SAND2018-11874C

Uncertainty Quantification in Computational Models of Large Scale Physical Systems

Habib N. Najm

Sandia National Laboratories
Livermore, CA, USA
hnnajm@sandia.gov

Seminar

National Research Council - Institute of Marine Engineering (CNR-INM)
Rome, Italy
Oct 25, 2018

andia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly own subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

Acknowledgement

B.J. Debusschere, M. Reagan, R.D. Berry, K. Sargsyan, C. Safta,
K. Chowdhary, M. Khalil, X. Huan, M. Eldred, G. Geraci, T. Casey, J. Oefelein,
G. Lacaze, Z. Vane, L. Hakim
– Sandia National Laboratories, CA

R.G. Ghanem – U. South. California, Los Angeles, CA
O.M. Knio – KAUST, Thuwal, Saudi Arabia & Duke Univ., Durham, NC
O.P. Le Maître – CNRS, Paris, France
Y.M. Marzouk – Mass. Inst. of Tech., Cambridge, MA

This work was supported by:

- DOE Office of Basic Energy Sciences, Div. of Chem. Sci., Geosci., & Biosci.
- DOE Office of Advanced Scientific Computing Research (ASCR)
- DOE ASCR Scientific Discovery through Advanced Computing (SciDAC) program
- DOE ASCR Applied Mathematics program
- DARPA
- Sandia National Laboratories, LDRD

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525. The views expressed in the article do not necessarily represent the views of the U.S. Department of Energy or the United States Government.

Outline

1. Introduction
2. PC Smoothing
3. Sparse Regression
4. Multilevel Multifidelity
5. Scramjet Application
6. Closure

Definition of Uncertainty Quantification (UQ)

UQ is the end-to-end estimation and analysis of uncertainty in:

models and their parameters

- assimilation of experimental/observational data
- model fitting and parameter estimation

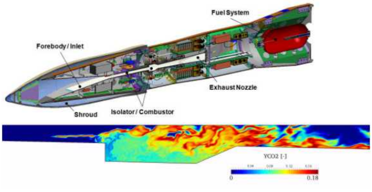
model predictions

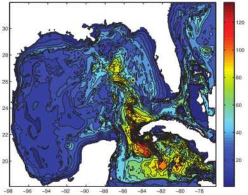
- forward propagation of parametric uncertainty to model outputs
- Analysis, comparison and selection among alternate plausible models

The Case for Uncertainty Quantification

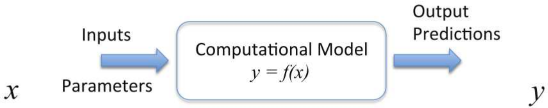
UQ is needed in:

- Assessment of confidence in computational predictions
- Validation and comparison of scientific/engineering models
- Robust design optimization under uncertainty
- Use of computational predictions for decision-support
- Assimilation of observational data and model construction
- Multiscale and multiphysics model coupling



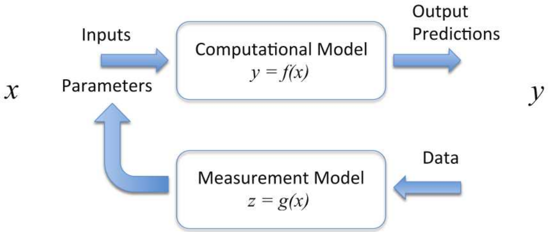


Uncertainty Quantification and Computational Science



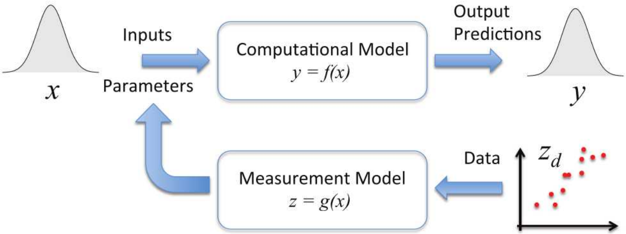
Forward problem

Uncertainty Quantification and Computational Science



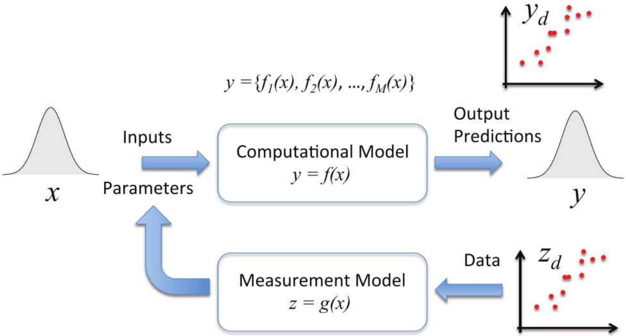
Inverse & Forward problems

Uncertainty Quantification and Computational Science



Inverse & Forward UQ

Uncertainty Quantification and Computational Science



Inverse & Forward UQ

Model validation & comparison, Hypothesis testing

Probabilistic Forward UQ – $y = f(x)$

Represent uncertain quantities using probability theory

Random sampling, Monte Carlo

- Generate random samples $\{x^i\}_{i=1}^N$ from the PDF of x, $p(x)$
- Bin the corresponding $\{y^i\}$ to construct $p(y)$
- Not feasible for computationally expensive $f(x)$
  - slow convergence of MC/QMC methods
    $\Rightarrow$ very large N required for reliable estimates

Build a cheap surrogate for $f(x)$, then use Monte Carlo/others

- Collocation – interpolants
- Regression – fitting
- Galerkin methods
  - Polynomial Chaos (PC) methods

Polynomial Chaos Expansion (PCE)

