter influence on boiling rate, a key crud predictor
- Dakota correlation coefficients: strong influence of core operating parameters (pressure more important than previously thought)
- Dittus-Bolter correlation model may dominate model form sensitivities (also nonlinear effects of ExpPBM)
- Scatter plots help visualize trend in input/output relationships



sensitivity of mass evaporation rate (max) to operating parameters

Cantilever Beam Sensitivity Analysis



- What are some global sensitivity analysis questions you could ask for the cantilever beam?
- What kinds of bounds or variable characterizations would you use?
- What might you expect the results to be?
- Beam computational model:

weight (*area = w*t*)

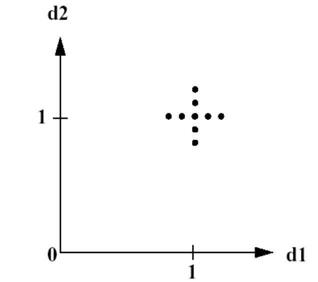
$$stress = \frac{600}{wt^2}Y + \frac{600}{w^2t}X \le R$$

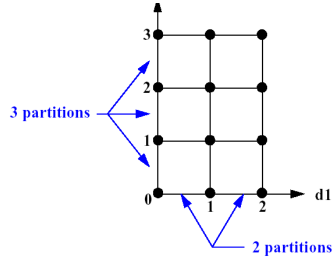
$$displacement = \frac{4L^3}{Ewt}\sqrt{\left(\frac{Y}{t^2}\right)^2 + \left(\frac{X}{w^2}\right)^2} \le D_0$$

Given values of *w, t, R, E, X, Y,* Dakota's `mod_cantilever` driver computes *area, stress-R, displacement-*D_0

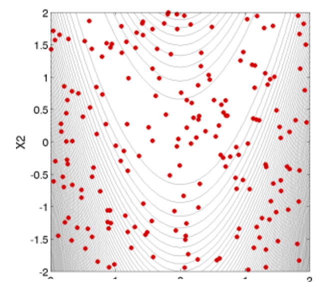
Global Sensitivity Analysis in Dakota

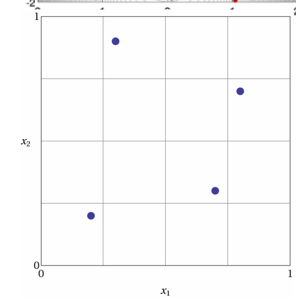
- Assess effect of input variables considered jointly over their whole range. Dakota process:
  - Specify variables: lower and upper bounds
  - Specify method: e.g., uniform random sampling
  - Specify responses: compute response value at each sample point
  - Run Dakota and analyze input/output relationships





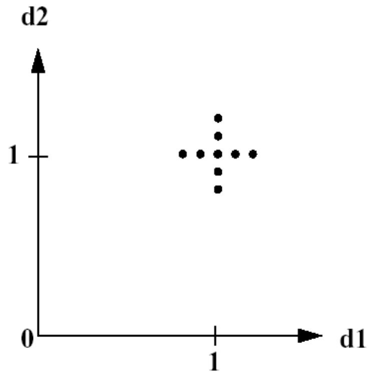
- Sample designs (methods) available:
  - Parameter studies: list, centered, grid, vector, user
  - Random sampling: Monte Carlo, Latin hypercube, Quasi-MC, CVT
  - DOE/DACE: Full-factorial, orthogonal arrays, Box-Behnken, CCD
  - Morris one-at-a-time
  - Sobol indices via variance-based decomposition, polynomial chaos



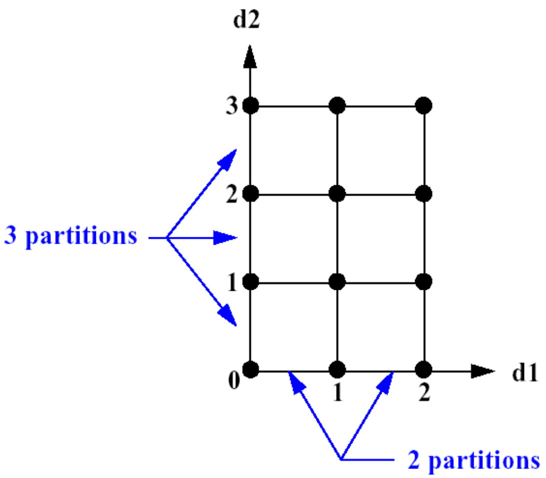


- Metrics: trends, correlations, main/interaction effects, Sobol indices, importance factors/local sensitivities

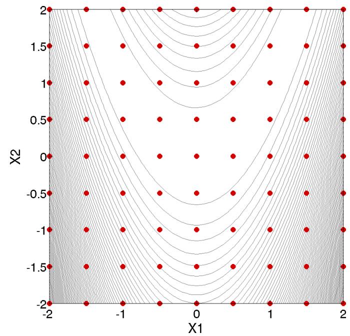
Basic Dakota SA for Cantilever: Centered and Grid Parameter Studies



- Start at nominal values, perturb up and down in each coordinate direction
- Specify the parameter variations, which responses to study



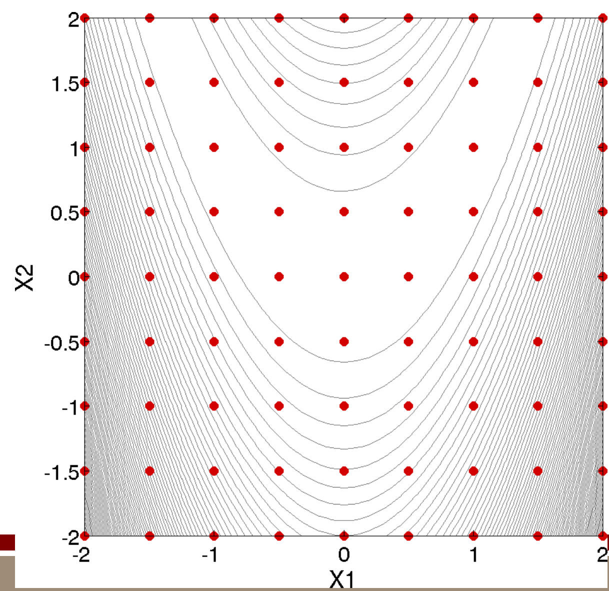
- Construct grid with a certain number of partitions in each dimension
- What are benefits/drawbacks of these methods?



Example: uniform grid over [-2.0, 2.0]

Exercise: Multi-dimensional Parameter Study

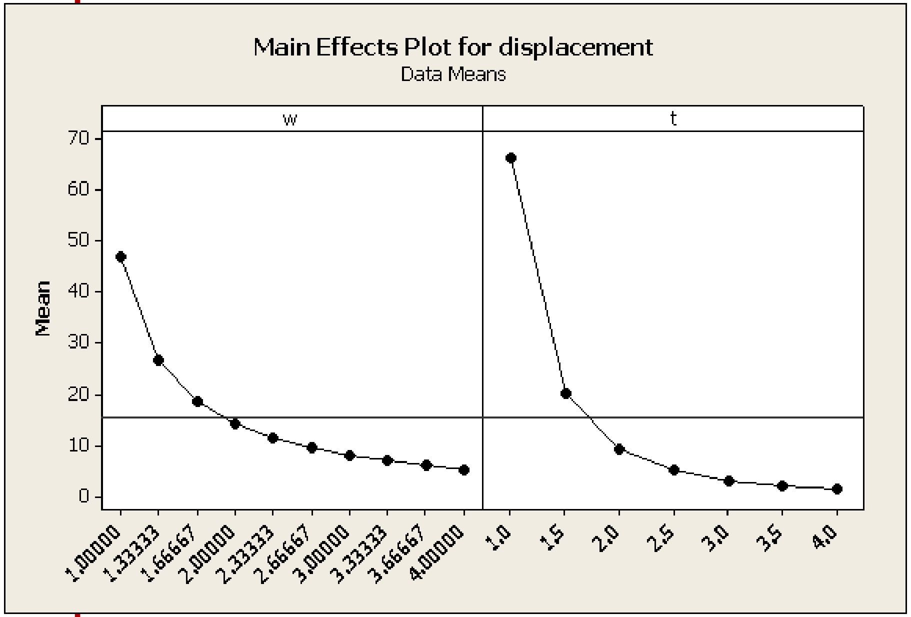
- Goal: understand how responses *area, stress, and displacement* vary with respect to the inputs *w* and *t* on a grid of points.
- *Exercise: run the mod_cantilever computational model at a grid of points over [1.0, 4.0] using the* `multidim_parameter_study` *method*
- Try 9 points in one dimension, 6 in the other
- See method and variable commands in Dakota reference manual



Example: uniform grid over [-2.0, 2.0]

Dakota Input File and Results: Cantilever Multi-dimensional Parameter Study

```
# examples/cantilever/cantilever_grid.in
strategy,
single_method
  graphics,tabular_graphics_data

method,
multidim_parameter_study
  partitions = 9 6

variables,
active design
continuous_design = 2
  lower_bounds   1.0      1.0
  upper_bounds   4.0      4.0
  descriptors       'w'      't'
continuous_state = 4
  lower_bounds  40000. 29.E+6 500. 1000.
  upper_bounds  40000. 29.E+6 500. 1000.
  descriptors      'R'    'E'    'X'    'Y'

interface,
direct
  analysis_driver = 'mod_cantilever'

responses,
num_objective_functions = 3
    response_descriptors = 'area' 'stress' 'displacement'
no_gradients
no_hessians
```

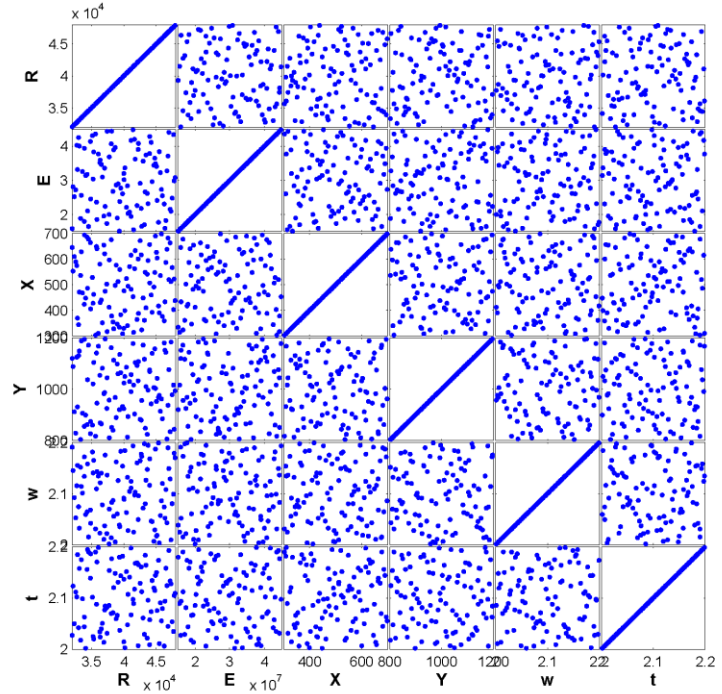


Dakota tabular data plotted with Minitab

What are benefits/drawbacks?

