SIAM Conference on Computational Science and Engineering

SAND2017-1673C

Ensemble Grouping strategies for embedded Stochastic Collocation

Marta D'Elia, M.S. Ebeida, and E. Phipps (SNL*), A.A. Rushdi (UC Davis)

SIAM CSE 17 – Atlanta, GA, February 27th 2017

*Sandia National Labs, NM – Sandia National Laboratories is a multi mission laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

CCR Center for Computing Research

Sandia National Laboratories

INTRODUCTION

UQ in computational modeling and simulation

quantifying uncertainties is a foundational component of predictive simulation

➡ development and analysis of many UQ methods

- random sampling,
- stochastic collocation
- stochastic Galerkin
- ...

INTRODUCTION

UQ in computational modeling and simulation

quantifying uncertainties is a foundational component of predictive simulation

➡ development and analysis of many UQ methods

- random sampling,
- stochastic collocation
- stochastic Galerkin
- ...

in **large-scale** scientific computing

- high-dimensional uncertain input spaces
- localized and/or non-smooth behavior

➡ research on **reducing the number of samples**

- adaptive sampling methods
- compressed sensing, tensor methods
- multi-level methods
- ...

INTRODUCTION

not enough!

the bottleneck is in the **sample evaluation:** applying these methodologies to large-scale scientific computing problems is often prohibitively expensive

INTRODUCTION

not enough!

the bottleneck is in the **sample evaluation:** applying these methodologies to large-scale scientific computing problems is often prohibitively expensive

our goal: reduce the cost of the evaluation of each sample

➡ propagate together multiple samples through a computational simulation: **embedded ensemble propagation**

E. Phipps, MD, H.C. Edwards, M. Hoemmen, J. Hu, S. Rajamanickam, *Embedded Ensemble Propagation for Improving Performance, Portability and Scalability of Uncertainty Quantification on Emerging Computational Architectures*, SIAM Journal of Sci Comp, 2017

idea: sample-dependent scalars in the code are replaced with small arrays

➡ the cost of assembling and solving the ensemble linear system is **substantially smaller** compared to the sequential case

INTRODUCTION

what can go wrong?

the total number of linear solver iterations for ensemble systems may be **strongly influenced** by which samples comprise the ensemble

➡ critical to the success is the **grouping** of samples into ensembles

INTRODUCTION

what can go wrong?

the total number of linear solver iterations for ensemble systems may be **strongly influenced** by which samples comprise the ensemble

➡ critical to the success is the **grouping** of samples into ensembles

contribution

- analyze a case study where the linear solver iterations significantly vary from sample to sample
- design grouping strategies that maximize the computational gain brought by the ensemble propagation

MD, H.C. Edwards, J. Hu, E. Phipps, S. Rajamanickam, *Ensemble Grouping Strategies for Embedded Stochastic Collocation Methods Applied to Anisotropic Diffusion Problems*, submitted, 2016

Outline

- (brief) **Introduction** to Stochastic Collocation methods
- **Background:** Embedded Ensemble Propagation
- Grouping strategies
- **Case study:** a highly anisotropic diffusion problem
- Future work

Introduction to Stochastic Collocation methods

Based on M. Gunzburger, C. Webster, G. Zhang. Stochastic finite element methods for partial differential equations with random input data. *Acta Numerica (2014), pp. 521–650, 2014*

PROBLEM SETTING

A stochastic elliptic PDE

- $D \subset \mathbb{R}^d$ $(d = 1, 2, 3)$: bounded domain with boundary ∂D
- $(\Omega, \mathcal{F}, \mathbb{P})$: complete probability space

Find $u : \overline{D} \times \Omega$ such that almost surely

$$\begin{cases} \mathcal{L}(a)u = f & \boldsymbol{x} \in D \\ \mathcal{B}u = g & \boldsymbol{x} \in \partial D, \end{cases}$$

where

- $\mathcal{L}$ – elliptic operator defined on D and parametrized by $a(\boldsymbol{x}, \omega)$
- $f(\boldsymbol{x}, \omega)$ – forcing term with $\boldsymbol{x} \in D$ and $\omega \in \Omega$
- $\mathcal{B}$ – boundary operator
- $g(\boldsymbol{x}, \omega)$ – boundary data with $\boldsymbol{x} \in \partial D$ and $\omega \in \Omega$

PROBLEM SETTING

Assumptions on the parameters

A. $f(\boldsymbol{x},\omega)$ and $g(\boldsymbol{x},\omega)$ are not affected by uncertainty

PROBLEM SETTING

Assumptions on the parameters

A. $f(\boldsymbol{x}, \omega)$ and $g(\boldsymbol{x}, \omega)$ are not affected by uncertainty

B. $a(\boldsymbol{x}, \omega)$ is bounded from above and below with probability 1

C. $a(\boldsymbol{x}, \omega)$ can be written as

$a(\boldsymbol{x}, \omega) = a(\boldsymbol{x}, \boldsymbol{y}(\omega))$ in $D \times \Omega$ where

$\boldsymbol{y}(\omega) = (y_1(\omega) \dots y_N(\omega)) \in \mathbb{R}^N$ random vector, uncorr. components

PROBLEM SETTING

Random parameter satisfying B and C:

truncated Karhunen-Loève (KL) expansion of the random field

Mercer's theorem: the second–order correlated random field $a(\boldsymbol{x}, \omega)$ with continuous covariance function $cov(\boldsymbol{x}, \boldsymbol{x}')$ can be written as

$$a(\boldsymbol{x}, \omega) = \mathbb{E}[a(\boldsymbol{x}, \cdot)] + \sum_{n=1}^{\infty} \sqrt{\lambda_n}\, b_n(\boldsymbol{x}) y_n(\omega),$$