- Model uncertain quantities as random variables (RVs)
- Given a *germ* $\boldsymbol{\xi}(\omega) = \{\xi_1, \cdots, \xi_n\}$ – a set of *i.i.d.* RVs
  - where $p(\boldsymbol{\xi})$ is uniquely determined by its moments

Any RV in $L^2(\Omega, \mathfrak{S}(\boldsymbol{\xi}), P)$ can be written as a PCE:

$$u(\boldsymbol{x}, t, \omega) = f(\boldsymbol{x}, t, \boldsymbol{\xi}) \simeq \sum_{k=0}^{P} u_k(\boldsymbol{x}, t) \Psi_k(\boldsymbol{\xi}(\omega))$$

- $u_k(\boldsymbol{x}, t)$ are mode strengths
- $\Psi_k()$ are multivariate functions orthogonal w.r.t. $p(\boldsymbol{\xi})$

Essential Use of Polynomial Chaos Expansions in UQ

Strategy:

- Represent model parameters/solution as random variables
- Construct PCEs for uncertain parameters
- Evaluate PCEs for model outputs

Advantages:

- Computational efficiency
- Utility
  - Moments: $\mathrm{E}(u) = u_0$, $\mathrm{var}(u) = \sum_{k=1}^{P} u_k^2 \langle \Psi_k^2 \rangle$, ...
  - Global Sensitivities – fractional variances, Sobol' indices
  - Surrogate for forward model

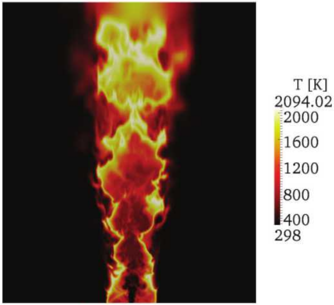
Requirement:

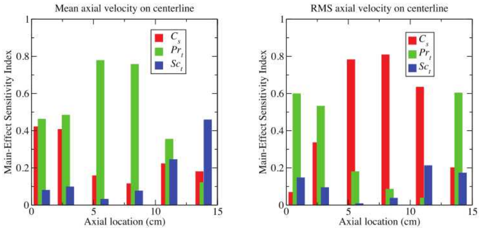
- RVs in L^2, *i.e.* with finite variance, on $(\Omega, \mathfrak{S}(\boldsymbol{\xi}), P)$

UQ in LES computations: turbulent bluff-body flame

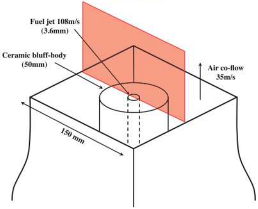
with M. Khalil, G. Lacaze, & J. Oefelein, Sandia Nat. Labs

- CH_4-H_2 jet, air coflow, 3D flow
- Re=9500, LES subgrid modeling
- 12×10^6 mesh cells, 1024 cores
- 3 days run time, 2×10^5 time steps
- 3 uncertain parameters (C_s, Pr_t, Sc_t)
- 2^{nd}-order PC, 25 sparse-quad. pts





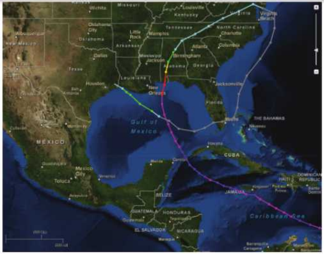
Main-Effect Sensitivity Indices

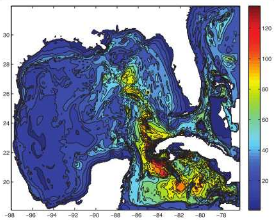


J. Oefelein & G. Lacaze, SNL

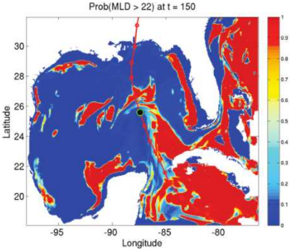
UQ in Ocean Modeling – Gulf of Mexico

A. Alexanderian, J. Winokur, I. Sraj, O.M. Knio, Duke Univ.
A. Srinivasan, M. Iskandarani, Univ. Miami; W.C. Thacker, NOAA





- Hurricane Ivan, Sep. 2004
- HYCOM ocean model (hycom.org)
- Predicted Mixed Layer Depth (MLD)
- Four uncertain parameters, *i.i.d.* U
  - subgrid mixing & wind drag params
- 385 sparse quadrature samples



(Alexanderian *et al.*, Winokur *et. al.*, *Comput. Geosci.*, 2012, 2013)

High dimensionality is a major challenge in forward UQ

- High dimensionality is the result of
  - Large number of uncertain parameters/inputs
  - Large number of degrees of freedom in random field inputs
- PCE sparse-quadrature requires an unfeasible number of model evaluations for very high dimensional systems
- MC requires similarly large number of samples when the number of important dimensions is very high
- Typically, physical model output quantities of interest are *smooth*
  - Only a small number of inputs are important
- In this case, the way out is:
  - Use global sensitivity analysis (GSA) with MC to identify important parameters
  - Use PCE sparse-quadrature on the reduced dimensional space for accurate forward UQ

Global Sensitivity Analysis (GSA)

- Random sampling-based
  - Define parametric PDFs
  - Sample them
    - Monte Carlo (MC) sampling
    - Quasi-Monte Carlo (QMC) methods
      e.g. Latin Hypercube sampling (LHS) ...
  - Run forward model for each sample
  - Evaluate statistics/PDFs of output observables
  - Sensitivity information
    - Scatter plots; Correlation measures; Regression
    - Importance Measures; Sobol' sensitivity indices
- Response surface construction based on samples
  - followed by sensitivity and/or UQ
- Analysis of Variance (ANOVA)
  - High Dimensional Model Representation (HDMR)