Workhorse SA Method: Random Sampling

- Generate space filling design (typically Monte Carlo or Latin hypercube with samples = 2x or 10x number of variables)
- Run model at each point
- Analyze input/output relationships with
  - Correlation coefficients
    - Simple correlation: strength and direction of a linear relationship between variables
    - Partial correlation: like simple correlation but adjusts for the effects of the other variables
    - Rank correlations: simple and partial correlations performed on "rank" of data
  - Regression and resulting coefficients
  - Variance-based decomposition
  - Importance factors



Two-dimensional projections of LHD for Cantilever
(plotted with Matlab)

Dakota Input File: Cantilever LHS Study

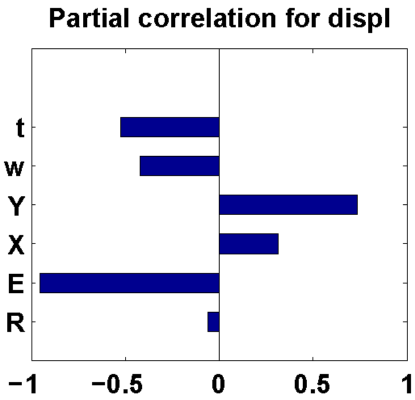
```
# Dakota INPUT FILE – examples/cantilever/cantilever_sa.in
strategy,
    single_method
    tabular_graphics_data
    graphics
method,
    sampling
    sample_type lhs
    seed =52983
    samples = 100
variables,
    uniform_uncertain = 6
      upper_bounds   48000   45.E+6  700.  1200.  2.2    2.2
      lower_bounds    32000.   15.E+6  300.    800.   2.0     2.0
      descriptors  'R' 'E' 'X' 'Y' ’w' ’t'
interface,
    direct
    analysis_driver = 'mod_cantilever'
responses,
    num_response_functions = 3
    response_descriptors = ’area' 'stress' 'displacement'
    no_gradients
    no_hessians
```

Global Sampling Results for Cantilever

Partial Correlations for Cantilever

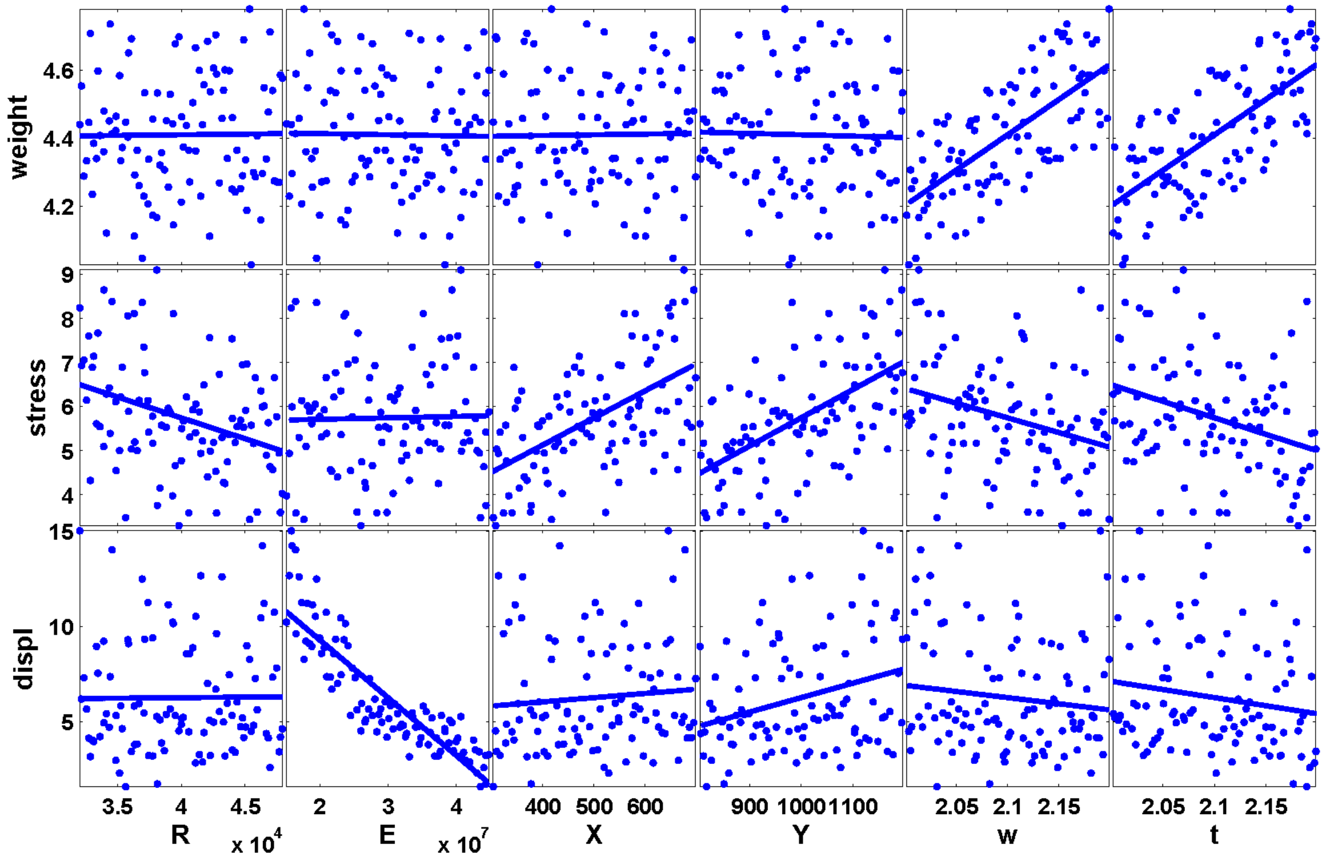
	area	stress	displ
R	0.14	-0.99	-0.06
E	-0.03	0.02	-0.95
X	-0.01	1.00	0.31
Y	0.05	1.00	0.74
w	1.00	-0.98	-0.42
t	1.00	-0.99	-0.52

correlation coefficients from Dakota console output (colored w/ Excel)



(plotted with Matlab)

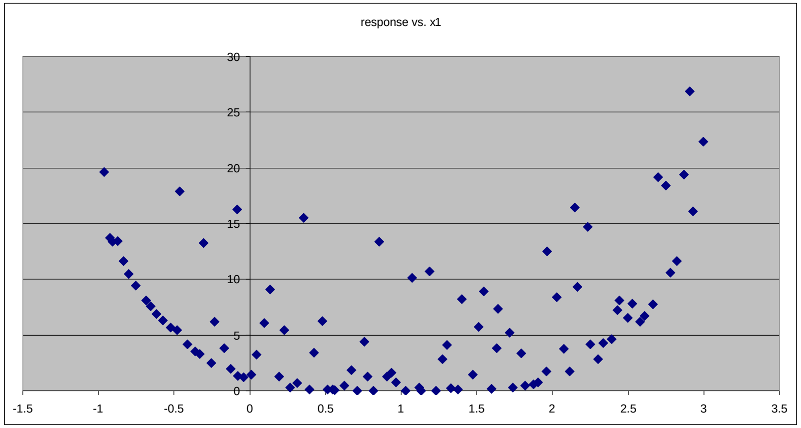
Dakota tabular data plotted in Matlab (can used Mintab, JMP, Excel, etc.)



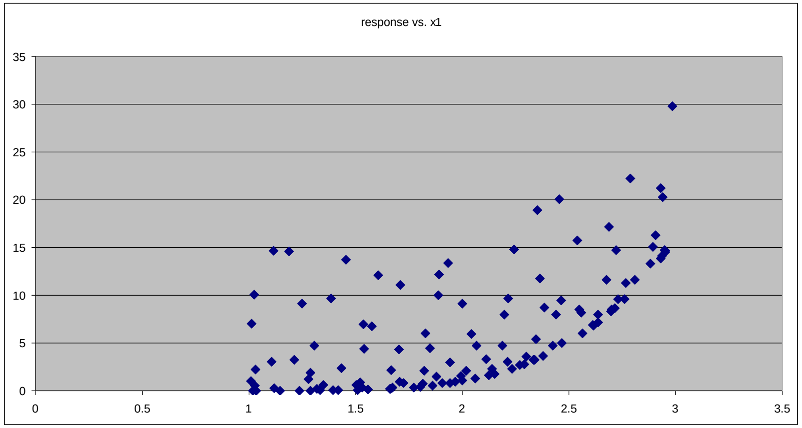
Limitations of SA methods

- Results are very dependent on the input bounds (example below)
- Grid studies are nice for generating plots and visualization surfaces but do not scale well with input dimension
- Trying to assess global trends with a limited number of samples: can miss local behavior

Bounds = [-1, 3]



Bounds = [1, 3]



Other SA Approaches Require Changing Method

- Dakota Reference Manual guides in specifying keywords

```
method,
sampling
    sample_type lhs
    seed =52983
    samples = 100
```

LHS Sampling

```
method,
    dace oas
    main_effects
    seed =52983
    samples = 500
```

Main Effects Analysis using Orthogonal Arrays

```
method,
sampling
    sample_type lhs
    seed =52983
    samples = 500
    variance_based_decomp
```

Variance-based Decomposition using LHS Sampling

```
method,
    psuade_moat
    partitions = 3
    seed =52983
    samples = 100
```

Morris One-At-a-Time

Dakota Sensitivity Analysis Summary

- What? Understand code output variations as input factors vary; main effects and key parameter interactions.
- Why? Identify most important variables and their interactions
- How? What Dakota methods are relevant? What results?

Category	Dakota method names	univariate trends	correlations	modified mean, s.d.	main effects Sobol inds.	importance factors / local sensis
Parameter studies	centered, vector, list	P				
	grid		D		P	
Sampling	sampling, dace lhs, dace random, fsu_quasi_mc, fsu_cvt with variance_based_decomp...	P	D		D	
DACE (DOE-like)	dace {oas, oa_lhs, box_behnken, central_composite}		D		D	
MOAT	psuade_moat			D		
PCE, SC	polynomial_chaos, stoch_collocation				D	D
Mean value	local_reliability					D

multi-purpose!

D: Dakota
P: Post-processing (3^{rd} party tools)

- Also see Dakota Usage Guidelines in User's Manual

Sensitivity Analysis References

- Saltelli A., Ratto M., Andres T., Campolongo, F., et al., Global Sensitivity Analysis: The Primer, Wiley, 2008.
- J. C. Helton and F. J. Davis. Sampling-based methods for uncertainty and sensitivity analysis. Technical Report SAND99-2240, Sandia National Laboratories, Albuquerque, NM, 2000.
- Sacks, J., Welch, W.J., Mitchell, T.J., and Wynn, H.P. Design and analysis of computer experiments. Statistical Science 1989; 4:409–435.
- Oakley, J. and O'Hagan, A. Probabilistic sensitivity analysis of complex models: a Bayesian approach. J Royal Stat Soc B 2004; 66:751–769.

