SOMEWHAT BEYOND THE BASICS FOR UNCERTAINTY QUANTIFICATION IN THE PARAMETERIZED PDE SETTING

Max Gunzburger
Florida State University

ETICS 2017 Research School on Uncertainty in Scientific Computing
Porquerolles, October, 2017

Supported by
US Department of Energy Office of Science
US Air Force Office of Scientific Research

Contents the course

- A brief review of the basics of uncertainty quantification in the parametric PDE setting

- Global polynomial approximation
  - more on sparse grids
  - best s-term approximation
  - discrete least-squares approximation
  - compressed sensing approaches

- Piecewise polynomial approximation
  - adaptive methods based on hierarchical finite element and wavelet bases
  - application to high-dimensional discontinuity detection

- Multilevel and multifidelity methods

You can fool all the people some of the time

and some of the people all the time

but you cannot fool all the people all the time

Abraham Lincoln

"Narrow" definition of UQ used in the course

- Uncertainties are modeled probabilistically
- The input to output map is computational expensive
  - for us, that map involves solving PDEs
- Uncertainty enters into the PDEs through a finite set of random parameters

The curse of dimensionality

- Consider total degree interpolation in hypercubes

  interpolation in N dimensions using
  polynomials of total degree at most p

  e.g., $N = 2$, $p = 1 \Rightarrow a + bx + cy$

  and

  interpolation in N dimensions using tensor products
  polynomials of degree at most p in each direction
  (e.g., tensor product bases)

  e.g., $N = 2$, $p = 1 \Rightarrow a + bx + cy + dxy$

- For the same p, both have same the approximation properties
  - for interpolating smooth functions, the rate of convergence is the same

- What about the complexity?

$N =$ number of variables	$p =$ maximal degree of polynomials	$M =$ number of degrees of freedom	
		using total degree polynomial basis	using tensor product basis
3	3	20	64
	5	56	216
5	3	56	1,024
	5	252	7,776
10	3	286	1,048,576
	5	3,003	60,046,176
20	3	1,771	$> 1\times10^{12}$
	5	53,130	$> 3\times10^{15}$
100	3	176,851	$> 1\times10^{60}$
	5	96,560,646	$> 6\times10^{77}$
		$\Uparrow$	$\Uparrow$
		$\frac{(N+p)!}{N!p!}$	$(p+1)^N$

$\Longrightarrow$ $M =$ number of function evaluations (e.g., PDE solves)

- The curse of dimensionality =

  the explosive growth in the number of parameter degrees of freedom

  and therefore in the number of PDE
  solves needed for a certain accuracy

  as the number of parameters N and the degree p of the polynomials increases

- If

  one does not take advantage of any knowledge of the function
  being approximated other than it is "smooth enough"

  – total degree interpolation is relatively much more economical compared to total tensor product interpolation

  $\Longrightarrow$ even total degree interpolation suffers greatly from the curse

A BRIEF REVIEW OF THE BASICS OF UNCERTAINTY QUANTIFICATION IN THE PARAMETRIC PDE SETTING

Max Gunzburger - Florida State University

Types of noise

- Random parameters
- White noise
- Colored (or correlated) noise

Random parameters

- Random parameters – a (finite) set of parameters $\boldsymbol{y} \in \mathbb{R}^N$ appearing in the model
  - chemical reaction rates, material properties, diffusivities, permeabilities, etc. are often uncertain
  - independent $\Rightarrow$ the PDF $\varrho(\boldsymbol{y}) = \prod_{n=1}^{N} \varrho_n(y_n)$
  - dependent $\Rightarrow$ joint distribution is not separable
  - N could be large $\Rightarrow$ there could be many random parameters in a problem, e.g., lots of reacting chemical species
- Almost always, the PDF is assumed to be known for each parameter
  - in actuality, almost always, the PDF is not known

- One is lucky if one knows a range for the parameters
  - in which case, one usually assumes a uniform distribution over that range

- Not knowing anything, one usually assumes a Gaussian distribution, or, if one wants the parameter to be positive, a log normal distribution
  - for the Gaussian case, one usually assumes that the mean and variance are known
  - for the log-normal case, one assumes a mean and variance for a Gaussian distribution and then exponentiates it
  - again, in actuality, one may not really know the mean and variance

- Of course, such

  uncertainties in the uncertainties

  has spawned the now rather big industry of

  stochastic parameter identification

- The various parameters may have very different PDFs, whether known or assumed
  - for this reason

    if one samples the parameters to determine
    realizations of the solution of a PDE

    then

    one should sample anisotropically

- Another industry spawned in cases where the number of parameters is large is determining "unimportant" or "non-influential" parameters whose uncertainties need not be modeled

  - screening methods, sensitivity analyses, ANOVA, expert opinions, experimental data, ...

  - warning: two parameters that are both non-influential on their own, can have influential interactions

  - warning: papers that claim that 1,000 parameters can be treated efficiently usually use examples where there are, in fact, very few significantly influential parameters

White noise

- White noise – random fields whose value at any point in a domain and/or at any time is sampled independently (according of a single PDF $\varrho(\omega)$) from its value at any other point and or any other time (values are i.i.d.)
  - (infinite) stochastic process
  - can be expressed in terms of an infinite number of parameters (e.g., Weiner's polynomial chaos expansions)
  - the most common case (and the one that makes the most sense) is Gaussian white noise
  - in a simulation, has to be approximated in terms of a finite number of parameters

- For simplicity, we assume the PDF is centered and its variance σ^2 is a constant

  – this is one reason why Gaussian white noise makes the most sense

    - only the Gaussian PDF is completely determined by its mean and variance

- The covariance of the white noise field is then given by

$$\mathrm{Cov}_{white}(\mathbf{x}, \mathbf{x}') = \sigma^2 \delta(\mathbf{x} - \mathbf{x}')$$

  – thus, the variance $\mathrm{Var}_{white}(\mathbf{x}) = \mathrm{Cov}_{white}(\mathbf{x}, \mathbf{x})$ of a white noise random field is infinite

  – translations:

    - white noise fields have a flat spectrum

    - white noise fields have infinite energy

    - white noise random fields are not physically realizable

- But white noise is (by far) the most used model for randomness
  - is it a sane thing to have inputs into, e.g., a PDE, that have infinite variance?
  - why do we get away with using white noise inputs?
- White noise is truncated
  - physically, the spectrum is not really flat for all frequencies
  - discretization truncates white noise
    - discretized white noise is much smoother than white noise itself
  - for elliptic and parabolic PDEs and other dissipative systems, the solution is much smoother than than the inputs
- One means for approximating white noise fields is to use orthogonal polynomial approximations

  $\Longleftarrow$ Weiner's polynomial chaos expansions

- Grid based discretizations are a popular method for discretizing white noise

  – let $\eta_{white}(\mathbf{x};\omega)$ denote a white noise random field whose values at points in the spatial domain D are i.i.d. samples drawn from a one-dimensional PDF $\varrho(\omega)$ having variance σ^2, and for simplicity, zero mean

  – let $\{D_n\}_{n=1}^N$ denote the cells of a meshing of D and let $|D_n|$ denote the volume of D_n

  – then, a grid-based discretization of the white noise field is defined as

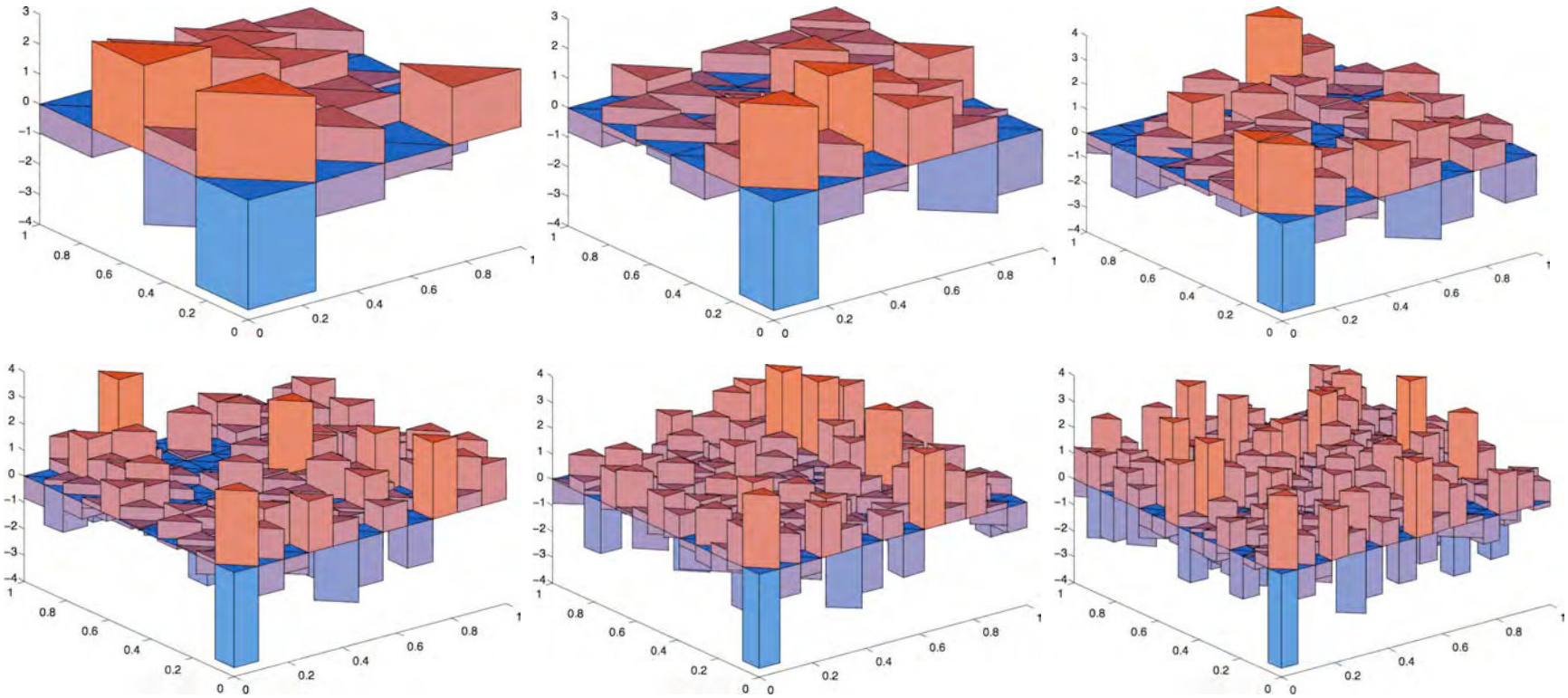
$$\eta_{white}^N(\mathbf{x};\boldsymbol{y}) = \mu_{white}(\mathbf{x}) + \sigma \sum_{n=1}^{N} \frac{1}{\sqrt{|D_n|}} \mathbb{1}_n(\mathbf{x}) y_n(\omega) \approx \eta_{white}(\mathbf{x};\omega)$$

  where $y_n(\omega)$, $n = 1, \ldots, N$, are i.i.d. samples drawn from $\varrho(\omega)$

  - discretized white noise is finite dimensional

    it is expressed in terms of a finite number of parameters

  – this is a piecewise constant approximation



Realizations of grid-based discretized white noise over in a square subdivided into 32, 72, 128, 242, 338, and 512 triangles

– a piecewise constant function is much smoother
than a white noise function

– discretized white noise has finite variance

$$\mathrm{Var}^N_{white}(\mathbf{x}) = \frac{\sigma^2}{|D_n|} \quad \text{for } \mathbf{x} \in D_n$$

- note that as the grid size tends to zero,
  the variance of discretized white noise goes to infinity

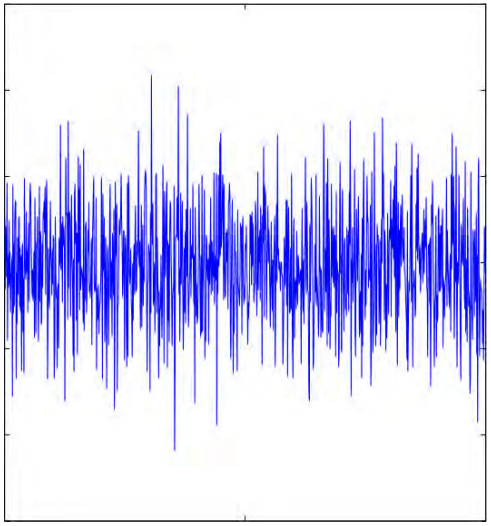
– but, for finite grid sizes, nothing bad happens

- you do not get any NAN's

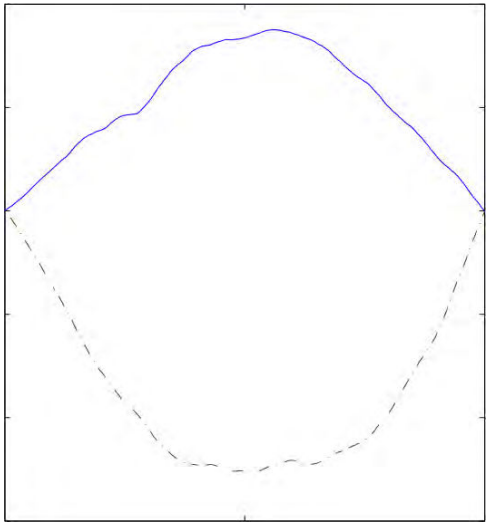
• Is there a notion of convergence?

$$\mathrm{Cov}^N_{white}(\mathbf{x}, \mathbf{x}') \to \mathrm{Cov}_{white}(\mathbf{x}, \mathbf{x}') = \delta(\mathbf{x} - \mathbf{x}') \quad \text{as} \quad \max_n |D_n| \to 0$$

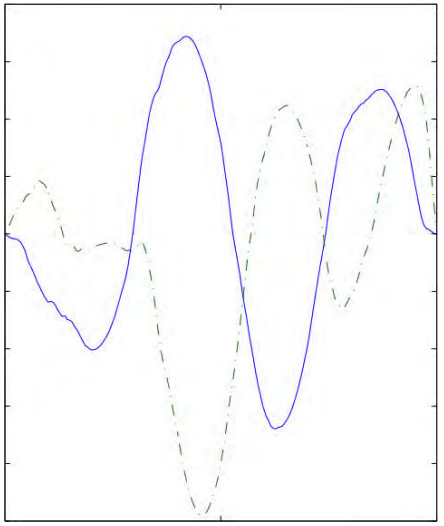
• What about the solution of a PDE?



white noise input



additive noise solutions



multiplicative noise solutions

A realization of the white noise input in one-dimension and two realizations of the solution of a simple two-point boundary value problem for a second-order ordinary differential equation with additive and linear multiplicative noise. The ranges of the ordinates in the three plots are $[-150, 150]$, $[-0.3, 0.2]$, *and* $[-0.1, -.08]$.

– solutions of elliptic (and parabolic) differential equations are much smoother than the input white noise

- Note that the number N of random parameters is intimitely tied to the grid size
  - as the grid size decreases, the number or parameters increases
  - in fact, $N = O(h^{-d})$ where d is the spatial dimension
  - in fact, N could be humongous

- Why do we get away with large N?
  - because we are doing Monte Carlo sampling of the parameter hypercube $\Gamma = \prod_{n=1}^{N} \Gamma_n$
    - we have $\Gamma_n = \Gamma_\omega$ for all n
    - where $\Gamma_\omega =$ the support of $\varrho(\omega)$

Correlated (colored) noise

- Correlated (or colored) noise – random fields whose value at any point in a domain and/or at any time is sampled according of a single PDF $\varrho(\omega)$ but which are not independent
  - the values of the correlated noise field are identically distributed
  - but the values are not independent
  - along with a function that determines the mean value of the field at each point,
    - the field comes with a correlation or covariance function that constrains the values of the field at points
  - (infinite) stochastic process
  - can be expressed in terms of an infinite number of parameters
  - in a simulation, has to be approximated in terms of a finite number of parameters

- Now the uncertainties in the uncertainties are even worse
  - one seldom knows the mean function, and even more so, the correlation function
  - sometimes one can assume a type of realistic noise for which the correlation is known
    - Brownian noise, pink noise, fractional Brownian motion, mean-reverting noise (Ornstein-Ulenbeck process), ...
  - most often, though, one simply assumes a correlation or covariance function, e.g.,

$$\begin{aligned}
\text{exponential} \quad & \text{Cov}(\mathbf{x};\mathbf{x}'') &&= e^{-\frac{|\mathbf{x}-\mathbf{x}'|}{L}} \\
\text{Gaussian} \quad & \text{Cov}(\mathbf{x};\mathbf{x}') &&= e^{-(\mathbf{x}-\mathbf{x}')^T\Sigma^{-1}(\mathbf{x}-\mathbf{x}')} \\
\text{Matérn} \quad & \text{Cov}(\mathbf{x};\mathbf{x}') &&= \\
& && \sigma^2 \frac{1}{\Gamma(\nu)2^{\nu-1}} \left(\sqrt{2\nu}\frac{|\mathbf{x}-\mathbf{x}'|}{L}\right)^{\nu} K_\nu \left(\sqrt{2\nu}\frac{|\mathbf{x}-\mathbf{x}'|}{L}\right)
\end{aligned}$$

- note that covariance function identification problem is a really big problem
  - e.g., in discretized models, one has to identify a matrix whose size depends on the number or elements or nodes in a grid

- Working directly with mean and covariance functions is not easy
  - e.g., how does one sample according the a given covariance function?

- One instead determines a representation of the noise in terms of random parameters
  - because correlated noise is an (infinite) stochastic process, the number of parameters in any representation is necessarily infinite
  - of course, in practice, one has to use truncated expansions so that only a finite number of parameters are involved
  - of course, truncating an infinite expansion introduces an error
    - this source of error is often ignored

- Ignoring this error (as is often done) is not a good thing
  - the truncation error should be commensurate with the spatial discretization error

    $\Rightarrow$ the number of terms N one should keep should be connected to the spatial grid size h
  - for correlated noise, the tie between N and h is weaker than that for white noise discretizations, but it is still there

- There are several means available for representing correlated noises in terms of random parameters, including global orthogonal polynomial expansions, grid-based methods, ...

  – perhaps the most popular means is to use Karhunen-Loève expansions

  – KL expansions are based on the eigenpairs $\{\lambda_n, b_n(\mathbf{x})\}_{n=1}^{\infty}$ of the covariance function

    - of course, in practice, one uses the eigenpairs of a discretized covariance function

    - this is another source of error that is almost always ignored

  – truncated KL expansions retain the dominant eigenpairs

  – thus, for correlated random field with a given covariance function and, for simplicity, zero mean, we have $\Longrightarrow$

$\underbrace{\eta(\mathbf{x},\omega)}$

"given" correlated random field with

values drawn from a centered PDF $\varrho(\omega)$

$$= \underbrace{\sum_{n=1}^{\infty} \sqrt{\lambda_n} b_n(\mathbf{x}) y_n(\omega)}$$

KL expansion

λ_n^h in non-increasing order

parameters $y_n(\omega)$ uncorrelated and drawn from $\varrho(\omega)$

$\underbrace{\approx}$

errors due to truncation of the KL expansion and spatial discretization of the covariance function

$$\underbrace{\sum_{n=1}^{N} \sqrt{\lambda_n^h} b_n^h(\mathbf{x}) y_n(\omega)} = \underbrace{\eta^{N,h}(\mathbf{x},\omega)}$$

truncated and spatially discretized KL expansion

discretized random field

- KL expansions do two wonderful things
  - they represent a random field (for which only the mean and covariance functions are known) in terms of parameters
    - having a parametric representation means we can evaluate a realization of the random field by merely choosing values for the random parameters
  - the random parameters in KL expansions are uncorrelated
    - this is great because we did not know how to, e.g., sample correlated parameters

- But, do we really know how to sample uncorrelated parameters? No we don't
  - what's the problem? $\Longrightarrow$ independent $\Rightarrow$ uncorrelated but uncorrelated $\not\Rightarrow$ independent

– example:

- let y_1 be a random variable uniformly distributed in $[-1, 1]$
- let $y_2 = y_1^2$
- clearly, y_1 and y_2 are not independent
- however $\text{Cov}(y_1, y_2) = 0$
  $\Rightarrow$ y_1 and y_2 are uncorrelated

– in fact, one can guarantee that a set uncorrelated random variables is independent only if the random variables are multivariate Gaussian

– this fact is almost always ignored

- for other PDFs, the uncorrelated random variables in a KL expansion are assumed to be independent
- usually this assumption is made tacitly without an explicit statement

- Log-normal correlated random fields are of interest because
  - they reputedly arise in practice, e.g., in subsurface media
  - they are convenient to use for coefficients of elliptic equations because they are positive
    - whereas, of course, a Gaussian distributed coefficient would not be
  - in the Karhunen-Loève setting, it is often assumed that

$$a(\mathbf{x};\boldsymbol{y}) \approx a_{min} + e^{\mu(\mathbf{x})+\sum_{n=1}^{N}\sqrt{\lambda_n}\,b_n(\mathbf{x})\,y_n} \qquad (*)$$

    where $\{y_n(\omega)\}_{n=1}^{N}$ are uncorrelated multivariate Gaussian random variables, and are therefore independent

– question: is the given covariance function

for the log-normal field

or is it for

the Gaussian field in the exponent?

– so, a potential issue with this approach that is seldom addressed is that

- if one uses the given mean and covariance functions to determine the truncated KL expansion of the Gaussian random field appearing as the exponent

then

-the resulting log-normal random field will have different mean and covariance functions which may not be the the physical one

– if one wants

- the log-normally distributed coefficient $a(\mathbf{x};\boldsymbol{y})$ given by (*) to correspond to a given mean function $\widetilde{\nu}(\mathbf{x})$ and covariance function $\widetilde{\mathsf{Cov}}(\mathbf{x},\mathbf{x}')$

then

- the mean $\mu(\mathbf{x})$ and covariance function $\mathsf{Cov}(\mathbf{x},\mathbf{x}')$ of the Gaussian field whose KL expansion is the exponent in (*) should be chosen as

$$\mu(\mathbf{x}) = \ln\left(\widetilde{\nu}(\mathbf{x})\right) - \frac{1}{2}\ln\left(1 + \frac{\widetilde{\mathsf{Cov}}(\mathbf{x},\mathbf{x})}{\left(\widetilde{\nu}(\mathbf{x})\right)^2}\right)$$

and

$$\mathsf{Cov}(\mathbf{x},\mathbf{x}') = \ln\left(1 + \frac{\widetilde{\mathsf{Cov}}(\mathbf{x},\mathbf{x}')}{\widetilde{\nu}(\mathbf{x})\widetilde{\nu}(\mathbf{x}')}\right)$$

- For general non-Gaussian correlated random fields, our only recourses are to either

  – assume that the KL parameters are independent (what is usually done)

    - in which case we simply express the random field
      in terms of its KL expansion

or

  – express the non-Gaussian random field in terms of a Gaussian random field and then use the KL expansion for the latter

  – the second approach requires knowledge of the CDF of the non-Gaussian random field

Solving PDEs with random inputs using stochastic Galerkin methods

- Some lingo
  - stochastic finite element methods
    $=$
    using a FEM for spatially discretizing a PDE with random inputs
  - stochastic Galerkin methods
    $=$
    discretization with respect to the random parameters is effected by a Galerkin method
  - weak form: given $\kappa(\mathbf{x}, \boldsymbol{y})$ and $f(\mathbf{x}, \boldsymbol{y})$, seek $u(\mathbf{x}, \boldsymbol{y}) \in X \times Y$ such that

$$\int_\Gamma \int_D \kappa \nabla u \cdot \nabla v \, d\mathbf{x} d\boldsymbol{y} = \int_\Gamma \int_D f v \, d\mathbf{x} d\boldsymbol{y} \quad \forall \, v(\mathbf{x}, \boldsymbol{y}) \in X \times Y$$

spatial function space X e.g., $H_0^1(D)$

parameter function space Y e.g., $L^2_\varrho(\Gamma) \Longleftarrow \int_\Gamma \big(f(\boldsymbol{y})\big)^2 \varrho(\boldsymbol{y}) \, d\boldsymbol{y} < \infty$

- Finite dimensional approximation subspaces
  - let $X_J \subset X$ with basis $\{\phi(\mathbf{x})\}_{j=1}^{J}$

    let $Y_M \subset Y$ with basis $\{\psi(\mathbf{x})\}_{m=1}^{M}$

  - then, we seek approximations $u_{J,M}(\mathbf{x}, \boldsymbol{y})$ of $u(\mathbf{x}, \boldsymbol{y})$ of the form

$$u_{J,M}(\mathbf{x}, \boldsymbol{y}) = \sum_{m=1}^{M} \sum_{j=1}^{J} c_{jm} \phi_j(\mathbf{x}) \psi_m(\boldsymbol{y}) \qquad (**)$$

  - then, the coefficients c_{jm}, $j = 1, \ldots, J$ and $m = 1, \ldots, M$, are found by solving

$$\int_{\Gamma} \int_{D} a(\nabla u_{J,M} \cdot \nabla \phi_{j'}) \psi_{m'} \, d\mathbf{x}\mathbf{t} d\boldsymbol{y} = \int_{\Gamma} \int_{D} f \phi_{j'} \psi_{m'} \, d\mathbf{x} d\boldsymbol{y} \qquad (***)$$

$$\forall\, j' = 1, \ldots, J,\ m' = 1, \ldots, M$$

  - note that to determine an approximation of $u(\mathbf{x}, \boldsymbol{y})$ at all points in the spatial domain D and parameter domain Γ only a single solution of (***) is needed

- Suppose that $F\big(u(\mathbf{x},\boldsymbol{y})\big)$ denotes an output of interest depending on a solution $u(\mathbf{x},\boldsymbol{y})$ of the PDE

  - e.g., F could be u itself or functionals of u such as spatial and/or temporal averages, maximum values, etc.

  – then, the quantity of interest is some statistical information about the output of interest F that is determined from

$$\int_\Gamma G\big(\, F\big(u(\mathbf{x},\boldsymbol{y})\big)\,\big)\,\varrho(\boldsymbol{y})\,d\boldsymbol{y}$$

  where G indicates what sort of statistics one wants to see, e.g.,

  $G(F) = F$ if we want the expected value $\mathbb{E}[F]$ of F

  $G(F) = F^2 - (\mathbb{E}[F])^2$ if we want the variance $\mathbb{V}[F]$ of F

  – then, we approximate by using the approximation $u_{J,M}$ in place of u $\quad\Longrightarrow$

$$\int_\Gamma G\big(\, F\big(u(\mathbf{x},\boldsymbol{y})\big)\,\big)\,\varrho(\boldsymbol{y})\,d\boldsymbol{y} \approx \int_\Gamma G\big(\, F\big(u_{JM}(\mathbf{x},\boldsymbol{y})\big)\,\big)\,\varrho(\boldsymbol{y})\,d\boldsymbol{y} \qquad (****)$$

- Thus, to obtain desired statistical quantity of interest, one single discrete system has to be solved
  - however, note that (***) is a system of JM equations in JM unknowns
- Of course, the right-hand side of (****), i.e., the approximate quantity of interest

$$\int_\Gamma G\big(\, F\big(u_{JM}(\mathbf{x},\boldsymbol{y})\big)\,\big)\,\varrho(\boldsymbol{y})\,d\boldsymbol{y}$$

has to be approximated as well because, in general, we cannot evaluate the integral exactly

  - e.g., using a Monte Carlo approach, we would have the approximation

$$\int_\Gamma G\big(\, F\big(u_{JM}(\mathbf{x},\boldsymbol{y})\big)\,\big)\,\varrho(\boldsymbol{y})\,d\boldsymbol{y} \approx \frac{1}{M}\sum_{m=1}^{M} G\big(\, F\big(u_{JM}(\mathbf{x},\boldsymbol{y}_m)\big)\,\big)\,\varrho(\boldsymbol{y}_m)$$

if $\boldsymbol{y}_m$ is randomly uniformly sampled

- Many types of bases for Y_M have been considered

  - global polynomial, local polynomial, wavelet, ...