Hi-dimension with large-scale computational models

When the number of feasible samples for GSA is highly limited due to computational costs:

- Reliable MC-estimation of sensitivity indices requires regularization
- Presuming smoothness, use MC samples to fit a PCE, which is subsequently used to estimate the sensitivity indices
- Employ ℓ_1-norm constrained regression to discover a sparse PCE
  - compressive sensing
- Employ Multilevel Monte Carlo (MLMC), as well as Multilevel Multifidelity (MLMF) methods
  - Optimal combination of coarse/fine mesh and low/high fidelity models to minimize computational costs for a given accuracy

Similarly for forward PC UQ:

- Employ generalized adaptive non-isotropic sparse quadrature with MLMF methods on reduced dimensional input space

Learning from Limited Sparse Noisy Data

- We often need to use data to estimate quantities of interest
- Experimental data can be inherently noisy
  - Randomness in physical system
  - Instrument noise
- It can also be sparse – data gaps
  - e.g. data (x_i, y_i) on small subsets of some domain $\mathcal{D}$
- Even when the data is from computations, it can be "noisy"
  - Statistics computed from simulation outputs can be noisy due to limited averaging time windows
  - Discrepancies between computational simulation outputs and some fit-model of interest can be treated as noise
- Generally, learning from small number of data points, especially when sparse and noisy, can be a major challenge
  - Little information available from data

Learning from data with limited information

- Generally we address "Learning" in the context of estimation of statistics, and fitting
- Consider data $(x_1^{(i)}, \cdots, x_n^{(i)}, y^{(i)})$, $i = 1, \cdots, N$
  - $x \in \mathbb{R}^n$ are independent variables
  - $y \in \mathbb{R}$ is the dependent variable
- We might be interested in statistics $s = (s_1, \cdots, s_n)$
  - with, e.g., $s_i = V[E(y|x_i)]$
- Or fitting function $y = f(x; c)$, to estimate $c = (c_0, \cdots, c_P)$
  - with, e.g. $f(x; c) = \sum_{k=0}^{P} c_k \Psi_k(x)$
  - with either least-squares fitting, or Bayes
- In all cases, the accuracy of the estimate of c or s depends on the "quality" of the data
  - data noise, coverage, range, quantity

Estimation of GSA Sobol' Indices with PC regularization

- When the number of samples is small, the GSA sensitivity indices can be computed with improved accuracy, relying on regularization
- Use regression with MC samples to fit a Legendre-Uniform PCE to the data

$$u(\boldsymbol{\xi}) = \sum c_k \Psi_k(\boldsymbol{\xi})$$

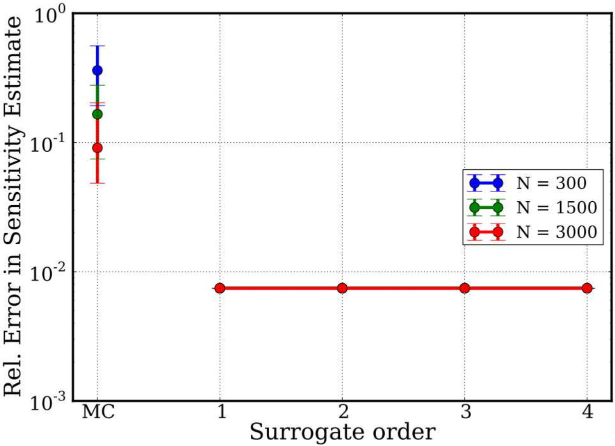
- Use PCE to evaluate Sobol Indices directly

Sargsyan, 2017

- Example results illustrate significant improvement over the direct estimation from samples