λ_n: eigenvalues, in decreasing order, of cov

b_n: corresponding eigenfunctions

$y_n(\omega) \in \mathbb{R}$: uncorrelated random variables

PROBLEM SETTING

Random parameter satisfying B and C:

truncated Karhunen-Loève (KL) expansion of the random field

Mercer's theorem: the second–order correlated random field $a(\boldsymbol{x},\omega)$ with continuous covariance function $cov(\boldsymbol{x},\boldsymbol{x}')$ can be written as

$$a(\boldsymbol{x},\omega) = \mathbb{E}[a(\boldsymbol{x},\cdot)] + \sum_{n=1}^{\infty} \sqrt{\lambda_n}\, b_n(\boldsymbol{x}) y_n(\omega),$$

λ_n: eigenvalues, in decreasing order, of cov

b_n: corresponding eigenfunctions

$y_n(\omega) \in \mathbb{R}$: uncorrelated random variables

Truncated KL expansion: truncation of the summation to the N-th term:

$$a(\boldsymbol{x},\omega) \approx \mathbb{E}[a(\boldsymbol{x},\cdot)] + \sum_{n=1}^{N} \sqrt{\lambda_n}\, b_n(\boldsymbol{x}) y_n(\omega)$$

PROBLEM SETTING

Goal of Uncertainty Quantification

determine *statistical information* about an output of interest that depends on the solution, i.e. a functional $G(\boldsymbol{y})$

Examples:
- $G(\boldsymbol{y}) = \frac{1}{|D|}\int_D u(\boldsymbol{x},\boldsymbol{y})\,d\boldsymbol{x}$
- $G(\boldsymbol{y}) = \max_{\boldsymbol{x}\in D} u(\boldsymbol{x},\boldsymbol{y})$

Quantity of interest: moments of $G(\boldsymbol{y})$

$$\text{e.g. QOI} = \mathbb{E}\left[G(\boldsymbol{y})\right] = \int_\Gamma G(\boldsymbol{y})\rho(\boldsymbol{y})\,d\boldsymbol{y}$$

STOCHASTIC COLLOCATION METHODS

Stochastic collocation (SC) methods:
nonintrusive stochastic sampling methods based on decoupled deterministic solves

STOCHASTIC COLLOCATION METHODS

Stochastic collocation (SC) methods:
nonintrusive stochastic sampling methods based on decoupled deterministic solves

SC methods in a nutshell:

- let $u_h(\cdot, \boldsymbol{y})$ be the semi-discrete approximation of $u(\boldsymbol{x}, \boldsymbol{y})$ for $\boldsymbol{y} \in \Gamma$
- main idea:
  - collocate $u_h(\cdot, \boldsymbol{y})$ on a suitable set of samples $\{\boldsymbol{y}_m\}_{m=1}^{M} \subset \Gamma$, i.e. determine M semi-discrete solutions
  - use those solution to construct a **global polynomial** to represent the fully discrete approximation $u_{h,M}(\boldsymbol{x}, \boldsymbol{y})$

STOCHASTIC COLLOCATION METHODS

Stochastic collocation (SC) methods:
nonintrusive stochastic sampling methods based on decoupled deterministic solves

SC methods in a nutshell:

- let $u_h(\cdot, \boldsymbol{y})$ be the semi-discrete approximation of $u(\boldsymbol{x}, \boldsymbol{y})$ for $\boldsymbol{y} \in \Gamma$

- main idea:
  - collocate $u_h(\cdot, \boldsymbol{y})$ on a suitable set of samples $\{\boldsymbol{y}_m\}_{m=1}^M \subset \Gamma$, i.e. determine M semi-discrete solutions
  - use those solution to construct a **global polynomial** to represent the fully discrete approximation $u_{h,M}(\boldsymbol{x}, \boldsymbol{y})$

- polynomial: **interpolatory**

  set of points $\{\boldsymbol{y}_m\}_{m=1}^M$ + basis functions $\{\psi_m(\boldsymbol{y})\}_{m=1}^M \in \mathcal{P}_{(p)}(\Gamma)$

  $\rightarrow$ fully discrete approximation $u_{h,M}(\boldsymbol{x}, \boldsymbol{y}) = \sum_{m=1}^{M} c_m(\boldsymbol{x}) \psi_m(\boldsymbol{y})$.

STOCHASTIC COLLOCATION METHODS

Fully discrete approximation

$$u_{h,M}(\boldsymbol{x},\boldsymbol{y}) = \sum_{m=1}^{M} c_m(\boldsymbol{x})\psi_m(\boldsymbol{y})$$

for **Lagrange** interpolation

$$c_m(\boldsymbol{x}) = u_h(\boldsymbol{x},\boldsymbol{y}_m) \quad \Rightarrow \quad u_{h,M}(\boldsymbol{x},\boldsymbol{y}) = \sum_{m=1}^{M} u_h(\boldsymbol{x},\boldsymbol{y}_m)\psi_m(\boldsymbol{y})$$

GENERALIZED SPARSE GRIDS

One-dimensional approximation

- $l \in \mathbb{N}_+$: one-dimensional level of approximation
- $\{y_k^l\}_{k=1}^{m(l)}$: sequence of one-dimensional interpolation points
- $m(l)$: number of collocation points at level l

1D interpolation operator: for a continuous function v

$$\mathcal{I}_l[v](y) = \sum_{k=1}^{m(l)} v(y_k^l)\psi_k^l(y), \quad l = 1, 2, \dots$$