- The most studied type of bases for Y_M are polynomial chaos bases

  polynomial chaos

  $=$

  global orthogonal polynomial approximation in parameter space

  – more precisely, products of one-dimensional global orthogonal polynomials are used

- PC $=$ polynomial chaos $\Leftarrow$ Gaussian PDF

  $\Rightarrow$ use Hermite polynomials (orthogonal with respect to the PDF)

- gPC $=$ generalized polynomial chaos $\Leftarrow$ other PDFs

  $\Rightarrow$ use polynomials that are orthogonal with respect to the PDF

- Good things about global orthogonal polynomial approximation

  – because of the hierarchical nature of one-dimensional global orthogonal polynomials

    - the degree of the k^{th} polynomial is exactly $k-1$

    a basis of multivariate orthogonal polynomials can be constructed that exactly spans the space of total degree polynomials

  – by solving a single discrete system

    - no sampling is needed

    one obtains an approximation of the solution of a PDE with random inputs at every point in the spatial domain and every point in the parameter domain

- Bad things about global orthogonal polynomial approximation

  – in general (e.g., for nonlinear problems) one has to solve a $JM \times JM$ discrete system

    - for problems that are linear in the solution function, it is possible to (partially) sparsify the discrete system

    - substantial effort (mostly in the linear case) has been devoted to developing efficient means for solving polynomials chaos discrete systems

Solving PDEs with random inputs using sampling methods

- Sampling schemes
  - sample M points $\{\boldsymbol{y}_m\}_{m=1}^M$ in the parameter domain Γ
  - for $m = 1, \ldots, M$, solve the spatially approximated PDE

$$\int_D a(\mathbf{x}, \boldsymbol{y}_m) \nabla u^h(\mathbf{x}, \boldsymbol{y}_m) \cdot \nabla v^h(\mathbf{x})\, d\mathbf{x} = \int_D f(\mathbf{x}, \boldsymbol{y}_m) v^h(\mathbf{x})\, d\mathbf{x}$$

    to obtain the set of spatially approximated solutions $\{u^h(\mathbf{x}, \boldsymbol{y}_m)\}_{m=1}^M$
  - statistical information about the solution of the PDE, or more often about functionals of the solution of the PDE, can be obtained from the set of solutions in much the same way as for the stochastic Galerkin approach

– thus, to determine statistical information in the sampling setting

- one solves M discrete systems of size $J \times J$

$\Longleftarrow$ non-intrusive methods

compared to the polynomial chaos case for which

- one solves a single discrete system of size $JM \times JM$

$\Longleftarrow$ intrusive methods

• What should guide the selection of sampling points?

– if one is interested in using them the determine statistical information

- so that quadrature rules for approximating integrals over the parameter domain are used

then

- the points should be selected so that they are good for quadrature

- if one is interested in using the points to build surrogates, e.g., interpolants, then
  - the points should be selected so that they are good for surrogate construction

- The list of sampling schemes is, of course, very, very, very long
  - among the ones we do not discuss is nonintrusive polynomial chaos approximations
    - still use orthogonal polynomials
    - but now select a set of points at which the approximation is "sampled"

- Aside: sampling schemes are stochastic Galerkin schemes
  - pick a set of points in Γ
  - approximate the solution of the PDE by using the Lagrange polynomials
    - the polynomials that are 1 at one point and 0 at all the others

    as a basis (this also a global polynomial approximation)
  - then, the $JM \times JM$ stochastic Galerkin discrete system becomes block diagonal
    - it reduces to M discrete systems, each of size $J \times J$

- The most popular algorithm (by far) for treating uncertainty, despite the fact that
  - convergence is in expectation
    $\Rightarrow$ more sampling does not necessarily make the error smaller
  - convergence is very slow
    $\Rightarrow \approx \frac{\sigma}{\sqrt{M}}$ $\sigma^2 =$ variance $M =$ number of samples
- Popular because
  - simple to implement
  - sequential sampling, i.e., the next point sampled does not care about where the previously sampled points are located
  - convergence is independent of the dimension N, i.e., of the number of random parameters
    $\Longrightarrow$ MC does not suffer from the curse of dimensionality

- Monte Carlo is used a lot to approximate integrals
  - certainly not good for interpolation
  - for integration, MC is indeed simple to implement, e.g.,

    one randomly samples M points $\{\boldsymbol{y}_m\}_{m=1}^M$ in Γ and then

$$\int_\Gamma f(\boldsymbol{y})\varrho(\boldsymbol{y})\,d\boldsymbol{y} \approx \begin{cases} \dfrac{1}{M}\displaystyle\sum_{m=1}^{M} f(\boldsymbol{y}_m) \\ \qquad \text{if one samples according to the PDF } \varrho(\boldsymbol{y}) \\ \dfrac{1}{M}\displaystyle\sum_{m=1}^{M} f(\boldsymbol{y}_m)\varrho(\boldsymbol{y}_m) \\ \qquad \text{if one samples uniformly} \end{cases}$$

  - is this really easy to implement for large N? even if Γ is a hypercube?

- Suppose one samples uniformly and suppose the PDF $\varrho(\boldsymbol{y})$ is short tailed (goes to zero pretty fast)
  - lots of sampling [and doing the expensive function evaluation $f(\boldsymbol{y}_m)$] is done where $\varrho(\boldsymbol{y})$ is small
  - this can get worse as N increases
    - recall that the volume of the ball inscribed in a hypercube goes to zero as $N \to \infty$
- Suppose one samples according to the PDF $\varrho(\boldsymbol{y})$
  - for general $\varrho(\boldsymbol{y})$, not easy to do, even in one dimension
  - in high dimension, only realistic for separable PDFs, i.e., $\varrho(\mathbf{y}) = \prod_{n=1}^{N} \varrho_n(\mathbf{y}_n)$, assuming one can efficiently sample the one dimensional PDFs
  - easiest to do for radial PDFs, i.e., $\varrho(\mathbf{y}) = \varrho(|\mathbf{y} - \mathbf{y}_0|)$ such as spherical Gaussian PDFs

Quasi-Monte Carlo

- The slow convergence of Monte Carlo has spawned a huge industry in alternatives that are "better" than MC but for which we still have, e.g.,

$$\int_\Gamma f(\boldsymbol{y})\varrho(\boldsymbol{y})\,d\boldsymbol{y} \approx \frac{1}{M}\sum_{m=1}^{M} f(\boldsymbol{y}_m)\varrho(\boldsymbol{y}_m)$$

  but now the sample points $\{\boldsymbol{y}_m\}_{m=1}^M$ are sampled in different ways
  - both deterministic and probabilistic, some sequential and some not, alternative sampling strategies have been invented
    - variance reduction techniques
    - QMC sequences – Halton, Sobol, Faure, ... ad infinitum; Hammersley
    - Latin hypercube
    - importance sampling
    - stratified sampling
    - lattice sampling
    - othogonal arrays
    - multilevel Monte Carlo
    - ⋮
    - ad nauseam

- Many of these methods have an

$$\text{error} \propto \frac{(\ln M)^{N+s}}{M}$$

  - for "small" N, get linear convergence

    - better than the $1/\sqrt{M}$ convergence of MC

  - for "large" N, the logarithmic term dominates

    - drats! the curse of dimensionality bites us again

"Better" quadrature rules

- Everyone knows that in one dimension and for smooth integrands one can do "better" with weighted rules compared to simple averaging rules

$$\int_\Gamma f(\boldsymbol{y})\varrho(\boldsymbol{y})\,d\boldsymbol{y} \approx \begin{cases} \displaystyle\sum_{m=1}^{M} w_m f(\boldsymbol{y}_m) \\ \qquad \text{if one samples according to the PDF } \varrho(\boldsymbol{y}) \\ \displaystyle\sum_{m=1}^{M} f(\boldsymbol{y}_m)\varrho(\boldsymbol{y}_m) \\ \qquad \text{if one samples uniformly} \end{cases}$$

  - of course, in one dimension, Gauss rules are the most beautiful example of how to define "better" rules
    - this is why in (very!) low dimension, tensor products of Gauss rules have proven to be very useful

  - but as we saw, tensor products should be avoided like ebola

- What about interpolatory rules based on total degree interpolation?
  - good quadrature points in a hypercube, even in low dimensions, are not known

- Then along came Smolyak

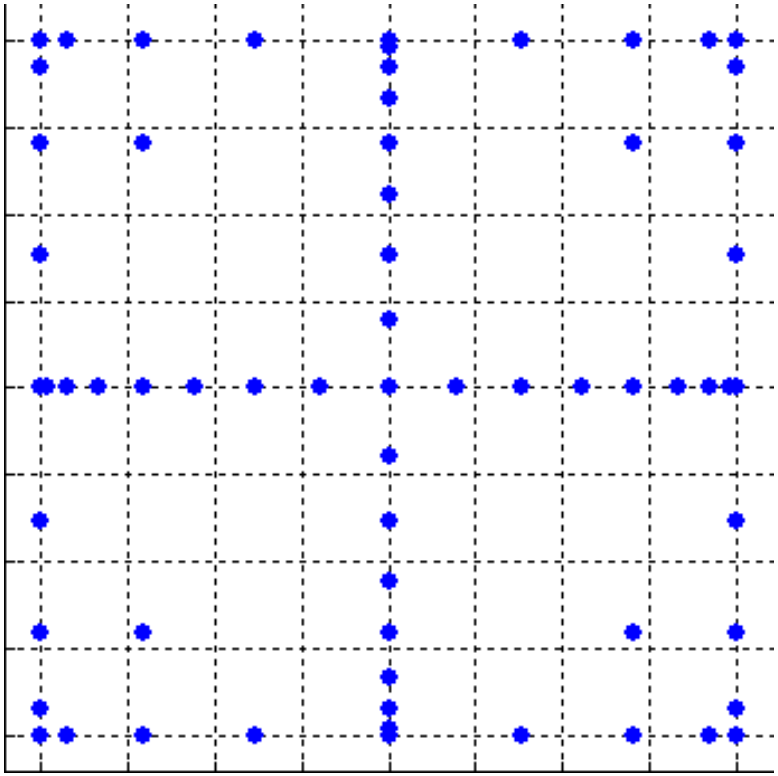
Stochastic collocation methods

- Smolyak (or sparse) grids (for interpolation and quadrature) are a judiciously chosen subset of tensor product grids
  - for the same precision
    - for interpolating or integrating the same polynomial space exactly

    Smolyak grids
    - require more points that does total degree interpolation or quadrature
    - require much fewer points than does tensor product interpolation or quadrature

	total degree		sparse grid		tensor product
degrees of freedom	$\frac{(N+p)!}{N!p!}$	$<$	$O\left(p(\ln p)^{N-1}\right)$	$\ll$	$(p+1)^N$

For three types of grids in $\mathbb{R}^N$, degrees of freedom for interpolatory quadrature rules having the same convergence behavior as total degree interpolatory quadrature using polynomials of degree at most p



A 65 point sparse grid

- Note the big holes in the grid, i.e., volumetric coverage is very bad
  - why do sparse grids work?

- If the function being interpolated or integrated is very smooth, the big holes do not matter
  - sparse grids beat Monte Carlo, quasi-Monte Carlo, etc. for moderate dimensions N
    - e.g., for $N = 10$ or so, sparse grids beat MC, etc. badly

- The need for smoothness

M	SG estimate	SG error	MC estimate	MC error
1	4.000	1.167	0.00000	5.16771
13	64.000	58.832	0.00000	5.16771
85	-42.667	47.834	3.01176	2.15595
389	-118.519	123.686	4.77121	0.39650
1457	148.250	143.082	5.15216	0.01555
4865	-24.682	29.850	5.41994	0.25226
exact	5.16771	–	5.16771	–

Comparison of sparse grid and Monte Carlo approximations of the integral of a function with a jump discontinuity.

In what we have discussed so far, where are we with respect to the curse of dimensionality?

- If our goal is to beat Monte Carlo for approximating statistical integrals
  - made some progress
  - still have far to go
- If our goal is to construct surrogates
  - still have even further to go

ORTHOGONAL POLYNOMIAL APPROXIMATIONS

Max Gunzburger - Florida State University

Prepared with
Clayton Webster, Hoang Tran, and Guannan Zhang - Oak Ridge National Laboratory

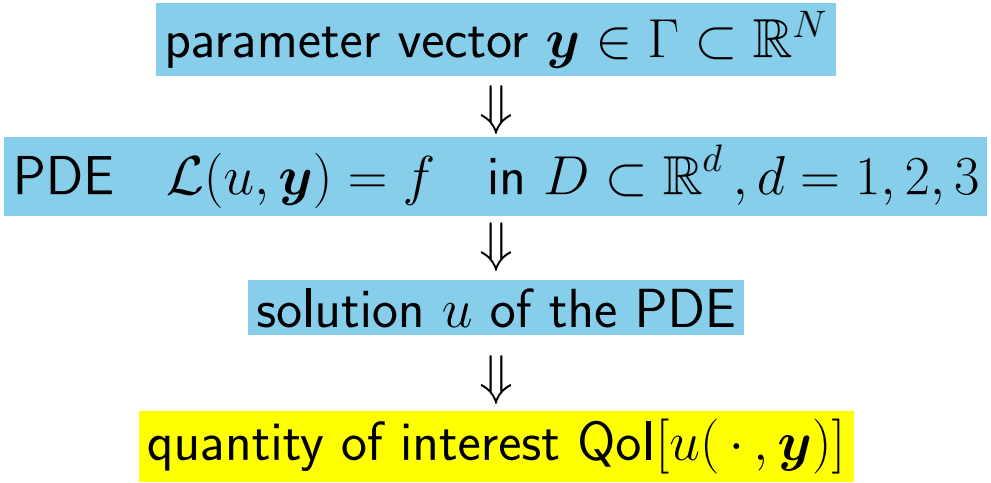
SETTING

High-dimensional parameterized PDEs

- PDE $\Longrightarrow \mathcal{L}(u, \boldsymbol{y}) = f \quad$ in $D \subset \mathbb{R}^d, d = 1, 2, 3$
  - the operator $\mathcal{L}$, linear or nonlinear, depends on a
    vector of d parameters $\boldsymbol{y} = (y_1, y_2, \ldots, y_N) \in \Gamma = \prod_{n=1}^{N} \Gamma_n$
    which can be deterministic or stochastic

- In the deterministic setting, the parameters $\boldsymbol{y}$ are known
  - the goal is, given $\boldsymbol{y} \in \Gamma$, to quickly determine an approximation of the solution map $\boldsymbol{y} \mapsto u(\,\cdot\,, \boldsymbol{y})$

- In the stochastic setting, the parameters $\boldsymbol{y}$ are subject to uncertainty
  due to, e.g., measurement error, incomplete description, unresolved scales
  modeled as a random vector $\boldsymbol{y} \in \Gamma$ with joint PDF $\varrho(\boldsymbol{y}) = \prod_{n=1}^{N} \varrho_n(y_n)$
  - the goal is to quantify the uncertainty in u or some statistical quantity depending on u, e.g., $\quad \mathbb{E}[u], \quad \mathbb{V}[u], \quad \mathbb{P}[u > u_0] = \mathbb{E}[\mathbf{1}_{\{u > u_0\}}]$


parameter vector $\boldsymbol{y} \in \Gamma \subset \mathbb{R}^N$
PDE $\mathcal{L}(u, \boldsymbol{y}) = f$ in $D \subset \mathbb{R}^d, d = 1, 2, 3$
solution u of the PDE
quantity of interest QoI$[u(\cdot, \boldsymbol{y})]$

- Deterministic Poisson problem

$$-\nabla \cdot \big(a(\mathbf{x})\nabla u(\mathbf{x})\big) = f(\mathbf{x}) \quad \forall \mathbf{x} \in D \subset \mathbb{R}^d \qquad u(\mathbf{x}) = 0 \quad \forall \mathbf{x} \in \partial D$$

  - assuming $a_{\min} = \min\limits_{\mathbf{x}\in D} a(\mathbf{x}) > 0$, by the Lax-Milgram lemma $\Longrightarrow$

    there exists a unique solution $u \in \mathcal{V} = H_0^1(D)$ with

$$\|u\|_{\mathcal{V}} = \|\nabla u\|_{L^2(D)} \leq \frac{1}{a_{\min}} \|f\|_{\mathcal{V}'}$$

- Let

$$\Gamma = [-1,1]^N,\ \boldsymbol{y} = (y_1, \ldots, y_N) \in \Gamma$$

assume $a = a(\mathbf{x}, \boldsymbol{y})$ depends on the parameter vector $\boldsymbol{y}$

and consider the parameterized boundary value problem

for any $\boldsymbol{y} \in \Gamma$, find $u(\,\cdot\,, \boldsymbol{y}) : D \to \mathbb{R}$ such that

$$-\nabla \cdot (a(\,\cdot\,, \boldsymbol{y}) \nabla u(\,\cdot\,, \boldsymbol{y})) = f(\,\cdot\,) \text{ in } D \qquad u(\,\cdot\,, \boldsymbol{y}) = 0 \text{ on } \partial D$$

- Invoke
  - continuity and coercivity (CC) assumption

$$0 < a_{\min} \le a(\mathbf{x}, \boldsymbol{y}) \le a_{\max}, \quad \forall \mathbf{x} \in \overline{D} \text{ and } \boldsymbol{y} \in \Gamma$$

- Lax-Milgram $\Longrightarrow$ existence and uniqueness of solution $u \in \mathcal{V} \otimes L^2(\Gamma)$

  - analytic continuation (AN) assumption

- the complex continuation of a, represented as the map $a : \mathbb{C}^N \to L^\infty(D)$, is an $L^\infty(D)$-valued analytic function on $\mathbb{C}^N$

- Examples

  1. $a(\mathbf{x}, \boldsymbol{y}) = a_0(\mathbf{x}) + \sum_{n=1}^{N} y_n \psi_n(\mathbf{x})$ $\Leftarrow$ affine
  2. $a(\mathbf{x}, \boldsymbol{y}) = a_0(\mathbf{x}) + \left(\sum_{n=1}^{N} y_n \psi_n(\mathbf{x})\right)^q, \quad q \in \mathbb{N}$ $\Leftarrow$ non-affine
  3. $a(\mathbf{x}, \boldsymbol{y}) = a_0(\mathbf{x}) + \exp\left(\sum_{n=1}^{N} y_n \psi_n(\mathbf{x})\right)$ $\Leftarrow$ motivated by truncated Karhunen-Loève expansion for the exponent

- Assume $a(\mathbf{x}, \boldsymbol{y})$ satisfies CC and AN; then

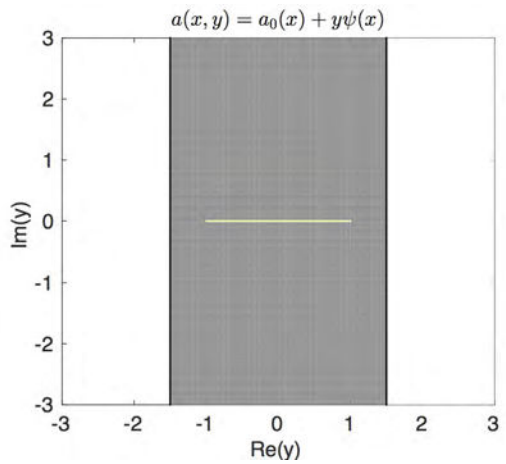
  $\boldsymbol{z} \mapsto u(\boldsymbol{z})$ is well defined and analytic in an open neighborhood of Γ in $\mathbb{C}^N$

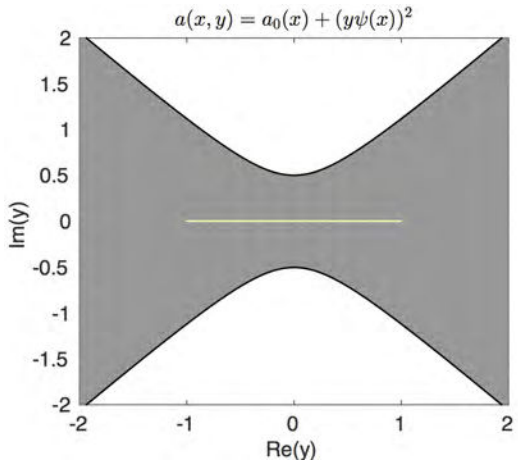
– if $a(\mathbf{x}, \boldsymbol{y})$ satisfies CC and AN, then for some open neighborhood $\widehat{\Gamma}$ of Γ in $\mathbb{C}^N$

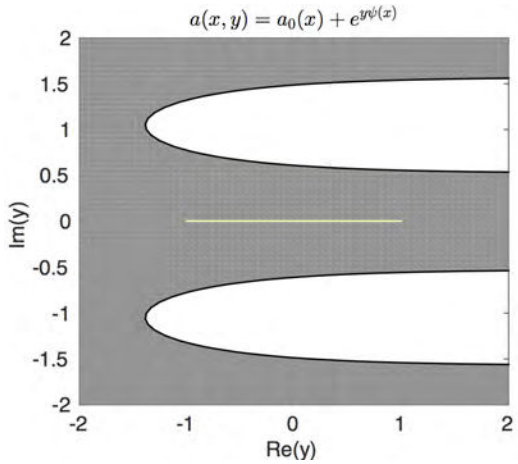
$$0 < \delta \leq Re(a(\mathbf{x}, \boldsymbol{z})) \quad \forall \mathbf{x} \in \overline{D} \text{ and } \boldsymbol{z} \in \widehat{\Gamma}$$

– $u(\boldsymbol{z})$ is well-defined and analytic in $\widehat{\Gamma}$

– we refer to $\widehat{\Gamma}$ as the domain of uniform ellipticity

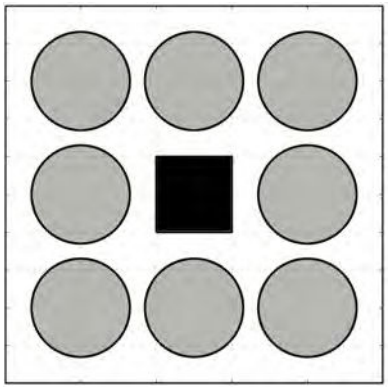


,


,


Domains of complex uniform ellipticity for some random fields
Here Γ is the line segment

- Example. We have 8 inclusions $D_1, D_2, \ldots, D_8$ within the $D = [0,1]^2$



  - assume a is a piecewise constant function over D and on each D_i the value of a is uniformly random within $[c - c_i, c + c_i]$ for some c and c_i satisfying $0 < c_i < c$

  - a can be represented as

$$a(\mathbf{x}, \boldsymbol{y}) = a_0(\mathbf{x}) + \sum_{n=1}^{8} y_i \psi_i(\mathbf{x}),$$

    where $a_0(\mathbf{x}) = c \qquad \psi_i = c_i \chi_{D_n} \qquad \boldsymbol{y} = (y_1, \ldots, y_8) \in \Gamma = [-1,1]^8$

GLOBAL POLYNOMIAL APPROXIMATIONS

- Observation: Solutions are often smooth with respect to the parameters

  Consequence: $u(\boldsymbol{y}, \cdot)$ can be approximated with respect to $\boldsymbol{y}$ by multivariate global polynomials

- Let $\boldsymbol{\nu} = (\nu_1, \ldots, \nu_N) \in \mathbb{N}_0^N$ and let $\Lambda \subset \mathbb{N}_0^N$ denote a multiindex set
  - these serve to define the multivariate polynomial space

$$\mathcal{P}_\Lambda(\Gamma) = \text{span}\left\{ \prod_{n=1}^{N} y_n^{\nu_n} \quad \text{with } \boldsymbol{\nu} \in \Lambda \right\}$$

  - let $\{\boldsymbol{\Psi}_{\boldsymbol{\nu}}\}_{\boldsymbol{\nu}\in\Lambda}$ denote a basis for $\mathcal{P}_\Lambda(\Gamma)$
    - e.g., multivariate Taylor, Legendre, Jacobi, Hermite, Lagrange, etc.
  - then, for $u_\Lambda \in \mathcal{V} \otimes \mathcal{P}_\Lambda(\Gamma)$, we have the representation

$$u_\Lambda(\mathbf{x}, \boldsymbol{y}) = \sum_{\boldsymbol{\nu}\in\Lambda} c_{\boldsymbol{\nu}}(\mathbf{x}) \boldsymbol{\Psi}_{\boldsymbol{\nu}}(\boldsymbol{y})$$

- Remarks
  - global polynomial approximations take advantage of the smooth dependence of u on the parameters
  - in this case, global polynomial approximations result in faster convergence than MC approximations
  - the evaluation of u_Λ requires the computation of coefficients $c_{\boldsymbol{\nu}}$
  - for now, we do not explicitly consider spatial discretizations

- Taylor basis
  - the Taylor expansion of $u(\mathbf{x}, \boldsymbol{y})$ is given by

$$u(\mathbf{x}, \boldsymbol{y}) = \sum_{\boldsymbol{\nu} \in \mathbb{N}_0^N} t_{\boldsymbol{\nu}}(\mathbf{x}) \boldsymbol{y}^{\boldsymbol{\nu}}$$

    where

$$\boldsymbol{y}^{\boldsymbol{\nu}} = \prod_{n=1}^{N} y_n^{\nu_n} \quad \text{and} \quad t_{\boldsymbol{\nu}}(\,\cdot\,) = \frac{1}{\boldsymbol{\nu}!} \partial_{\boldsymbol{y}}^{\boldsymbol{\nu}} u(\,\cdot\,, \mathbf{0}) \in \mathcal{V}$$

  - then u is approximated by the truncated expansion

$$u_\Lambda(\mathbf{x}, \boldsymbol{y}) = \sum_{\boldsymbol{\nu} \in \Lambda} t_{\boldsymbol{\nu}}(\mathbf{x}) \boldsymbol{y}^{\boldsymbol{\nu}}$$

Orthonormal bases

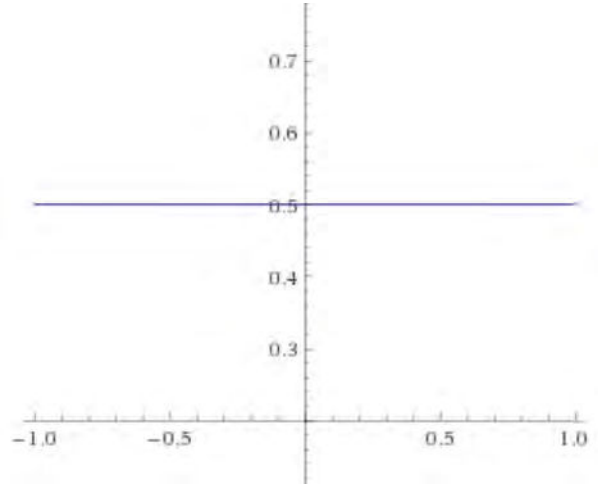
- For stochastic Galerkin, we choose $\{\boldsymbol{\Psi}_{\boldsymbol{\nu}}\}_{\boldsymbol{\nu}\in\Lambda}$ as an orthonormal basis
  - let $\varrho = \varrho(\boldsymbol{y})$ be a weight function on Γ $\Longleftarrow$ for us $\varrho =$ PDF

    $\Longrightarrow$ nonnegative and integrable in Γ
  - let $L^2_\varrho(\Gamma) = \left\{ f \ : \ \Gamma \to \mathbb{R}, \ \int_\Gamma f^2(\boldsymbol{y})\varrho(\boldsymbol{y})d\boldsymbol{y} < \infty \right\}$
  - define the inner product and norm on $L^2_\varrho(\Gamma) \Longrightarrow$ for $f_1,\ f_2 \in L^2_\varrho(\Gamma)$

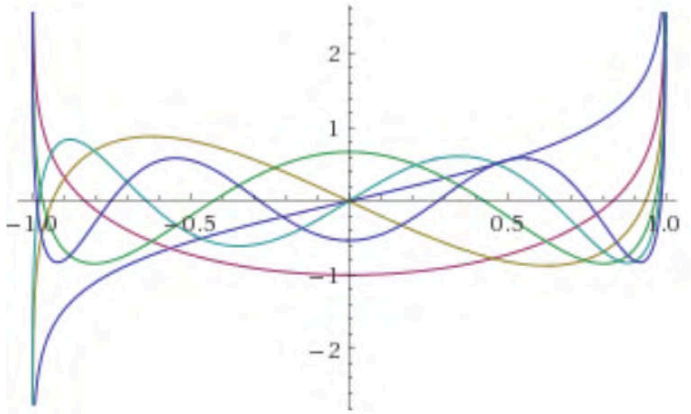
    $$\langle f_1, f_2\rangle = \int_\Gamma f_1(\boldsymbol{y}) f_2(\boldsymbol{y})\varrho(\boldsymbol{y})d\boldsymbol{y} \qquad \|f_1\|_{L^2_\varrho(\Gamma)} = \Big(\int_\Gamma |f_1(\boldsymbol{y})|^2 \varrho(\boldsymbol{y})d\boldsymbol{y} \Big)^{1/2}$$
  - the set $\{\boldsymbol{\Psi}_{\boldsymbol{\nu}}\}_{\boldsymbol{\nu}\in\Lambda}$ is called orthogonal with respect to ϱ if

    $$\langle \boldsymbol{\Psi}_{\boldsymbol{\nu}}, \boldsymbol{\Psi}_{\boldsymbol{\nu}'}\rangle = 0, \quad \text{if } \ \boldsymbol{\nu} \neq \boldsymbol{\nu}'$$
  - in addition, $\{\boldsymbol{\Psi}_{\boldsymbol{\nu}}\}_{\boldsymbol{\nu}\in\Lambda}$ is called orthonormal if $\langle \boldsymbol{\Psi}_{\boldsymbol{\nu}}, \boldsymbol{\Psi}_{\boldsymbol{\nu}}\rangle = 1 \quad \forall \boldsymbol{\nu} \in \Lambda$