Estimation of GSA Sobol' Indices with PC regularization

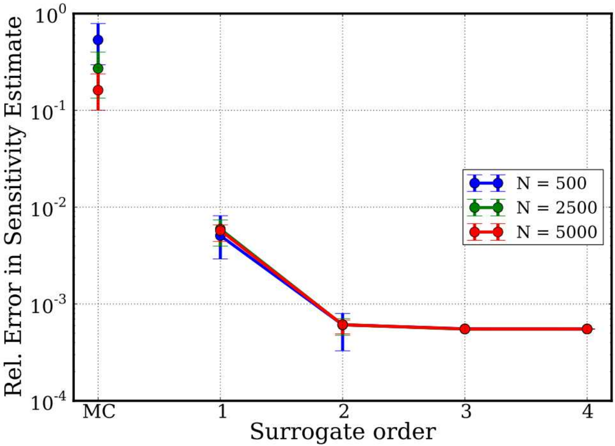
$d = 1$



Sargsyan, 2017

Estimation of GSA Sobol' Indices with PC regularization

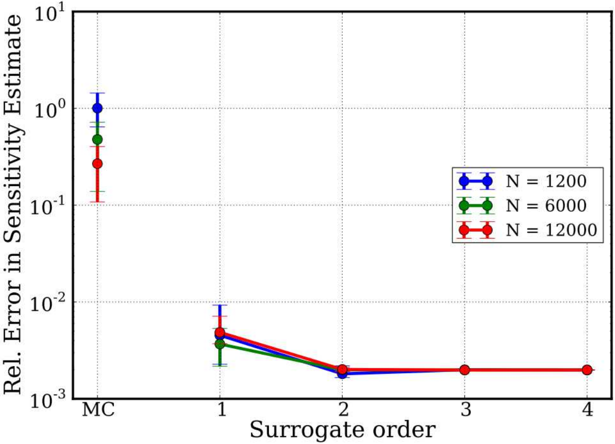
$d = 3$



Sargsyan, 2017

Estimation of GSA Sobol' Indices with PC regularization

$d = 10$



Sargsyan, 2017

Need for regularization in fitting/inversion

- Sometimes the fitting problem is ill-posed ... non-uniqueness
  - Multiple values of c give the same $f(x;c)$ over a range of x
- More often the problem is ill-conditioned
  - The amount of information in the data is small relative to the number of parameters we are interested in learning
    - For example $N << P + 1$
- Ill-conditioning can manifest itself in sensitivity to
  - Initial guess choice in least-squares fitting
  - Prior choice in Bayesian fitting
  - Specific choice of data set
- Ill-conditioning can lead to poor convergence in the iterative solution or to poor mixing in MCMC
- Regularization is often useful to deal with these challenges

Regularization in Learning from Data

- Regularization involves adding some information into the problem
- Of course, the choice of the form of $f(x;c)$, or the value of P, also provide explicit regularization
- Regularization allows enforcement of desired traits in the solution
  - Smoothness, positivity, ...
  - Introduces bias, destroys consistency
- Example: Tikhonov-type regularization:

$$c = \operatorname*{argmin}_{c'} \left(\|f(x,c) - y\|_2^2 + \alpha L(c') \right)$$

- How to choose regularization form, L, α ? – Somewhat arbitrary
- $L(t) := \|t\|_2^2 \Rightarrow$ favor solutions with small ℓ_2 norm
- $L(t) := \| \max_x \nabla_x f(x,t)\|_2^2 \Rightarrow$ favor smooth functions
- $L(t) := \|t\|_0 \Rightarrow$ favor solutions with small # of non-zero elements
  - sparse solutions

Regularization in Learning from Data

- Employing the ℓ_0 constraint is a problem because the resulting constrained optimization is not convex
- It turns out that it's possible to use the ℓ_1-norm, with good sparsity selection
    - This problem is indeed convex
- Hence the use of the ℓ_1 norm in sparse regression
    - Compressive Sensing, LASSO
    - Bayesian compressive sengin, Bayesian LASSO

Sparse regression

Model:

$$y = f(x) \simeq \sum_{k=0}^{K-1} c_k \Psi_k(x)$$

with $x \in \mathbb{R}^n$, Ψ_k max order p, and $K = (p+n)!/p!/n!$

- N samples $(x_1, y_1), \ldots, (x_N, y_N)$
- Estimate K terms $c_0, \ldots, c_{K-1}$, s.t.

$$\text{min } ||\boldsymbol{y} - \boldsymbol{A}\boldsymbol{c}||_2^2$$

where $\boldsymbol{y} \in \mathbb{R}^N$, $\boldsymbol{c} \in \mathbb{R}^K$, $\boldsymbol{A}_{ik} = \Psi_k(x_i)$, $\boldsymbol{A} \in \mathbb{R}^{N \times K}$

With $N << K \Rightarrow$ under-determined

- Need some form of regularization

Regularization – Compressive Sensing (CS)

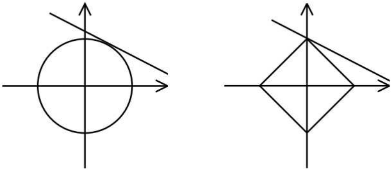
- ℓ_2-norm – Tikhonov regularization; Ridge regression:

$$\min \{\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{c}\|_2^2 + \|\boldsymbol{c}\|_2^2\}$$

- ℓ_1-norm – Compressive Sensing; LASSO; basis pursuit

$$\begin{aligned} \min \{\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{c}\|_2^2 + \|\boldsymbol{c}\|_1\} & \\ \min \{\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{c}\|_2^2\} & \quad \text{subject to } \|\boldsymbol{c}\|_1 \leq \epsilon \\ \min \{\|\boldsymbol{c}\|_1\} & \quad \text{subject to } \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{c}\|_2^2 \leq \epsilon \end{aligned}$$

$\Rightarrow$ discovery of sparse signals



Multilevel Multifidelity (MLMF) Methods – 1

We evolved MC estimation of the Sobol' sensitivity indices in two steps

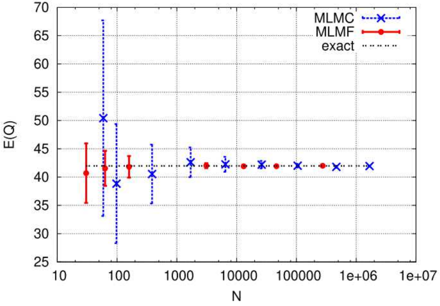
- We introduced the fitting of a PCE as a primary step, before using the PCE to estimate the sensitivity indices
  - This helps, but more is needed
- We introduced the regularization of the PCE fitting by introducing a norm of the solution vector into the objective function, or constraint
  - ℓ_2 and ℓ_1 norm
- ℓ_1-norm constrained regression facilitates the discovery of a *sparse* PCE that has good fit to the data
  - This helps, providing fitted sparse PCEs given available data

Multilevel Multifidelity (MLMF) Methods – 2

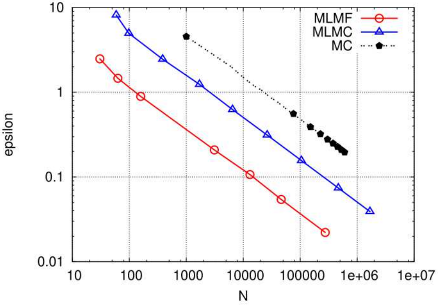
In particular, when the computational model is quite expensive, we still seek more reduction in the required number of samples