- Dakota User's Manual
  - Parameter Study Capabilities
  - Design of Experiments Capabilities/Sensitivity Analysis
  - Uncertainty Quantification Capabilities (for MC/LHS sampling)
- Corresponding Reference Manual sections

Dakota Uncertainty Quantification (UQ)

- UQ goals and examples
- Select Dakota examples for UQ:
  - Monte Carlo sampling
  - Local and global reliability
  - Polynomial chaos expansions / stochastic collocation
  - *Mixed aleatory-epistemic approaches*
  - *Probabilistic design*

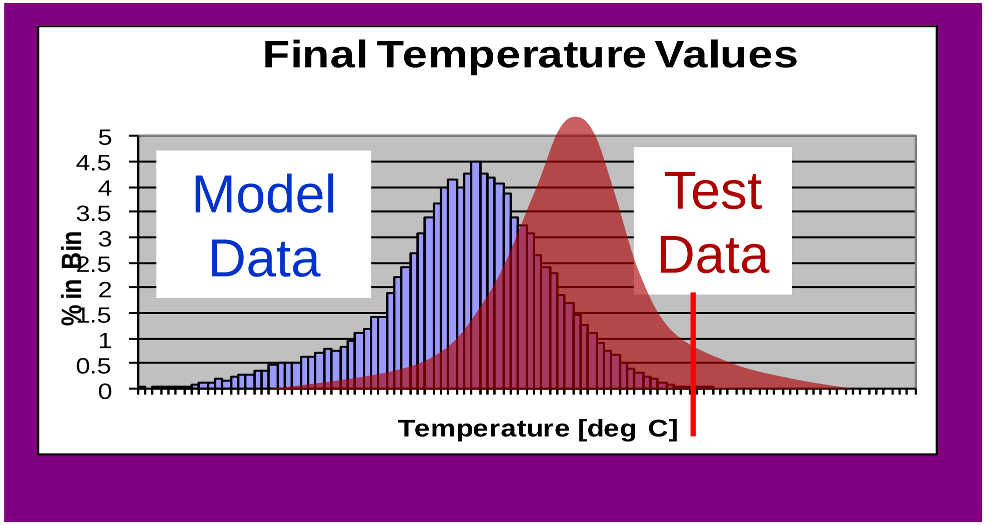
Why Perform Uncertainty Quantification?

- What? Determine variability, distributions, statistics of code outputs, given uncertainty in input factors
- Why? Assess likelihood of typical or extreme outcomes. Given input uncertainty…
  - Determine mean or median performance of a system
  - Assess variability in model response
  - Find probability of reaching failure/success criteria (reliability metrics)
  - Assess range/intervals of possible outcomes
- Assess how close uncertainty-endowed code predictions are to
  - Experimental data
    (validation, is model sufficient *for the intended application?)*
  - Performance expectations or limits
    (quantification of margins and uncertainties)

Many Potential Uncertainties in Simulation and Validation

- physics/science parameters
- statistical variation, inherent randomness
- model form / accuracy
- material properties
- manufacturing quality
- operating environment, interference
- initial, boundary conditions; forcing
- geometry / structure / connectivity
- experimental error (measurement error, measurement bias)
- numerical accuracy (mesh, solvers); approximation error
- human reliability, subjective judgment, linguistic imprecision

Parameterized…



The effect of these on model outputs should be integral to an analyst's deliverable: *best estimate PLUS uncertainty!*

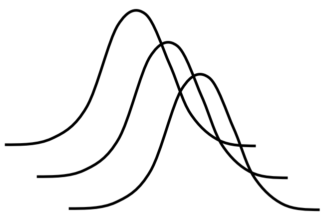
Forward Parametric Uncertainty Quantification

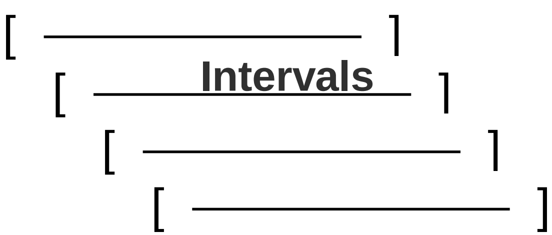
- **Identify and characterize uncertain variables (may not be normal, uniform)**
- ***Forward propagate: quantify the effect that (potentially correlated) uncertain (nondeterministic) input variables have on model output:***

<u>Input Variables *u*</u> (physics parameters, geometry, initial and boundary conditions) → **Computational Model** → **Variable Performance Measures *f(u)***

Uncertainties on inputs

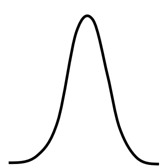
- Parameterized distributions: normal, uniform, gumbel, etc.
- Means, standard deviations
- PDF, CDF from data
- Intervals
- Belief structures





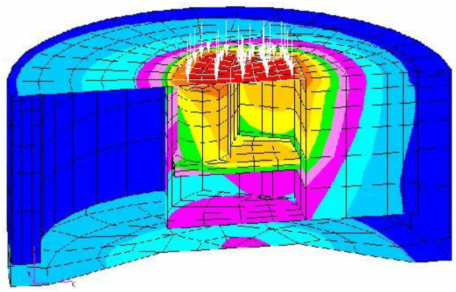
Uncertainties on outputs

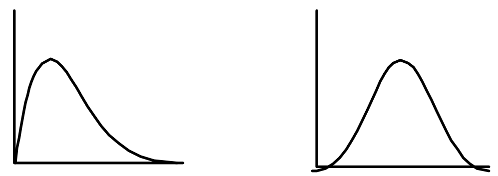
- Means, standard deviations
- Probabilities
- Reliabilities
- PDF, CDF
- Intervals
- Belief, plausibility

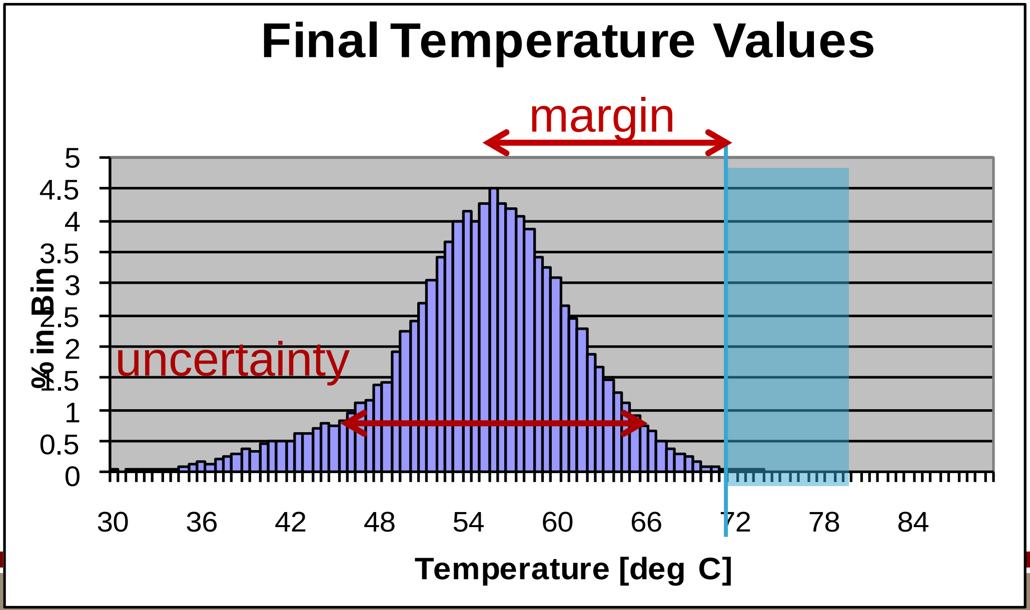


Example: Thermal Uncertainty Quantification

- Device subject to heating (experiment or computational simulation)
- Uncertainty in composition/ environment (thermal conductivity, density, boundary), parameterized by $u_1, \ldots, u_N$
- Response temperature $f(u)=T(u_1, \ldots, u_N)$ calculated by heat transfer code







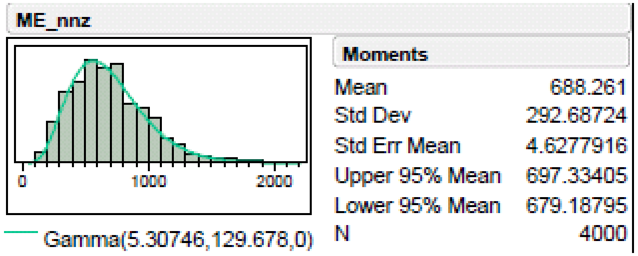
Given distributions of $u_1,\ldots,u_N$, UQ methods calculate statistical info on outputs:

- Mean(T), StdDev(T), Probability($T \geq T_{critical}$)
- Probability distribution of temperatures
- *Correlations (trends) and sensitivity of temperature*

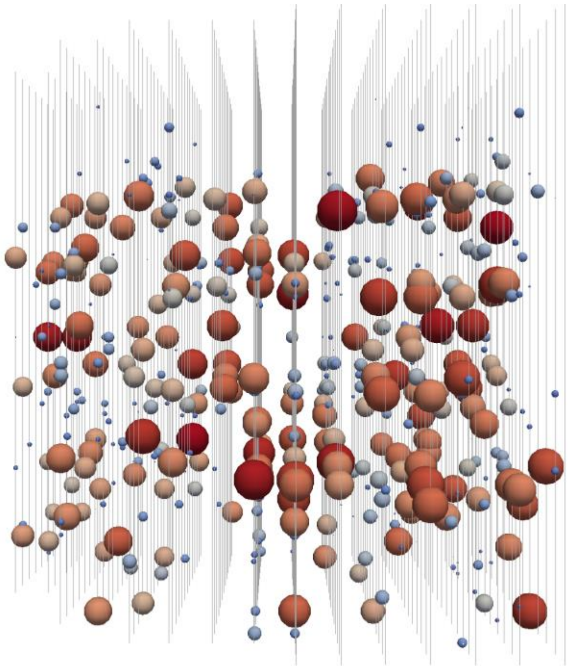
Example: Uncertainty in Boiling Rate in Nuclear Reactor Core

Method	ME_nnz		ME_meannz		ME_max	
	Mean	Std Dev	Mean	Std Dev	Mean	Std Dev
LHS (40)	651.225	297.039	127.836	27.723	361.204	55.862
LHS (400)	647.33	286.146	127.796	25.779	361.581	51.874
LHS (4000)	688.261	292.687	129.175	25.450	364.317	50.884
PCE (Θ(2))	687.875	288.140	129.151	25.7015	364.366	50.315
PCE (Θ (3))	688.083	292.974	129.231	25.3989	364.310	50.869
PCE (Θ (4))	688.099	292.808	129.213	25.4491	364.313	50.872

mean and standard deviation of key metrics

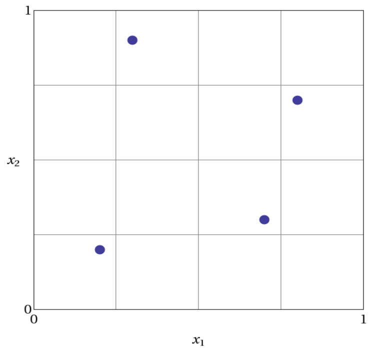


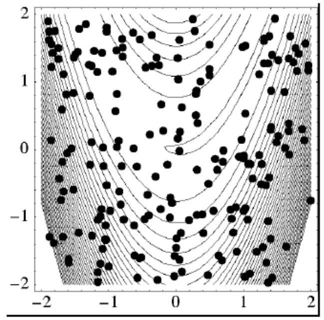
normally distributed inputs need not give rise to normal outputs…

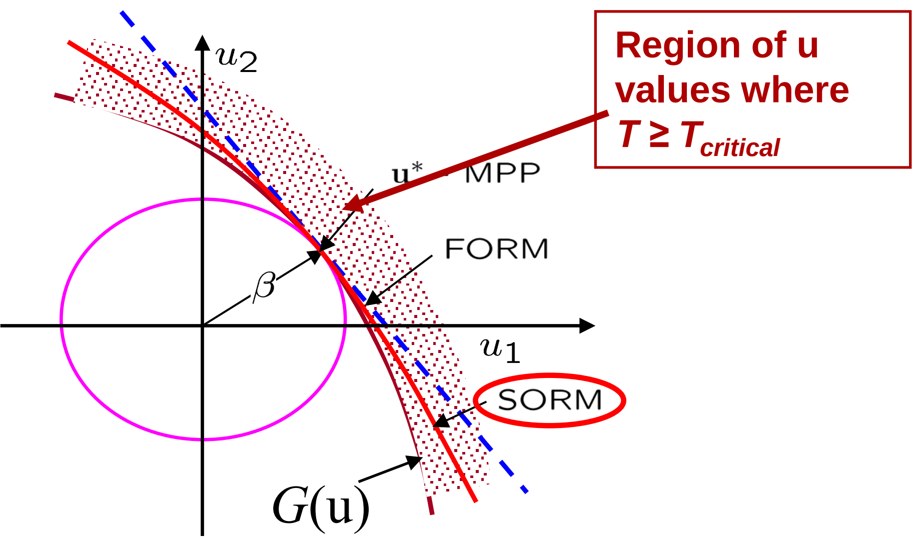


anisotropic uncertainty distribution in boiling rate throughout quarter core model