- The univariate Legendre polynomials $\{L_n\}_{n=1}^{\infty}$ are orthogonal polynomials over the interval $\Gamma = [-1, 1]$ with respect to the weight function $\varrho(y) = \frac{1}{2}$



weight



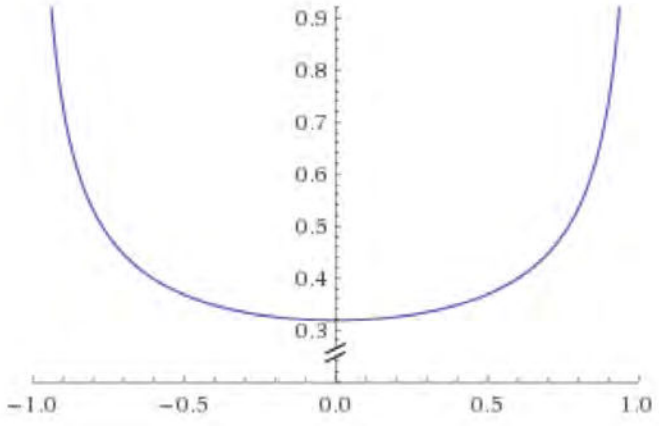
polynomials

– the multivariate Legendre polynomials $\{L_{\boldsymbol{\nu}}\}_{\boldsymbol{\nu} \in \mathbb{N}_0^N}$, given by

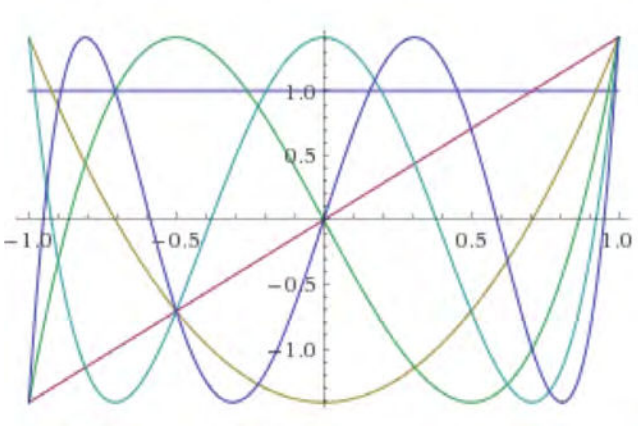
$$L_{\boldsymbol{\nu}}(\boldsymbol{y}) = \prod_{n=1}^{N} L_{\nu_n}(y_n) \qquad \forall \boldsymbol{\nu} \in \mathbb{N}_0^N$$

are orthogonal polynomials over the domain $\Gamma = [-1, 1]^N$ with respect to the weight function $\varrho(\boldsymbol{y}) = (1/2)^N$

- The univariate Chebyshev polynomials $\{T_j\}_{j=1}^{\infty}$ are orthogonal polynomials over the interval $\Gamma = [-1, 1]$ with respect to the weight function $\varrho(y) = \pi^{-1}(1 - y^2)^{-1/2}$



weight



polynomials

  - the multivariate Chebyshev polynomials $\{T_{\boldsymbol{\nu}}\}_{\boldsymbol{\nu} \in \mathbb{N}_0^N}$, given by

$$L_{\boldsymbol{\nu}}(\boldsymbol{y}) = \prod_{n=1}^{N} L_{\nu_n}(y_n) \qquad \forall \boldsymbol{\nu} \in \mathbb{N}_0^N$$

  are orthogonal polynomials over the domain $\Gamma = [-1, 1]^N$ with respect to the weight function $\varrho(\boldsymbol{y}) = \pi^{-N} \prod_{n=1}^{N} \frac{1}{(1-y_n^2)^{-1/2}}$

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

Some members of the Askey family of orthogonal polynomials

- Some well-know results
  - for any function $f \in L^2_\varrho(\Gamma)$

$$f(\boldsymbol{y}) = \sum_{\boldsymbol{\nu} \in \mathbb{N}_0^N} \widehat{f}_{\boldsymbol{\nu}} \boldsymbol{\Psi}_{\boldsymbol{\nu}}(\boldsymbol{y}) \quad \text{with } \widehat{f}_{\boldsymbol{\nu}} = \langle f, \boldsymbol{\Psi}_{\boldsymbol{\nu}} \rangle$$

  - Parseval's equality $\Longrightarrow$ $\|f\|^2_{L^2_\varrho(\Gamma)} = \sum_{\boldsymbol{\nu} \in \mathbb{N}_0^N} |\widehat{f}_{\boldsymbol{\nu}}|^2$

  - for $\Lambda \subset \mathbb{N}_0^N$ and $f_\Lambda(\boldsymbol{y}) = \sum_{\boldsymbol{\nu} \in \Lambda} \widehat{f}_{\boldsymbol{\nu}} \boldsymbol{\Psi}_{\boldsymbol{\nu}}(\boldsymbol{y})$

$$\|f - f_\Lambda\|_{L^2_\varrho(\Gamma)} = \min_{\widetilde{f} \in \mathcal{P}_\Lambda(\Gamma)} \|f - \widetilde{f}\|_{L^2_\varrho(\Gamma)}$$

- For a solution $u \in \mathcal{V} \otimes L^2_\varrho(\Gamma)$

$$u(\mathbf{x}, \boldsymbol{y}) = \sum_{\boldsymbol{\nu} \in \mathbb{N}_0^N} \widehat{u}_{\boldsymbol{\nu}}(\mathbf{x}) \boldsymbol{\Psi}_{\boldsymbol{\nu}}(\boldsymbol{y}) \quad \text{with} \quad \widehat{u}_{\boldsymbol{\nu}}(\mathbf{x}) = \langle u(\mathbf{x}, \cdot\,), \boldsymbol{\Psi}_{\boldsymbol{\nu}} \rangle$$

  – Parseval's equality $\Longrightarrow \|u\|^2_{\mathcal{V} \otimes L^2_\varrho(\Gamma)} = \sum_{\boldsymbol{\nu} \in \mathbb{N}_0^N} \|\widehat{u}\|^2_{\mathcal{V}}$

  – the best approximation of u out of $\mathcal{V} \otimes \mathcal{P}_\Lambda(\Gamma)$ is

$$u_\Lambda(\mathbf{x}, \boldsymbol{y}) := \sum_{\boldsymbol{\nu} \in \Lambda} \widehat{u}_{\boldsymbol{\nu}}(\mathbf{x}) \boldsymbol{\Psi}_{\boldsymbol{\nu}}(\boldsymbol{y})$$

  for which we have

$$\big\|u - u_\Lambda\big\|_{\mathcal{V} \otimes L^2_\varrho(\Gamma)} = \min_{\widetilde{u} \in \mathcal{V} \otimes \mathcal{P}_\Lambda(\Gamma)} \big\|u - \widetilde{u}\big\|_{\mathcal{V} \otimes L^2_\varrho(\Gamma)}$$

- What is a good polynomial approximation subspace?

- Natural choice in one dimension

$$\Lambda = \{1, 2, 3, \ldots, p\} \quad \text{and} \quad u_\Lambda(\mathbf{x}, \boldsymbol{y}) := \sum_{j=1}^{p} \widehat{u}_j(\mathbf{x}) \boldsymbol{\Psi}_j(\boldsymbol{y})$$

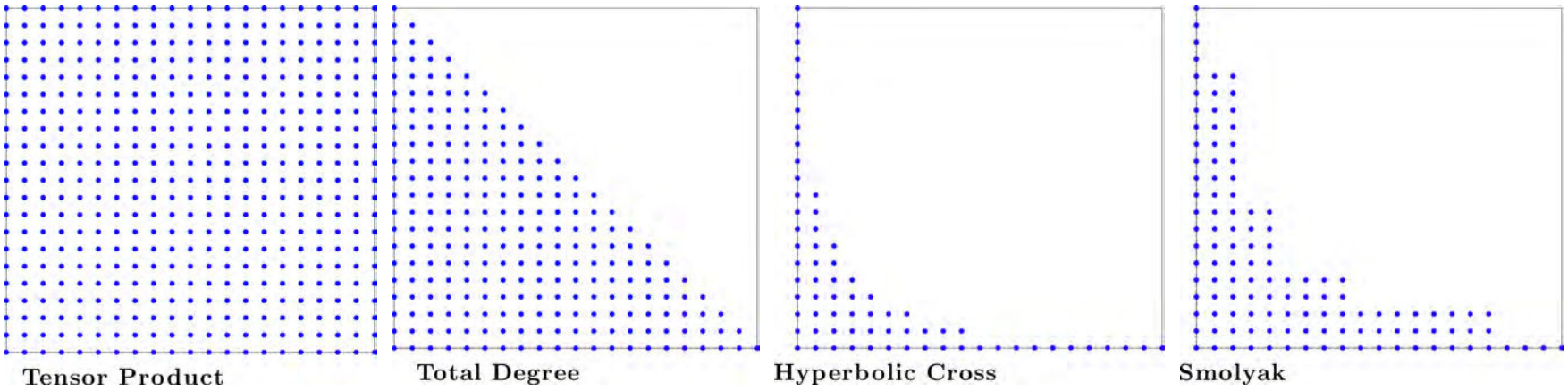
- Several choices for the multi-index $\boldsymbol{\nu} \in \Lambda(p)$

tensor product (TP) $\quad \max_{1 \le n \le N} \nu_n \le p$

total degree (TD) $\quad \sum_{n=1}^{N} \nu_n \le p$

hyperbolic cross (HC) $\quad \prod_{n=1}^{N} (\nu_n + 1) \le p + 1$

Smolyak (SM) $\quad \sum_{n=1}^{N} f(\nu_n) \le f(p)$ with $f(\nu) = \lceil \log_2(\nu) \rceil$, $\nu \ge 2$



- TD, HC and SM all partially alleviate the curse of dimensionality with respect to TP methods

$$\#\Lambda^{TP}(p) = (p+1)^N$$

$$\#\Lambda^{TD}(p) = \frac{(N+p)!}{N!p!}$$

$$\#\Lambda^{HC}(p) \lesssim \min\{p^3 4^N, p^{2+\log(N)}\}$$

$$\#\Lambda^{SM}(p) = O\big(p(\ln p)^{N-1}\big)$$

- But beating tensor products is not a big thing
  - we want to do better than that
  - ideally, we would like to find, given an error tolerance, a polynomial approximation that has the fewest possible terms that meets that tolerance

- Before looking into this, we mention an improvement on straightforward sparse grid approximation

ANISOTROPIC SPARSE GRIDS

- Anisotropic representation of sparse grids
  - introduce weight vector $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_N) \in \mathbb{R}_+^N$ with $\alpha_{\min} = 1$

    Tensor product (TP): $\max_{1 \le i \le d} \alpha_n \nu_n \le p$

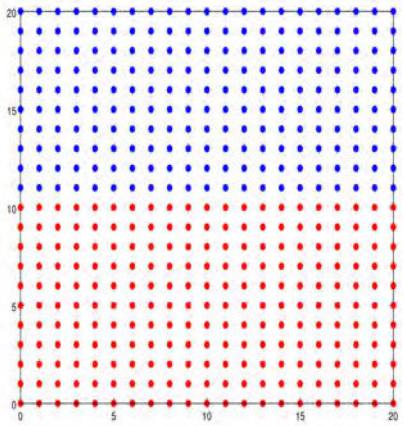
    Total degree (TD): $\sum_{i=1}^N \alpha_n \nu_n \le p$

    Hyperbolic cross (HC): $\prod_{n=1}^N (\nu_n + 1)^{\alpha_n} \le p + 1$

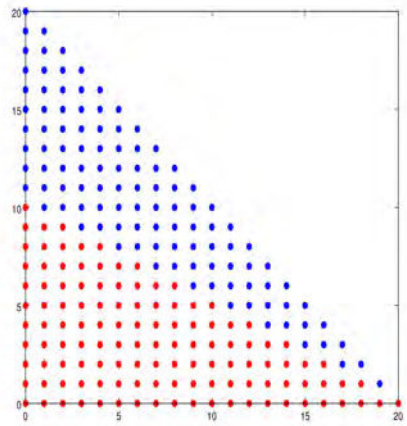
    Smolyak (SM): $\sum_{n=1}^N \alpha_n f(\nu_n) \le f(p)\}$
    with $f(\nu) = \lceil \log_2(\nu) \rceil$, $\nu \ge 2$

- In some cases the weights can be determined
  - e.g., in KL expansions, the coefficients are proportional to the decaying eigenvalues of the correlation matrix

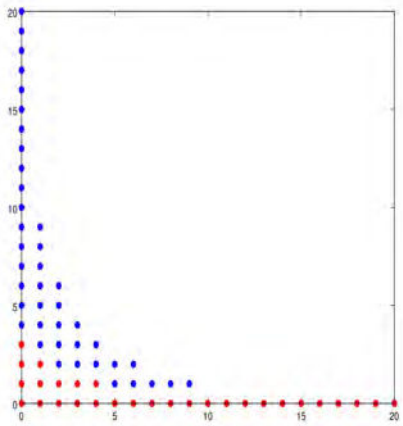
- In some cases, the weights can be determined adaptively



TP



TD



HC

Blue + red: standard grids

Red: anisotropic grids with $\alpha_1 = 1$ and $\alpha_2 = 1/2$

BEST S-TERM APPROXIMATIONS

Defining best s-term approximations

- One easily obtains the estimates
  - for Taylor expansions

$$\|u - u_\Lambda\|_{L^\infty(\Gamma,\mathcal{V})} = \sup_{\boldsymbol{y}\in\Gamma} \Big\| \sum_{\boldsymbol{\nu}\notin\Lambda} t_{\boldsymbol{\nu}} \boldsymbol{y}^{\boldsymbol{\nu}} \Big\|_{\mathcal{V}} \le \sup_{\boldsymbol{y}\in\Gamma} \sum_{\boldsymbol{\nu}\notin\Lambda} \|t_{\boldsymbol{\nu}} \boldsymbol{y}^{\boldsymbol{\nu}}\|_{\mathcal{V}} = \sum_{\boldsymbol{\nu}\notin\Lambda} \|t_{\boldsymbol{\nu}}\|_{\mathcal{V}}$$

  - for Legendre expansions

$$\big\|u - u_\Lambda\big\|^2_{\mathcal{V}\otimes L^2_\varrho(\Gamma)} = \sum_{\boldsymbol{\nu}\notin\Lambda} \|\widehat{u}_{\boldsymbol{\nu}}\|^2_{\mathcal{V}}$$

- Best S-term approximations
  - choose $\Lambda = \Lambda_S^{\rm opt} \Longrightarrow$ the set of s largest coefficients $\|t_{\boldsymbol{\nu}}\|_{\mathcal{V}}$ or $\|\widehat{u}_{\boldsymbol{\nu}}\|_{\mathcal{V}}$
  - $\Lambda_S^{\rm opt}$ provides a theoretical approximation
    - generally not realizable in practice
    - you have to solve for the complete expression to ascertain which are the largest coefficients!

- Analyticity of the solution revisited
  - polydisc: $\mathcal{O}_{\boldsymbol{\gamma}} = \bigotimes_{n=1}^{N} \{z_n \in \mathbb{C},\ |z_n| \leq \gamma_n\}$

    polyellipse: $\mathcal{E}_{\boldsymbol{\gamma}} = \bigotimes_{n=1}^{N} \left\{ z_n \in \mathbb{C},\ \frac{z_n + z_n^{-1}}{2} \ :\ |z_n| = \gamma_n \right\}$

  - assume $a(\mathbf{x}, \boldsymbol{y})$ is affine in $\boldsymbol{y}$ and satisfies CC

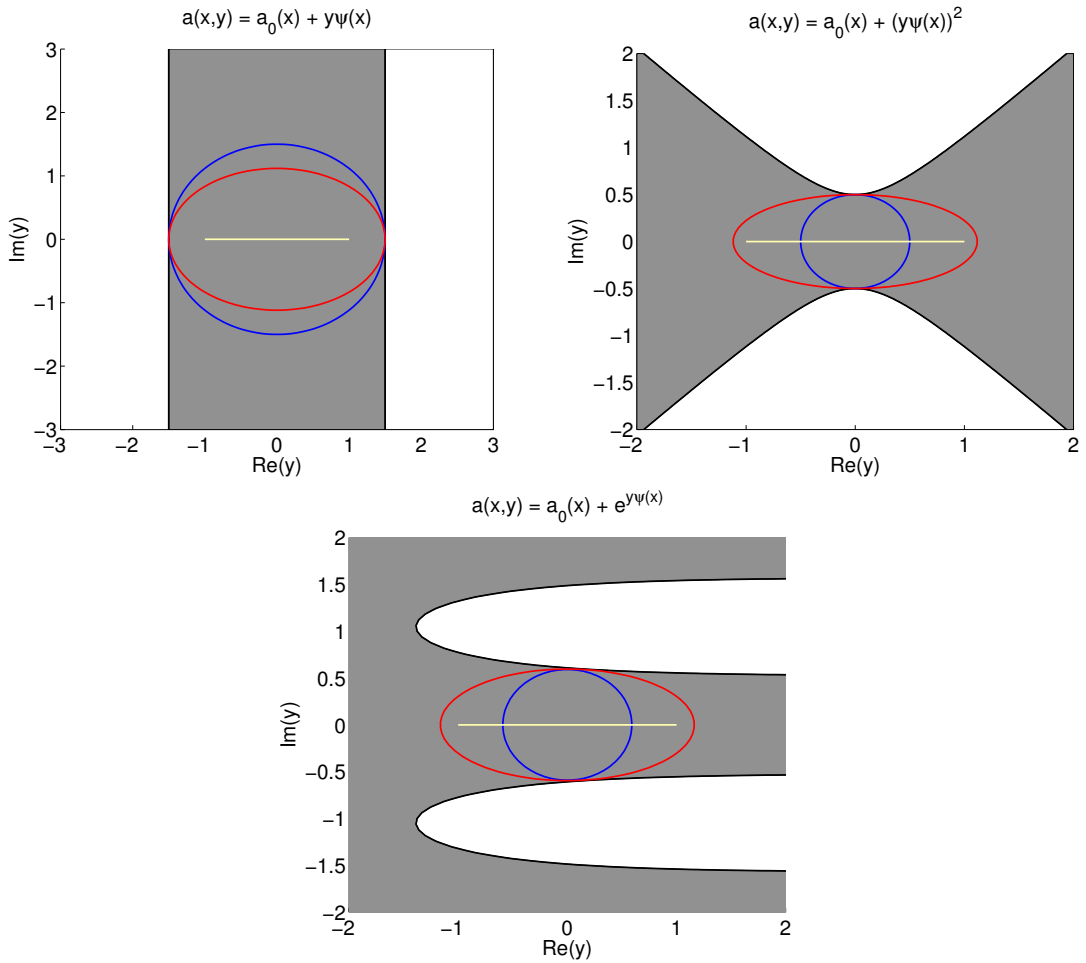
    $\Longrightarrow$ $\widehat{\Gamma}$ contains some polydisc $\mathcal{O}_{\boldsymbol{\gamma}}$ with $\boldsymbol{\gamma} = (\gamma_n)_{1 \leq n \leq N}$, $\gamma_n > 1\ \ \forall n$

    $\Longrightarrow$ $\{\|t_{\boldsymbol{\nu}}\|_{\mathcal{V}}\}_{\boldsymbol{\nu}}$ is ℓ^q-summable for all $q < 1$

  - assume $a(\mathbf{x}, \boldsymbol{y})$ satisfies CC and AN

    $\Longrightarrow$ the domain of uniform ellipticity $\widehat{\Gamma}$ contains some polyellipse $\mathcal{E}_{\boldsymbol{\gamma}}$ with $\boldsymbol{\gamma} = (\gamma_n)_{1 \leq n \leq N}$, $\gamma_n > 1\ \ \forall n$

    $\Longrightarrow$ $\{\|\widehat{u}_{\boldsymbol{\nu}}\|_{\mathcal{V}}\}_{\boldsymbol{\nu}}$ is ℓ^q-summable for all $q < 1$



Domains of complex uniform ellipticity for some random fields

- Although the best S-term approximation is not realizable, it's good to know how good it is so that we judge the quality of realizable methods

Coefficient bounds

- Taylor coefficient estimate
  - assume $\boldsymbol{z} \mapsto u(\boldsymbol{z})$ is analytic in an open neighborhood of $\mathcal{O}_{\boldsymbol{\gamma}}$ with $\gamma_n > 1 \ \forall n$
  - assume $\Re(a(\mathbf{x}, \boldsymbol{z})) \geq \delta > 0 \ \forall \mathbf{x} \in \overline{D}, \ \boldsymbol{z} \in \mathcal{O}_{\boldsymbol{\gamma}}$

  then
  - $\sum_{\boldsymbol{\nu} \in \mathbb{N}_0^N} t_{\boldsymbol{\nu}} \boldsymbol{y}^{\boldsymbol{\nu}}$ converges uniformly towards $u(\boldsymbol{y})$ in Γ
  - decay of Taylor coefficients $\quad \|t_{\boldsymbol{\nu}}\|_{\mathcal{V}} \leq C_\delta \boldsymbol{\gamma}^{-\boldsymbol{\nu}}$

    where $C_\delta = \frac{\|f\|_{\mathcal{V}'}}{\delta}$ and $\boldsymbol{\gamma}^{-\boldsymbol{\nu}} = \gamma_1^{-\nu_1} \cdots \gamma_n^{-\nu_n}$

- Legendre coefficient estimate
  - assume $\boldsymbol{z} \mapsto u(\boldsymbol{z})$ is analytic in an open neighborhood of $\mathcal{E}_{\boldsymbol{\gamma}}$ with $\gamma_n > 1 \ \forall n$
  - assume $\Re(a(\mathbf{x}, \boldsymbol{z})) \geq \delta > 0 \ \forall \mathbf{x} \in \overline{D}, \ \boldsymbol{z} \in \mathcal{E}_{\boldsymbol{\gamma}}$

  then
  - $\sum_{\boldsymbol{\nu} \in \mathbb{N}_0^N} \widehat{u}_{\boldsymbol{\nu}} L_{\boldsymbol{\nu}}(\boldsymbol{y})$ converges towards $u(\boldsymbol{y})$ in $\mathcal{V} \otimes L^2(\Gamma)$ and
  - decay of Legendre coefficients $\|\widehat{u}_{\boldsymbol{\nu}}\|_{\mathcal{V}} \leq C_{\boldsymbol{\gamma},\delta} \boldsymbol{\gamma}^{-\boldsymbol{\nu}} \prod_{n=1}^N \sqrt{2\nu_n + 1}$

    where $C_{\boldsymbol{\gamma},\delta} = \frac{\|f\|_{\mathcal{V}'}}{\delta} \prod_{n=1}^N \frac{\ell(\mathcal{E}_{\gamma_n})}{4(\gamma_n - 1)}$
    $\ell(\mathcal{E}_{\gamma_n})$ denoting the perimeter of $\mathcal{E}_{\gamma_n}$

- Optimal Taylor coefficient upper bounds
  - let $\mathcal{A}$ denote the set of all $(\boldsymbol{\gamma}, \delta)$, $\gamma_n > 1$, $\delta > 0$, such that the polydisc $\mathcal{O}_\gamma$ is contained in $\widehat{\Gamma}$
  - assume $\Re(a(\mathbf{x}, \boldsymbol{z})) \geq \delta > 0$, $\forall \mathbf{x} \in \overline{D}$, $\boldsymbol{z} \in \mathcal{O}_\gamma$

then

$$\|t_{\boldsymbol{\nu}}\|_{\mathcal{V}} \leq C_\delta \boldsymbol{\gamma}^{-\boldsymbol{\nu}} \qquad \forall (\boldsymbol{\gamma}, \delta) \in \mathcal{A}$$

  - optimal Taylor coefficient bounds

$$\inf_{(\boldsymbol{\gamma}, \delta) \in \mathcal{A}} C_\delta \boldsymbol{\gamma}^{-\boldsymbol{\nu}}$$

- Optimal Legendre coefficient upper bounds
  - let $\mathcal{A}$ denote the set of all $(\boldsymbol{\gamma}, \delta)$, $\gamma_n > 1$, $\delta > 0$, such that the polyellipse $\mathcal{E}_\gamma$ is contained in $\widehat{\Gamma}$
  - assume $\Re(a(\mathbf{x}, \boldsymbol{z})) \geq \delta > 0$, $\forall \mathbf{x} \in \overline{D}$, $\boldsymbol{z} \in \mathcal{E}_{\boldsymbol{\gamma}}$)

then

$$\|\widehat{u}_{\boldsymbol{\nu}}\|_{\mathcal{V}} \leq C_{\boldsymbol{\gamma},\delta} \boldsymbol{\gamma}^{-\boldsymbol{\nu}} \prod_{n=1}^{N} \sqrt{2\nu_i + 1} \qquad \forall (\boldsymbol{\gamma}, \delta) \in \mathcal{A}$$

  - optimal Legendre coefficient bounds

$$\inf_{(\boldsymbol{\gamma},\delta)\in\mathcal{A}} C_{\boldsymbol{\gamma},\delta} \boldsymbol{\gamma}^{-\boldsymbol{\nu}} \prod_{n=1}^{N} \sqrt{2\nu_i + 1}.$$

- Pessimistic convergence estimate for best S-term approximations
  - let

$$C(q) = \|(\|t_{\boldsymbol{\nu}}\|_{\mathcal{V}})_{\boldsymbol{\nu}}\|_{\ell^q} \quad \text{for Taylor}$$

$$C(q) = \|(\|\widehat{u}_{\boldsymbol{\nu}}\|_{\mathcal{V}})_{\boldsymbol{\nu}}\|_{\ell^q} \quad \text{for Legendre}$$

  - then we have the estimates

$$\text{Taylor:} \quad \|u - u_{\Lambda_S^{\mathrm{opt}}}\|_{L^\infty(\Gamma,\mathcal{V})} \leq C(q) S^{1-\frac{1}{q}} \quad \text{for all } 0 < q < 1$$

$$\text{Legendre:} \quad \left\|u - u_{\Lambda_S^{\mathrm{opt}}}\right\|_{\mathcal{V}\otimes L^2_\varrho(\Gamma)} \leq C(q) S^{\frac{1}{2}-\frac{1}{q}} \quad \text{for all } 0 < q < 1$$

  - Legendre approximation features faster convergence rate than that of Taylor expansion
  - smaller q $\quad\Longrightarrow\quad$ stronger convergence rate but bigger $C(q)$.
  - $C(q)$ is implicit

Refined error estimate for best S-term approximations

- Let $B(\boldsymbol{\nu})$ denote the upper bound for $\|t_{\boldsymbol{\nu}}\|_{\mathcal{V}}$ or $\|\widehat{u}_{\boldsymbol{\nu}}\|_{\mathcal{V}}$
  - define $b(\boldsymbol{\nu}) = -\ln B(\boldsymbol{\nu})$ so that $B(\boldsymbol{\nu}) = e^{-b(\boldsymbol{\nu})}$

- Let $\Lambda_S^{\mathrm{Qopt}}$ denote the quasi-optimal index set corresponding to s largest coefficient bounds (not the coefficients themselves)

- We have $\|c_{\boldsymbol{\nu}}\|_{\mathcal{V}} \leq e^{-b(\boldsymbol{\nu})}$
  - then

$$\sup_{\boldsymbol{y}\in\Gamma} \left\| u - \sum_{\boldsymbol{\nu}\in\Lambda_S^{\mathrm{opt}}} c_{\boldsymbol{\nu}} \boldsymbol{\Psi}_{\boldsymbol{\nu}} \right\|_{\mathcal{V}} \leq \sum_{\boldsymbol{\nu}\notin\Lambda_S^{\mathrm{opt}}} \|c_{\boldsymbol{\nu}}\|_{\mathcal{V}} \leq \sum_{\boldsymbol{\nu}\notin\Lambda_S^{\mathrm{Qopt}}} e^{-b(\boldsymbol{\nu})}$$

  where $c_{\boldsymbol{\nu}}$ stands for either $t_{\boldsymbol{\nu}}$ or $\widehat{u}_{\boldsymbol{\nu}}$

- Assumption on coefficient bounds (AoB)

  The map $b : [0, \infty)^N \to \mathbb{R}$ satisfies

  - $b(\mathbf{0}) = 0$
  - b is continuous in $[0, \infty)^N$
  - the mapping $\tau \mapsto \frac{1}{\tau} b(\tau \boldsymbol{\nu})$ is either

    increasing in $(0, \infty)$ for all $\boldsymbol{\nu} \in [0, \infty)^N$

    or

    decreasing in $(0, \infty)$ for all $\boldsymbol{\nu} \in [0, \infty)^N$
  - there exists $0 < c < C$ such that $c|\boldsymbol{\nu}| < b(\boldsymbol{\nu}) < C|\boldsymbol{\nu}|$ as $\boldsymbol{\nu} \to \infty$