ψ_k^l: Lagrange fundamental polynomial of degree $p_l = m(l) - 1$

GENERALIZED SPARSE GRIDS

Multi-dimensional approximation

- $\mathbf{l} = (l_1 \; \dots \; l_N) \in \mathbb{N}_+^N$: a multi-index
- $L \in \mathbb{N}_+$: total level of the sparse grid approximation

GENERALIZED SPARSE GRIDS

Multi-dimensional approximation

- $\mathbf{l} = (l_1 \ldots l_N) \in \mathbb{N}_+^N$: a multi-index
- $L \in \mathbb{N}_+$: total level of the sparse grid approximation

N-dimensional operator at level l: tensor product of 1D operators

N-dimensional sparse grid operator: sum of all operators for $l = 1, 2, \ldots L$

GENERALIZED SPARSE GRIDS

Adaptive grid generation (1D)

e_{jli}: error estimate at $(\boldsymbol{x}_j, y_{li})$=($j$-th DOF, i-th sample at level l)

τ: user-defined error tolerance

Algorithm

at each successive interpolation level

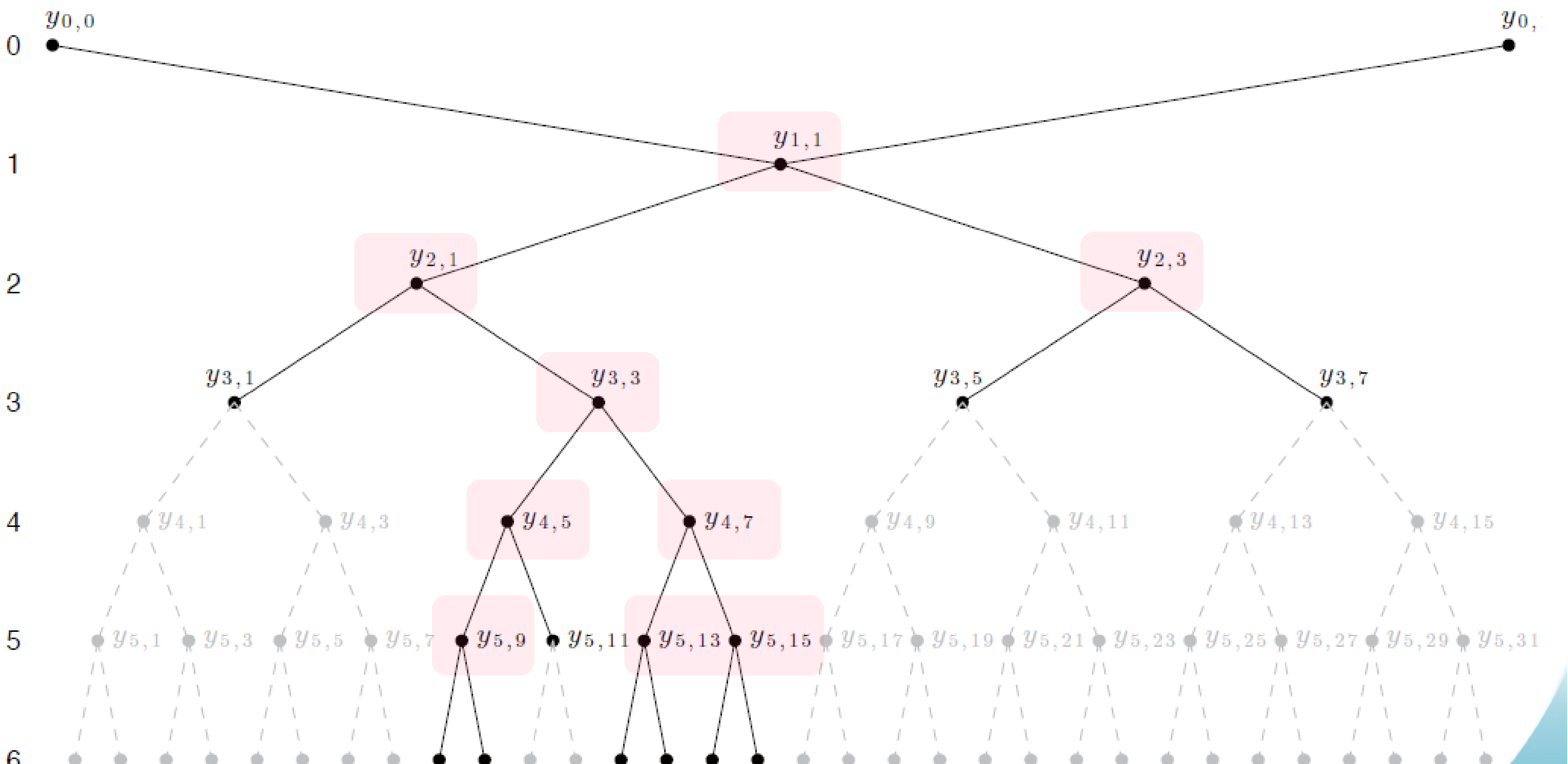
1. **evaluate** e_{jli}

   IF $\max_j |e_{jli}| \geq \tau$

2. **refine** the grid around y_{li} adding the two neighbor points

GENERALIZED SPARSE GRIDS

Adaptive grid generation (1D)



$\max_j |e_{jli}| \geq \tau$

GENERALIZED SPARSE GRIDS

Adaptive grid generation (1D)

e_{jli}: error estimate at $(\boldsymbol{x}_j, y_{li})$=($j$-th DOF, i-th sample at level l)

τ: user-defined error tolerance

Algorithm

at each successive interpolation level

1. **evaluate** e_{jli}

   IF $\max_j |e_{jli}| \geq \tau$

2. **refine** the grid around y_{li} adding the two neighbor points

N-dimensional case: same procedure! (keeping in mind that every point in the sample space has N neighbors)