Three Core Dakota UQ Methods



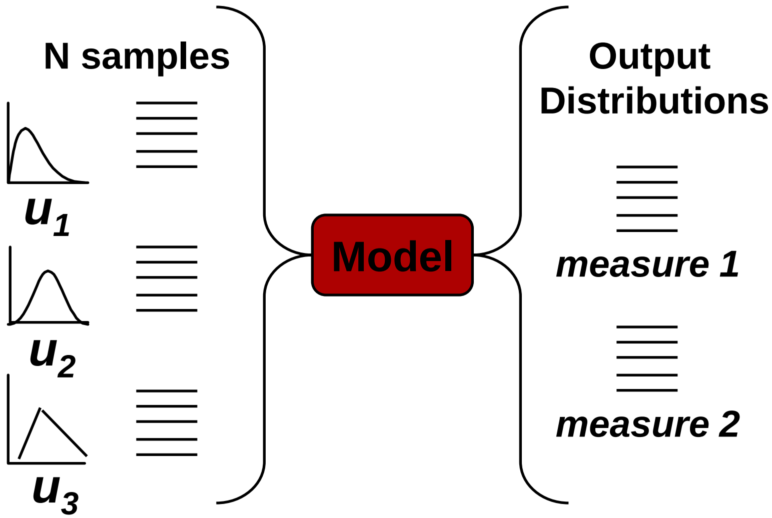




- Sampling (Monte Carlo, Latin hypercube): robust, easy to understand, slow to converge / resolve statistics
- Reliability: good at calculating probability of a particular behavior or failure / tail statistics; efficient, some methods are only local
- Stochastic Expansions (PCE/SC global approximations): efficient tailored surrogates, statistics often derived analytically, *far more efficient than sampling for reasonably smooth functions*

Black-box UQ Workhorse: Random Sampling Methods

Given distributions of $u_1,\ldots,u_N$, sampling-based methods calculate sample statistics, e.g., on temperature $T(u_1,\ldots,u_N)$:



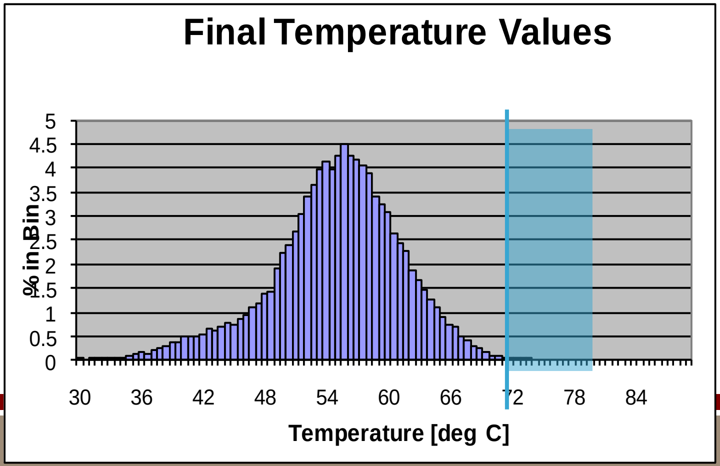
- **sample mean**

$$\bar{T} = \frac{1}{N}\sum_{i=1}^{N} T(u^i)$$

- **sample variance**

$$T_{\sigma^2} = \frac{1}{N}\sum_{i=1}^{N}\left[T(u^i) - \bar{T}\right]^2$$

- **full PDF(probabilities)**



- Monte Carlo sampling, Quasi-Monte Carlo
- Centroidal Voroni Tessalation (CVT)
- Latin hypercube (stratified) sampling: better convergence; stability across replicates

Robust, but slow convergence: $O(N^{-1/2})$, independent of dimension *(in theory)*

Example:
Cantilever Beam UQ with Sampling

- Dakota study with LHS
- Determine mean system response, variability, margin to failure given
  - Yield stress R ~ Normal(40000, 2000)
  - Young's modulus E ~ Normal(2.9e7, 1.45e6)
  - Horizontal load X ~ Normal(500, 100)
  - Vertical load Y ~ Normal(1000, 100)
  - *(Dakota supports a wide range of distribution types)*
- Hold width and thickness at 2.5
- Compute with respect to thresholds with `probability_levels` or `response_levels`
  - What is the probability(stress < 20000)?

Dakota Input:
LHS Sampling for Cantilever Beam

```
method,
    sampling
      sample_type lhs
      samples = 10000  seed = 12347
      num_probability_levels = 0 17 17
      probability_levels =
      .001 .01 .05 .1 .15 .2 .3 .4 .5 .6 .7 .8 .85 .9 .95 .99 .999
       .001 .01 .05 .1 .15 .2 .3 .4 .5 .6 .7 .8 .85 .9 .95 .99 .999
      cumulative distribution

variables,
      continuous_design = 2
         initial_point 2.5 2.5
         upper_bounds 10.0 10.0
         lower_bounds 1.0 1.0
         descriptors ’w' ‘t'
      normal_uncertain = 4
         means = 40000. 29.E+6 500. 1000.
         std_deviations = 2000. 1.45E+6 100. 100.
         descriptors = 'R' 'E' 'X' 'Y'

responses,
      num_response_functions = 3
      descriptors = 'area' 'stress' 'displacement'
      no_gradients
      no_hessians
```

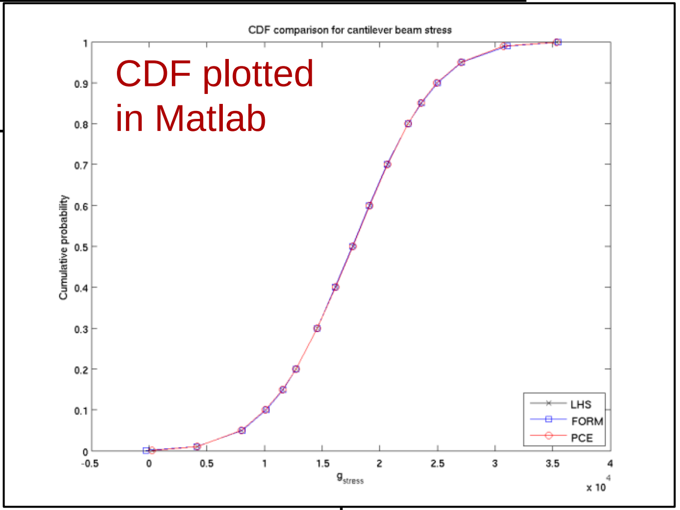
Dakota Output:
LHS Sampling for Cantilever Beam

- Moments and confidence intervals

```
Statistics based on 10000 samples:

Moment-based statistics for each response function:
                            Mean           Std Dev          Skewness          Kurtosis
          area  6.2500000000e+00  0.0000000000e+00  0.0000000000e+00 -3.0000000000e+00
        stress  1.7599759864e+04  5.7886440706e+03 -2.2153567379e-02 -2.9903700042e+00
  displacement  1.7201261575e+00  4.0670385498e-01  1.7796424852e-01 -2.9899089153e+00

95% confidence intervals for each response function:
                    LowerCI_Mean      UpperCI_Mean    LowerCI_StdDev    UpperCI_StdDev
          area  6.2500000000e+00  6.2500000000e+00  0.0000000000e+00  0.0000000000e+00
        stress  1.7486290789e+04  1.7713228938e+04  5.7095204696e+03  5.8700072185e+03
  displacement  1.7121539434e+00  1.7280983716e+00  4.0114471657e-01  4.1242034152e-01
```

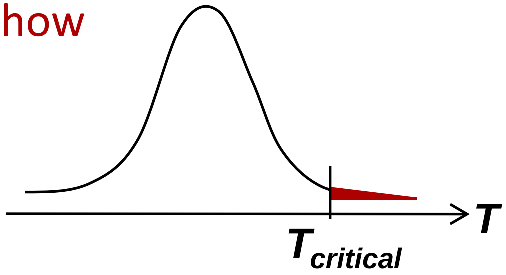
- CDF (and PDF) data

```
Level mappings for each response function:
Cumulative Distribution Function (CDF) for g_stress:
     Response Level  Probability Level  Reliability Index
     --------------  -----------------  -----------------
   2.4921421856e+02   1.0000000000e-03
   4.1489075797e+03   1.0000000000e-02
   7.9708753041e+03   5.0000000000e-02
   1.0090342657e+04   1.0000000000e-01
   1.1589780322e+04   1.5000000000e-01
   1.2731567123e+04   2.0000000000e-01
   1.4564078343e+04   3.0000000000e-01
```



Challenge: Calculating Potentially Small Probability of Failure

- Given uncertainty in materials, geometry, and environment, how to determine likelihood of failure: *Probability*($T \geq T_{critical}$)?
- Perform 10,000 LHS samples and count how many exceed threshold;
  (better) perform adaptive importance sampling



Mean value: make a linearity (and possibly normality) assumption and project; great for many parameters with efficient derivatives!

$$\mu_T = T(\mu_u)$$

$$\sigma_T = \sum_i \sum_j Cov_u(i,j) \frac{dg}{du_i}(\mu_u) \frac{dg}{du_j}(\mu_u)$$

Reliability: directly determine input variables which give rise to failure behaviors by solving an optimization problem for a most probable point (MPP) of failure

$$\text{minimize} \quad u^T u$$

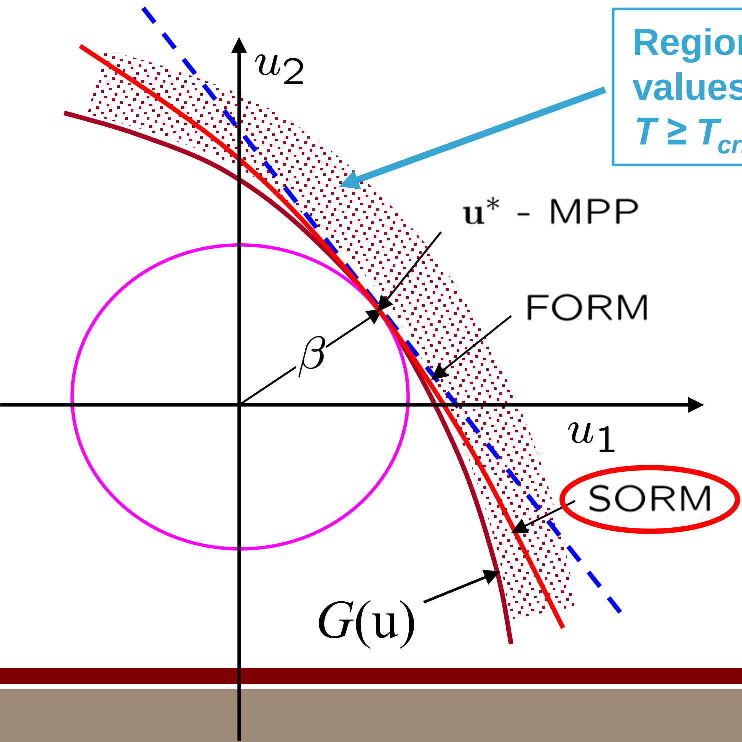
$$\text{subject to} \quad T(u) = T_{critical}$$