- Given $\boldsymbol{\gamma} = (\gamma_n)_{1 \le i \le N}$ with $\gamma_n > 1 \ \forall n$
  - let $\boldsymbol{\lambda} = (\lambda_n)_{1 \le n \le N}$ such that $\lambda_n = \log \gamma_n \ \forall n$

- AoB is satisfied by

$$B(\boldsymbol{\nu}) = \boldsymbol{\gamma}^{-\boldsymbol{\nu}} \qquad b(\boldsymbol{\nu}) = \sum_{n=1}^{N} \lambda_n \nu_n$$

$$B(\boldsymbol{\nu}) = \boldsymbol{\gamma}^{-\boldsymbol{\nu}} \prod_{n=1}^{N} \sqrt{2\nu_n + 1} \qquad b(\boldsymbol{\nu}) = \sum_{n=1}^{N} \left(\lambda_n \nu_n - \tfrac{1}{2} \log(2\nu_n + 1)\right)$$

$$B(\boldsymbol{\nu}) = \inf_{\boldsymbol{\gamma},\delta} \; C_\delta \boldsymbol{\gamma}^{-\boldsymbol{\nu}} \qquad b(\boldsymbol{\nu}) = \sup_{\boldsymbol{\gamma},\delta} \left(\sum_{n=1}^{N} \lambda_n \nu_n - \log C_\delta \right)$$

$$B(\boldsymbol{\nu}) = \inf_{\boldsymbol{\gamma},\delta} \; C_{\gamma,\delta} \boldsymbol{\gamma}^{-\boldsymbol{\nu}} \prod_{n=1}^{N} \sqrt{2\nu_n + 1} \qquad b(\boldsymbol{\nu}) = \sup_{\boldsymbol{\gamma},\delta} \left(\sum_{n=1}^{N} \left(\lambda_n \nu_n - \tfrac{1}{2} \log(2\nu_n + 1)\right) - \log C_{\gamma,\delta} \right)$$

$$B(\boldsymbol{\nu}) = \boldsymbol{\gamma}^{-\boldsymbol{\nu}} \frac{|\boldsymbol{\nu}|!}{\boldsymbol{\nu}!} \qquad b(\boldsymbol{\nu}) = \sum_{n=1}^{N} \lambda_n \nu_n - \log \frac{\Gamma(|\boldsymbol{\nu}|+1)}{\prod_{n=1}^{N} \Gamma(\nu_n+1)}$$

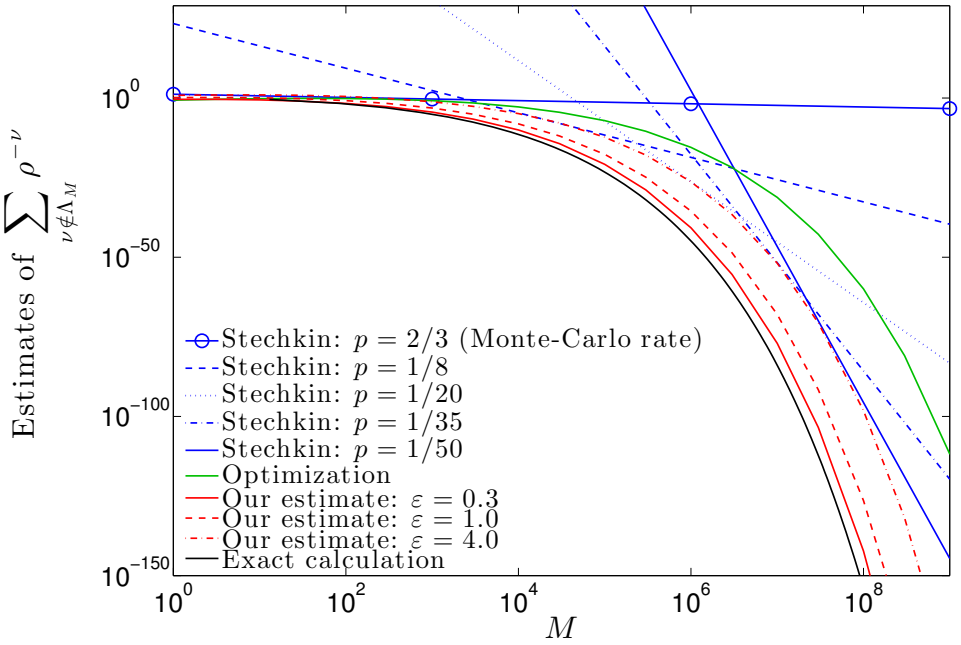
  - until recently, analysis has been limited to the first case

- Consider the Taylor series $\sum_{\boldsymbol{\nu}\in\mathbb{N}_0^N} t_{\boldsymbol{\nu}}\boldsymbol{y}^{\boldsymbol{\nu}}$ of u
  - recall that
$$\|t_{\boldsymbol{\nu}}\|_{\mathcal{V}} \leq C\boldsymbol{\gamma}^{-\boldsymbol{\nu}} \quad \forall \boldsymbol{\nu} \in \mathbb{N}_0^N$$
  - denote by Λ_S^{opt} the set of indices corresponding to s largest coefficients
- Then, assuming AoB, we have that
  - for any $\varepsilon > 0$, there exists $S_\varepsilon > 0$ depending on ε such that
$$\sup_{\boldsymbol{y}\in\Gamma} \left\| u(\boldsymbol{y}) - \sum_{\boldsymbol{\nu}\in\Lambda_S^{\text{opt}}} t_{\boldsymbol{\nu}}\boldsymbol{y}^{\boldsymbol{\nu}} \right\|_{\mathcal{V}} \leq C_u(\varepsilon) S \exp\left(-\left(\frac{SN! \prod_{n=1}^N \lambda_n}{(1+\varepsilon)} \right)^{1/N} \right)$$
    for all $S > S_\varepsilon$
    $\Longrightarrow$ we achieve sub-exponential convergence rates of the form $S\exp(-(\kappa S)^{1/N})$ with optimized κ
  - this is the best estimate obtained so far, improving on results obtain based on "Stechkin estimation"

- Example 1. Estimate the truncation error of $\sum_{\boldsymbol{\nu}\in\mathbb{N}_0^4} \boldsymbol{\gamma}^{-\boldsymbol{\nu}}$, where

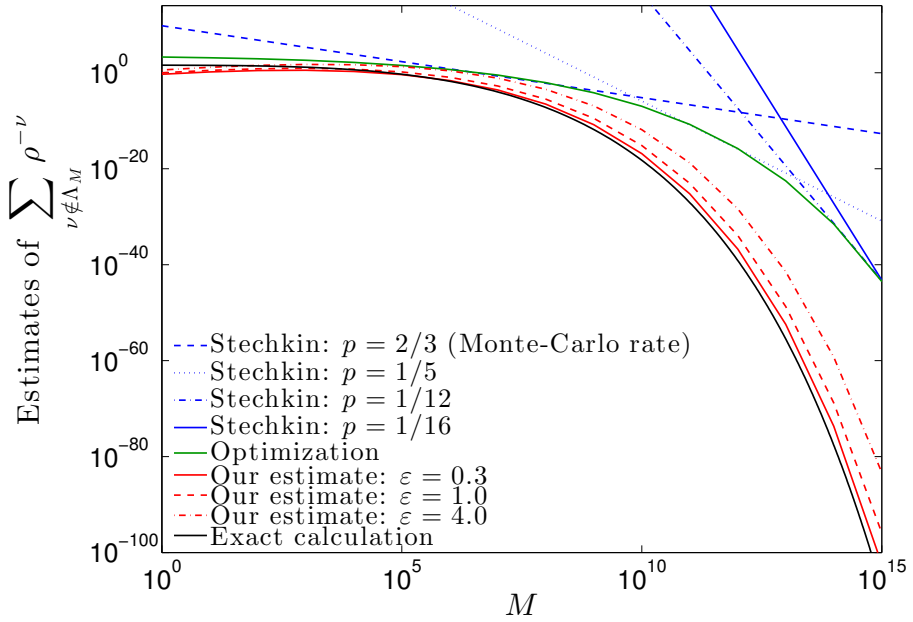
$$\gamma_1 = \gamma_2 = 2, \quad \gamma_3 = 4, \quad \gamma_4 = 16$$

  - we have a 4-dimensional parameter space
  - this problem arises in error analysis of best S-term Taylor approximations for parameterized elliptic PDEs with non-overlapping basis functions
  - the best theoretical coefficient bound has the form: $\|t_{\boldsymbol{\nu}}\|_{\mathcal{V}} \leq \boldsymbol{\gamma}^{-\boldsymbol{\nu}}$



- Example 2. Estimate the error of $\sum_{\boldsymbol{\nu}\in\mathbb{N}_0^8} \boldsymbol{\gamma}^{-\boldsymbol{\nu}}$, where

$$\gamma_n = 2, \qquad n = 1, \ldots, 8$$



- A lower bound for the error in best S-term approximation can also be obtained
  - AoB is not enough
  - Taylor polynmial approximations satisfy the additional assumptions
  - the lower bound looks like

$$\text{error} \geq C_{lower} S^{1-\frac{1}{N}} \exp(-(\kappa_{lower} s)^{1/N})$$

    as opposed to the upper bound

$$\text{error} \leq C_{upper} S \exp(-(\kappa_{upper} s)^{1/N})$$

QUASI-OPTIMAL BEST S-TERM POLYNOMIAL APPROXIMATIONS

- The optimal best S-term approximation is in general not computable
- Instead, what can be done is to determine a quasi-optimal approximation
  - first
    - derive estimates for the sizes of the coefficients
  - then
    - throw out the coefficients for which the estimated values are small
  - then
    - construct an approximate solution involving only the remaining coefficients
- This strategy will be effective only if one can obtain sharp estimates on the sizes of the coefficients
  - ideally, one would want sharp upper and lower bounds on the coefficients

- Still not useful estimates
  - determining the radii γ is possible only in a few cases
  - in case basis functions $\{\psi_i\}$ have non-overlapping supports (as is the case for "inclusion problems") $\boldsymbol{\gamma}$ is independent of $\boldsymbol{\nu}$ and can be found easily
  - advantages
    - practical surrogate for the best S-term approximation
    - index sets can be determined near optimally and adaptively
  - challenge
    - sharp bounds on the coefficients are still very difficult and are problem dependent

REFERENCES

- J. Beck, F. Nobile, L. Tamellini, R. Tempone, *Convergence of quasi-optimal stochastic Galerkin methods for a class of PDEs with random coefficients*, Computers and Mathematics with Applications, 67, 2014, 732-751

- A. Chkifa, A. Cohen, R. DeVore, C. Schwab, *Sparse adaptive Taylor approximation algorithms for parametric and stochastic elliptic PDEs*, Model. Math. Anal. Numer. 47, 2013, 253-280

- A. Cohen, R. DeVore, C. Schwab, *Analytic regularity and polynomial approximation of parametric and stochastic elliptic PDEs*, Analysis and Applications, 9, 2011, 11-47.

- M. Gunzburger, C. Webster, G. Zhang, *Stochastic finite element methods for PDEs with random input data*, Acta Numerica, 23, 2014, 521-650.

- H. Tran, C. Webster, G. Zhang, *Analysis of quasi-optimal polynomial approximations for parameterized PDEs with deterministic and stochastic coefficients*, Numerische Mathematik, 37, 2017, 451-493.

DISCRETE LEAST-SQUARES APPROXIMATIONS

Max Gunzburger - Florida State University

Prepared with
Clayton Webster, Hoang Tran, and Guannan Zhang - Oak Ridge National Laboratory

- Objective: Determine a global polynomial approximation of $h \in L^2_\varrho(\Gamma)$

$$h(\boldsymbol{y}) = \sum_{\boldsymbol{\nu} \in \mathbb{N}_0^N} \widehat{h}_{\boldsymbol{\nu}} \boldsymbol{\Psi}_{\boldsymbol{\nu}}(\boldsymbol{y}) \quad \text{with } \widehat{h}_{\boldsymbol{\nu}} = \langle h, \boldsymbol{\Psi}_{\boldsymbol{\nu}} \rangle$$

- Given a multiindex set Λ, let

$$\mathcal{P}_\Lambda(\Gamma) = \operatorname{span} \left\{ \prod_{n=1}^N y_n^{\nu_n} \quad \text{with} \quad \boldsymbol{\nu} \in \Lambda \right\} \subset L^2_\varrho(\Gamma)$$

- We denote the best approximation of h out of $\mathcal{P}_\Lambda(\Gamma)$ by

$$h_\Lambda(\boldsymbol{y}) = \sum_{\boldsymbol{\nu} \in \Lambda} \widehat{h}_{\boldsymbol{\nu}} \boldsymbol{\Psi}_{\boldsymbol{\nu}}(\boldsymbol{y})$$

  for which we have

$$\|h - h_\Lambda\|_\varrho = \min_{q \in \mathcal{P}_\Lambda(\Gamma)} \|h - q\|_\varrho$$

- In general, we only know h at a set of points $\{\boldsymbol{y}_m\}_{m=1}^M$ in Γ

Discrete least square (DLS) approximations

- The discrete least square (DLS) approximation is given by

$$h^{LS} = \arg\min_{q \in \mathcal{P}_\Lambda(\Gamma)} \sum_{m=1}^{M} |h(\boldsymbol{y}_m) - q(\boldsymbol{y}_m)|^2$$

- Let
  - Λ denote a multi-index set and let $S = \#\Lambda = \dim[\mathcal{P}_\Lambda(\Gamma)]$
  - $\{\boldsymbol{\Psi}_{\boldsymbol{\nu}}\}_{\boldsymbol{\nu} \in \Lambda}$ be a set of orthogonal polynomials

- Denote the discrete least-square approximation by

$$h^{LS} = \sum_{\boldsymbol{\nu} \in \Lambda} c_{\boldsymbol{\nu}} \boldsymbol{\Psi}_{\boldsymbol{\nu}}(\boldsymbol{y})$$

- The vector of coefficients $\boldsymbol{c}_\Lambda = (c_{\boldsymbol{\nu}})_{\boldsymbol{\nu}\in\Lambda} \in \mathbb{R}^S$ is the solution of the minimization problem

$$\boldsymbol{c}_\Lambda = \underset{\boldsymbol{z}_\Lambda=(z_{\boldsymbol{\nu}})_{\boldsymbol{\nu}\in\Lambda}\in\mathbb{R}^S}{\arg\min} \sum_{m=1}^{M} \left| h(\boldsymbol{y}_m) - \sum_{\boldsymbol{\nu}\in\Lambda} z_{\boldsymbol{\nu}} \boldsymbol{\Psi}_{\boldsymbol{\nu}}(\boldsymbol{y}_m) \right|^2$$

- Of course, $\boldsymbol{c}_\Lambda$ is the solution of the matrix problem

$$\boldsymbol{G}\boldsymbol{c}_\Lambda = \boldsymbol{h}$$

where $\boldsymbol{G}$ is the $S \times S$ matrix

$$\boldsymbol{G}_{\boldsymbol{\nu},\boldsymbol{\nu}'} = \frac{1}{M} \sum_{m=1}^{M} \boldsymbol{\Psi}_{\boldsymbol{\nu}}(\boldsymbol{y}_m) \boldsymbol{\Psi}_{\boldsymbol{\nu}'}(\boldsymbol{y}_m)$$

and $\boldsymbol{h}$ is the $S \times 1$ vector

$$\boldsymbol{h}_{\boldsymbol{\nu}} = \frac{1}{M} \sum_{m=1}^{M} h(\boldsymbol{y}_m) \boldsymbol{\Psi}_{\boldsymbol{\nu}}(\boldsymbol{y}_m)$$

  - for stability, $\boldsymbol{G}$ needs to be well conditioned

- Observation
  - assume $\boldsymbol{y}_m$ is randomly sampled according to the measure ϱ
  - then

$$\text{for } M \to \infty \qquad \Longrightarrow \qquad \boldsymbol{G}_{\boldsymbol{\nu},\boldsymbol{\nu}'} \to \int_{\Gamma} \boldsymbol{\Psi}_{\boldsymbol{\nu}}(\boldsymbol{y}) \boldsymbol{\Psi}_{\boldsymbol{\nu}'}(\boldsymbol{y}) \varrho(\boldsymbol{y}) d\boldsymbol{y} = \delta_{\boldsymbol{\nu},\boldsymbol{\nu}'}$$

$$\Longrightarrow \qquad \mathbb{E}(\boldsymbol{G}) = \boldsymbol{I}$$

- How does one quantify the proximity of the matrices $\boldsymbol{G}$ and $\boldsymbol{I}$?
- How large does the number of samples M have to be to insure stability?

- Let

$$K(\Lambda) = \sup_{\boldsymbol{y}\in\Gamma} \sum_{\boldsymbol{\nu}\in\Lambda} |\boldsymbol{\Psi}_{\boldsymbol{\nu}}(\boldsymbol{y})|^2 \qquad \Longleftarrow \qquad \text{property of the basis}$$

  and recall the spectral norm

$$|||\boldsymbol{G}||| = \max_{\boldsymbol{z}\neq 0} \frac{|\langle \boldsymbol{G}\boldsymbol{z}, \boldsymbol{z}\rangle|}{\|\boldsymbol{z}\|^2}$$

- Then, for $0 < \delta < 1$

$$\mathbb{P}(|||\boldsymbol{G} - \boldsymbol{I}||| \leq \delta) > 1 - 2S \exp\left(-\frac{c_\delta M}{K(\Lambda)}\right)$$

  where $c_\delta = \delta + (1-\delta)\log(1-\delta) > 0$

- Note that

$$|||\boldsymbol{G}-\boldsymbol{I}||| \leq \delta \Longleftrightarrow \max_{\boldsymbol{z}\neq 0} \frac{|\langle \boldsymbol{G}\,\boldsymbol{z},\boldsymbol{z}\rangle - \|\boldsymbol{z}\|^2|}{\|\boldsymbol{z}\|^2} \leq \delta$$

$$\Longleftrightarrow (1-\delta)\|\boldsymbol{z}\|^2 \leq \langle \boldsymbol{G}\,\boldsymbol{z},\boldsymbol{z}\rangle \leq (1+\delta)\|\boldsymbol{z}\|^2 \quad \forall \boldsymbol{z}\in\mathbb{R}^S$$

$$\Longleftrightarrow (1-\delta)\|\boldsymbol{z}\|^2 \leq \|\boldsymbol{A}\boldsymbol{z}\|^2 \leq (1+\delta)\|\boldsymbol{z}\|^2 \quad \forall \boldsymbol{z}\in\mathbb{R}^S$$

where $\boldsymbol{A}$ is an $M\times S$ sampling matrix with

$$\boldsymbol{A}_{m,\boldsymbol{\nu}} = \frac{1}{\sqrt{M}}\Psi_{\boldsymbol{\nu}}(\boldsymbol{y}_m)$$

- Then we have the isometry property:
  - for $0<\delta<1, \qquad c_\delta = \delta + (1-\delta)\log(1-\delta) > 0$
  - and with probability exceeding $1-2S\exp\left(-\frac{c_\delta M}{K(\Lambda)}\right)$

  we have that

$$(1-\delta)\|\boldsymbol{z}\|^2 \leq \|\boldsymbol{A}\boldsymbol{z}\|^2 \leq (1+\delta)\|\boldsymbol{z}\|^2 \quad \forall \boldsymbol{z}\in\mathbb{R}^S$$

- $\boldsymbol{A}$ satisfying the isometry property (IP)
  - set $\delta = \frac{1}{2}$ and M such that $\frac{M}{\log M} \geq \frac{K(\Lambda)(1+r)}{c_{1/2}}$

    $\Longrightarrow$ IP holds with probability $\geq 1 - 2M^{-r}$
  - set $M \geq \frac{K(\Lambda)}{c_\delta}\left(\log(2S) + \log(\frac{1}{\gamma})\right)$

    $\Longrightarrow$ holds with probability $\geq 1 - \gamma$

- Error estimate
  - assume $|h(\boldsymbol{y})| \leq L$ for all $\boldsymbol{y} \in \Gamma$
  - if, for any $r > 0$, M satisfies $\quad \frac{M}{\log M} \geq \frac{K(\Lambda)(1+r)}{c_{1/2}}$

  then

$$\mathbb{E}(\|h - h^{LS}\|^2) \lesssim \underbrace{\|h - h_\Lambda\|^2}_{\text{best approximation error on } \Lambda} + L^2 M^{-r}$$

- Estimating $K(\Lambda)$ in the 1D setting $\Lambda = \{0, 1, \ldots, S-1\}$
  - recall that $K(\Lambda) = \sup_{\boldsymbol{y}\in\Gamma} \sum_{\boldsymbol{\nu}\in\Lambda} |\boldsymbol{\Psi}_{\boldsymbol{\nu}}(\boldsymbol{y})|^2 = \sum_{\boldsymbol{\nu}\in\Lambda} \|\boldsymbol{\Psi}_{\boldsymbol{\nu}}\|_{L^\infty}^2$

- trigonometric polynomials $\Psi_j(y)$

$$\|\Psi_j\|_{L^\infty} = 1 \quad \Rightarrow \quad K(\Lambda) = S$$

stability condition: $\dfrac{M}{\log(M)} \gtrsim S$

- Legendre polynomials $L_j(y)$

$$\|L_j\|_{L^\infty} = \sqrt{2j+1} \quad \Rightarrow \quad K(\Lambda) = \sum_{j=0}^{S-1} (2j+1) = S^2$$

stability condition: $\dfrac{M}{\log(M)} \gtrsim S^2$

- Chebyshev polynomials $T_j(y)$

$$T_0 = 1,\ \|T_j\|_{L^\infty} = \sqrt{2} \ \text{ if } \ j \geq 1 \quad \Rightarrow \quad K(\Lambda) = 2S - 1$$

stability condition: $\dfrac{M}{\log(M)} \gtrsim S$

- Estimating $K(\Lambda)$ in the multivariate setting
  - an index set $\Lambda \subset \mathbb{N}_0^N$ is called a lower set or a downward closed set if

$$(\boldsymbol{\nu} \in \Lambda \text{ and } \boldsymbol{\nu}' \leq \boldsymbol{\nu}) \implies \boldsymbol{\nu}' \in \Lambda,$$

    where $\boldsymbol{\nu}' \leq \boldsymbol{\nu}$ means that $\nu_n' \leq \nu_n$ for all $1 \leq n \leq N$

    - a generalization of the set $\Lambda = \{0, \ldots, S-1\}$ in 1D

    - for smooth functions, good index sets are often lower sets

  - assume Λ is a lower set $\implies$

    - Legendre: $K_L(\Lambda) = \sum_{\boldsymbol{\nu} \in \Lambda} \prod_{n=1}^{N} (2\nu_n + 1) \leq (\#\Lambda)^2$

    - Chebyshev: $K_T(\Lambda) = \sum_{\boldsymbol{\nu} \in \Lambda} 2^{\#(\mathrm{supp}(\boldsymbol{\nu}))} \leq (\#\Lambda)^{\frac{\log 3}{\log 2}}$

  - stability conditions

$$\frac{M}{\log(M)} \gtrsim S^2 \qquad \text{for Legendre}$$

$$\frac{M}{\log(M)} \gtrsim S^{\frac{\log 3}{\log 2}} \qquad \text{for Chebyshev}$$

Application to parameterized PDEs

1. Generate M random samples $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_M$ according to the measure ϱ

2. For each $\boldsymbol{y}_m$, solve the spatially discretized parameterized PDEs $\mathcal{L}(u, \boldsymbol{y}_m) = f$ to obtain the solution $u^h(\boldsymbol{y}_m) \in \mathcal{V}^h$

3. $u^{LS} = \sum_{\boldsymbol{\nu} \in \Lambda} c_{\boldsymbol{\nu}}(x) \boldsymbol{\Psi}_{\boldsymbol{\nu}}(\boldsymbol{y})$, where

$$(c_{\boldsymbol{\nu}})_{\boldsymbol{\nu} \in \Lambda} = \underset{\boldsymbol{z} = (z_{\boldsymbol{\nu}}) \in (\mathcal{V}^h)^S}{\arg\min} \sum_{m=1}^{M} \left\| u(\boldsymbol{y}_m) - \sum_{\boldsymbol{\nu} \in \Lambda} z_{\boldsymbol{\nu}} \boldsymbol{\Psi}_{\boldsymbol{\nu}}(\boldsymbol{y}_m) \right\|_{\mathcal{V}}^2$$

4. $\boldsymbol{c} = (c_{\boldsymbol{\nu}})_{\boldsymbol{\nu} \in \Lambda}$ is the solution of

$$\boldsymbol{G}\boldsymbol{c} = \boldsymbol{u}$$

where $\boldsymbol{G}$ is an $S \times S$ matrix and $\boldsymbol{u}$ is an $S \times 1$ vector in $(\mathcal{V}^h)^S$ given by

$$\boldsymbol{G}_{\boldsymbol{\nu}, \boldsymbol{\nu}'} = \frac{1}{m} \sum_{m=1}^{M} \boldsymbol{\Psi}_{\boldsymbol{\nu}}(\boldsymbol{y}_m) \boldsymbol{\Psi}_{\boldsymbol{\nu}'}(\boldsymbol{y}_m) \qquad \boldsymbol{u}_{\boldsymbol{\nu}} = \frac{1}{m} \sum_{m=1}^{M} u(\boldsymbol{y}_m) \boldsymbol{\Psi}_{\boldsymbol{\nu}}(\boldsymbol{y}_m)$$

- Pros
  - non-intrusive method
  - mitigates Runge phenomena

- Cons
  - number of samples is larger than the degree of freedom
  - accuracy is sensitive to the choice of polynomial space

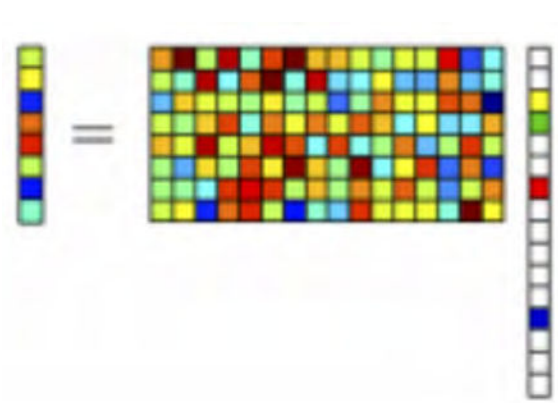
COMPRESSED SENSING-BASED APPROXIMATIONS

Max Gunzburger - Florida State University

Prepared with
Clayton Webster, Hoang Tran, and Guannan Zhang - Oak Ridge National Laboratory

INTRODUCTION

- Recover signals/functions from an underdetermined system
  - want to find a solution of the matrix problem



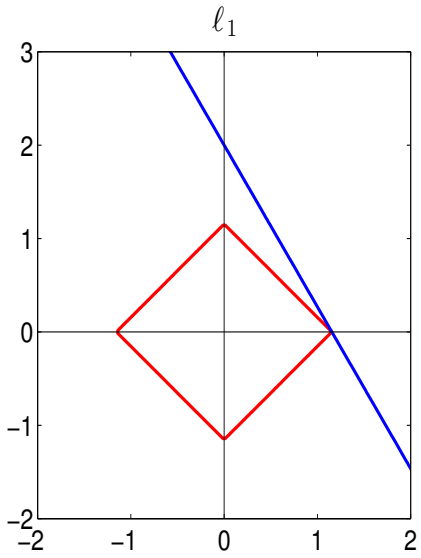
linear algebra class $\Longrightarrow$ find minimum norm solution

compressed sensing $\Longrightarrow$ find solution vector for which many of the components are zero

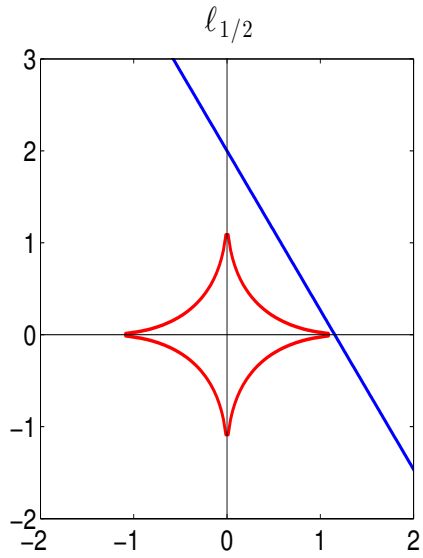
  - initially developed for signal recovery

- Sparse signals are recovered via sparsity-inducing norms

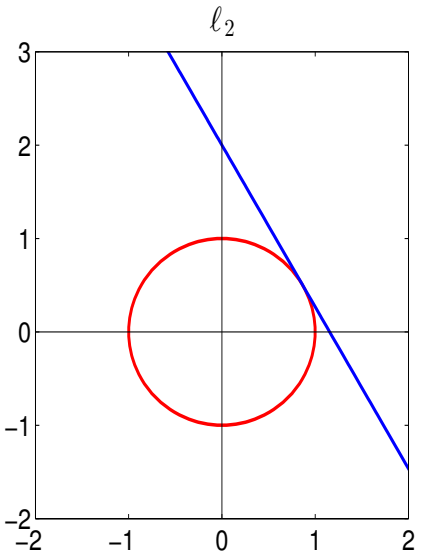
$$\boldsymbol{c} = \arg\min \|\boldsymbol{z}\|_q \quad \text{subject to} \quad \boldsymbol{u} = \mathcal{A}\boldsymbol{z} \qquad \text{with } 0 < q \leq 1$$