- Multilevel Multifidelity (MLMF) methods allow further savings by combining information judiciously from low/high-resolution and how/high-fidelity models
- Use many low resolution/fidelity model computations and a minimal necessary number of high resolution/fidelity model computations to achieve target accuracy with MC

Heat equation–MLMF vs. MLMC vs. plain MC

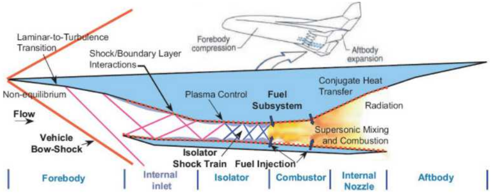


Expected Value

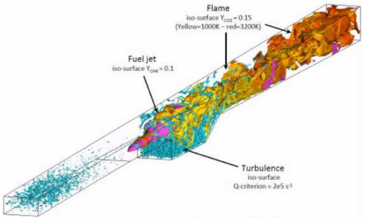


Accuracy ε

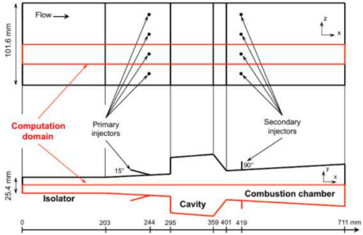
Supersonic Combusting Ramjet (scramjet)

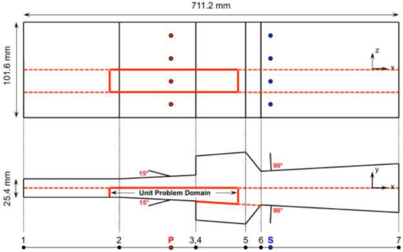


In flight



Numerical model





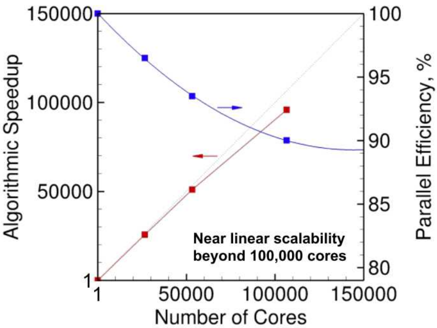
Scramjet–24 uncertain parameters

Parameter	Symbol	Range
Inflow boundary conditions		
Inlet		
Stagnation pressure	$p_{0,i}$	1.48 MPa $\pm 5\%$
Stagnation temperature	$T_{0,i}$	1550 K $\pm 5\%$
Mach number	M_i	$2.51 \pm 10\%$
Turbulence intensity	$I_i = u'_i/U_i$	$[0.0 - 0.05]$
Turbulence intensity ratio	$I_r = v'_i/u'_i$	1.0
Turbulence length scale	L_i	$[0.0 - 8.0]$mm
Boundary layer thickness	δ_i	$[2.0 - 6.0]$mm
Fuel injection $(36\%CH_4, 64\%C_2H_4)$		
Mass flux	$\dot{m}_f$	7.37×10^{-3} kg/s $\pm 10\%$
Static Temperature	T_f	300.0 K $\pm 5\%$
Mach Number	M_f	$1.0 \pm 5\%$
Turbulence intensity	$I_f = u'_f/U_f$	$[0.025 - 0.075]$
Turbulence length scale	L_f	$[0.02 - 1.0]$ mm
Wall boundary conditions		
Wall Temperature	T_w	Profile from KLE Expansion (10 params)
Turbulence model parameters		
Static Smagorinsky		
Modified Smagorinsky constant	C_R	$[0.01 - 0.016]$
Turbulent Prandtl number	Pr_t	$[0.5 - 1.7]$
Turbulent Schmidt number	Sc_t	$[0.5 - 1.7]$

LES Performed using RAPTOR Code Framework

Joe Oefelein – Sandia National Labs. – currently at Georgia Tech

- **Theoretical framework … (Comprehensive physics)**
  - Fully-coupled, compressible conservation equations
  - Real-fluid equation of state (high-pressure phenomena)
  - Detailed thermodynamics, transport and chemistry
  - Multiphase flow, spray
  - Dynamic SGS modeling (No Tuned Constants)
- **Numerical framework … (High-quality numerics)**
  - Staggered finite-volume differencing (non-dissipative, discretely conservative)
  - Dual-time stepping with generalized preconditioning (all-Mach-number formulation)
  - Detailed treatment of geometry, wall phenomena, transient BC's
- **Massively-parallel … (Highly-scalable)**
  - Demonstrated performance on full hierarchy of HPC platforms (e.g., scaling on ORNL CRAY XK7 TITAN architecture shown below)
  - Selected for early science campaign on next generation SUMMIT platform (ORNL Center for Accelerated Application Readiness, 2015 – 2018)



Global sensitivity analysis: Sobol indices

Global sensitivity analysis (GSA) (Saltelli:2004,2008)

- For a given quantify of interest (QoI) ...
- QoI variance decomposed into contributions from each parameter
- Sobol indices rank parameters by their contributions (Sobol:2003)

Total effect $$S_{T_i} = \frac{\mathbb{E}_{\lambda_{\sim i}}[\text{Var}_{\lambda_i}(f(\lambda)|\lambda_i)]}{\text{Var}(f(\lambda))}$$

S_{T_i} small $\Rightarrow$ low impact parameter $\Rightarrow$ fix value (i.e. dim. eliminated)

How to compute?