GENERALIZED SPARSE GRIDS

PROS:

- complete decoupling of spatial and probabilistic discretizations
- very easy to implement (codes for PDEs used as black boxes)
- embarassingly parallelizable

GENERALIZED SPARSE GRIDS

PROS:

- complete decoupling of spatial and probabilistic discretizations
- very easy to implement (codes for PDEs used as black boxes)
- embarassingly parallelizable

CONS:

- perform well **only** when $u(\boldsymbol{x}, \boldsymbol{y})$ is smooth wrt $\boldsymbol{y}$
- fail to approximate solutions that have an irregular dependence

GENERALIZED SPARSE GRIDS

PROS:

- complete decoupling of spatial and probabilistic discretizations
- very easy to implement (codes for PDEs used as black boxes)
- embarassingly parallelizable

CONS:

- perform well **only** when $u(\boldsymbol{x}, \boldsymbol{y})$ is smooth wrt $\boldsymbol{y}$
- fail to approximate solutions that have an irregular dependence

➡ **LOCAL SC** methods:

the basis functions are **locally supported** piecewise polynomials

$\{\psi_m\}_{m=1}^{M}$ is a piecewise hierarchical polynomial basis

Sandia National Laboratories

Numerical solution via ENSAMBLES

E. Phipps, MD, H.C. Edwards, M. Hoemmen, J. Hu, S. Rajamanickam, *Embedded Ensemble Propagation for Improving Performance, Portability and Scalability of Uncertainty Quantification on Emerging Computational Architectures*, SIAM Journal on Sci Comp, 2017

A NEW STRATEGY

a few considerations:

- **problem:** in large-scale, high-performance scientific computing, the dominant cost is solving the PDE at each interpolation point

  the cost of each sample evaluation can be so large that stochastic collocation for a little more than a handful of random variables y_n is intractable!

A NEW STRATEGY

a few considerations:

- **problem:** in large-scale, high-performance scientific computing, the dominant cost is solving the PDE at each interpolation point

  the cost of each sample evaluation can be so large that stochastic collocation for a little more than a handful of random variables y_n is intractable!

- **idea:** improve the performance of the method **"opening up the box"** and exploiting the structure within each PDE evaluation.

⇒ **EMBEDDED ENSEMBLE PROPAGATION**

A NEW STRATEGY

➡ **EMBEDDED ENSEMBLE PROPAGATION**

note: in scientific simulations there is a huge amount of data and computation that is the same for each realization of the uncertain input data (e.g. the mesh)

idea: reuse this information by **propagating multiple samples** (ensembles) at a time exploiting features of modern and emerging computer architectures

ENSEMBLE PROPAGATION in finite element simulations

Finite element discretization

- continuous problem $\mathcal{L}(a)u = f$
- discretization: for $\boldsymbol{y}_m$, $m = 1, \dots M$

$$(\star) \qquad L_m \mathbf{U}_m = \mathbf{F}, \qquad L_m \in \mathbb{R}^{J \times J},\ \mathbf{U}_m \in \mathbb{R}^J,\ \mathbf{F} \in \mathbb{R}^J,$$

J: number of spatial degrees of freedom

ENSEMBLE PROPAGATION in finite element simulations

Finite element discretization

- continuous problem $\mathcal{L}(a)u = f$
- discretization: for $\boldsymbol{y}_m$, $m = 1, \ldots M$

$$(\star) \qquad L_m \mathbf{U}_m = \mathbf{F}, \qquad L_m \in \mathbb{R}^{J\times J},\ \mathbf{U}_m \in \mathbb{R}^J,\ \mathbf{F} \in \mathbb{R}^J,$$

J: number of spatial degrees of freedom

- given an ensemble size S, solve $(\star)$ for S samples $\boldsymbol{y}_{m_1} \ldots \boldsymbol{y}_{m_S}$:

$$L_{m_1}\mathbf{U}_{m_1} = \mathbf{F},\ \ \ldots\ \ L_{m_S}\mathbf{U}_{m_S} = \mathbf{F}$$

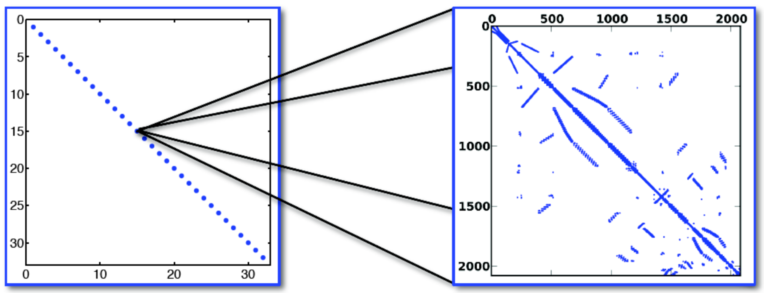
or equivalently

$$\left(\sum_{i=1}^{S} e_i e_i^T \otimes L_{m_i}\right)\left(\sum_{i=1}^{S} e_i \otimes \mathbf{U}_{m_i}\right) = \sum_{i=1}^{S} e_i \otimes \mathbf{F}$$

e_i: i^{th} column of the $S \times S$ identity matrix

A *mathematically* equivalent formulation

$$\left(\sum_{i=1}^{S} e_i e_i^T \otimes L_{m_i}\right)\left(\sum_{i=1}^{S} e_i \otimes \mathbf{U}_{m_i}\right) = \sum_{i=1}^{S} e_i \otimes \mathbf{F}$$