ℓ_1



$\ell_{1/2}$



ℓ_2

APPLICATION TO PARAMETERIZED PDES

- Approximate $u(x, \boldsymbol{y})$ by

$$u(\mathbf{x}, \boldsymbol{y}) \simeq u^{\#}(\mathbf{x}, \boldsymbol{y}) = \sum_{\boldsymbol{\nu} \in \Lambda} c_{\boldsymbol{\nu}}(\mathbf{x}) \boldsymbol{\Psi}_{\boldsymbol{\nu}}(\boldsymbol{y})$$

- Non-intrusive approach: compute $u(\,\cdot\,, \boldsymbol{y}_m)$ for a set of samples $\{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_M\}$ in Γ

- $\boldsymbol{c} = (c_{\boldsymbol{\nu}})_{\boldsymbol{\nu} \in \Lambda}$ satisfies

$$u(\mathbf{x}, \boldsymbol{y}_m) = \sum_{\boldsymbol{\nu} \in \Lambda} c_{\boldsymbol{\nu}}(\mathbf{x}) \boldsymbol{\Psi}_{\boldsymbol{\nu}}(\boldsymbol{y}_m) \quad \forall m = 1, \ldots, M$$

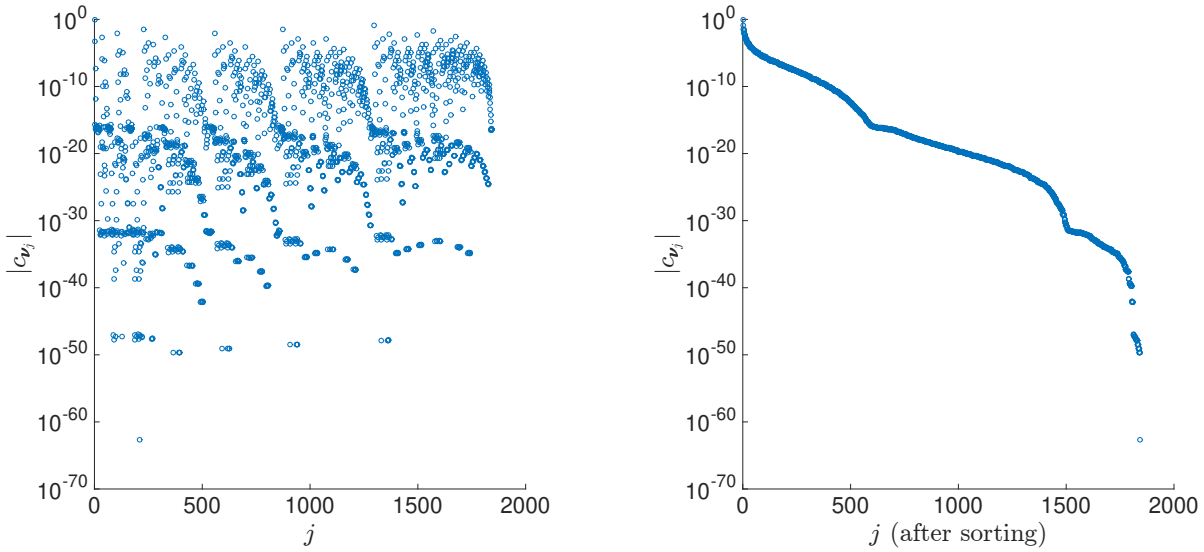
$$\Longleftrightarrow \qquad \boldsymbol{u} = \boldsymbol{A}\boldsymbol{c}$$

where

$$\boldsymbol{A} = (\boldsymbol{\Psi}_{m,\boldsymbol{\nu}}) = (\boldsymbol{A}_{\boldsymbol{\nu}}(\boldsymbol{y}_m))_{\substack{m \in [M] \\ \boldsymbol{\nu} \in \Lambda}} \qquad \boldsymbol{u} = (u(\,\cdot\,, \boldsymbol{y}_m))_{m \in [M]} \in \mathcal{V}^M$$

- Solve $\boldsymbol{u} = \boldsymbol{A}\boldsymbol{c}$

- Observations
  - each measurement $u(\boldsymbol{y}_m)$ $\quad\Longleftrightarrow\quad$ a PDE solve
  - $\boldsymbol{c} = (c_{\boldsymbol{\nu}})_{\boldsymbol{\nu}\in\Lambda}$ decays fast ("approximately sparse")



  - important coefficients often have low indices
  - however, we don't know the shape of the correct index set

- Main idea:
  - approximate $\boldsymbol{u}$ on a big polynomial subspace $\mathcal{P}_{\Lambda_0}(\Gamma)$ with Λ_0 possibly far from optimal

    - let $S_0 = \#(\Lambda_0)$

  - undersampling

    generate $M \ll S_0$ samples $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_M$ and solve for $u(\boldsymbol{y}_1), \ldots, u(\boldsymbol{y}_M)$

  - reconstruction

    determine $\boldsymbol{c} = (c_{\boldsymbol{\nu}})_{\boldsymbol{\nu}\in\Lambda_0}$ from the underdetermined system $\boldsymbol{u} = \boldsymbol{A}\boldsymbol{c}$ using a compressed sensing algorithm

    $$\boldsymbol{c} = \underset{\boldsymbol{z}\in\mathcal{V}^N}{\arg\min} \|\boldsymbol{z}\|_1 \quad \text{such that} \quad \boldsymbol{u} = \boldsymbol{A}\boldsymbol{z}$$

    where $\|\boldsymbol{z}\|_1 = \sum_{\boldsymbol{\nu}\in\Lambda_0} \|z_{\boldsymbol{\nu}}\|_{\mathcal{V}}$

- Let

$$\Theta = \sup_{\boldsymbol{\nu} \in \Lambda_0} \|\boldsymbol{\Psi}_{\boldsymbol{\nu}}\|_\infty$$

denote the uniform bound for the orthonormal system of basis functions and let $\boldsymbol{A} \in \mathbb{R}^{M \times S_0}$ be the random sampling matrix $\boldsymbol{A} = (\boldsymbol{\Psi}_{m,\boldsymbol{\nu}})$

  - then, if

$$M \geq C\Theta^2 S \log^2(S) \log(S_0)$$

the best S-term approximation is recovered

  - note the seemingly very mild dependence on S_0

- We have $\Theta = 1$ for Fourier, Hadamard, circulant, etc. matrices
  - to reconstruct best S-term approximation, need $\simeq S$ samples
- For polynomial approximations, Θ can be prohibitively high
  - Chebyshev basis: $\Theta = 2^{S_0/2}$
  - Legendre basis: $\Theta \gtrsim S_0$
  - preconditioned Legendre basis: $\Theta = 2^{S_0/2}$
- If we constrain the compressed sensing minimization problem so that only lower sets Λ are considered

  i.e., we have $$u_{lower} = \underset{\substack{\#(\Lambda)=S \\ \Lambda \text{ lower}}}{\arg\min} \|u - u_\Lambda\|_1$$

  we obtain an approximation for which
  - the sample complexity is reduced
  - the best lower S-term is as good as the best S-term approximation
  - the effect of Runge's phenomenon is reduced

- We now have

$$M \geq CK(S)\log^2(K(S))\log(S_0)$$

  - with

    - for Legendre: $K(S) \leq S^2$

    - for Chebyshev: $K(S) \leq S^{\frac{\log 3}{\log 2}}$

- We can choose the multi-index set Λ_0 as a hyperbolic cross $\mathcal{H}_s$ which is the union of all lower sets of cardinality S, i.e.,

$$\mathcal{H}_s = \left\{ \boldsymbol{\nu} = (\nu_1, \ldots, \nu_N) \in \mathbb{N}_0^N \ : \ \prod_{n=1}^{N} (\nu_n + 1) \leq S \right\}$$

  - note that $S_0 = \#(\mathcal{H}_s) \leq 2S^3 4^N$

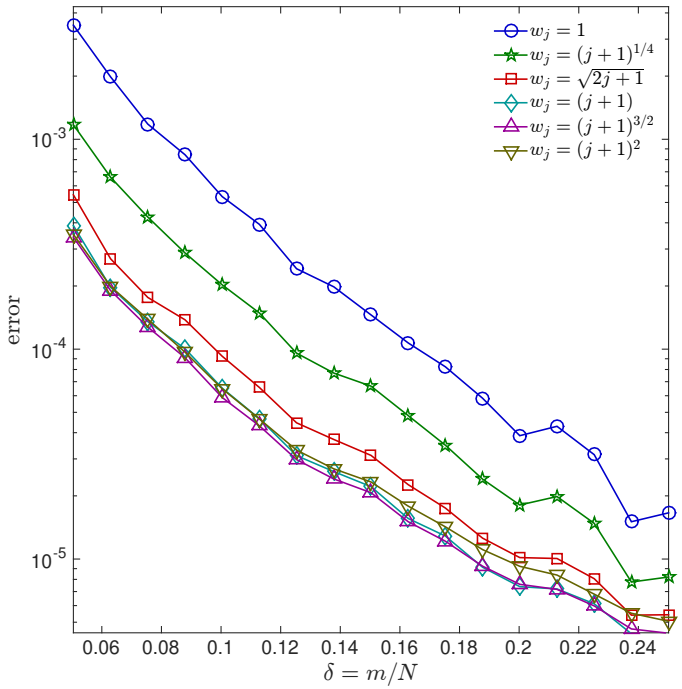
- We can do even better if we use weighted ℓ_1 minimization
  - choose the specific weights $\omega_\nu = \|\Psi_\nu\|_\infty$
  - then solve the problem

    find $u^{CS}(\boldsymbol{y}) = \sum_{\boldsymbol{\nu}\in\Lambda_0} c_{\boldsymbol{\nu}}\Psi_{\boldsymbol{\nu}}(\boldsymbol{y})$, where $\boldsymbol{c} = (c_{\boldsymbol{\nu}})_{\boldsymbol{\nu}\in\Lambda_0}$ is the solution of

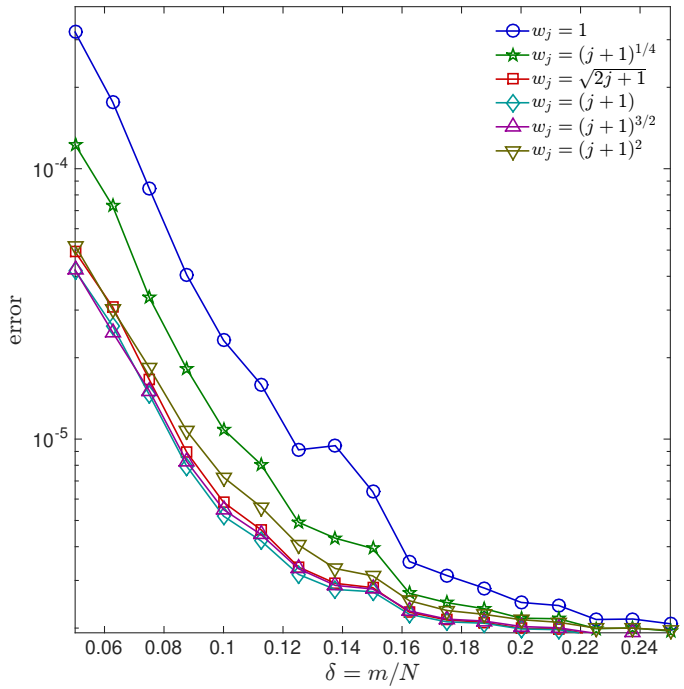
    $$\min \sum_{\boldsymbol{\nu}\in\Lambda_0} \omega_{\boldsymbol{\nu}} \|z_{\boldsymbol{\nu}}\|_{\mathcal{V}} \quad \text{subject to } \boldsymbol{u} = \boldsymbol{A}\boldsymbol{z}$$

  - the weights favor low indices and penalize high indices

- Example 1: $u(\boldsymbol{y}) = \dfrac{\prod_{n=1}^{N/2} \cos(8y_n/2^i)}{\prod_{n=N/2+1}^{d}(1-y_n/4^n)}$

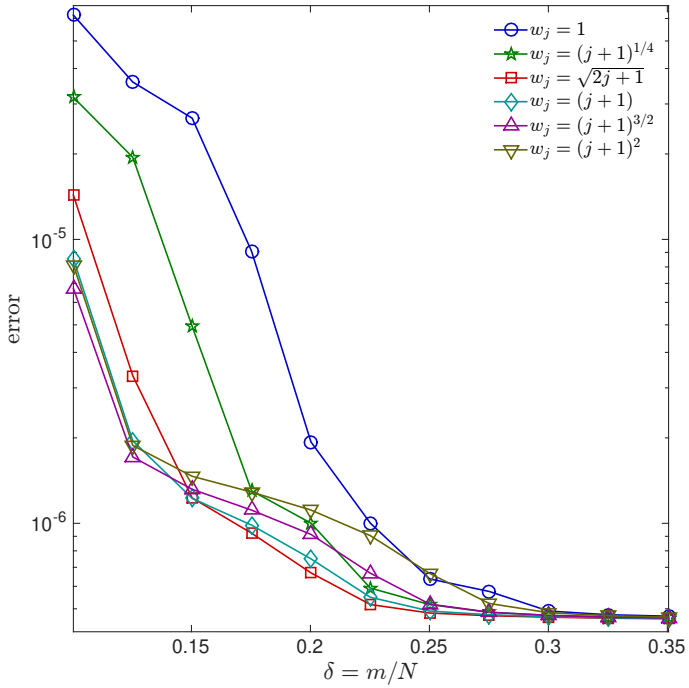


$N = 8,\ S_0 = 1843$

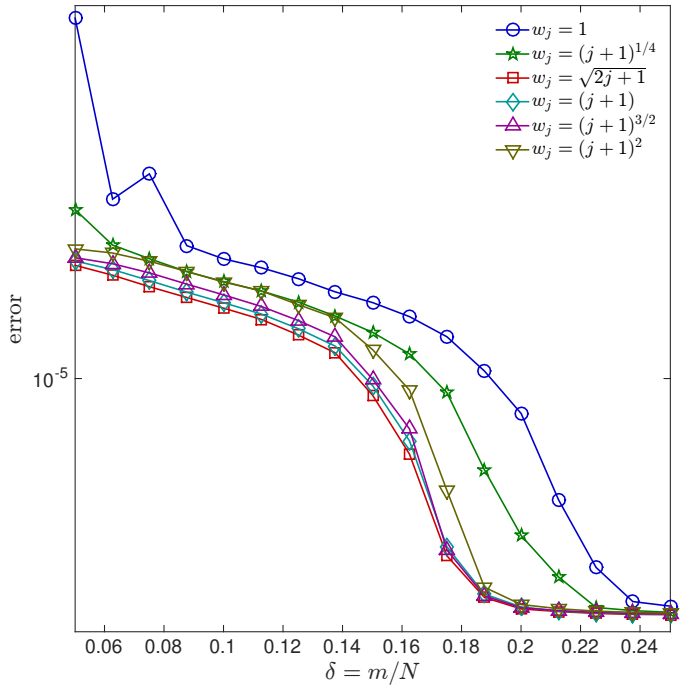


$N = 16,\ S_0 = 4129$

- Example 2: $u(\boldsymbol{y}) = \exp\left(-\frac{\sum_{n=1}^{N}\cos(y_n)}{8N}\right)$

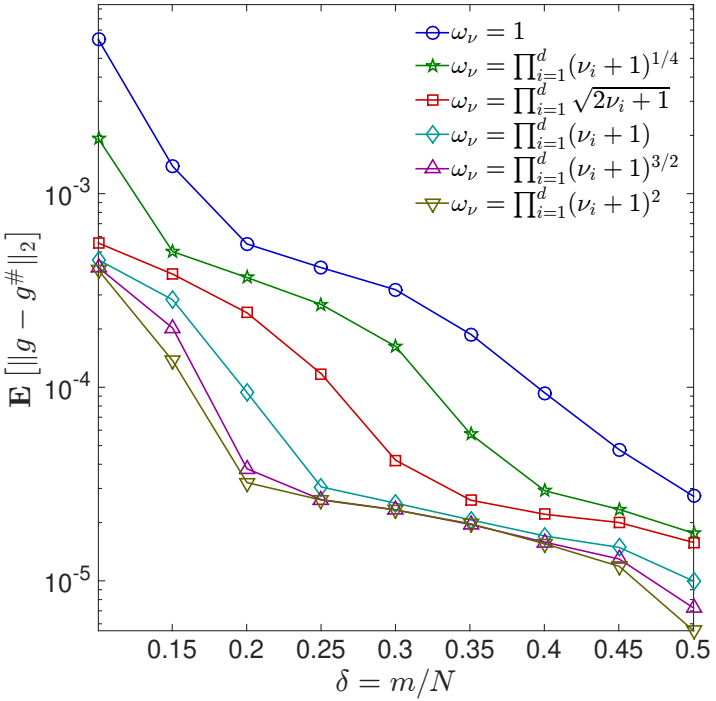


$N = 8,\ S_0 = 1843$

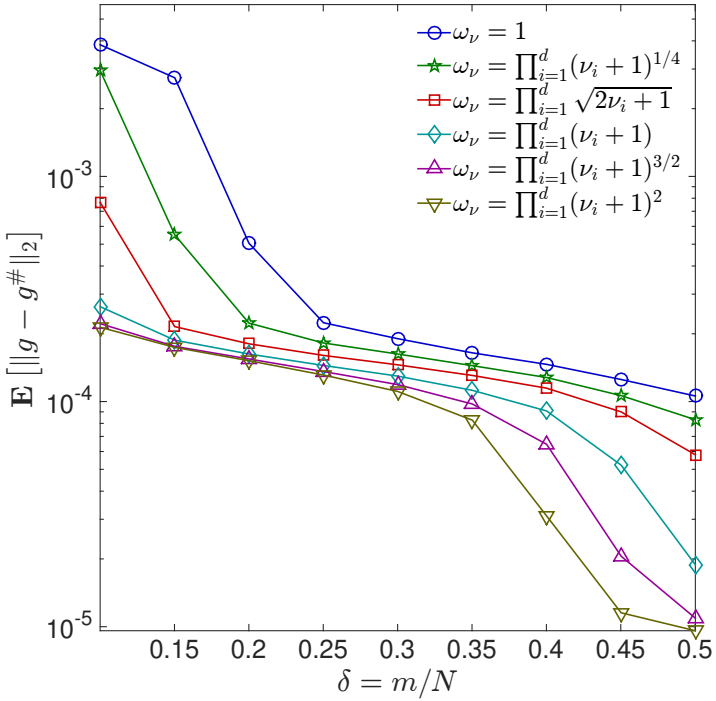


$N = 16,\ S_0 = 4129.$

- Example 3: $u(\boldsymbol{y}) = \exp\left(-\frac{\sum_{n=1}^{N} y_n}{2N}\right)$



$N = 8,\ S_0 = 1843$



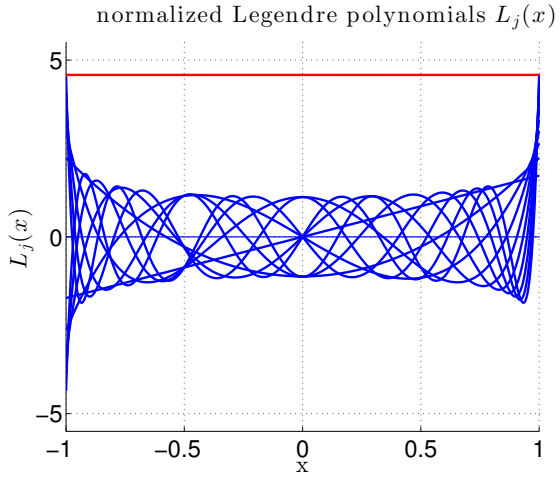
$N = 16,\ S_0 = 4129$

- But even the improved bound is pessimistic
- Consider 1D Legendre expansion on $\Lambda_0 = \{0, 1, \ldots, S_0 - 1\}$
  - current theories give that the number of samples needed is $M \geq \Theta^2 S \times$ log factor where $\Theta = \sqrt{2S_0 - 1}$
  - numerical experiments show that there is some successful recovery with underdetermined Legendre systems
  - the number of measurements for guaranteed recovery guarantee should not depend on S_0 or the maximum polynomial degree
  - recent result in 1D says that it is enough to have

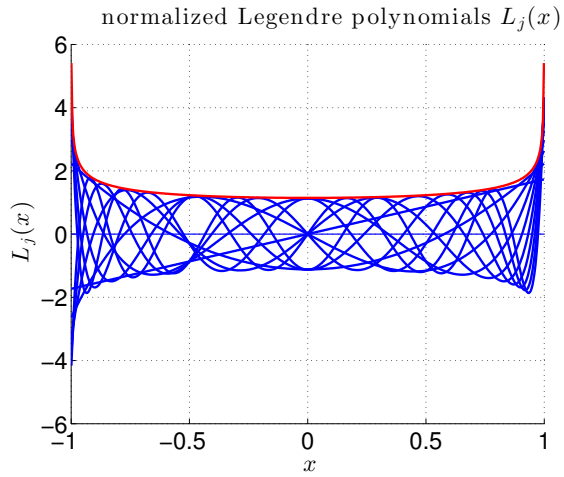
    $$M \geq S^2 \times \text{log factor}$$

    independent of the polynomial degree and of S_0

– the key idea used in for deriving the improved estimate on the number of samples needed is to use a better bound for the Legendre poynomials



previous theories use uniform bound



latest result uses envelope bound

REFERENCES

- J. Bourgain, *An improved estimate in the restricted isometry problem*, in Geometric Aspects of Functional Analysis, Lecture Notes in Mathematics, 2014, 65-70.

- A. Chkifa, A. Cohen, G. Migliorati, F. Nobile, and R. Tempone, *Discrete least squares polynomial approximation with random evaluations - application to parametric and stochastic elliptic PDEs*, ESAIM: M2AN, 49, 2015, 815-837.

- A. Chkifa, N. Dexter, H. Tran, and C. Webster, *Polynomial approximation via compressed sensing of high-dimensional functions on lower sets*, Mathematics of Computation, to appear in 2017.

- A. Cohen, M. Davenport, and D. Leviatan, *On the Stability and Accuracy of Least Squares Approximations*, Found. Comput. Math., 13, 2013, 819-834.

- A. Chernov and D. Dũng, *New explicit-in-dimension estimates for the cardinality of high-dimensional hyperbolic crosses and approximation of functions having mixed smoothness*, Journal of Complexity, 32, 2016, 92-121.

- S. Foucart and H. Rauhut, *A Mathematical Introduction to Compressive Sensing*, Birkhäuser, 2013.

- H. Rauhut and C. Schwab, *Compressive sensing Petrov-Galerkin approximation of high dimensional parametric operator equations*, Math. Comp., 86, 2016, 661-700.

- H. Rauhut and R. Ward, *Interpolation via weighted l1-minimization*, Applied and Computational Harmonic Analysis, 40, 2016, 321-351.

- H. Tran and C. Webster, *A uniform bound of sample complexity for sparse Legendre approximation via ℓ_1 minimization in arbitrary polynomial subspace*, preprint 2016.

LOCALLY ADAPTIVE SPARSE GRID METHODS AND APPLICATIONS

Max Gunzburger - Florida State University

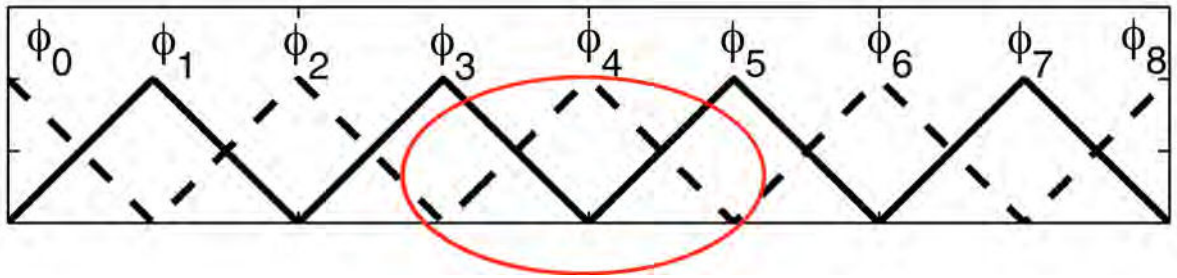
Prepared with
Clayton Webster, Hoang Tran, and Guannan Zhang - Oak Ridge National Laboratory

LOCAL SPARSE GRID INTERPOLATION

Hierarchical piecewise polynomial bases

- We begin by constructing hierarchical bases in one dimension
  - we use the hat functions to explain the construction
    - later we will consider other bases
  - the hat function having support $(y_{\ell,i} - h_\ell, y_{\ell,i} + h_\ell)$ is defined by

$$\psi_{\ell,i}(y) = \psi\left(\frac{y + 1 - ih_\ell}{h_\ell}\right)$$



- Delta property $\Longleftarrow$ each basis function is zero at other grid points

$$\Longrightarrow \qquad \psi_{\ell,i}(y_{\ell,i'}) = \delta_{ii'}$$

- A sequence of nodal bases can be generated by defining a sequence of mesh sizes $\{h_\ell\}_{\ell=0,1,\ldots}$
  - the most common choice is halve the mesh size

$$h_{\ell+1} = \frac{h_\ell}{2} \qquad \text{for } \ell = 0, 1, \ldots$$

  - the number of nodes (which equals the number of grid points) is then

$$R_0 = 2 \qquad \text{and} \qquad R_\ell = 2^\ell + 1 \quad \text{for } \ell = 1, 2, \ldots$$

- Let V_ℓ denote the space spanned by the nodal basis on level $\ell \implies$

$$V_0 = \text{span}\left\{\psi_{0,0}(y)\,, \psi_{0,1}(y)\right\}$$

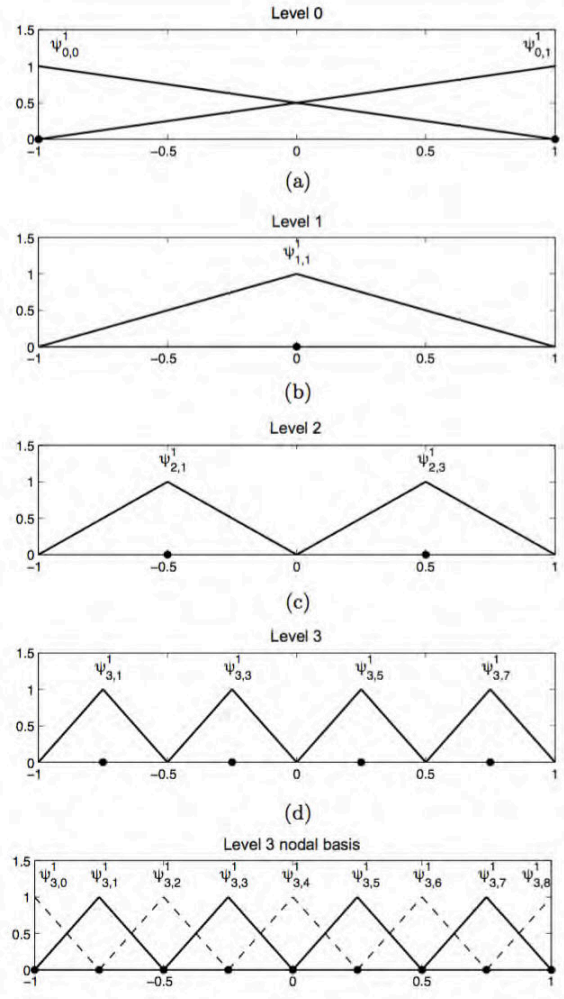
$$V_\ell = \text{span}\left\{\psi_{\ell,i}(y) \;:\; i = 0, \ldots, 2^\ell\right\} \quad \text{for } \ell = 1, 2, \ldots$$

  - due to the dyadic rule, the sequence $\{V_\ell\}$ is nested $\implies$

$$V_0 \subset V_1 \subset \cdots \subset V_\ell \subset V_{\ell+1} \subset \cdots \subset V$$

  - due to the nesting structure of $\{V_\ell\}$, we can define the incremental subspace

$$W_\ell = V_\ell \setminus V_{\ell-1} \quad \implies \quad V_\ell = V_{\ell-1} \oplus W_l$$



- We then have a hierarchical subspace splitting of V_ℓ given by

$$V_\ell = V_0 \oplus W_1 \oplus \cdots \oplus W_\ell \quad \text{for} \quad \ell = 1, 2, \ldots$$