- Monte Carlo estimators (Saltelli:2002,2010) still prohibitive for LES
- **Our approach:** construct affordable surrogate models via polynomial chaos expansion (PCE)

Polynomial chaos expansions

A QoI (output) random variable can be expanded as follows:

$$f(\lambda(\xi)) = \sum_{\beta \in \mathcal{J}} c_\beta \Psi_\beta(\xi)$$

- c_β: PCE coefficients
- ξ: reference random vector (e.g., uniform, Gaussian)
- Ψ_β: multivariate orthonormal polynomial (e.g., Legendre, Hermite)
- β: multi-index, reflects order of polynomial basis

Orthonormality property
$\Rightarrow$ extract Sobol indices analytically from coefficients (no Monte Carlo!):

$$S_{T_i} = \frac{1}{\text{Var}(f(\lambda))} \sum_{\beta \in \mathcal{J}:\beta_i>0} c_\beta^2 \qquad \text{where} \qquad \text{Var}(f(\lambda)) = \sum_{0 \neq \beta \in \mathcal{J}} c_\beta^2$$

Sparse polynomial chaos expansions

Non-intrusive regression to compute expansion coefficients $Gc = f$:

$$\underbrace{\begin{bmatrix} \Psi_{\beta^1}(\xi^{(1)}) & \cdots & \Psi_{\beta^N}(\xi^{(1)}) \\ \vdots & & \vdots \\ \Psi_{\beta^1}(\xi^{(M)}) & \cdots & \Psi_{\beta^N}(\xi^{(M)}) \end{bmatrix}}_{G} \underbrace{\begin{bmatrix} c_{\beta^1} \\ \vdots \\ c_{\beta^N} \end{bmatrix}}_{c} = \underbrace{\begin{bmatrix} f(\lambda(\xi^{(1)})) \\ \vdots \\ f(\lambda(\xi^{(M)})) \end{bmatrix}}_{f},$$

Challenges:

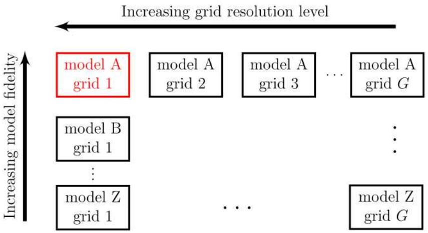
- Very few LES flow solves (data) available
- Large number of polynomial PCE basis (e.g. total-order degree 3 in 24 dimensions: 2925 terms)
- Extremely under-determined system ($N \gg M$)

Our approach: use compressed sensing to find sparse solution

$$\min_c \frac{1}{2}\| c \|_1 + \tau \| Gc - f \|_2^2$$

discover and retain only basis terms with high magnitude coefficients

Multilevel and multifdelity forms



Telescopic sum:

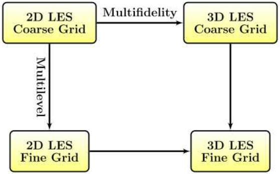
$$f_L(\lambda) = f_0(\lambda) + \sum_{\ell=1}^{L} f_{\Delta_\ell}(\lambda)$$

- ℓ indicates different grid levels or fidelity of models
- Δ_ℓ indicates difference between models ℓ and $\ell - 1$

Function approximation:

$$f_L(\lambda) \approx \hat{f}_L(\lambda) = \hat{f}_0(\lambda) + \sum_{\ell=1}^{L} \hat{f}_{\Delta_\ell}(\lambda)$$

High-D – ML/MF UQ Results



Two model forms and two mesh discretization levels

- Model form: 2D (LF) and 3D (HF) LES
- Meshes: $d/8$ and $d/16$

The P1 problem is considered (24 inputs).
Five QoIs extracted over a plane at $x/d = 100$.

- $\mathbb{E}_{y,t}$ stagnation pressure ($P_{0,mean}$)
- $\mathbb{E}_{y}$ RMS$_t$ stagnation pressure ($P_{0,rms}$)
- $\mathbb{E}_{y,t}$ Mach number (M_{mean})
- $\mathbb{E}_{y,t}$ turbulent kinetic energy (TKE$_{mean}$)
- $\mathbb{E}_{y,t}$ scalar dissipation rate (χ_{mean})

	2D	3D
$d/8$	1	204
$d/16$	25.5	1844

Relative computational cost for the model forms and discretization levels.

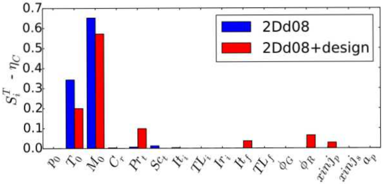
Optimize statistical accuracy given a limited number of high fidelity model evaluations by leveraging cheaper lower fidelity simulations.

GSA: uncertain + design parameter space (16d); 2D flow

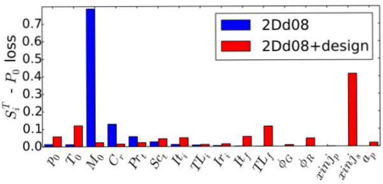
Setup

- Design parameters:
  - global equivalence ratio (ϕ_G)
  - ratio of equivalence ratios primary/secondary (ϕ_R)
  - location of primary and secondary injectors ($xinj_P$, $xinj_S$)
  - angle of primary injector (a_P).
- 220 simulations