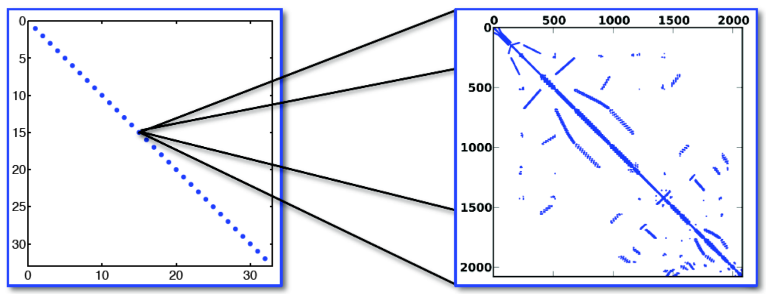


all spatial DOF for a given sample $\boldsymbol{y}_{m_i}$ are ordered consecutively

ENSEMBLE PROPAGATION in finite element simulations

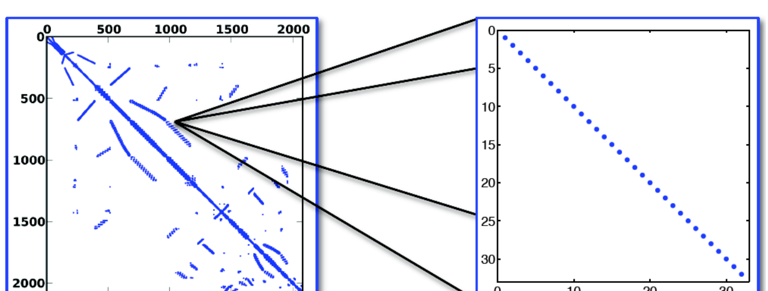
A *mathematically* equivalent formulation

$$\left(\sum_{i=1}^{S} e_i e_i^T \otimes L_{m_i}\right)\left(\sum_{i=1}^{S} e_i \otimes \mathbf{U}_{m_i}\right) = \sum_{i=1}^{S} e_i \otimes \mathbf{F}$$



all spatial DOF for a given sample $\boldsymbol{y}_{m_i}$ are ordered consecutively

$$\left(\sum_{i=1}^{S} L_{m_i} \otimes e_i e_i^T\right)\left(\sum_{i=1}^{S} \mathbf{U}_{m_i} \otimes e_i\right) = \sum_{i=1}^{S} \mathbf{F} \otimes e_i$$



DOF for all samples are ordered consecutively for a given spatial DOF

ENSEMBLE PROPAGATION in finite element simulations

Advantage: the new formulation can be solved efficiently by **replacing** each sample-dependent quantity with a length-S array

ENSEMBLE PROPAGATION in finite element simulations

Advantage: the new formulation can be solved efficiently by **replacing** each sample-dependent quantity with a length-S array

Consequences:

- Sample independent quantities automatically reused

  (e.g. mesh, matrix graph, etc) $\Rightarrow$ **reduction** of

  computation, memory usage, memory traffic

  because they are

  computed, stored, loaded once per ensemble

ENSEMBLE PROPAGATION in finite element simulations

Advantage: the new formulation can be solved efficiently by **replacing** each sample-dependent quantity with a length-S array

Consequences:

- Sample independent quantities automatically reused (e.g. mesh, matrix graph, etc) $\Rightarrow$ **reduction** of

  computation, memory usage, memory traffic

  because they are

  computed, stored, loaded once per ensemble

- Random memory accesses of sample-dependent quantities replaced by contiguous accesses of ensemble arrays.

  Example: this effect, combined with reuse of the sparse matrix graph can result in 50% reduction in cost of matrix-vector products

ENSEMBLE PROPAGATION in finite element simulations

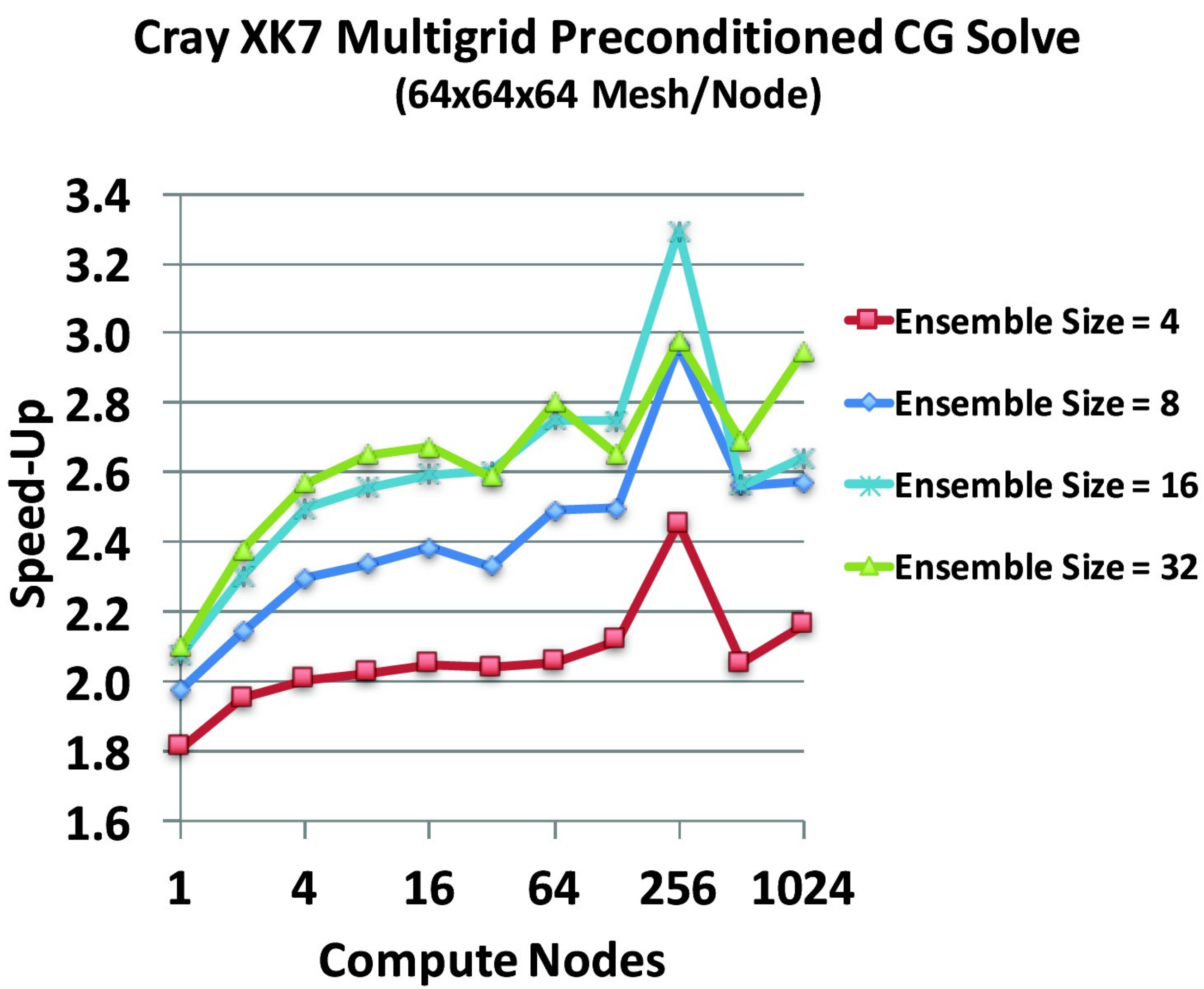
Consequences:

- arithmetic on ensemble arrays can be naturally mapped to fine-grained vector parallelism (present in most computer architectures today)

- # of distributed memory communication steps of sample-dependent information: reduced by a factor of S
- size of each communication message: increased by a factor of S

ENSEMBLE PROPAGATION: performance results

Results for ISOTROPIC diffusion: SPEED-UP for different ensemble size S



ENSEMBLE PROPAGATION in finite element simulations

Note: in the previous tests the number of CG iterations is **independent of the sample value**

$\Rightarrow$ the number of CG iterations for each ensemble is **independent of the choice of samples grouped together**

ENSEMBLE PROPAGATION in finite element simulations

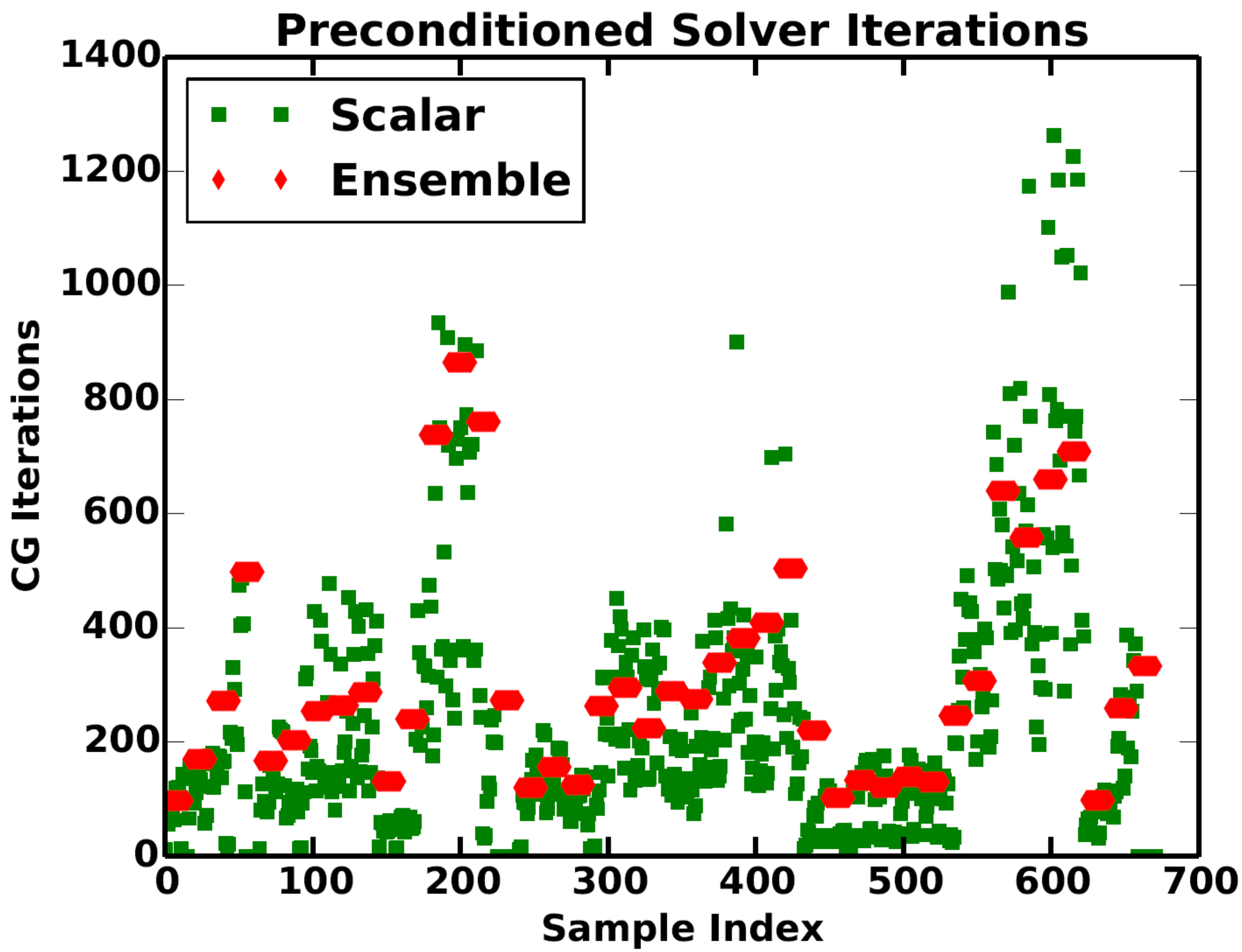
Note: in the previous tests the number of CG iterations is **independent of the sample value**