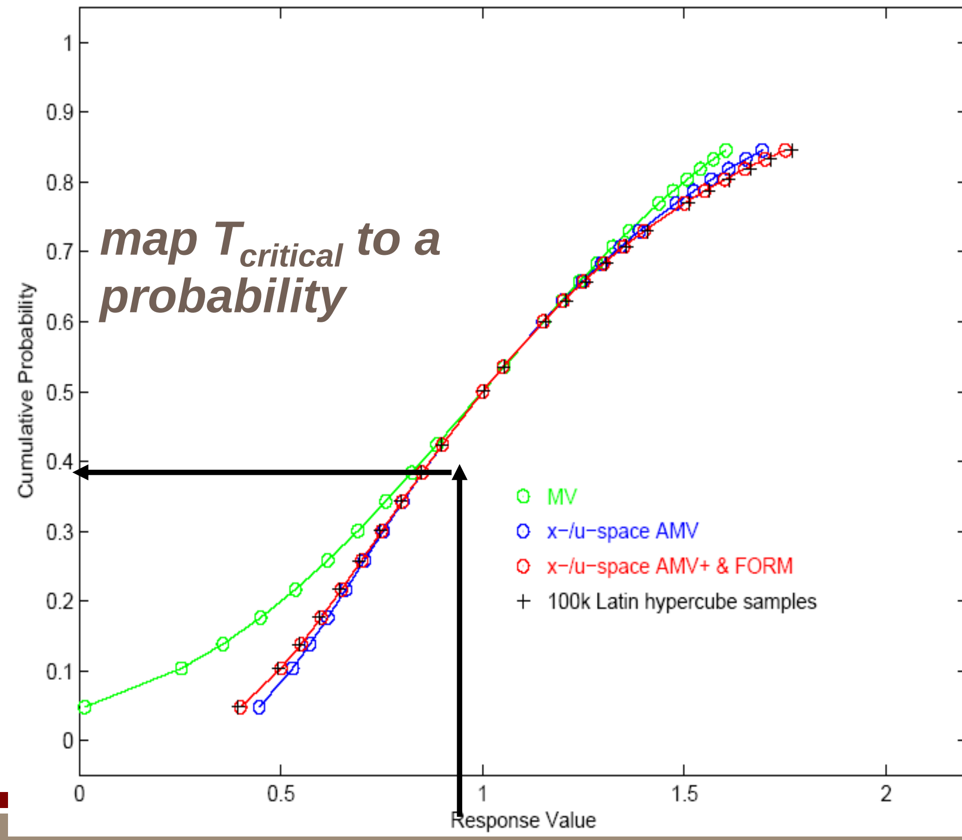
Analytic Reliability: MPP Search

Perform optimization in uncertain variable space to determine Most Probable Point (of response or failure occurring).

Reliability Index Approach (RIA)

$$\begin{array}{ll} \text{minimize} & \mathbf{u}^T\mathbf{u} \\ \text{subject to} & G(\mathbf{u}) = \bar{z} \end{array}$$



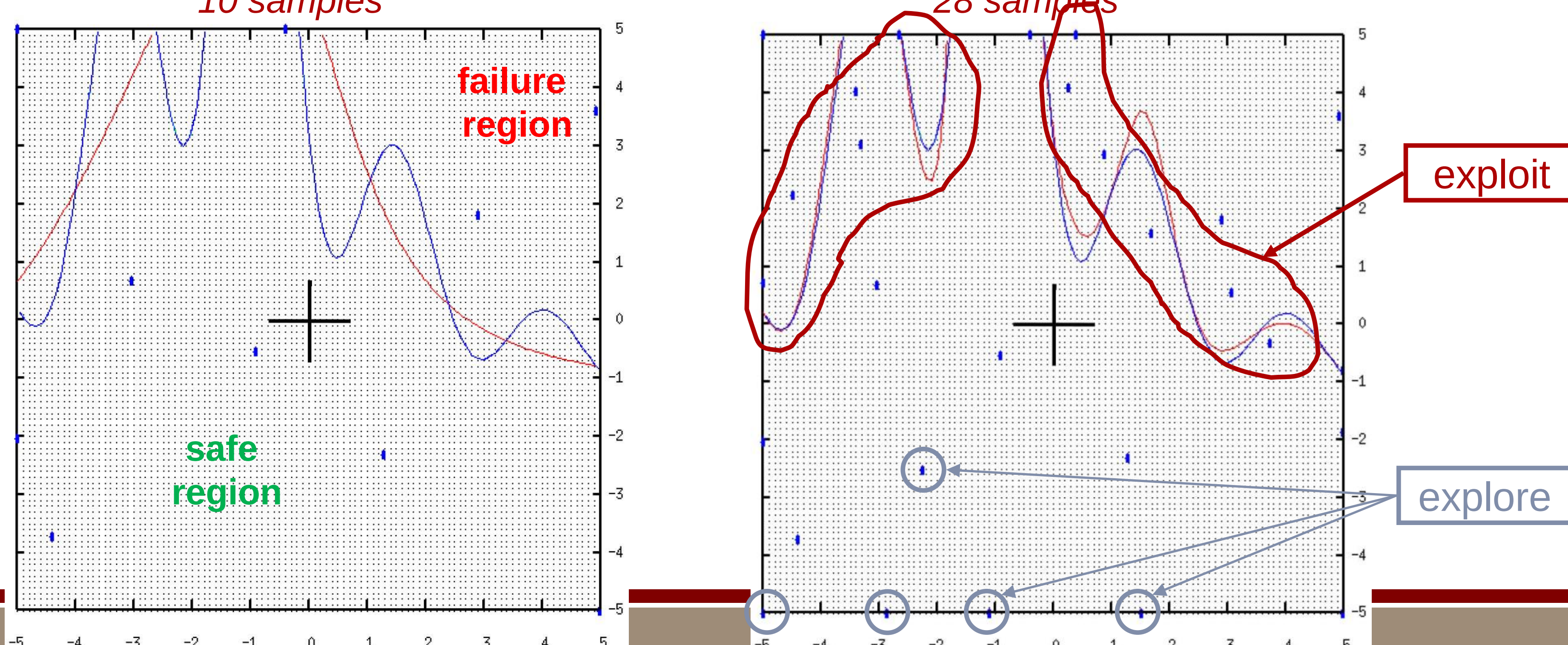


Efficient Global Reliability Analysis Using Gaussian Process Surrogate + MMAIS

- Efficient global optimization (EGO)-like approach to solve optimization problem
- Expected feasibility function: balance exploration with local search near failure boundary to refine the GP
- Cost competitive with best local MPP search methods, yet better probability of failure estimates; addresses nonlinear and multimodal challenges

Gaussian process model (level curves) of reliability limit state with

10 samples *28 samples*



Generalized Polynomial Chaos Expansions (PCE)

Approximate the response using orthogonal polynomial basis functions defined over standard random variables

$$R = \sum_{j=0}^{P} \alpha_j \Psi_j(\boldsymbol{\xi})$$

$$\alpha_j = \frac{\langle R, \Psi_j \rangle}{\langle \Psi_j^2 \rangle} = \frac{1}{\langle \Psi_j^2 \rangle} \int_{\Omega} R\, \Psi_j\, \varrho(\boldsymbol{\xi})\, d\boldsymbol{\xi}$$

- Intrusive or non-intrusive
- Wiener-Askey Generalized PCE: optimal basis selection leads to exponential convergence of statistics

Distribution	Density function	Polynomial	Weight function	Support range
Normal	$\frac{1}{\sqrt{2\pi}} e^{\frac{-x^2}{2}}$	Hermite $He_n(x)$	$e^{\frac{-x^2}{2}}$	$[-\infty, \infty]$
Uniform	$\frac{1}{2}$	Legendre $P_n(x)$	1	$[-1, 1]$
Beta	$\frac{(1-x)^{\alpha}(1+x)^{\beta}}{2^{\alpha+\beta+1} B(\alpha+1,\beta+1)}$	Jacobi $P_n^{(\alpha,\beta)}(x)$	$(1-x)^{\alpha}(1+x)^{\beta}$	$[-1, 1]$
Exponential	e^{-x}	Laguerre $L_n(x)$	e^{-x}	$[0, \infty]$
Gamma	$\frac{x^{\alpha} e^{-x}}{\Gamma(\alpha+1)}$	Generalized Laguerre $L_n^{(\alpha)}(x)$	$x^{\alpha} e^{-x}$	$[0, \infty]$

- Can also numerically generate basis orthogonal to empirical data (PDF/histogram)

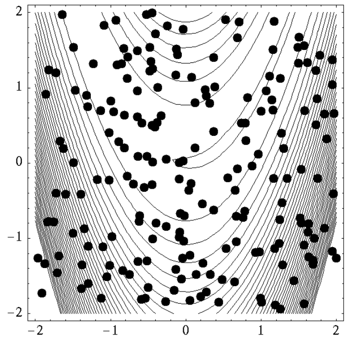
Sample Designs to Form Polynomial Chaos or Stochastic Collocation Expansions

Random sampling: PCE

Expectation (sampling):
- **Sample w/in distribution of *x***
- **Compute expected value of product of *R* and each Y_j**

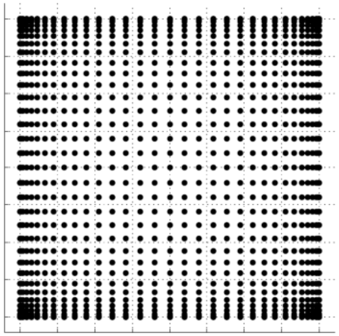
Linear regression ("point collocation"):

$$\boxed{\Psi\alpha = R}$$

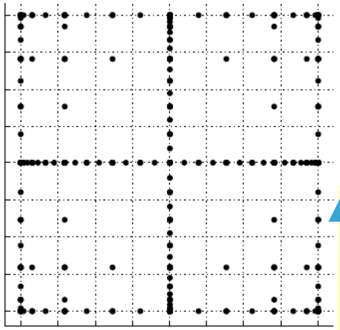


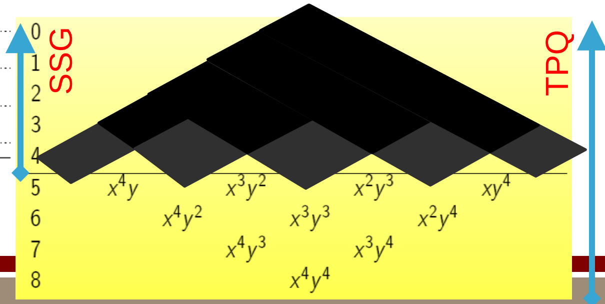
Tensor-product quadrature: PCE/SC

Tensor product of 1-D integration rules, e.g., Gaussian quadrature



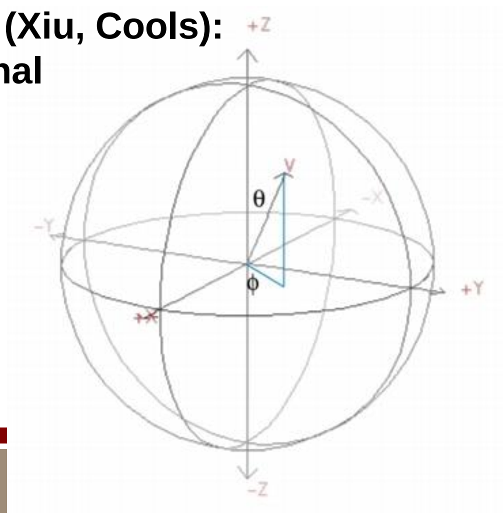
Smolyak Sparse Grid: PCE/SC





Cubature: PCE

Stroud and extensions (Xiu, Cools): optimal multidimensional integration rules



Changes for Reliability, PCE

```
method,
      local_reliability
        mpp_search no_approx
        num_probability_levels = 0 17 17
        probability_levels =
       .001 .01 .05 .1 .15 .2 .3 .4 .5 .6 .7 .8
.85 .9 .95 .99 .999
       .001 .01 .05 .1 .15 .2 .3 .4 .5 .6 .7 .8
.85 .9 .95 .99 .999
        cumulative distribution

responses,
      descriptors = 'area' 'stress'
                    'displacement'
      num_response_functions = 3
      analytic_gradients
      no_hessians
```

```
method,
      polynomial_chaos
        sparse_grid_level = 2 #non_nested
        sample_type lhs seed = 12347
        samples = 10000
        num_probability_levels = 0 17 17
        probability_levels =
       .001 .01 .05 .1 .15 .2 .3 .4 .5 .6 .7 .8
.85 .9 .95 .99 .999
       .001 .01 .05 .1 .15 .2 .3 .4 .5 .6 .7 .8
.85 .9 .95 .99 .999
        cumulative distribution
```