  - each W_ℓ contains about half of the basis functions of the associated V_ℓ
  - non-overlapping property

    for $l \geq 1$, the supports of the basis functions in W_ℓ do not overlap

- The hierarchical and the nodal bases span the same subspace V_ℓ

- The hierarchical basis only possesses a partial delta property $\Longrightarrow$

  - the basis functions corresponding to a specific level possess the delta property with respect to its own level and coarser levels, but not with respect to finer levels

Hierarchical piecewise linear interpolation

- We build, in one dimension, the piecewise linear interpolant of a function with respect to the level L grid

- For each level $\ell = 1, \ldots, L$

  the interpolant of a function $g(y)$ with respect to the grid points $\{y_{\ell,i}\}_{i=0}^{2^\ell}$ and the corresponding nodal basis $\{\psi_{\ell,i}(y)\}_{i=0}^{2^\ell}$

  is given by

$$\mathcal{I}_\ell[g](y) = \sum_{i=0}^{2^\ell} g(y_{\ell,i})\psi_{\ell,i}(y) \in V_\ell$$

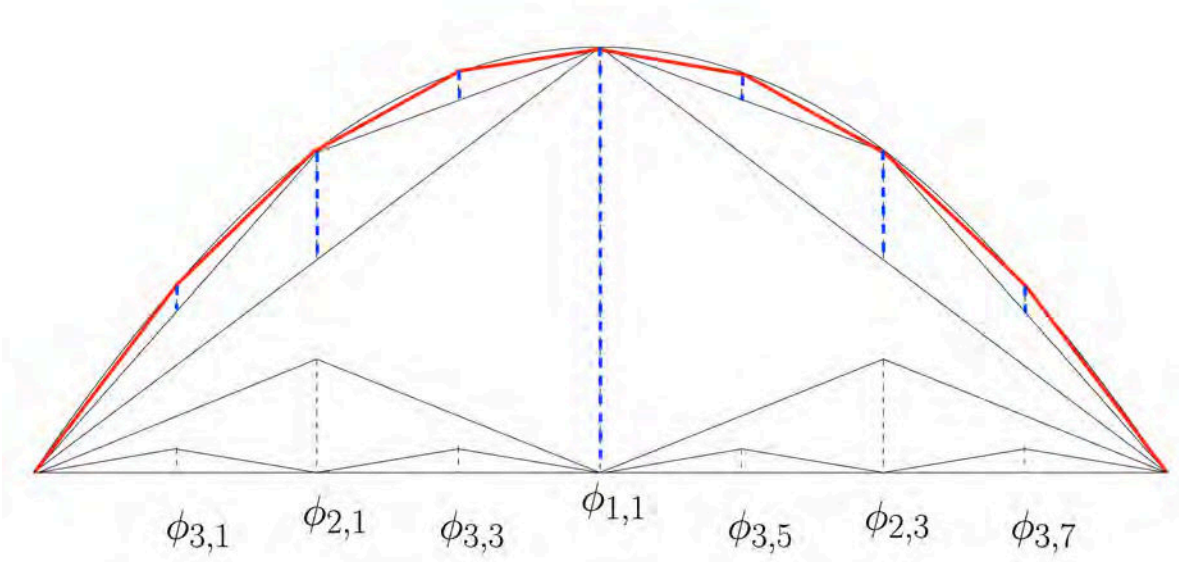
- Due to the nesting property $V_\ell = V_{\ell-1} \oplus W_\ell$, we have $\mathcal{I}_{\ell-1}[g] = \mathcal{I}_\ell\big[\mathcal{I}_{\ell-1}[g]\big]$, based on which we define the incremental interpolation operator

$$\Delta_l[g] = \mathcal{I}_\ell[g] - \mathcal{I}_{\ell-1}[g] = \mathcal{I}_\ell\big[g - \mathcal{I}_{\ell-1}[g]\big] = \sum_{i=0}^{2^\ell} c_{\ell,i}\psi_{\ell,i}(y)$$

  where $c_{\ell,i} = g(y_{\ell,i}) - \mathcal{I}_{\ell-1}\big[g(y_{\ell,i})\big]$

- We then have

$$\mathcal{I}_L[g] = \mathcal{I}_{L-1}[g] + \Delta_L[g] = \cdots = \mathcal{I}_0[g] + \sum_{\ell=1}^{L} \Delta_\ell[g]$$



For $L = 3$*:*

solid red line $\Longrightarrow$ *the piecewise linear interplolant*

dashed blue lines $\Longrightarrow$ *the absolute value of the coefficients* $c_{\ell,i}$*,* $i = 1, \ldots, 2^3$

- The one-dimensional hierarchical polynomial basis can be extended to the N-dimensional parameter domain Γ using tensorization
  - the N-variate basis function $\psi_{\boldsymbol{\ell},\mathbf{i}}(\boldsymbol{y})$ associated with $\boldsymbol{y}_{\boldsymbol{\ell},\mathbf{i}} = (y_{\ell_1,i_1}, \ldots, y_{\ell_n,i_n})$ is defined using tensor products $\Rightarrow$

$$\psi_{\boldsymbol{\ell},\mathbf{i}}(\boldsymbol{y}) = \prod_{n=1}^{N} \psi_{\ell_n,i_n}(y_n)$$

    where $\boldsymbol{\ell} = (\ell_1, \ldots, \ell_N)$ and $\mathbf{i} = (i_1, \ldots, i_N)$ are multi-indices indicating the resolution level and the grid point within the level

- The multi-dimensional space spanned by the nodal basis of level $\boldsymbol{\ell}$ is then

$$\mathcal{V}_{\boldsymbol{\ell}} = \text{span}\{\psi_{\boldsymbol{\ell},\mathbf{i}}(\boldsymbol{y}) \ : \ i_n = 0, \ldots, 2^{\ell_n}, \ n = 1, \ldots, N\}$$

  - this space could be anisotropic, i.e., $\ell_n \neq \ell_{n'}$ for some $n \neq n'$

- The N-dimensional hierarchical incremental subspace $W_{\boldsymbol{\ell}}$ can be defined by

$$\mathcal{W}_{\boldsymbol{\ell}} = \bigotimes_{n=1}^{N} W_{\ell_n} = \text{span}\left\{\psi_{\boldsymbol{\ell},\mathbf{i}}(\boldsymbol{y}) \mid \mathbf{i} \in B_{\boldsymbol{\ell}}\right\}$$

  where the multi-index set $B_{\boldsymbol{\ell}}$ is defined by

$$B_{\boldsymbol{\ell}} = \left\{ \mathbf{i} \in \mathbb{N}^N \;:\; \begin{array}{ll} i_n \in \{1,3,5,\ldots,2^{\ell_n}-1\} & \text{for } n = 1,\ldots,N \quad \text{if } \ell_n > 0 \\ i_n \in \{0,1\} & \text{for } n = 1,\ldots,N \quad \text{if } \ell_n = 0 \end{array} \right\}$$

- A subspace $\mathcal{V}_{\mathcal{J}}$ of the tensorial space $\mathcal{V}_{\boldsymbol{\ell}}$ can be defined by choosing a subset $\mathcal{J}$ of multi-indices and then taking the the direct sum of the corresponding sets of $\mathcal{W}_{\boldsymbol{\ell}} \implies$

$$\mathcal{V}_{\mathcal{J}} = \bigoplus_{\boldsymbol{\ell} \in \mathcal{J}} \mathcal{W}_{\boldsymbol{\ell}}$$

  - $\mathcal{W}_{\boldsymbol{\ell}}$ contains the building blocks used to construct a subspace of the tensorial space $\mathcal{V}_{\boldsymbol{\ell}}$
  - the multi-index set $\mathcal{J}$ is the blueprint that determines which building blocks are use to construct $\mathcal{V}_{\mathcal{J}}$

Sparse grids

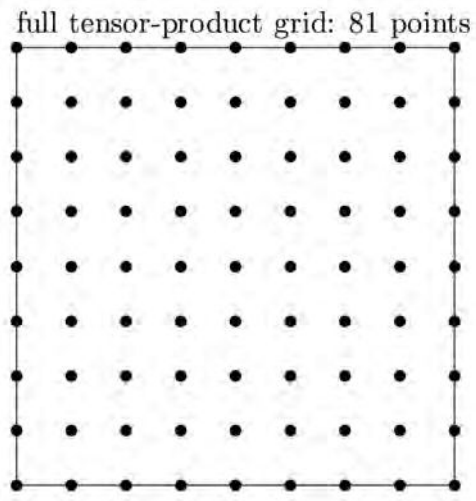
- In any subspace $\mathcal{V}_{\boldsymbol{\ell}}$ we can define a tensor-product interpolation operator

$$\mathcal{I}_{\boldsymbol{\ell}}[g] = \sum_{i_1=0}^{2^{\ell_1}} \cdots \sum_{i_N=0}^{2^{\ell_N}} g(y_{\ell_1,i_1}, \ldots, y_{\ell_N,i_N}) \left(\prod_{n=1}^{N} \psi_{\ell_n,i_n}(y_n) \right)$$

- In any subspace $\mathcal{W}_{\boldsymbol{\ell}}$, we can define a tensor-product incremental operator

$$\begin{aligned}
\Delta[g] &= \Delta_{\ell_1} \otimes \cdots \otimes \Delta_{\ell_N}[g] \\
&= \bigotimes_{n=1}^{N} \left(\mathcal{I}_{\ell_n} - \mathcal{I}_{\ell_n - 1} \right)[g] \\
&= \sum_{\boldsymbol{\alpha} \in \{0,1\}^N} \left((-1)^{|\boldsymbol{\alpha}|} \bigotimes_{n=1}^{N} \mathcal{I}_{\ell_n - \alpha_n}[g] \right)
\end{aligned}$$

where $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_N)$
and $|\boldsymbol{\alpha}| = \alpha_1 + \cdots + \alpha_N$



- The isotropic sparse grid interpolant is defined by choosing the index set $\mathcal{J}$ as

$$\mathcal{J}_L^{\mathrm{sg}} = \left\{ \boldsymbol{\ell} \in \mathbb{N}^N \ : \ |\boldsymbol{\ell}| = \ell_1 + \cdots + \ell_N \leq L \right\}$$

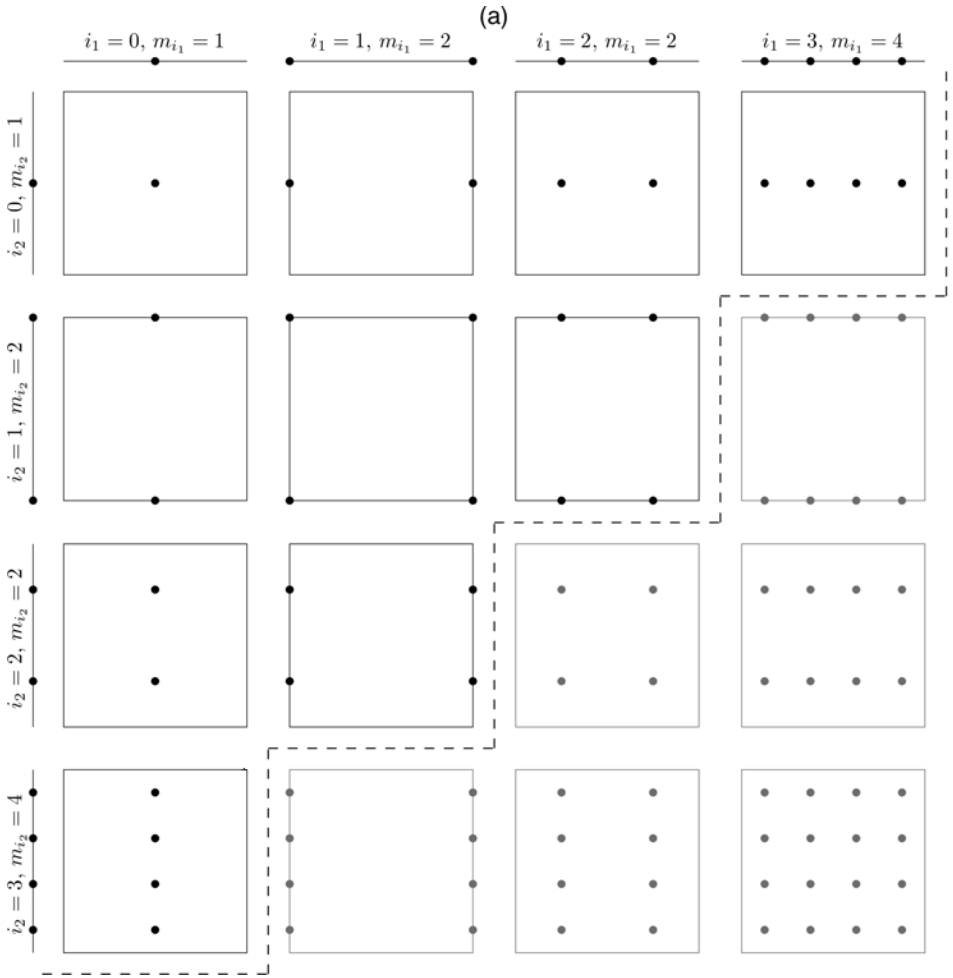
  – the corresponding polynomial subspace $\mathcal{V}_{\mathcal{J}}$ is given by

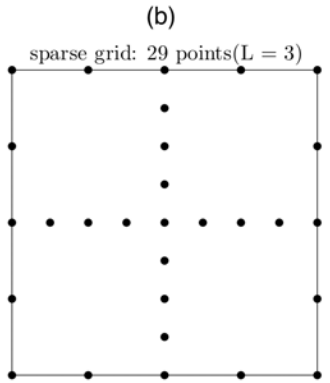
$$\mathcal{V}_{\mathcal{J}_\ell^{\mathrm{sg}}} = \bigoplus_{\boldsymbol{\ell} \in \mathcal{J}_L^{\mathrm{sg}}} W_{\boldsymbol{\ell}} = \bigoplus_{\ell=0}^{L} \bigoplus_{|\boldsymbol{\ell}|=\ell} W_{\boldsymbol{\ell}}$$

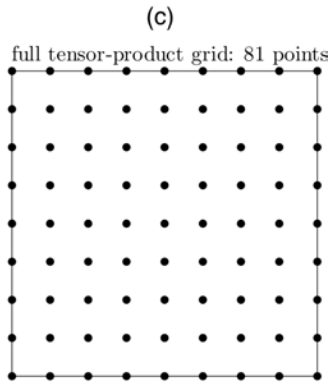
  – the sparse grid interpolant can then be naturally obtained by summing all the $\Delta_{\boldsymbol{\ell}}$ associated with $\mathcal{J}_L^{\mathrm{sg}}$

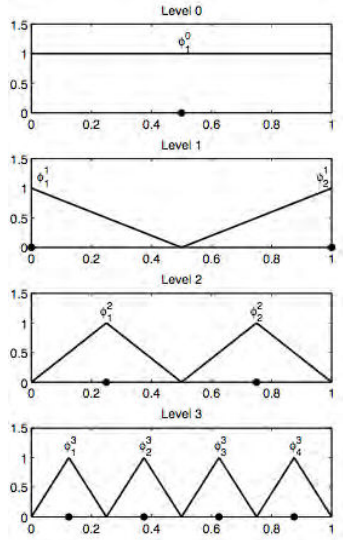
$$\mathcal{I}_\ell^{\mathrm{sg}}[g](\boldsymbol{y}) = \sum_{\ell=0}^{L} \sum_{|\boldsymbol{\ell}|=\ell} \underbrace{\Delta_{\ell_1} \otimes \cdots \otimes \Delta_{\ell_N}}_{\Delta_{\boldsymbol{\ell}}} [g](\boldsymbol{y})$$

- Sparse grid vs. full tensor product ($L = 3$)









Constructing sparse grid piecewise linear interpolants

- Computing the coefficients of a Lagrange interpolant is equivalent to solving a linear system

$$\boldsymbol{\Psi c} = \boldsymbol{g}$$

where Ψ_{ij} is the value of the j-th basis function evaluated at the i-th interpolation point

  - the interpolation matrix of a tensor product interpolant is the identity matrix due to the delta property $\Longleftarrow \quad \Psi_{ij} = \delta_{ij}$
  - the sparse grid interpolant can also be written as the linear combination of the basis functions in $\mathcal{V}_{\mathcal{J}_\ell^{\mathrm{sg}}}$

$$\mathcal{I}_L^{\mathrm{sg}}[g](\boldsymbol{y}) = \sum_{\ell=0}^{L} \sum_{|\boldsymbol{\ell}|=\ell} \Delta_{\ell_1} \otimes \cdots \otimes \Delta_{\ell_N}[g](\boldsymbol{y}) = \sum_{\ell=0}^{L} \sum_{|\boldsymbol{\ell}|=\ell} \sum_{\mathbf{i} \in B_{\boldsymbol{\ell}}} c_{\boldsymbol{\ell},\mathbf{i}}\, \psi_{\boldsymbol{\ell},\mathbf{i}}(\boldsymbol{y})$$

  - the coefficient matrix is no longer diagonal, but we would like to see if we could exploit the partial delta property to solve it explicitly (without recourse to a linear solver)

- The sparse grid interpolation can be written as a recursive process

$$\begin{aligned}\mathcal{I}_L^{\mathrm{sg}}[g](\boldsymbol{y}) &= \sum_{\ell=0}^{L} \sum_{|\boldsymbol{\ell}|=\ell} \Delta_{\ell_1} \otimes \cdots \otimes \Delta_{\ell_N}[g](\boldsymbol{y}) \\ &= \mathcal{I}_{L-1}^{\mathrm{sg}}[g](\boldsymbol{y}) + \sum_{|\boldsymbol{\ell}|=L} \Delta_{\ell_1} \otimes \cdots \otimes \Delta_{\ell_N}[g](\boldsymbol{y}) \\ &= \mathcal{I}_{L-1}^{\mathrm{sg}}[g](\boldsymbol{y}) + \sum_{|\boldsymbol{\ell}|=L} \sum_{\mathbf{i} \in B_{\boldsymbol{\ell}}} c_{\boldsymbol{\ell},\mathbf{i}}\, \psi_{\boldsymbol{\ell},\mathbf{i}}(\boldsymbol{y})\end{aligned}$$

- For any $\boldsymbol{\ell}$ satisfying $|\boldsymbol{\ell}| = L$ and any $\boldsymbol{\ell}'$ satisfying $|\boldsymbol{\ell}'| \leq L$, there exists one component $\ell_n > \ell'_n$ such that $\psi_{\ell_n,i_n}(y_{\ell'_n,j_n}) = 0$ due to the partial delta property
  - thus, we have

$$\psi_{\boldsymbol{\ell},\mathbf{i}}(\boldsymbol{y}_{\boldsymbol{\ell}',\boldsymbol{j}}) = 0 \quad \text{for} \quad \boldsymbol{\ell} \geq \boldsymbol{\ell}'$$

- Now, suppose we are given $\mathcal{I}^{\mathrm{sg}}_{L-1}[g](\boldsymbol{y})$ and add new points on level L
  - substituting any interpolation points $\boldsymbol{y}_{\boldsymbol{\ell},\mathbf{i}}$ satisfying $|\boldsymbol{\ell}| < L$, we have
$$\mathcal{I}^{\mathrm{sg}}_{L}[g](\boldsymbol{y}_{\boldsymbol{\ell},\mathbf{i}}) = \mathcal{I}^{\mathrm{sg}}_{L-1}[g](\boldsymbol{y}_{\boldsymbol{\ell},\mathbf{i}})$$
  - this means the new added basis functions on level L will not change the coefficients of $\mathcal{I}^{\mathrm{sg}}_{L-1}[g]$
- Substituting any new added point $\boldsymbol{y}_{\boldsymbol{\ell},\mathbf{i}}$ satisfying $|\boldsymbol{\ell}| = L$, we have
$$\mathcal{I}^{\mathrm{sg}}_{L-1}[g](\boldsymbol{y}_{\boldsymbol{\ell},\mathbf{i}}) + c_{\boldsymbol{\ell},\mathbf{i}}\psi_{\boldsymbol{\ell},\mathbf{i}}(\boldsymbol{y}) = g(\boldsymbol{y}_{\boldsymbol{\ell},\mathbf{i}})$$
  so that $c_{\boldsymbol{\ell},\mathbf{i}}$ can be computed explicitly by
$$c_{\boldsymbol{\ell},\mathbf{i}} = g(\boldsymbol{y}_{\boldsymbol{\ell},\mathbf{i}}) - \mathcal{I}^{\mathrm{sg}}_{L-1}[g](\boldsymbol{y}_{\boldsymbol{\ell},\mathbf{i}})$$
  - $c_{\boldsymbol{\ell},\mathbf{i}}$ is called the surplus
- What we have shown is that

  the sparse grid interpolation matrix is a lower triangular matrix

The error in sparse grid piecewise linear interpolation

- Define the mixed derivative and norm

$$\|g\|^2_{H^s_{\text{mix}}} = \sum_{\boldsymbol{\alpha}=0}^{s} |D^{\boldsymbol{\alpha}} g|_2^2 \qquad \text{where} \qquad D^{\boldsymbol{\alpha}} g = \frac{\partial^{|\boldsymbol{\alpha}|} g}{\partial y_1^{\alpha_1} \cdots \partial y_d^{\alpha_d}}$$

  and the associated space H^s_{mix}

$$H^s_{\text{mix}} = \left\{ g : \Gamma \to \mathbb{R} \; : \; \|g\|_{H^s_{\text{mix}}} < \infty \right\}$$

- For a function $g \in H^2_{\text{mix}}$, the error of the sparse grid interpolant the error of the full tensor product interpolant are

$$\|g - \mathcal{I}_L^{\text{sg}}[g]\|_{L^2} = \mathcal{O}\left(h_L^2 \log(h_L^{-1})^{N-1}\right) \qquad \|g - \mathcal{I}_L^{\text{tp}}[g]\|_{L^2} = \mathcal{O}\left(h_L^2\right)$$

- However the complexity comparison, i.e., the number of grid points, is

$$\#(\mathcal{V}_L^{\text{sg}}) = \mathcal{O}\left(h_L^{-1} \log(h_L^{-1})^{N-1}\right) \qquad \text{vs.} \qquad \#(\mathcal{V}_L^{\text{tp}}) = \mathcal{O}\left(h_L^{-N}\right)$$

ADAPTIVE LOCAL SPARSE GRID INTERPOLATION

- Adaptive mesh refinement (AMR) has been widely used to approximate functions with irregular behavior, e.g., steep gradients, sharp transitions, jump discontinuities, etc.

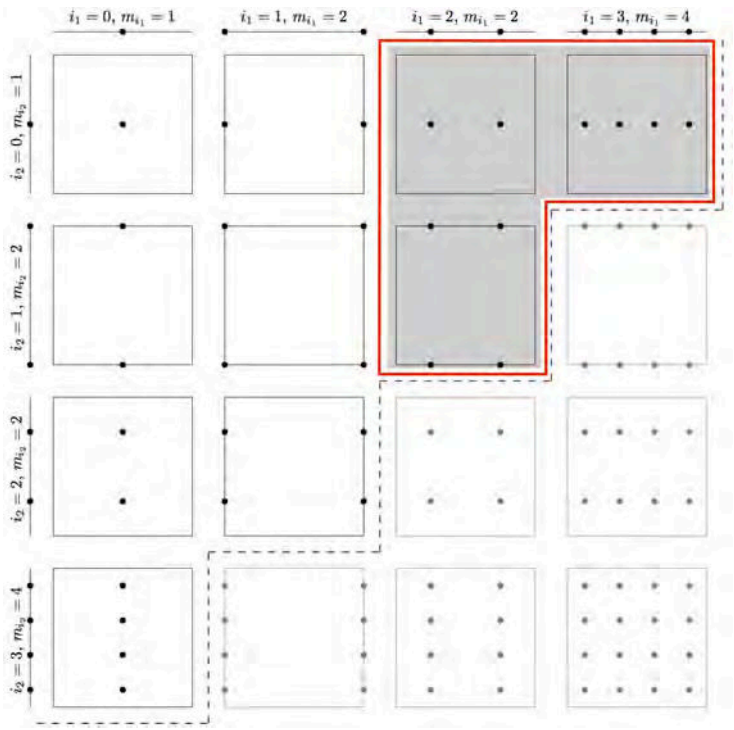
- The key to successful AMR is to exploit an a posteriori error indicator to estimate the error of the current approximation and to guide us as to where to add new grid points

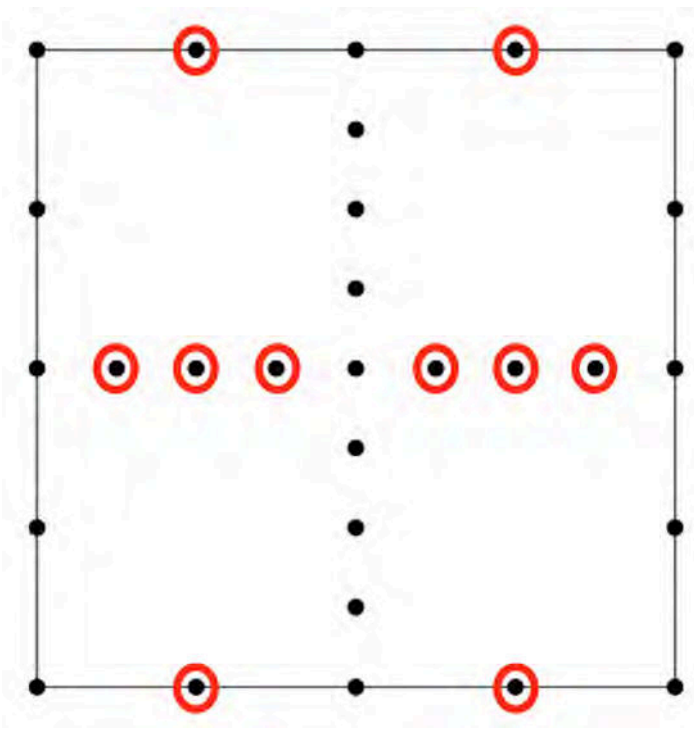
- Questions
  - can we do mesh refinement on sparse grids?
  - if so, what is the error indicator?