Combustion Efficiency

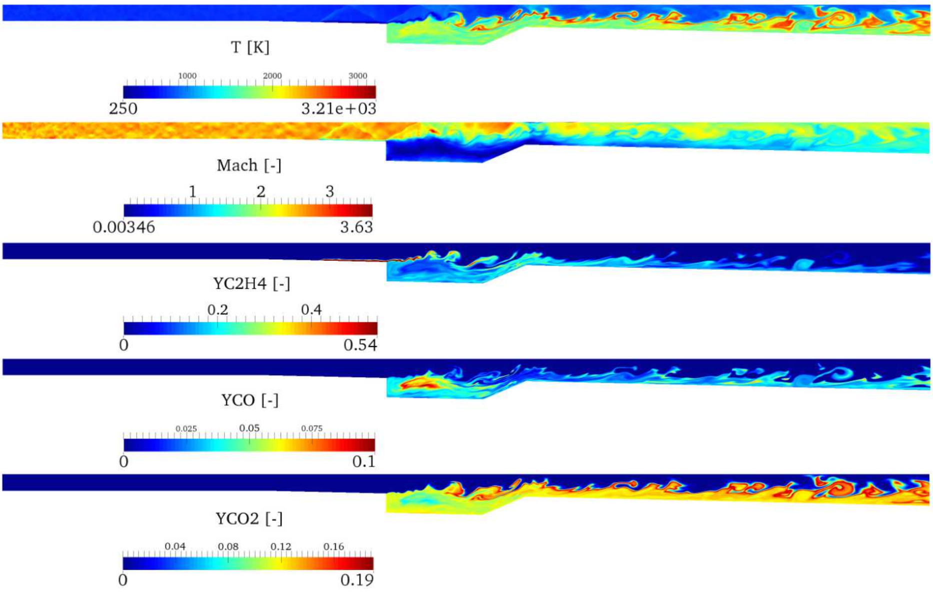


Stagnation Pressure Loss

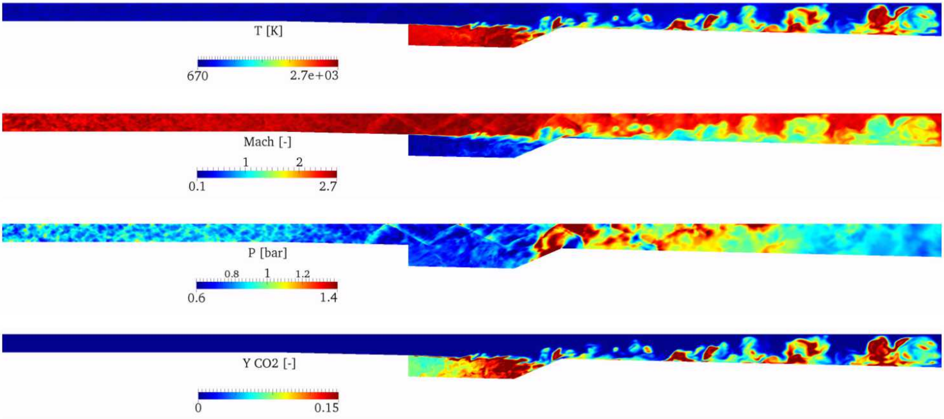


- Inlet Mach number (M_o) and stagnation temperature (T_o) remain dominant
- Location of the second injector important for ΔP_{stag}

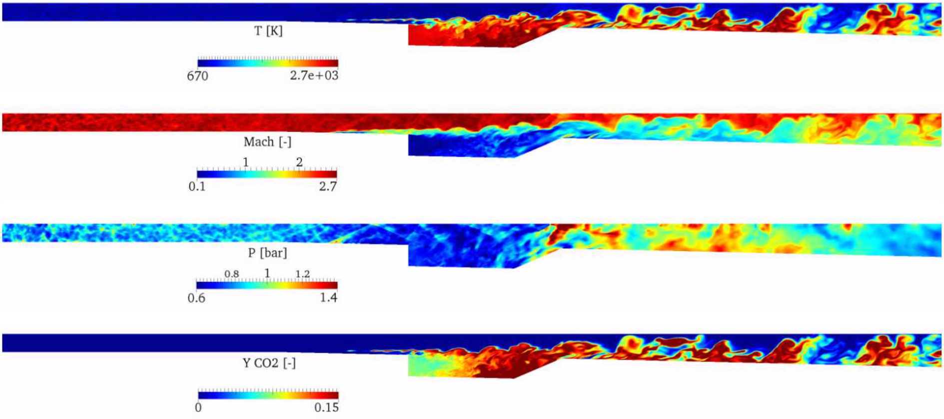
Instantaneous Flow Structure – 2D d32



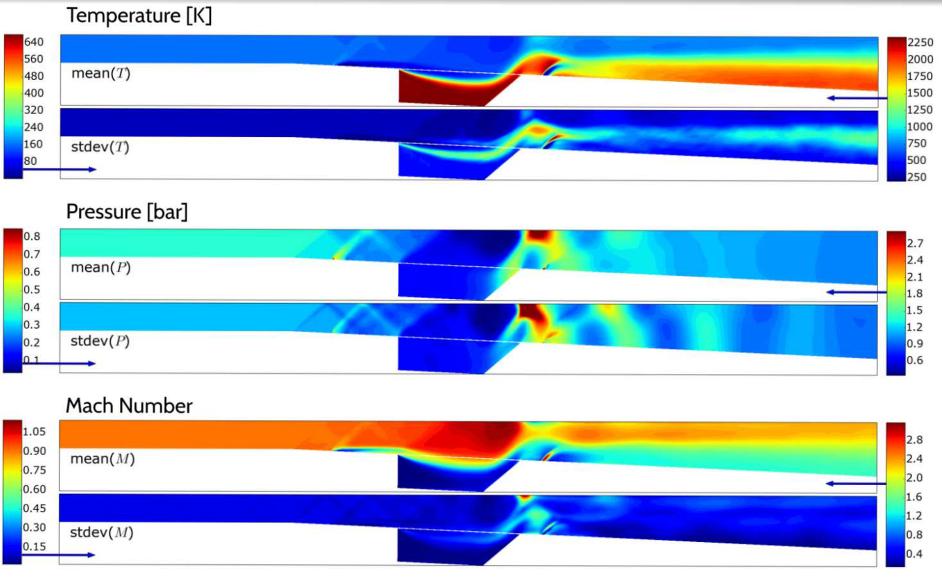
Instantaneous Flow Structure – z-0-cut – 3D d16