Uncertainty Quantification Research in Dakota: New algorithms bridge robustness/efficiency gap

Sandia National Laboratories

	Production	New	Under dev.	Planned	Collabs.
Sampling	**Latin Hypercube, Monte Carlo**	**Importance, Incremental**		**Bootstrap, Jackknife**	**FSU**
Reliability	***Local:* Mean Value, First-order & second-order reliability methods (FORM, SORM)**	***Global:* Efficient global reliability analysis (EGRA)**	**gradient-enhanced**	**recursive emulation, TGP**	***Local:* Notre Dame, *Global:* Vanderbilt**
Stochastic expansion		**PCE and SC with uniform & dimension-adaptive p-/h-refinement**	**Local adapt refinement, gradient-enhanced, compr sens**	**Discrete rv, orthogonal least interp.**	**Stanford, Purdue**
Other probabilistic			**Rand fields/ stoch proc**	**Dimension reduction**	**Cornell, Maryland**
Epistemic	**Interval-valued/ Second-order prob. (nested sampling)**	**Opt-based interval estimation, Dempster-Shafer**	**Bayesian, discrete/ model form**	**Imprecise probability**	**LANL, UT Austin**
Metrics & Global SA	**Importance factors, Partial correlations**	**Main effects, Variance-based decomposition**		**Stepwise regression**	**LANL**

Research: Adaptive Refinement, Gradient Enhancement

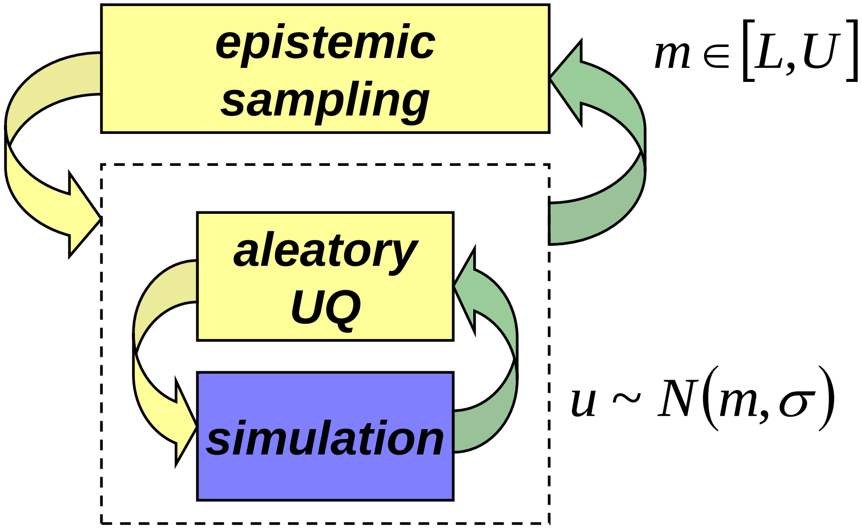
Adv. Deployment

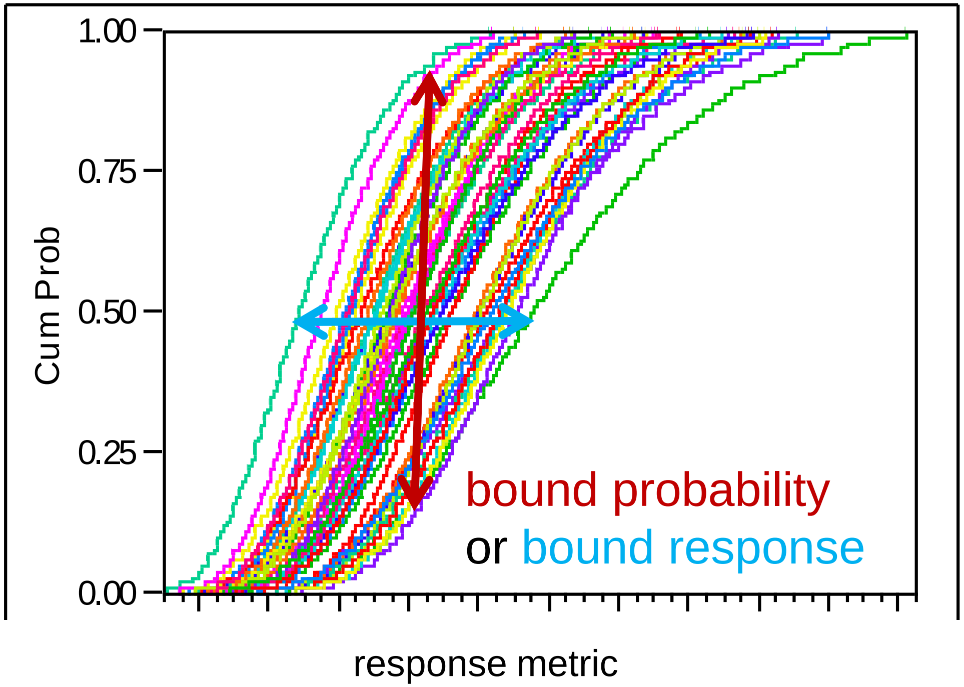
Fills Gaps

Aleatory/Epistemic UQ: Nested ("Second-order")Approaches

- Propagate over epistemic and aleatory uncertainty, e.g.,
  UQ with bounds on the mean of a normal distribution (hyper-parameters)
- Typical in regulatory analyses (e.g., NRC, WIPP)
- Outer loop: epistemic (interval) variables, inner loop UQ over aleatory (probability) variables; *potentially costly, not conservative*
- If treating epistemic as uniform, do not analyze probabilistically!

50 outer loop samples:
50 aleatory CDF traces





"Envelope" of CDF traces represents response epistemic uncertainty

Model Form UQ in Fluid/Structure Interactions

Low — Med — High

Discrete model choices for same physics:

- A clear hierarchy of fidelity (low to high)
- An ensemble of models that are all credible (lacking a clear preference structure)
  - With data: Bayesian model selection
  - Without data: epistemic model form uncertainty propagation

SA-RANS — KE-RANS-NBC — KE-RANS-DBC

- Combination:

Potential flow — Vortex lattice

SA-RANS — KE-RANS-NBC — KE-RANS-DBC

Smagorinsky-LES — Germano-LES

DNS

Horizontal Axis Wind Turbine ***Vertical Axis Wind Turbine***

wind turbine applications

Multifidelity UQ using Stochastic Expansions

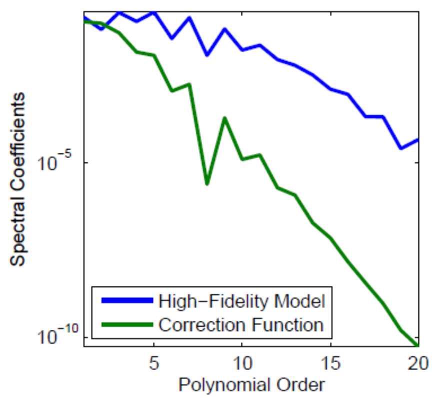
- High-fidelity simulations (e.g., RANS, LES) can be prohibitive for use in UQ
- Low fidelity "design" codes often exist that are predictive of basic trends
- Can we leverage LF codes w/i HF UQ in a rigorous manner? → global approxs. of model discrepancy

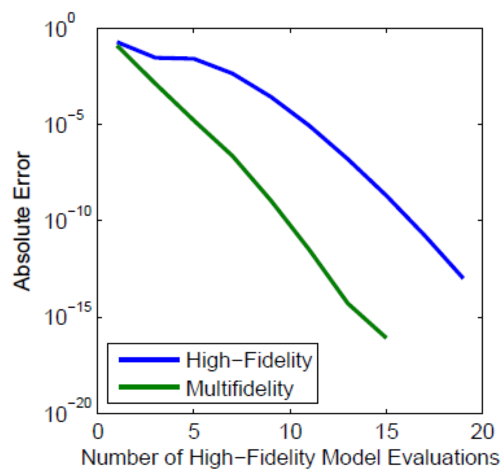
$$\boxed{\hat{f_{hi}}(\xi) = \sum_{j=1}^{N_{lo}} f_{lo}(\xi_j) L_j(\xi) + \sum_{j=1}^{N_{hi}} \Delta f(\xi_j) L_j(\xi)}$$

$$\boxed{N_{lo} >> N_{hi}}$$

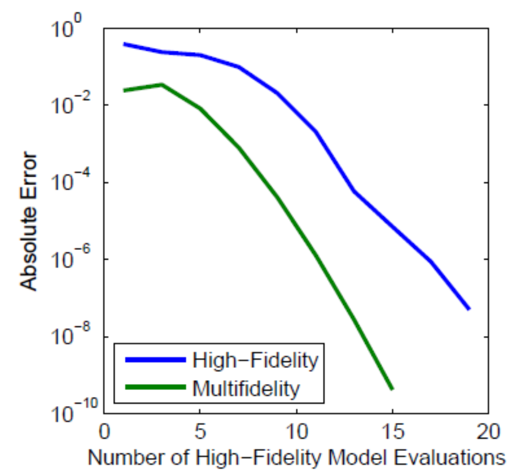
$$R_{\text{high}}(\xi) = e^{-0.05\xi^2} \cos 0.5\xi \underbrace{- 0.5e^{-0.02(\xi-5)^2}}_{\text{discrepancy}}$$

$$R_{\text{low}}(\xi) = e^{-0.05\xi^2} \cos 0.5\xi,$$





(a) Error in mean



(b) Error in standard deviation

Low fidelity

CACTUS: Code for Axial and Crossflow TUrbine Simulation

High fidelity: DG formulation for LES
Full Computational Fluid Dynamics/
Fluid-Structure Interaction

Dakota UQ: Summary, Relevant Methods

- What? Understand code output uncertainty / variability
- Why? Risk-informed decisions with variability, possible outcomes
- How? What Dakota methods are relevant?