$\Rightarrow$ the number of CG iterations for each ensemble is **independent of the choice of samples grouped together**

This is unlikely for more realistic problems: **the way samples are grouped into ensembles has a strong effect on the performance**

ENSEMBLE PROPAGATION in finite element simulations

This is unlikely for more realistic problems, e.g. **anisotropic** diffusion



ENSEMBLE PROPAGATION in finite element simulations

Note: in the previous tests the number of CG iterations is **independent of the sample value**

$\Rightarrow$ the number of CG iterations for each ensemble is **independent of the choice of samples grouped together**

This is unlikely for more realistic problems: **the way samples are grouped into ensembles has a strong effect on the performance**

$\Rightarrow$ it is **necessary** to develop **grouping strategies** to maximize the performace improvement brought by the ensemble propagation

Grouping strategies

MD, H.C. Edwards, J. Hu, E. Phipps, S. Rajamanickam, *Ensemble Grouping Strategies for Embedded Stochastic Collocation Methods Applied to Anisotropic Diffusion Problems*, submitted, 2016

SOME CONSIDERATIONS

Facts: • the convergence of the linear solver is almost always affected by the spectral properties of the matrices L_m

• spectra of FE matrices are strongly related to quantities such as
– **condition number**
– **spatial variations of the parameters** (total variation, magnitude of the gradient, strength of the anisotropy, etc.)

SOME CONSIDERATIONS

Facts: • the convergence of the linear solver is almost always affected by the spectral properties of the matrices L_m

• spectra of FE matrices are strongly related to quantities such as
– **condition number**
– **spatial variations of the parameters** (total variation, magnitude of the gradient, strength of the anisotropy, etc.)

• **different quantities affect different solvers:**
– CG is strongly affected by the condition number
– stretched and irregular grids affect the behavior of AMG

SOME CONSIDERATIONS

Facts: • the convergence of the linear solver is almost always affected by the spectral properties of the matrices L_m

• spectra of FE matrices are strongly related to quantities such as
– **condition number**
– **spatial variations of the parameters** (total variation, magnitude of the gradient, strength of the anisotropy, etc.)

• **different quantities affect different solvers:**
– CG is strongly affected by the condition number
– stretched and irregular grids affect the behavior of AMG

• **regardless** of the rearrangement of rows and columns, the spectra of the ensemble matrices are the **union of the spectra** of the matrices within each ensemble

SOME CONSIDERATIONS

Facts: • the convergence of the linear solver is almost always affected by the spectral properties of the matrices L_m

• spectra of FE matrices are strongly related to quantities such as
– **condition number**
– **spatial variations of the parameters** (total variation, magnitude of the gradient, strength of the anisotropy, etc.)

• **different quantities affect different solvers:**
– CG is strongly affected by the condition number
– stretched and irregular grids affect the behavior of AMG

• **regardless** of the rearrangement of rows and columns, the spectra of the ensemble matrices are the **union of the spectra** of the matrices within each ensemble

• **the convergence of the ensemble solver is in general poorer** than that of the solver applied to each sample individually

SOME CONSIDERATIONS

Question: how to **minimize** the deterioration of the convergence?

Strategy: group together samples whose FE matrices have similar spectral properties, i.e. that require a similar number of iterations

SOME CONSIDERATIONS

Question: how to **minimize** the deterioration of the convergence?

Strategy: group together samples whose FE matrices have similar spectral properties, i.e. that require a similar number of iterations

Challenge: **find indicators** for predicting which samples feature a similar convergence behavior

Highly anisotropic diffusion problems

MD, H.C. Edwards, J. Hu, E. Phipps, S. Rajamanickam, *Ensemble Grouping Strategies for Embedded Stochastic Collocation Methods Applied to Anisotropic Diffusion Problems*, submitted, 2016

MD, M. Ebeida, E. Phipps, A. Rushdi *Surrogate-based Ensemble Grouping Strategies for Embedded Stochastic Collocation Methods*, in preparation.

SOME CONSIDERATIONS

Diffusion equation:

$$\begin{cases} \mathcal{L}(a(\boldsymbol{y}))u = -\nabla \cdot (A(\cdot, \boldsymbol{y})\nabla u) = f & \boldsymbol{x} \in D, \boldsymbol{y} \in \Gamma \\ \mathcal{B}u = u = 0 & \boldsymbol{x} \in \partial D \end{cases}$$

forcing term: $f \in L^2(D)$:

diffusivity tensor: $A(\boldsymbol{x}, \cdot) = diag(a(\boldsymbol{x}, \boldsymbol{y}),\ \overline{a})$ (in 2D)

$a(\boldsymbol{x}, \boldsymbol{y})$: truncated KL approximation of a random field, i.e.

$$a(\boldsymbol{x}, \boldsymbol{y}) = a_{\min} + \widehat{a} \exp\left\{ \sum_{n=1}^{N} \sqrt{\lambda_n} b_n(\boldsymbol{x}) y_n \right\}$$