Instantaneous Flow Structure – z-inj-cut – 3D d16

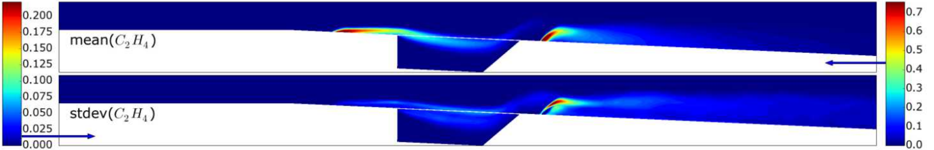


MC-Predicted Uncertainty in Mean Flow Quantities – 3D

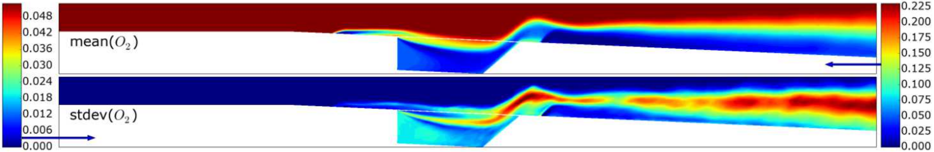


MC-Predicted Uncertainty in Mean Flow Quantities – 3D

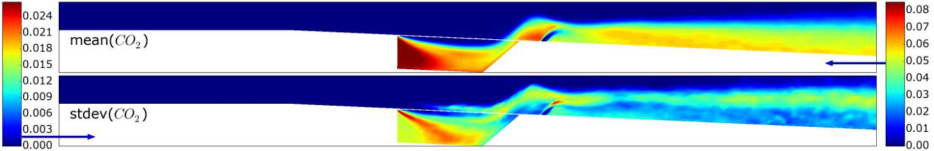
Ethane



Oxygen



Carbon Dioxide



Optimization Under Uncertainty – strategies

- Design parameters ϕ
- Uncertain parameters λ
- OUU statement ... example

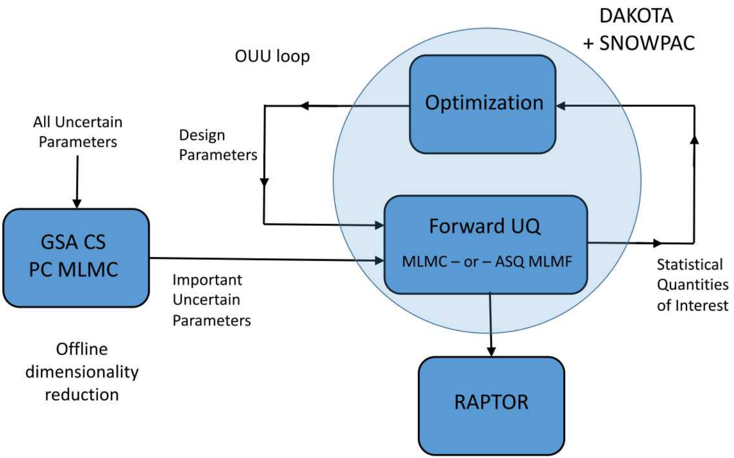
$$
\begin{aligned}
\phi^* &= \underset{\phi}{\operatorname{argmin}} \, \mathbb{E}_\lambda[f(\phi, \lambda)] \\
s.t. \quad & \mathbb{E}_\lambda[g(\phi, \lambda)] + 3\mathbb{V}^{\frac{1}{2}}[g(\phi, \lambda)] < \alpha \\
& \mathbb{E}_\lambda[h(\phi, \lambda)] - 3\mathbb{V}^{\frac{1}{2}}[h(\phi, \lambda)] > \beta
\end{aligned}
$$

  Many other statements, probabilistic, risk analysis, are possible
- Each step in the optimization strategy requires solving the forward UQ problem for the given ϕ
- Inherits and magnifies all the difficulties of forward UQ in high-dimensional complex models

Elements of OUU in large scale models

- We rely on two software libraries under development
  - DAKOTA – https://dakota.sandia.gov
  - SNOWPAC – https://bitbucket.org/fmaugust/nowpac.git
- Offline GSA/PC-smoothing/CS/MLMF and dimensionality reduction for uncertain parameters
- Reliance on surrogates and simplified models where possible
- Noise in objective function due to finite time-window averaging for flow statistics of interest
- Code/model failures are often encountered when exploring parameter spaces
- When noise is high and/or have failures, use MC/MLMC for forward UQ – build a Gaussian process over ϕ with MLMC – SNOWPAC
- If noise is small enough, and no failed samples, can use forward-UQ with PCE adaptive sparse quadrature (ASQ) and MLMF – DAKOTA
- DAKOTA handles overall optimization strategy

Optimization under uncertainty workflow



Closure

- Relevance of UQ in computational science
- Challenges
  - High dimensionality
  - Model complexity
  - Optimization under uncertainty
- Discussed
  - GSA, PC smoothing, CS, MLMC, MLMF
  - OUU, finite averaging noise, code failures
- Ongoing application for UQ & OUU in Scramjet design
- A highly multidisciplinary enterprise – applied math, probability, statistics, information theory, computations, data, physical modeling