character	method class	problem character	variants
aleatory	probabilistic sampling	nonsmooth, multimodal, modest cost, # variables	Monte Carlo, LHS, importance
	local reliability	smooth, unimodal, more variables, failure modes	mean value and MPP, FORM/SORM,
	global reliability	nonsmooth, multimodal, low dimensional	EGRA
	stochastic expansions	nonsmooth, multimodal, low dimension	polynomial chaos, stochastic collocation
epistemic	interval estimation	simple intervals	global/local optim, sampling
	evidence theory	belief structures	global/local evidence
both	nested UQ	mixed aleatory / epistemic	nested

- See Dakota Usage Guidelines in User’s Manual
- Analyze tabular output with third-party statistics packages

UQ References

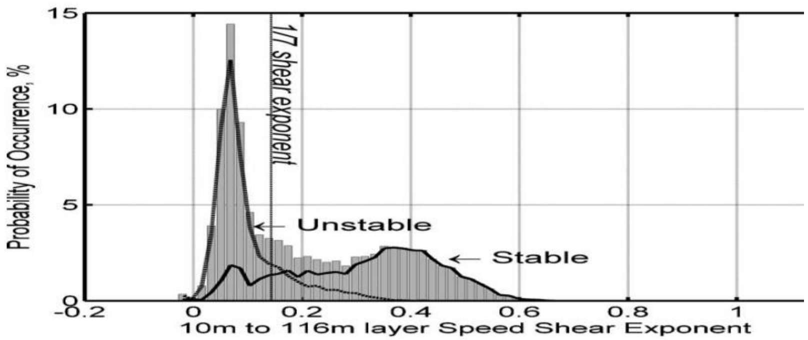
- SAND report 2009-3055. “Conceptual and Computational Basis for the Quantification of Margins and Uncertainty” J. Helton.
- Helton, JC, JD Johnson, CJ Sallaberry, and CB Storlie. “Survey of Sampling-Based Methods for Uncertainty and Sensitivity Analysis”, *Reliability Engineering and System Safety* 91 (2006) pp. 1175-1209
- Helton JC, Davis FJ. Latin Hypercube Sampling and the Propagation of Uncertainty in Analyses of Complex Systems. *Reliability Engineering and System Safety* 2003;81(1):23-69.
- Haldar, A. and S. Mahadevan. *Probability, Reliability, and Statistical Methods in Engineering Design* (Chapters 7-8). Wiley, 2000.
- Eldred, M.S., "Recent Advances in Non-Intrusive Polynomial Chaos and Stochastic Collocation Methods for Uncertainty Analysis and Design," paper AIAA-2009-2274 in Proceedings of the 11th AIAA Non-Deterministic Approaches Conference, Palm Springs, CA, May 4-7, 2009.

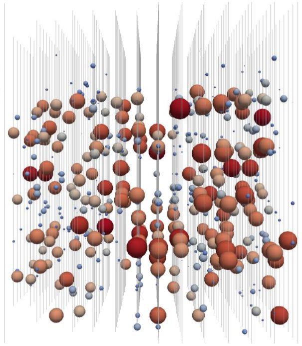
- Dakota User’s Manual: Uncertainty Quantification Capabilities
- Dakota Theory Manual
- Corresponding Reference Manual sections

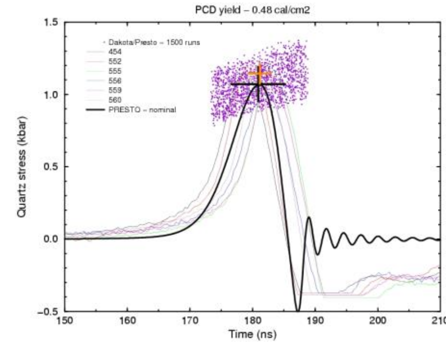
SAND2014-4255C

Exceptional service in the national interest

Sandia National Laboratories







Dakota Calibration Methods

Adam Stephens, Laura Swiler

Dakota Clinic at CSDMS May 22, 2014

U.S. DEPARTMENT OF ENERGY

NNSA National Nuclear Security Administration

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000. SAND NO. 2011-XXXXP

SAND2014-4255C

Calibration Background

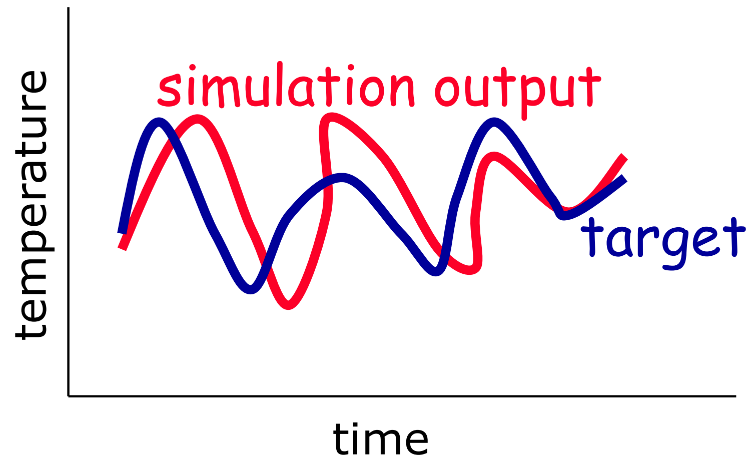
- Determine parameter values that maximize agreement between simulation response and target response (AKA parameter estimation, parameter identification, nonlinear least-squares)

$$SSE(\boldsymbol{\theta}) = \sum_{i=1}^{n} [(y_i - s_i(\boldsymbol{\theta}))]^2 = \sum_{i=1}^{n} [r_i(\boldsymbol{\theta})]^2$$

Sum of Squared Errors = sum of Residuals

Experimental data

Simulation output



Calibration

- Fit model to data
  - E.g., determine material model parameters such that predicted stress-strain curve matches one generated experimentally
- Other uses: determine control settings that enable a system to achieve a prescribed performance profile
- Calibration should be performed in a larger context of verification, sensitivity analysis, uncertainty quantification, and validation
- Calibration is often thought of as "inverse modeling" whereas uncertainty propagation (from uncertain inputs to model outputs) is called "forward modeling"
- Calibration is not validation! <u>Separate</u> data must be used to assess whether a calibrated model is valid

More about Calibration

- Can formulate the calibration problem as an optimization problem and either use global derivative-free or local gradient-based methods to solve it
- Global Methods:
  - Usually better at finding an overall minimum or set of minima.
  - Do not require the calculation of gradients which can be expensive, especially for high-dimensional problems.
  - Global methods often require more function evaluations than local methods.
  - We use DIRECT (DIviding RECTangles), a method that adaptively subdivides the space of feasible design points so as to guarantee that iterates are generated in the neighborhood of a global minimum in finitely many iterations.
  - With global methods, we hand the SSE to the optimizer as one objective to minimize:

$$SSE(\boldsymbol{\theta}) = \sum_{i=1}^{n} [(y_i - s_i(\boldsymbol{\theta})]^2 = \sum_{i=1}^{n} [r_i(\boldsymbol{\theta})]^2$$

More about Calibration

- Nonlinear least squares methods are local methods which exploit the structure of the SSE objective
- Gauss-Newton optimization methods are commonly used: these are a modification of Newton's method for root-finding.
- We use NL2SOL algorithm, which is more robust than many Gauss-Newton solvers which experience difficulty when the residuals at the solution are significant.
- These methods assume the residuals are near zero close to optimum: we ignore the term circled and only use gradients to approximate the Hessian matrix of second-derivatives
- These methods can be very efficient, converging in a few function evaluations

$$\frac{SSE}{2} = f(\theta) = \frac{1}{2} r(\theta)^T r(\theta) = \frac{1}{2}\left[s(\theta) - y\right]^T \left[s(\theta) - y\right]$$

$$\nabla f(\theta) = J(\theta)^T r(\theta); \quad J_{ij} = \frac{\partial r_i}{\partial \theta_j} \qquad \nabla^2 f(\theta) = J^T J + \sum_{i=1}^{n} r_i(\theta) \nabla^2 r_i(\theta)$$

Exercise: Calibrate Cantilever to Experimental Data

- Calibrate design variables *E, w, t* to data from all 3 responses
- *X, Y, R* fixed (state) at nominal values
- Use NL2SOL or OPT++ Gauss-Newton
- Key DAKOTA specs:
  - calibration_terms = 3
  - no constraints
  - least_squares_datafile

DATA	clean	with error
area	7.5	7.772
stress	2667	2658
displacement	0.309	0.320

```
cantilever_clean.dat
cantilever_witherror.dat
```

- ***For least-squares methods, application normally must return residuals $r_i(x)= s_i(x)- d_i$ to DAKOTA***
- ***Here we return the usual area, stress, displacement and specify a datafile and DAKOTA computes the residuals***

Potential Solution: Cantilever Least-Squares

```
# Calibrate to area, stress, and displacement data generated with
# E = 2.85e7, w = 2.5, t = 3.0

method
  nl2sol
    convergence_tolerance = 1.0e-6

variables
  continuous_design = 3
    upper_bounds  3.1e7 10.0 10.0
    initial_point 2.9e7 4.0  4.0
    lower_bounds  2.7e7 1.0  1.0
    descriptors   'E' 'w' 't'
  # Fix at nominal
  continuous_state = 3
    initial_state 40000 500 1000
    descriptors 'R' 'X' 'Y'

interface
  direct
    analysis_driver = 'mod_cantilever'

responses
  calibration_terms = 3
#    calibration_data_file = 'dakota_cantilever_clean.dat'
    calibration_data_file = 'dakota_cantilever_witherror.dat'
    descriptors = 'area' 'stress' 'displacement'
  analytic_gradients
  no_hessians
```

Confidence Intervals approximated by calculating the variance of the parameter vector as diagonal elements of:

$$\hat{\sigma}^2 (J^T J)^{-1}$$

```
CIs without error:
E: [ 2.850e+07, 2.850e+07 ]
w: [ 2.500e+00, 2.500e+00 ]
t: [ 3.000e+00, 3.000e+00 ]
```

```
CIs with error:
E: [ 1.992e+07, 4.190e+07 ]
w: [ 1.962e+00, 3.918e+00 ]
t: [ 1.954e+00, 3.309e+00 ]
```

Quick Guide for Calibration Method Selection

Category	DAKOTA method names	Continuous Variables	Categorical/ Discrete Variables	Bound Constraints	General Constraints
Gradient-Based Local (Smooth Response)	`nl2sol`	x		x	
	`nlssol_sqp, optpp_g_newton`	x		x	x
Gradient-Based Global (Smooth Response)	`hybrid strategy, multi_start strategy`	x		x	x
Derivative-Free Global (Nonsmooth Response)	`efficient_global, surrogate_based_global`	x		x	x

See Usage Guidelines in DAKOTA User's Manual.
Also, can apply any optimizer when doing derivative-free local or global calibration.

Bayesian Methods

- What is Bayesian analysis?
- How is it used in calibration?
- Why is it hard?
- What is the state-of-the-art?
- What calibration capabilities do we have in DAKOTA?