Strategy 1: Dimensional adaptive anisotropic sparse grids

- This strategy shares ideas used for global sparse grids
- Instead of having the same resolution along each direction, we can add the anisotropy by defining a weighted norm for the multi-index $\boldsymbol{\ell}$ $\Longrightarrow$

$$\mathcal{J}_L^{\text{aniso}} = \left\{ \boldsymbol{\ell} \in \mathbb{N}^N \ : \ |\boldsymbol{\ell}|_w = w_1 \ell_1 + \cdots + w_N \ell_N \leq L \right\}$$





Strategy 2: Local adaptive refinement

- Recall the expression of the surplus $c_{\boldsymbol{\ell},\mathbf{i}}$

$$c_{\boldsymbol{\ell},\mathbf{i}} = g(\boldsymbol{y}_{\boldsymbol{\ell},\mathbf{i}}) - \mathcal{I}^{\mathrm{sg}}_{L-1}[g](\boldsymbol{y}_{\boldsymbol{\ell},\mathbf{i}})$$

  that can be bounded by

$$|c_{\boldsymbol{\ell},\mathbf{i}}| \leq C 2^{-2|\boldsymbol{\ell}|}$$

  – thus, the surplus can be used as an error indicator to guide the refinement

- For a given threshold $\tau > 0$

  the level L interpolant $\mathcal{I}^{\mathrm{sg}}_{\tau,L}[g]$

  retains only the terms of the isotropic SG interpolant $\mathcal{I}^{\mathrm{sg}}_{L}[g]$

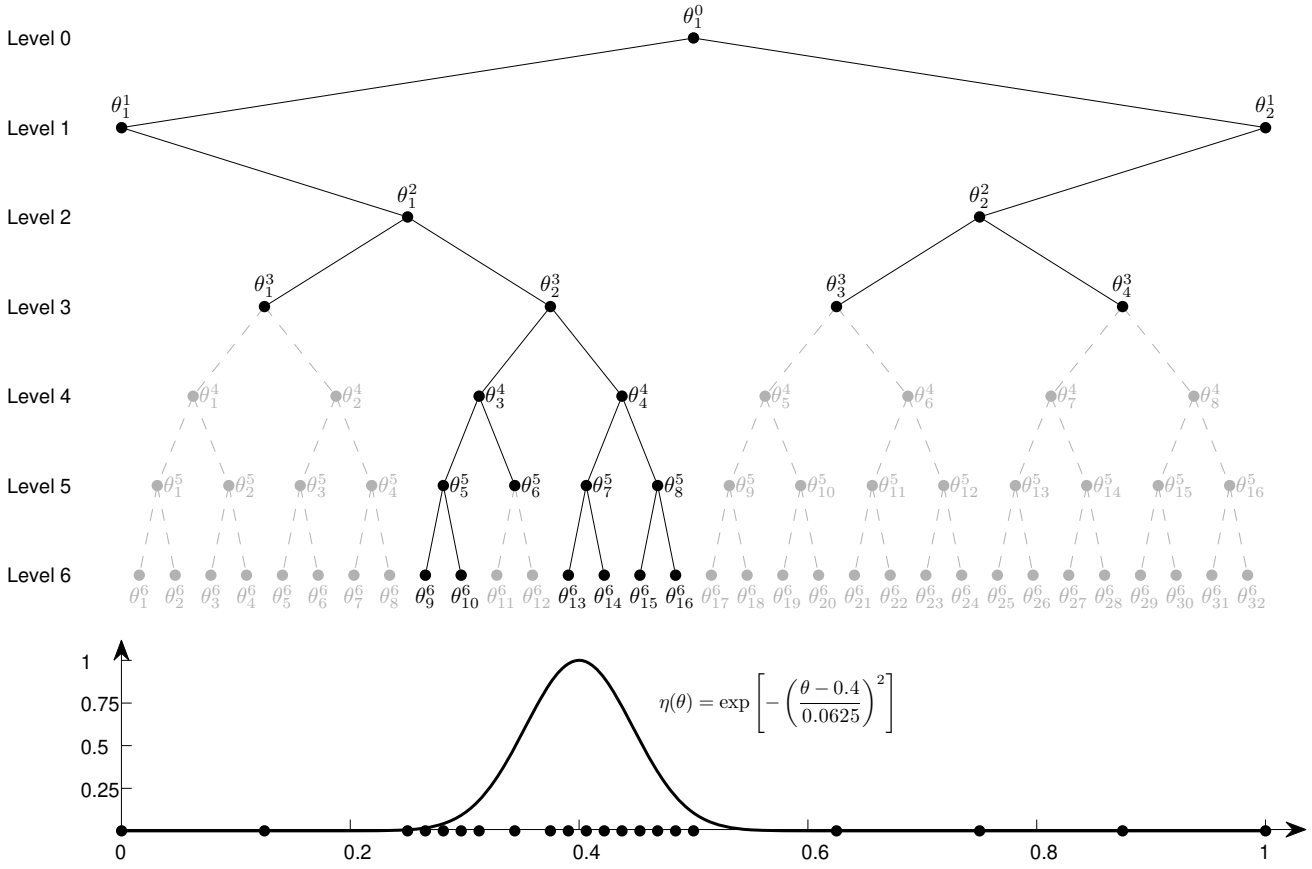
  for which the magnitudes of the corresponding surpluses are larger than τ

  – that is

$$\mathcal{I}^{\mathrm{sg}}_{\tau,L}[g](\boldsymbol{y}) = \sum_{\ell=0}^{L} \sum_{|\boldsymbol{\ell}|=\ell} \sum_{\mathbf{i}\in B^{\tau}_{\boldsymbol{\ell}}} c_{\boldsymbol{\ell},\mathbf{i}}\, \psi_{,\mathbf{i}}(\boldsymbol{y}) \quad \text{with} \quad B^{\tau}_{\boldsymbol{\ell}} = \{\mathbf{i} \in B \ : \ |c_{\boldsymbol{\ell},\mathbf{i}}| > \tau\}$$

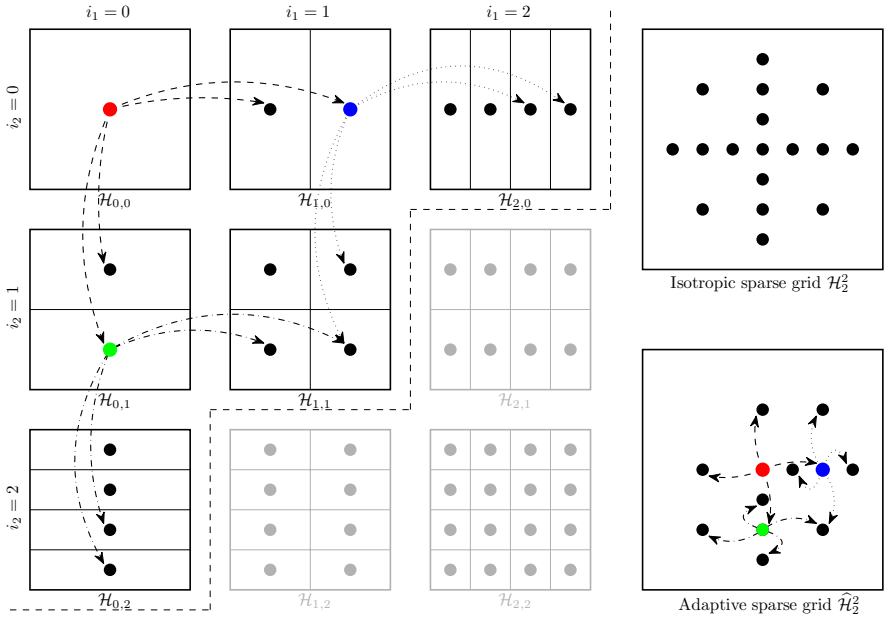
- One-dimensional example

  level 6 adaptive sparse grid with $\tau = 0.01$



The resulting adaptive grid has 21 points (black points) whereas the full grid has 65 points (black and gray points)

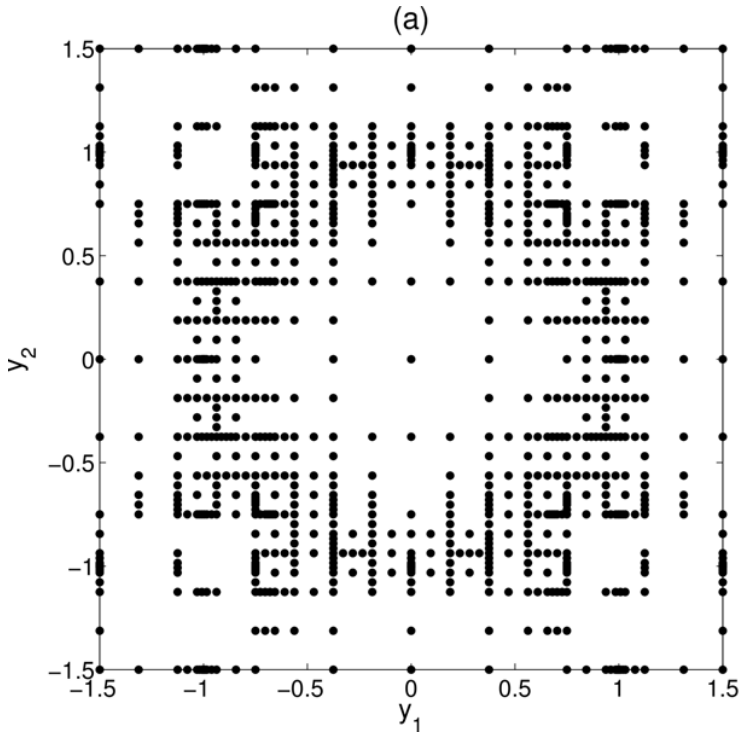
- Two-dimensional example
  level 2 sparse grid

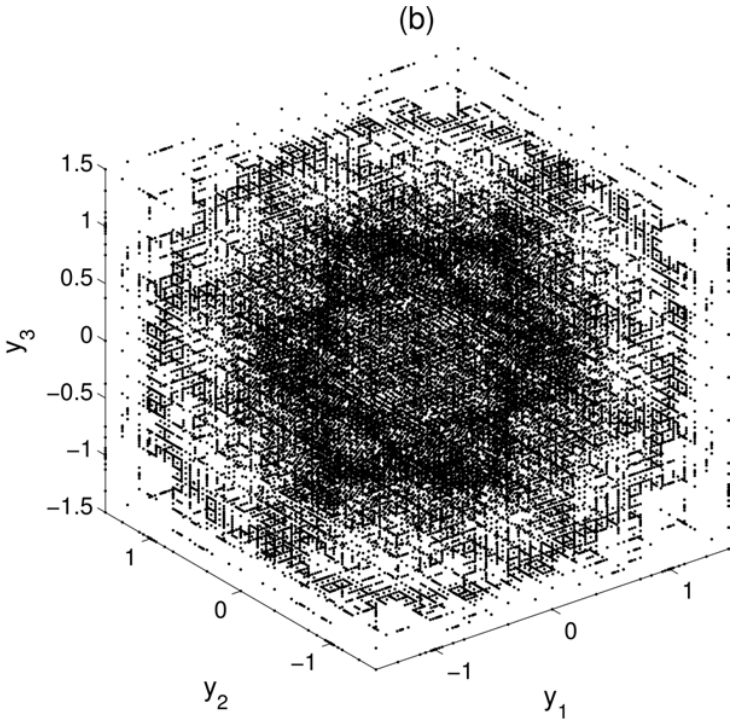


With adaptivity, each point that corresponds to a large surplus, e.g., the points in red, blue, or green, lead to 2 children points added in each direction resulting in the adaptive sparse grid having 12 points instead of the 17 point standard level sparse grid

- Characteristic functions in two and three dimensions
  - results for the characteristic function

$$g(\boldsymbol{y}) = \begin{cases} 1 & \sqrt{y_1^2 + \cdots + y_N^2} \leq 1 \\ 0 & \text{otherwise} \end{cases}$$

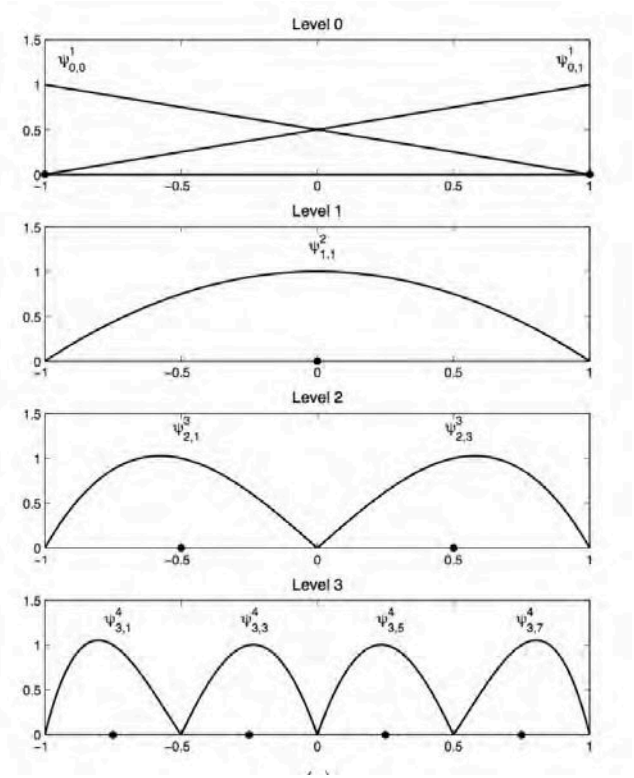


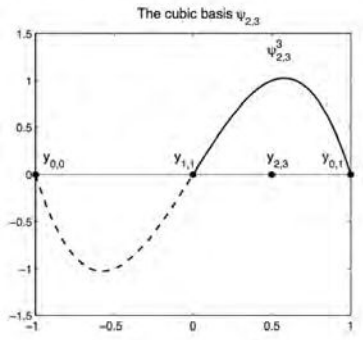


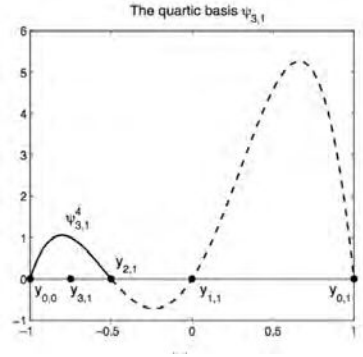
SPARSE GRIDS WITH OTHER TYPES OF BASIS FUNCTIONS

Higher-order hierarchical piecewise polynomials

- When defining high-order bases, we would like to retain the partial delta property
  - new basis functions on level ℓ vanish at all the level $\ell' \leq \ell$ level nodes except, of course, at the node corresponding to the basis function

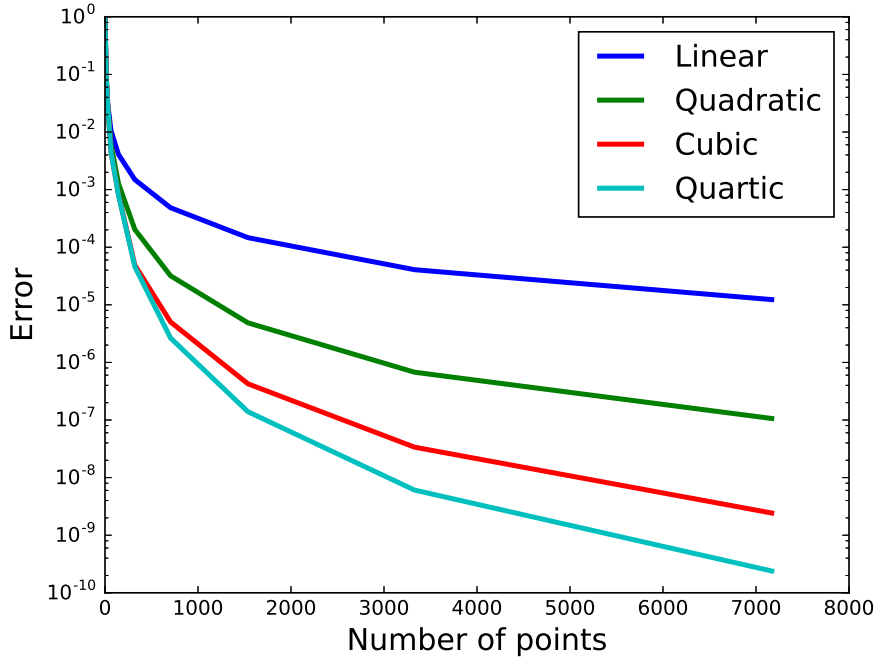






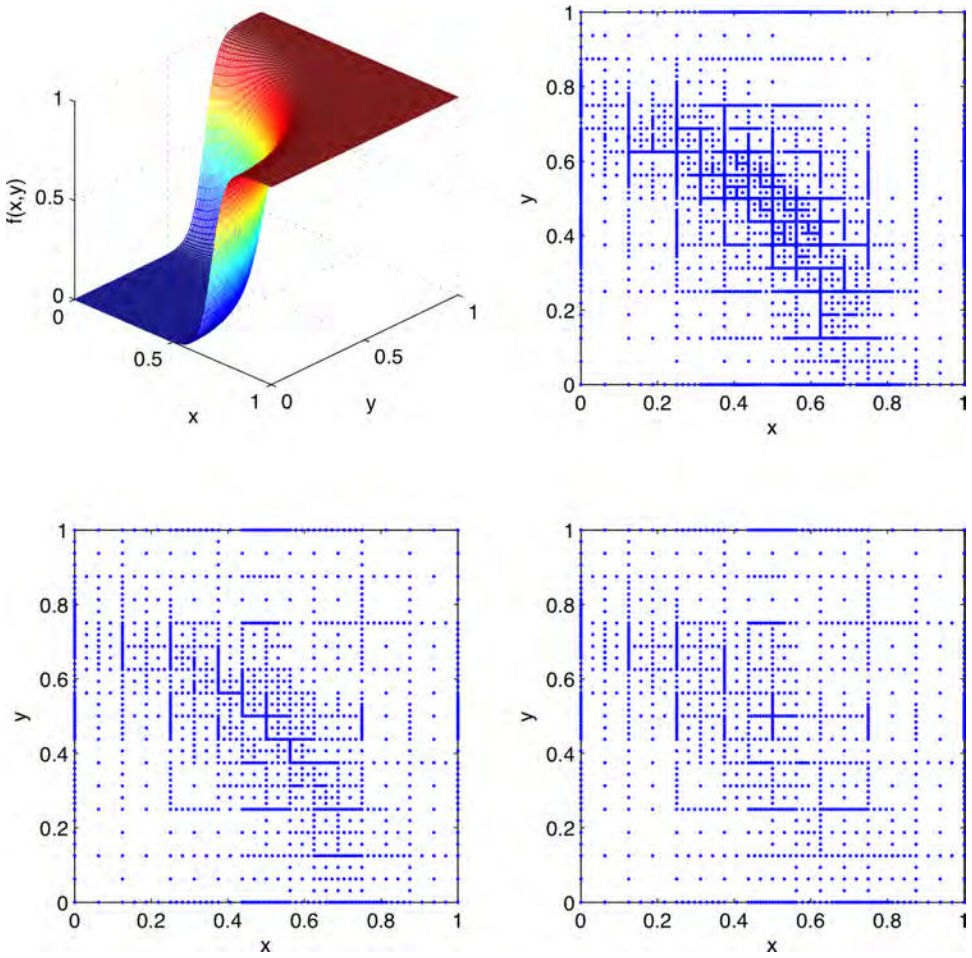
- For a function $g \in H^{p+1}_{\text{mix}}$, the error of the sparse grid interpolant $\mathcal{I}^{\text{sg}}_L[g]$ is

$$\|g - \mathcal{I}^{\text{sg}}_L[g]\|_{L^2} = \mathcal{O}\left(h_L^{p+1} \log(h_L^{-1})^{N-1}\right)$$



Error decay for the 2D function $g(\boldsymbol{y}) = \exp(-y_1^2 - y_2^2)$

- Adaptivity with high-order sparse-grid interpolation
  - linear, quadratic and cubic approximations with $tol = 10^{-3}$



Wavelet bases for sparse grids

- Motivation: The aforementioned hierarchical bases may have some stability issues when doing adaptivity
  - for example, for a $g(\boldsymbol{y})$ and for finite element bases, we have the upper bound

$$\|\mathcal{I}_L^{\mathrm{sg}}[g]\|_{L^2}^2 \leq C \sum_{\ell=0}^{L} \sum_{|\boldsymbol{\ell}|=l} \sum_{\mathbf{i} \in B_{\boldsymbol{\ell}}} |c_{\boldsymbol{\ell},\mathbf{i}}|^2$$

    which may be an over estimate, meaning that a big coefficient may only contribute very little to the approximation

- Ideally, we would like our basis to be a Riesz basis for which one has upper and lower bounds
  - i.e., there exists constants c and C independent of the level L such that

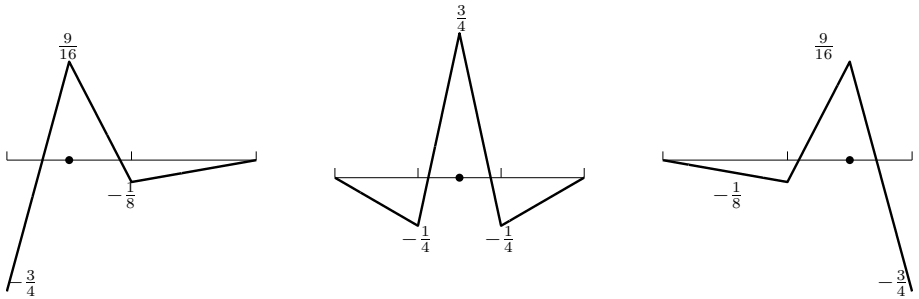
$$c \sum_{\ell=0}^{L} \sum_{|\boldsymbol{\ell}|=l} \sum_{\mathbf{i} \in B_{\boldsymbol{\ell}}} |c_{\boldsymbol{\ell},\mathbf{i}}|^2 \leq \|\mathcal{I}_L^{\mathrm{sg}}[g]\|_{L^2}^2 \leq C \sum_{\ell=0}^{L} \sum_{|\boldsymbol{\ell}|=l} \sum_{\mathbf{i} \in B_{\boldsymbol{\ell}}} |c_{\boldsymbol{\ell},\mathbf{i}}|^2$$

- Finite element bases are not Riesz bases
  - the lower bound does not hold
- However, compactly supported hierarchical piecewise polynomial wavelet bases that are Riesz bases can be constructed from a finite element hierarchical basis
- In the linear case, such second-generation wavelets are defined by

$$\phi_{\ell,i} = \psi_{\ell,i} - \frac{1}{4}\psi_{\ell-1,\frac{i-1}{2}} - \frac{1}{4}\psi_{\ell-1,\frac{i+1}{2}} \quad \text{for } 1 < i < 2^\ell - 1,\ i \text{ odd}$$

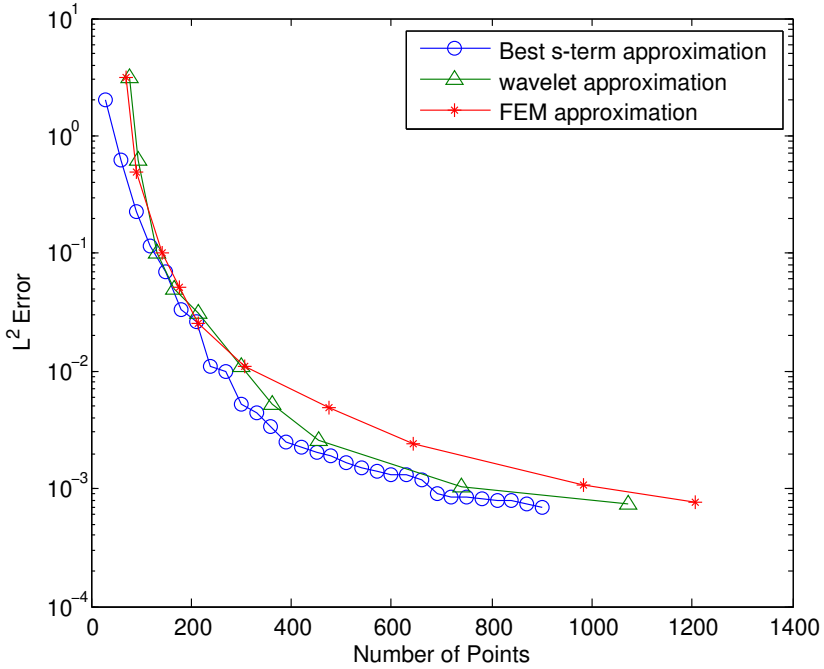
$$\phi_{\ell,i} = \psi_{\ell,i} - \frac{3}{4}\psi_{\ell-1,\frac{i-1}{2}} - \frac{1}{8}\psi_{\ell-1,\frac{i+1}{2}} \quad \text{for } i = 1$$

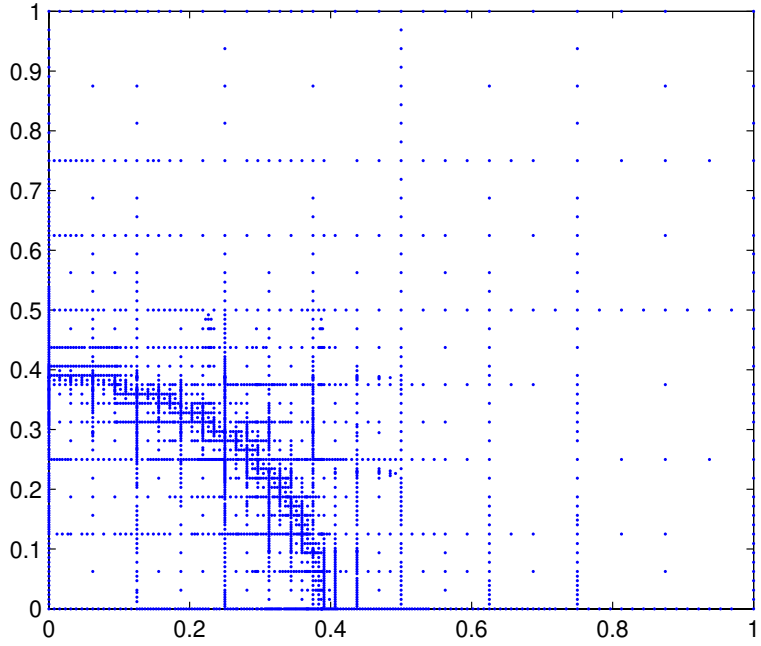
$$\phi_{\ell,i} = \psi_{\ell,i} - \frac{1}{8}\psi_{\ell-1,\frac{i-1}{2}} - \frac{3}{4}\psi_{\ell-1,\frac{i+1}{2}} \quad \text{for } i = 2^\ell - 1$$



- An example: the bivariate function

$$f(x,y) = \frac{1}{|0.15 - x^2 - y^2| + 0.1}$$





Compared to the adaptive finite element approximation, the L^2 error of the adaptive wavelet approximation is closer to that of the best s-term approximation

APPLICATION 1
HIERARCHICAL ACCELERATION OF STOCHASTIC COLLOCATION METHODS

Computational costs of stochastic collocation methods

- We consider the computational costs of using stochastic collocation methods for problems involving parameterized PDEs
  - if we have M collocation points, we are required to solve the PDE M times, once for each collocation point
  - here we consider linear elliptic equations and finite element methods
  - we denote the approximate finite element solution for the collocation point $\boldsymbol{y}_{\boldsymbol{\ell},\mathbf{i}}$ by

$$u_{J_h}(\mathbf{x}, \boldsymbol{y}_{\boldsymbol{\ell},\mathbf{i}}) = \sum_{j=1}^{J_h} u_{j,\boldsymbol{\ell},\mathbf{i}} \phi_j(\mathbf{x})$$

    where J_h denotes the number of finite element degrees of freedom

– we also assume that the stiffness matrices are symmetric and positive definite and that a conjugate gradient method is used to solve the linear systems, resulting in the approximation to the finite element solution given by

$$\widetilde{u}_{J_h}(\mathbf{x}, \boldsymbol{y}_{\boldsymbol{\ell},\mathbf{i}}) = \sum_{j=1}^{J_h} \widetilde{u}_{j,\boldsymbol{\ell},\mathbf{i}} \phi_j(x) \approx u_{J_j}(\mathbf{x}, \boldsymbol{y}_{\boldsymbol{\ell},\mathbf{i}})$$

where $\widetilde{\boldsymbol{u}}_{\boldsymbol{\ell},\mathbf{i}} = (\widetilde{u}_{1,\boldsymbol{\ell},\mathbf{i}}, \ldots, \widetilde{u}_{J_h,\boldsymbol{\ell},\mathbf{i}})^\top$ is the output of the solver corresponding to the collocation point $\boldsymbol{y}_{\boldsymbol{\ell},\mathbf{i}}$

- In the case of using conjugate gradient methods, the error $\boldsymbol{e}_{\mathbf{i}}{}^k = \boldsymbol{u}_{\mathbf{i}} - \boldsymbol{u}_{\mathbf{i}}^k$ is bounded by

$$\|\boldsymbol{e}_{\mathbf{i}}{}^k\|_{\boldsymbol{A}_{\mathbf{i}}} \leq 2 \left(\frac{\sqrt{\kappa_{\mathbf{i}}} - 1}{\sqrt{\kappa_{\mathbf{i}}} + 1} \right)^k \|\boldsymbol{e}_{\mathbf{i}}{}^0\|_{\boldsymbol{A}_{\mathbf{i}}}$$

- The total computational cost for constructing $\widetilde{u}_{J_h,M_\ell} \approx u_{J_h,M_\ell}$ is given by

$$\mathcal{C}_{\text{total}} = \sum_{l=0}^{L} \sum_{|\boldsymbol{\ell}|=l} \sum_{\mathbf{i} \in B} \mathcal{M}_{\mathbf{i}}$$

  where $\mathcal{M}_{\mathbf{i}}$ is the number of iterations needed at the collocation point $\boldsymbol{y}_{\mathbf{i}}$.

- The approximation $\widetilde{u}_{J_h,M_\ell}(x, \boldsymbol{y})$ can be represented in a hierarchical manner:

$$\widetilde{u}_{J_h,M_\ell}(x, \boldsymbol{y}) = \widetilde{u}_{J_h,M_{\ell-1}}(x, \boldsymbol{y}) + \sum_{g(\boldsymbol{\ell})=L} \sum_{\mathbf{i} \in B_{\mathbf{l}}} \widetilde{c}_{\mathbf{l},\mathbf{i}}(x) \cdot \psi_{\mathbf{l},\mathbf{i}}(\boldsymbol{y})$$

- At each collocation point $\boldsymbol{y}_{\mathbf{i}}$ on level L, $\boldsymbol{u}_{\mathbf{i}} = (u_{1,\mathbf{i}}, \ldots, u_{J_h,\mathbf{i}})^\top$ can be represented by

$$u_{j,\mathbf{i}} = u_{J_h,M_{\ell-1}}(x_j, \boldsymbol{y}_{\mathbf{i}}) + c_{j,\mathbf{i}}, \quad \text{for} \quad j = 1, \ldots, J_h$$

- Key idea: due to the decay of $|c_{j,\mathbf{i}}|$ as $|\boldsymbol{\ell}| \to \infty$, the initial guess for the CG solver is chosen as

$$\widetilde{\boldsymbol{u}}_{\mathbf{i}}{}^{0} = \left(\widetilde{u}_{J_h,M_{\ell-1}}(x_1,\boldsymbol{y}_{\mathbf{i}}),\ldots,\widetilde{u}_{J_h,M_{\ell-1}}(x_{J_h},\boldsymbol{y}_{\mathbf{i}})\right)^{\top}$$

where the error of such prediction is, for $j = 1,\ldots,J_h$,

$$\left|\widetilde{u}^{0}_{j,\mathbf{i}} - u(x_j,\boldsymbol{y}_{\mathbf{i}})\right| \leq \left|\widetilde{u}_{J_h,M_{\ell-1}}(x_j,\boldsymbol{y}_{\mathbf{i}}) - u_{J_h,M_{\ell-1}}(x_j,\boldsymbol{y}_{\mathbf{i}})\right| + c_{j,\mathbf{i}}$$

- Without hierarchical acceleration, the minimum cost $\mathcal{C}_{\min}$ for building the standard piecewise linear SG approximation $\widetilde{u}_{J_h,M_\ell}(\mathbf{x},\boldsymbol{y})$ with the prescribed accuracy $\varepsilon > 0$ can be bounded by

$$\mathcal{C}_{\min} \leq \frac{\alpha_1}{N}\left[\alpha_2 + \alpha_3\frac{\log_2\left(\frac{3C_{\mathrm{sg}}}{\varepsilon}\right)}{N}\right]^{\alpha_4 N}\left(\frac{3C_{\mathrm{sg}}}{\varepsilon}\right)^{\alpha_5}$$
$$\times\frac{1}{\log_2\left(\frac{\sqrt{\kappa}+1}{\sqrt{\kappa}-1}\right)}\left[\alpha_6\log_2\left(\frac{3C_{\mathrm{sg}}}{\varepsilon}\right) + \log_2(\sqrt{\kappa}\tau_0) + \alpha_7 N + \alpha_8\right]$$

where the constants $\alpha_1,\ldots,\alpha_8$ are independent of L, N and ε.