Bayesian Analysis

Bayesian analysis allows us to formally combine:

Earlier understanding of a phenomenon

Currently measured data

Updated degree of belief

We want to make a formal statistical inference about the probability distribution underlying a random phenomenon

Bayes' Theorem

Posterior probability:
Updated belief about E given the occurrence of a related event A

Likelihood Prior probability

$$\boldsymbol{P(E|\,A) = \frac{P(A|E)P(E)}{P(A)}}$$

$$\boldsymbol{P(E_i|\,A) = \frac{P(A|E_i)P(E_i)}{\sum_j P(A|E_j)P(E_j)}}$$

Bayes' Theorem for Continuous Variables

θ: uncertain parameter

y: observed data

Likelihood function

Prior PDF

Posterior PDF

$$f(\theta \mid y) = \frac{f(y \mid \theta)\,\pi(\theta)}{\boxed{\int_{-\infty}^{\infty} \underbrace{f(y \mid \theta)\,\pi(\theta)}_{L(\theta)}\, d\theta}}$$

Constant

Posterior $\propto$ Likelihood x Prior

Bayesian Calibration for Simulation Models

- Experimental data = Model output + error

$$d_i = G(\boldsymbol{\theta}, \mathbf{x}_i) + \varepsilon_i$$

- Error term incorporates measurement errors and modeling errors (can get more complex with a bias term)

$$d_i = G(\boldsymbol{\theta}, \mathbf{x}_i) + \delta(\mathbf{x}_i) + \varepsilon_i$$

- If we assume error terms are independent, zero mean Gaussian random variables with variance σ^2, the likelihood is:

$$L(\boldsymbol{\theta}) = \prod_{i=1}^{n} \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{(d_i - G(\boldsymbol{\theta}, \mathbf{x}_i))^2}{2\sigma^2} \right]$$

- How do we obtain the posterior?
  - It is usually too difficult to calculate analytically
  - We use a technique called Monte Carlo Markov Chain (MCMC)

Markov Chain Monte Carlo

Sandia National Laboratories

- In MCMC, the idea is to *generate a sampling density that is approximately equal to the posterior.* We want the sampling density to be the stationary distribution of a Markov chain.
- Metropolis-Hastings is a commonly used algorithm
- It has the idea of a "proposal density" which is used for generating X_{i+1} in the sequence, conditional on $X_{i.}$
- Implementation issues: How long do you run the chain, how do you know when it is converged, how long is the burn-in period, etc.?
- Acceptance rate is very important. Need to tune the proposal density to get an "optimal" acceptance rate, 45-50% for 1-D problems, 23-26% for high dimensional problems
- COMPUTATIONALLY VERY EXPENSIVE

Surrogate Models

- Since MCMC requires tens of thousands of function evaluations, it is necessary to have a fast-running surrogate model of the simulation
- Dakota has the capability for using the following surrogates in the Bayesian calibration:
  - Gaussian Processes
  - Polynomial Chaos Expansions
  - Stochastic Collocation
- Steps for a Bayesian analysis:
  - Take initial set of samples from simulation
    - Use LHS or Sparse Grid
  - Develop surrogate approximation of the simulation
  - Define priors on the input parameters (uniform currently)
  - Perform Bayesian analysis using MCMC
  - Generate and analyze posterior distributions

Why is Bayesian Calibration difficult?

- In general, parameter estimation / inverse problems are challenging:
  - Observations contain noise
  - Model is imperfect
  - Many combinations of parameter values yield comparable fits
  - Model is expensive
- Bayesian calibration can address all of the above. However, the MCMC can give poor results and is hard to diagnose. The surrogates fits can be poor. These problems are often highly sensitive to priors and the likelihood formulation.
- There are a variety of MCMC approaches. We currently support:
  - Metropolis-Hastings
  - Delayed Rejection/Adaptive Metropolis (DRAM)
  - Differential Evolution Adaptive Metropolis (DREAM)

Status of Bayesian Calibration Methods in DAKOTA

- **QUESO** is a library of UQ methods developed at the UT PECOS center.
  - We currently can perform Bayesian calibration of model parameters with a simulation directly (no emulator), with a Gaussian process emulator, or with a polynomial chaos or stochastic collocation emulator.
  - The user is allowed to specify scaling for the proposal covariance.
  - We can input data from a file to build a GP emulator. We have looked at building the GP based on initial LHS points plus points from multi-start NLLS. This appears to help significantly, since it increases points in high likelihood regions.
  - Four variations of DRAM for the MCMC chain generation: metropolis-hasting or adaptive metropolis, delayed rejection or no delayed rejection. Recently added Prudencio's multi-level MCMC algorithm.
  - To do:
    - Allow for parallel chains, including Prudencio's multi-level algorithm
    - Extend the capability to handle more complicated covariances for observational error.

Status of Bayesian Calibration Methods in DAKOTA

- **DREAM.** Initial implementation in Dakota (as of June, 2013). Allows for multiple chains. Allows use of the same set of surrogates.
- **GPMSA.** GPMSA (Gaussian Process Models for Simulation Analysis) is a code developed by Brian Williams, Jim Gattiker, Dave Higdon, et al. at LANL.
  - Original LANL code is in Matlab.
  - GPMSA was re-implemented in the QUESO framework. We have an initial wrapper to it in Dakota, but much of it is hardcoded, not ready for general applications yet.
  - Need a way to handle functional data.
- **Framework:**
  - We do have the capability to read in configuration variables (X)
  - We can incorporate one estimate of sigma for all experiments, or a particular sigma per each experiment
  - We do not handle or formulate a discrepancy term at this point.

DAKOTA Bayesian Example

DAKOTA INPUT FILE - dakota_bayes.in

```
method,
    bayes_calibration queso,
        emulator
          gp     emulator_samples = 50
#         pce    sparse_grid_level = 3
          samples = 5000 seed = 348
          rejection delayed
          metropolis adaptive
          proposal_covariance_scale = 0.01
#       calibrate_sigma

variables,
          continuous_design = 2
          lower_bounds = 0. 0.
          upper_bounds = 3. 3.

interface,
          system
          analysis_driver = 'text_book‘
```

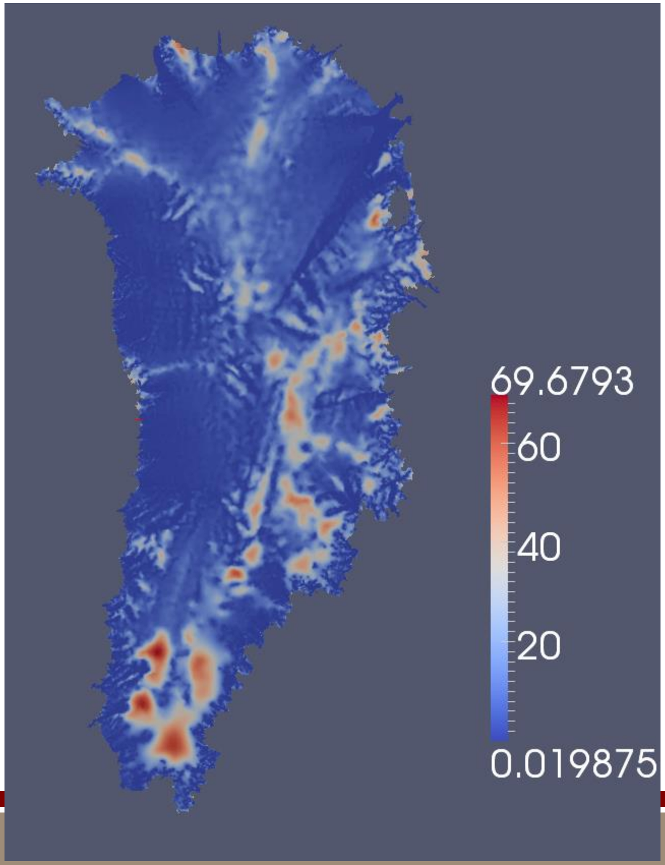
```
responses,
    calibration_terms = 1
    calibration_data_file = 'test10.txt‘
    freeform
     num_experiments = 1
     num_replicates = 10
     num_std_deviations = 1
      no_gradients
      no_hessians
```

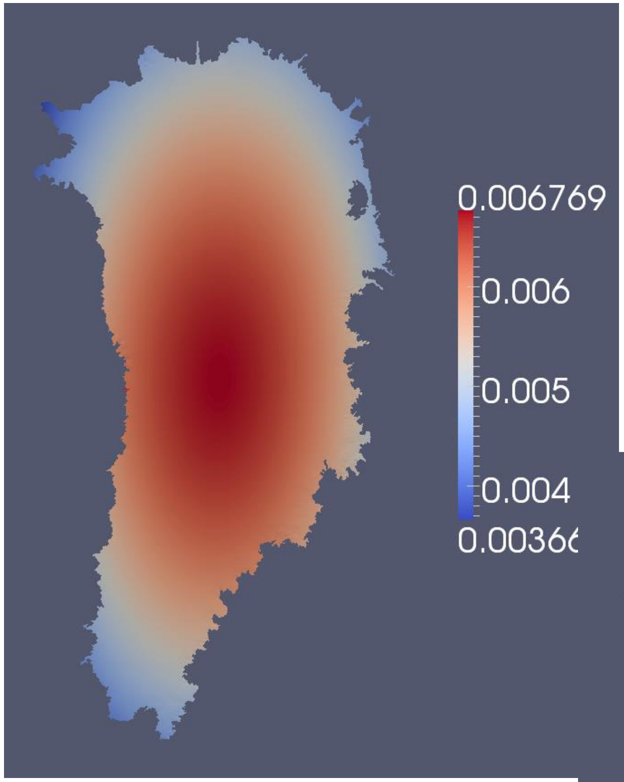
Data File: test10.txt

11.83039	1.0
11.94504	1.0
11.70863	1.0
12.19501	5.0
11.41225	1.0
10.86503	1.0
11.70797	1.0
11.54544	1.0
10.61684	1.0
10.94383	1.0

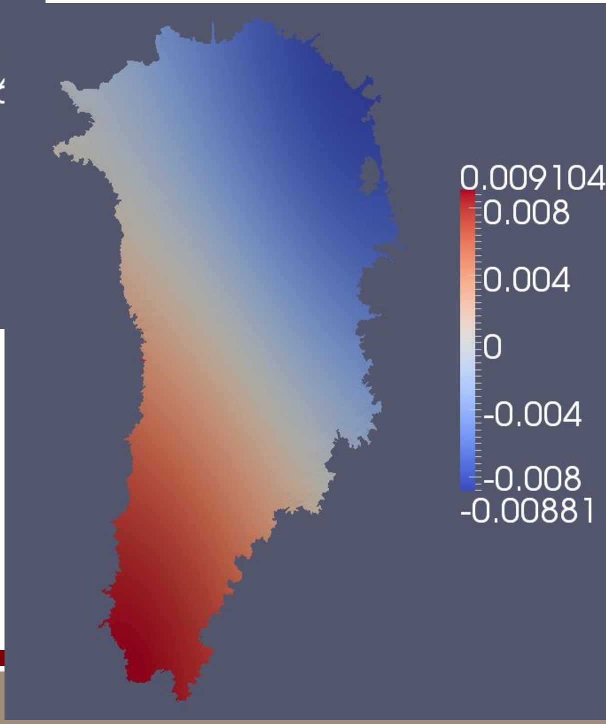
DAKOTA Example: Greenland Ice Model

Basal friction mean field
“Truth” model as observations





Two of the modes based on K-L expansions: posterior MAP estimates of the modes



Summary

- Bayesian calibration is conceptually attractive because it is able to give probabilistic estimates of model parameters and incorporates current information as well as historical data
- There is a “sweet spot” where it is useful: you need enough data to move the prior
- The state of the art is performing Bayesian analysis on a relatively small number of parameters, possibly using an emulator for expensive models, possibly including a discrepancy term
- There is much research in MCMC methods: it is easy to get a poor sampler and thus poor posterior estimate of parameters
  - Adaptive methods which build up information about the covariance between parameters
  - Methods do not perform well when there are parameters which don’t affect the output strongly
  - Issue of “where does the uncertainty get pushed” – into the model parameters or the error term?