- With hierarchical acceleration the minimum cost $\mathcal{C}_{\min}$ for building the standard piecewise linear SG approximation $\widetilde{u}_{J_h,M_\ell}(x,\boldsymbol{y})$ with the prescribed accuracy $\varepsilon > 0$ can be bounded by

$$\mathcal{C}_{\min} \leq \alpha_1 \left[\alpha_2 + \alpha_3 \frac{\log_2\left(\frac{2C_{\mathrm{sg}}}{\varepsilon}\right)}{N}\right]^{\alpha_4 N} \left(\frac{2C_{\mathrm{sg}}}{\varepsilon}\right)^{\alpha_5} \times \frac{1}{\log_2\left(\frac{\sqrt{\kappa}+1}{\sqrt{\kappa}-1}\right)} \left[2N - \log_2(N) + \alpha_9 + \log_2(\sqrt{\kappa})\right]$$

where the constants $\alpha_1, \ldots, \alpha_5$ and α_9 are independent of L, N, and ε

- Consider the 2D Poisson equation with random diffusivity and forcing term

$$\begin{cases} \nabla \cdot (a(\mathbf{x}, \boldsymbol{y}) \nabla u(\mathbf{x}, \boldsymbol{y})) = f(\mathbf{x}, \boldsymbol{y}) \quad \text{in} \quad [0,1]^2 \times \Gamma, \\ u(\mathbf{x}, \boldsymbol{y}) = 0 \quad \text{on} \quad \partial D \times \Gamma \end{cases}$$

where a and f arethe nonlinear functions of the random vector $\boldsymbol{y}$ given by

$$a(\mathbf{x}, \boldsymbol{y}) = 0.1 + \exp\left[y_1 \cos(\pi x_1) + y_2 \sin(\pi x_2)\right]$$

and

$$f(\mathbf{x}, \boldsymbol{y}) = 10 + \exp\left[y_3 \cos(\pi x_1) + y_4 \sin(\pi x_2)\right]$$

where y_n for $n = 1, 2, 3, 4$ are iid random variables following the uniform distribution $U([-1, 1])$

  - the quantity of interest is the mean value of the solution over $D \times \Gamma$

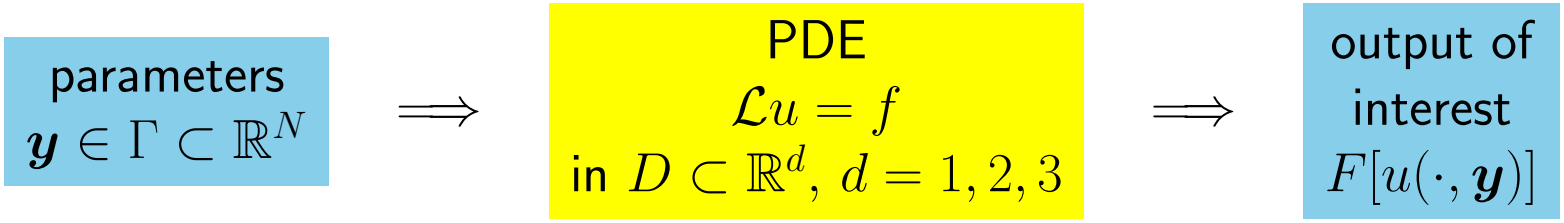
$$\text{QoI} = \mathbb{E}\left[\int_D u(\mathbf{x}, \boldsymbol{y}) d\mathbf{x}\right]$$

- The computational savings of the piecewise SG approach with hierarchical acceleration

Basis type	Error	# SG points	hSGSC cost	hSGSC+acceleration cost	hSGSC+acceleration saving
Linear	1.0e-2	377	13,841	7,497	45.8%
	1.0e-3	1,893	81,068	38,670	52.2%
	1.0e-4	7,777	376,287	167,832	55.3%
Quadratic	1.0e-3	701	29,874	11,877	60.2%
	1.0e-4	2,285	110,744	36,760	66.8%
	1.0e-5	6,149	329,294	100,420	69.5%
Cubic	1.0e-4	1,233	59,344	23,228	60.8%
	1.0e-5	3,233	172,845	57,777	66.5%
	1.0e-6	7,079	415,760	129,433	68.8%

APPLICATION 2
HIGH-DIMENSIONAL DISCONTINUITY DETECTION

Discontinuities in parameter space



- The PDE operator $\mathcal{L}$ depends on N parameters $\boldsymbol{y} = (y_1, y_2, \ldots, y_N) \in \Gamma$

- The parameters $\boldsymbol{y}$ may be affected by uncertainty (measurement error, incomplete description of parameters)
  - uncertainty is modeled by endowing the random vector $\boldsymbol{y}$ with a joint PDF $\varrho(\boldsymbol{y}) = \prod_{n=1}^{N} \varrho_n(y_n)$

- The output of interest $F(\boldsymbol{y}) = F(u(\boldsymbol{y}))$ is a functional of u which may
  - be a smooth function of $\boldsymbol{y}$
  - have steep gradients with respect to $\boldsymbol{y}$
  - have discontinuities with respect to $\boldsymbol{y}$

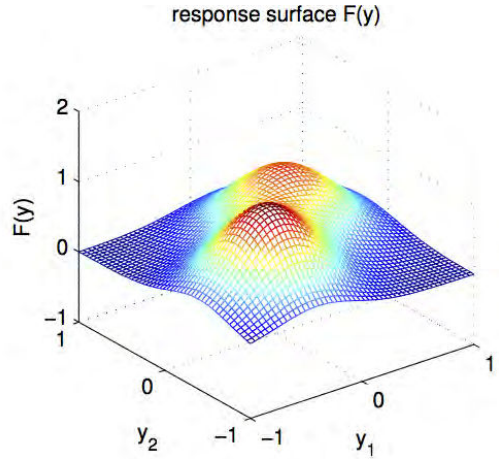
- Basic problem
  - for $F(\boldsymbol{y})$ with discontinuities in the parameter space Γ, we want to
    - identify surfaces of discontinuity
    - subdivide the geometry into subregions of smooth behavior
    - construct a piecewise approximation which is smooth over each subregion

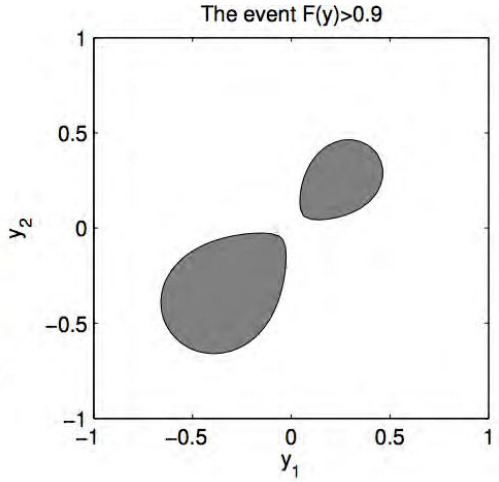
- Event probability problems
  - for any $F(\boldsymbol{y})$, continuous or discontinuous,
    - given the PDF $\rho(\boldsymbol{y})$ for the input parameter $\boldsymbol{y} \in \Gamma$
    - given the threshold F_0
    - given an output of interest $F(\boldsymbol{y}) = F(u(\boldsymbol{y}))$

  determine the probability of the event $F(\boldsymbol{y}) \geq F_0$

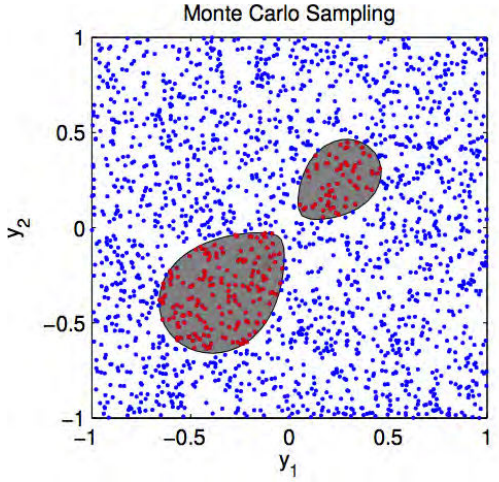
$$\mathbb{P}\left[F(u(\boldsymbol{y})) \geq F_0\right] = \int_{\Gamma} \chi_{\{F(\boldsymbol{y}) \geq F_0\}}(\boldsymbol{y}) \rho(\boldsymbol{y}) d\boldsymbol{y}$$

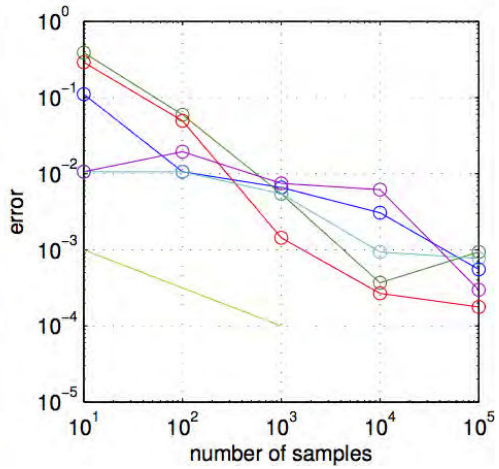
- Monte Carlo approach





no. of samples	MC estimate	MC error
1	0.000000	0.110691
10	0.200000	0.089309
100	0.090000	0.020691
1,000	0.106000	0.004691
10,000	0.108300	0.002391
100,000	0.110430	0.000261
1,000,000	0.110564	0.000127
exact	0.110691	



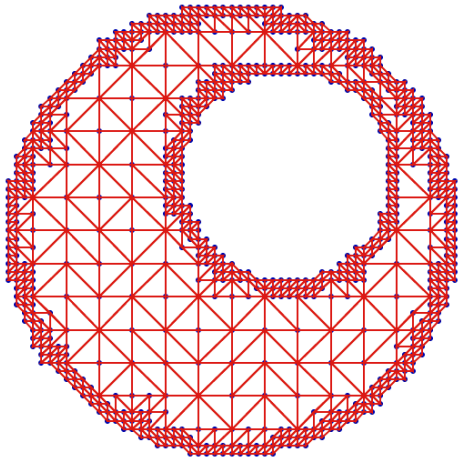


- Monte Carlo is slow to converge
- lots of solutions of the PDE are needed
- quadrature rules with global polynomial approximation do not work

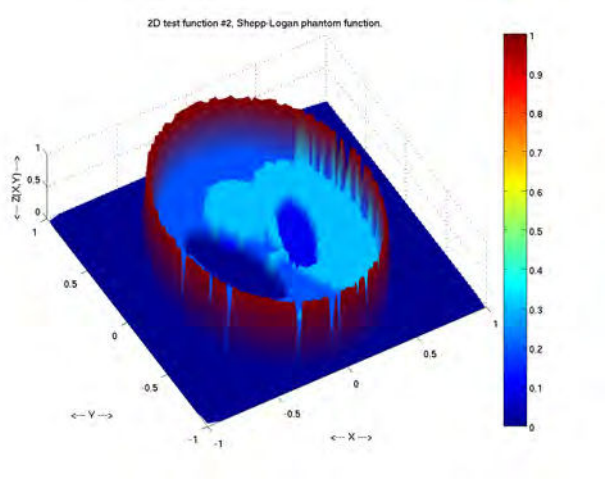
Some more background

- We are given a bounded domain $\Gamma \subset \mathbb{R}^N$ but we are interested in a subdomain Γ_0 which can only be described implicitly
  - e.g. $\Longrightarrow$ for given F_0, $\quad \Gamma_0 = \{\boldsymbol{y} \in \Gamma \ : \ \text{such that} \ F(\boldsymbol{y}) \geq F_0\}$
- Questions:
  - can we detect the boundary of Γ_0?
  - can we accurately and efficiently estimate the integral $\displaystyle\int_{\Gamma_0} \rho(\boldsymbol{y}) d\boldsymbol{y}$
- Our goals is to ameliorate the curse of dimensionality in
  - building approximations to the boundary of Γ_0
  - estimating the above integral faster than conventional Monte Carlo sampling and other approaches

- Existing methods for detecting jump discontinuities include
  - Monte Carlo sampling
  - adaptive triangular mesh refinement
  - discontinuous Galerkin methods
  - polynomial annihilation
  - adaptive hierarchical sparse grids



2D adaptive sparse grid
requires 5,925 points



3D adaptive sparse-grid
requires 21,501 points

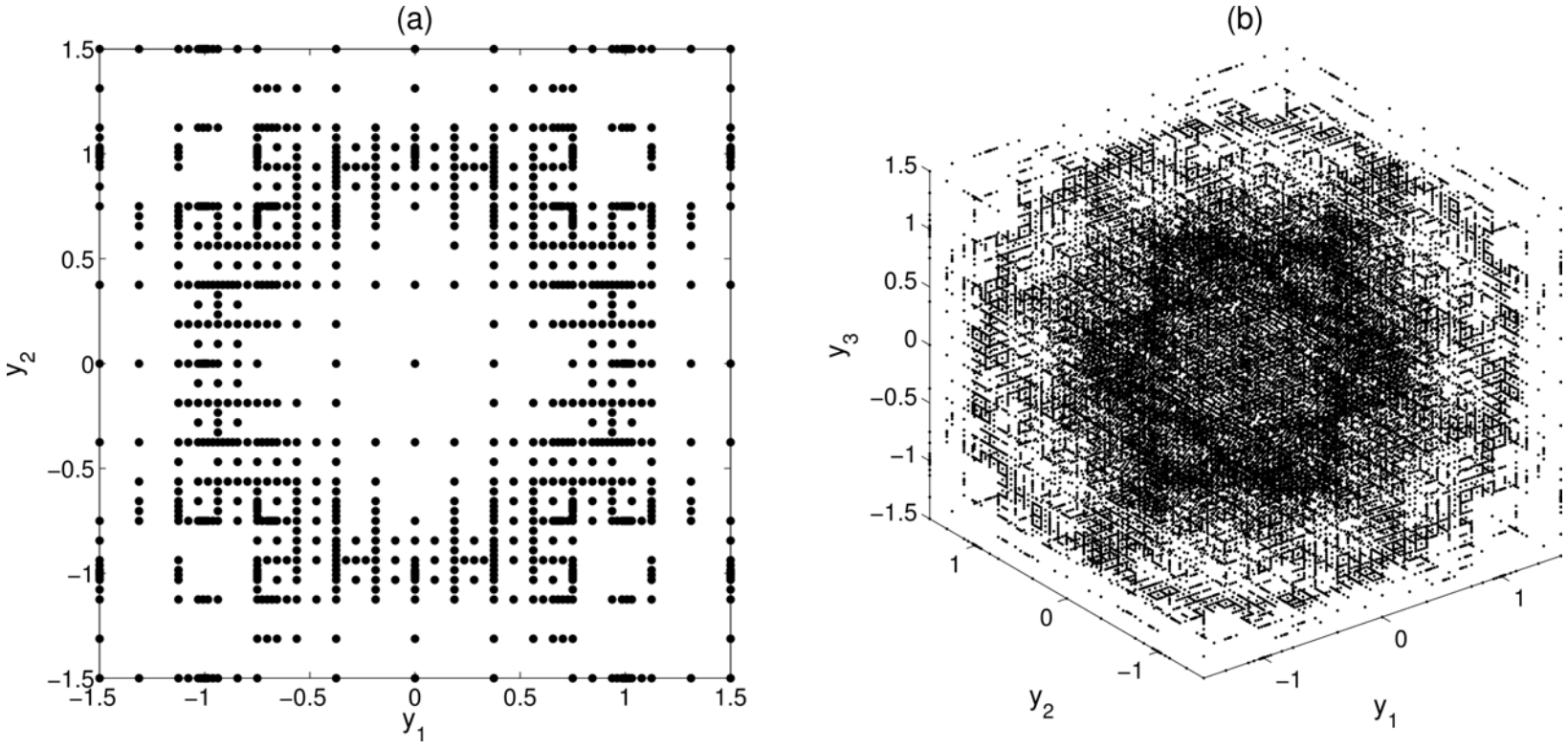
- In the sparse grid approach, an adaptive process based on surpluses is used to select a subset of the tensor product grid that is concentrated near the discontinuity surface
  - for discontinuous functions, the adaptive hierarchical sparse-grid method can incur very high cost, even in low dimensions

$$\Uparrow$$
$$\Uparrow$$
$$\Uparrow$$

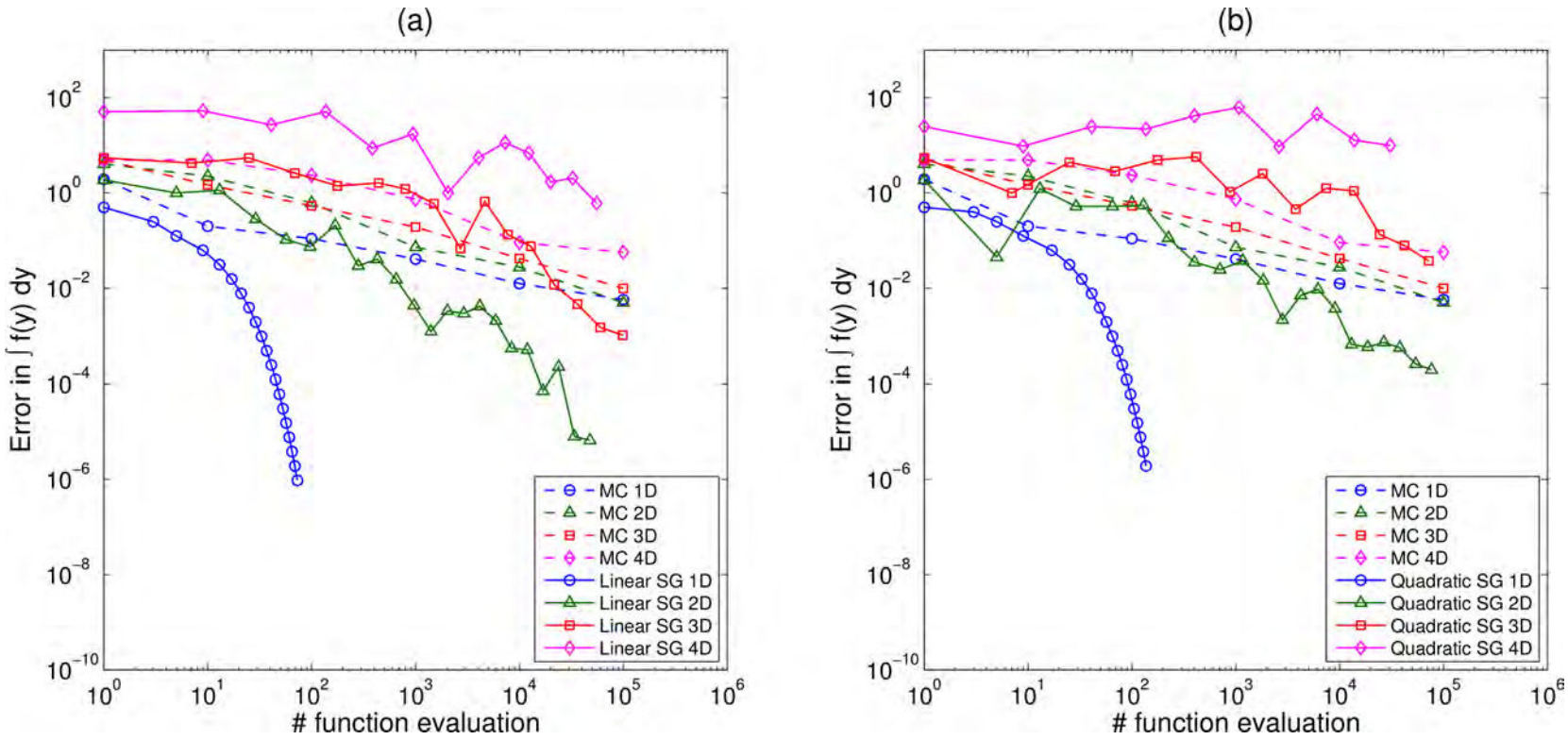
- the sparse-grid interpolant does not converge in L^∞ norm, which means the surplus does not decay to zero
- the adaptivity generates a dense grid around the $N-1$ dimensional discontinuity surface
- many grid points do not contribute much to the approximation
- high-order hierarchical basis functions are useless

- For the characteristic function for the ball

$$\Longrightarrow \quad g(\boldsymbol{y}) = \begin{cases} 1 & \sqrt{y_1^2 + \cdots + y_N^2} \leq 1 \\ 0 & \text{otherwise} \end{cases}$$



2D adaptive sparse grid requires 5,925 points; 3D adaptive sparse-grid requires 21,501 points



The error is measured by $\left| \int g(\boldsymbol{y}) d\boldsymbol{y} - \int \mathcal{I}_L^{\mathrm{sg}}[g](\boldsymbol{y}) d\boldsymbol{y} \right|$

A hyperspherical sparse-grid method for discontinuity detection

- Consider a bounded domain $\Gamma \subset \mathbb{R}^N$, a subdomain $\Gamma_0 \subset \Gamma$, and a characteristic function $g(\boldsymbol{y}) : \Gamma \to \mathbb{R}$ defined by

$$g(\boldsymbol{y}) = \begin{cases} 1 & \text{if } \boldsymbol{y} \in \Gamma_0 \subset \Gamma \\ 0 & \text{otherwise} \end{cases}$$

  - Γ_0 is the characteristic domain
  - $\partial\Gamma_0$ is the discontinuity surface described by an implicit equation $G(\boldsymbol{y}) = 0$ in Γ
    - e.g., a hypersphere can be represented by $G(\boldsymbol{y}) \equiv \sum_{n=1}^{N} y_n^2 - \lambda^2 = 0$

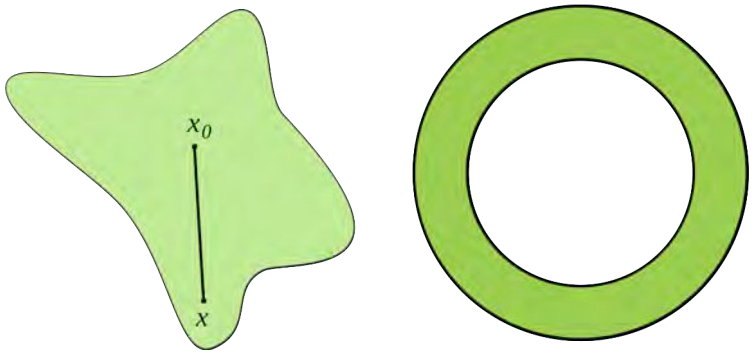
- The goal is to find two bounded domains $\Gamma_1 \subset \Gamma$ and $\Gamma_2 \subset \Gamma$ such that

$$\Gamma_1 \subset \Gamma_0 \subset \Gamma_2 \subset \Gamma \qquad \text{and} \qquad \text{dist}\,(\partial\Gamma_1, \partial\Gamma_2) \leq \varepsilon$$

  where ε is a prescribed accuracy

  - clearly, $g(\boldsymbol{y}) = 0$ for $\boldsymbol{y} \in \partial\Gamma_2$ and $g(\boldsymbol{y}) = 1$ for $\boldsymbol{y} \in \partial\Gamma_1$

- About the domain $\Gamma_0 \in \Gamma$, we assume
  - Γ_0 is a star-convex domain in Γ
  - a point $\boldsymbol{y}_0$ in Γ_0 is given such that for all $\boldsymbol{y}$ in Γ_0, the line segment $\{\boldsymbol{y}_0 + t\boldsymbol{y} \ : \ t \in [0,1]\}$ from $\boldsymbol{y}_0$ to $\boldsymbol{y}$ is in Γ_0



Left: a star-convex domain is not necessarily convex; Right: an annulus is not a star-convex domain (The two figures are from Wikipedia)

- A hyperspherical coordinate system is a generalization of the 2D polar and 3D spherical coordinate systems to N dimensions
  - we have
    - one radial coordinate r ranging over $[0, +\infty)$
    - one angular coordinate θ_{N-1} ranging over $[0, 2\pi)$
    - $N-2$ angular coordinates $\theta_1, \ldots, \theta_{N-2}$ ranging over $[0, \pi)$

- The conversion from hyperspherical coordinates centered at $\boldsymbol{y}_0$ to Cartesian coordinates is given by

$$
\begin{aligned}
y_1 &= y_{0,1} + r\cos(\theta_1) \\
y_2 &= y_{0,2} + r\sin(\theta_1)\cos(\theta_2) \\
y_3 &= y_{0,3} + r\sin(\theta_1)\sin(\theta_2)\cos(\theta_3) \\
&\vdots \\
y_{N-2} &= y_{0,N-2} + r\sin(\theta_1)\cdots\sin(\theta_{N-2})\cos(\theta_{N-1}) \\
y_{N-1} &= y_{0,N-1} + r\sin(\theta_1)\cdots\sin(\theta_{N-2})\sin(\theta_{N-1})
\end{aligned}
$$

- The hyperspherical sparse-grid method for discontinuity detections proceeds as follows
  - transform the Cartesian coordinates $y_1, \ldots, y_N$ to the hyperspherical coordinates $r, \theta_1, \ldots, \theta_{N-1}$ with the given origin point $\boldsymbol{y}_0$
  - each point $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_{N-1})$ corresponds to a ray in $\mathbb{R}^N$ out from $\boldsymbol{y}_0$ in a specific direction
    - due to the star-convexity of the domain Γ_0, there is only one jump discontinuity in each direction $\boldsymbol{\theta}$
  - $\partial\Gamma_0$ can be represented by a function $r = g(\boldsymbol{\theta})$ on the bounded N-1 dimensional domain

$$\Gamma_\theta = \prod_{n=1}^{N-1} [0, \pi] \times [0, 2\pi]$$

    where for any $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_{N-1}) \in \Gamma_\theta$, $(g(\boldsymbol{\theta}), \boldsymbol{\theta})$ is on $\partial\Gamma_0$
  - build an L-level sparse grid $\mathcal{H}_L^{N-1}$ on Γ_θ with a total of M grid points

$$\mathcal{H}_L^{N-1} = \{\boldsymbol{\theta}_m \in \Gamma_\theta \text{ for } m = 1, \ldots, M\}$$

– for an accuracy tolerance ε and for $m = 1, \ldots, M$, from $\boldsymbol{y}_0$, along the direction corresponding $\boldsymbol{\theta}_m \in \mathcal{H}_L^{N-1}$, use 1-D bisection method to find two values g_m^1 and g_m^2 such that

$$g_m^1 \leq g(\boldsymbol{\theta}_m) \leq g_m^2 \qquad \text{and} \qquad |g_m^1 - g_m^2| \leq \varepsilon$$

– build sparse-grid interpolants $g^1(\boldsymbol{\theta})$ and $g^2(\boldsymbol{\theta})$ based on $\{g_m^1, m = 1, \ldots, M\}$ and $\{g_m^2, m = 1, \ldots, M\}$, respectively

- we then have

$$\begin{aligned} \left(g^1(\boldsymbol{\theta}), \boldsymbol{\theta}\right) &\implies \partial\Gamma_1 \\ \left(g^2(\boldsymbol{\theta}), \boldsymbol{\theta}\right) &\implies \partial\Gamma_2 \end{aligned}$$

- number function evaluations $= \sum_{m=1}^{M}$ (number bisection trials for $\boldsymbol{\theta}_m$)

- according to smoothness of the hyper-surface $\partial\Gamma_0$, different types of basis functions can be used, e.g., high-order hierarchical finite element or wavelet bases