SOME CONSIDERATIONS

Facts: • the FE matrix corresponding to $A(\boldsymbol{x}, \boldsymbol{y}_m)$ is always spd

$\Rightarrow$ the discretized problem has a unique solution
$\Rightarrow$ it is suitable for an iterative solver based on CG

SOME CONSIDERATIONS

Facts: • the FE matrix corresponding to $A(\boldsymbol{x}, \boldsymbol{y}_m)$ is always spd

$\Rightarrow$ the discretized problem has a unique solution
$\Rightarrow$ it is suitable for an iterative solver based on CG

• the convergence of CG depends on the condition number
(very slow when L_m has a widespread spectrum)

$\Rightarrow$ use preconditioned CG (PCG)
AMG are often the preconditioner of choice for diffusion problems

SOME CONSIDERATIONS

Facts: • the FE matrix corresponding to $A(\boldsymbol{x}, \boldsymbol{y}_m)$ is always spd

$\Rightarrow$ the discretized problem has a unique solution
$\Rightarrow$ it is suitable for an iterative solver based on CG

• the convergence of CG depends on the condition number
(very slow when L_m has a widespread spectrum)

$\Rightarrow$ use preconditioned CG (PCG)
AMG are often the preconditioner of choice for diffusion problems

• **even for SPD matrices** a variety of issues can hamper the effectiveness of AMG algorithm, e.g.

– mesh stretching and irregular meshes
– highly anisotropic problem coefficients
– choice of discretization, etc.

GROUPING STRATEGIES

A. PARAMETER-BASED: the grouping depends on the values, in space, of the diffusion tensor in correspondence of a single sample

[1] MD, H.C. Edwards, J. Hu, E. Phipps, S. Rajamanickam,
Ensemble Grouping Strategies for Embedded Stochastic Collocation Methods Applied to Anisotropic Diffusion Problems, submitted, 2016

GROUPING STRATEGIES

A. PARAMETER-BASED: the grouping depends on the values, in space, of the diffusion tensor in correspondence of a single sample

Indicator: $I(\widetilde{\boldsymbol{y}}) = \|r(\boldsymbol{x},\widetilde{\boldsymbol{y}})\|_\infty$ where $r(\boldsymbol{x},\widetilde{\boldsymbol{y}}) = \dfrac{\lambda_{\max}(A(\boldsymbol{x},\widetilde{\boldsymbol{y}}))}{\lambda_{\min}(A(\boldsymbol{x},\widetilde{\boldsymbol{y}}))}$

r: ratio between max and min eigenvalues of the diffusion tensor
$\Rightarrow$ **intensity of the anisotropy** at each point in the spatial domain

max of r over D: measure of the anisotropy associated with $\widetilde{\boldsymbol{y}}$

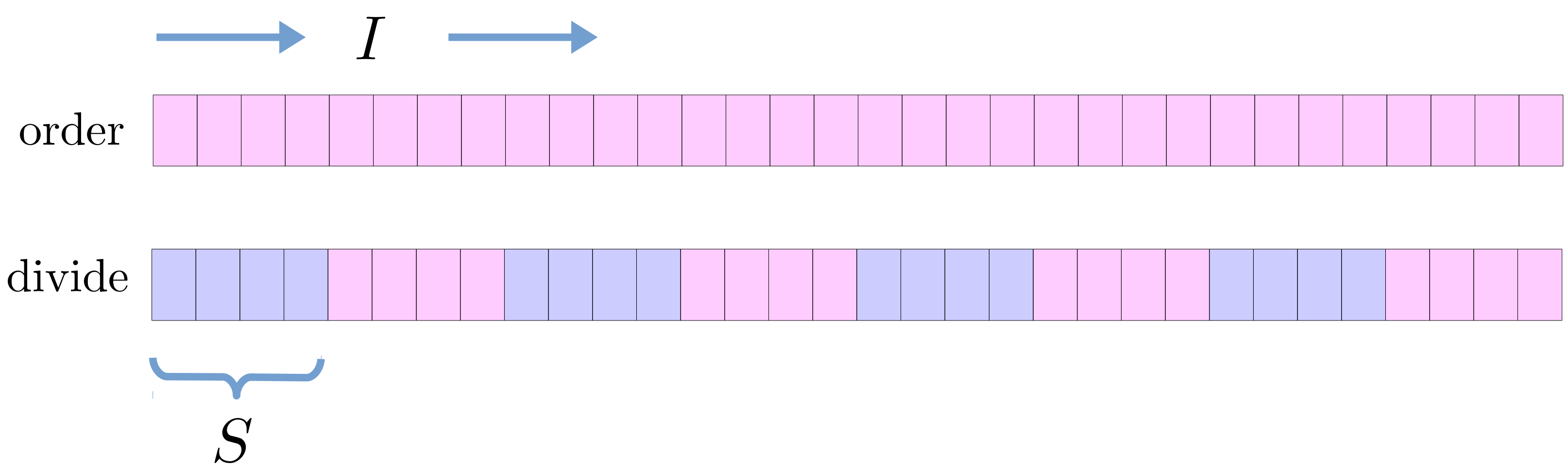
$$r(\boldsymbol{x},\widetilde{\boldsymbol{y}}) = \frac{\lambda_{\max}(A(\boldsymbol{x},\widetilde{\boldsymbol{y}}))}{\lambda_{\min}(A(\boldsymbol{x},\widetilde{\boldsymbol{y}}))} = \frac{a(\boldsymbol{x},\widetilde{\boldsymbol{y}})}{\overline{a}}$$

[1] MD, H.C. Edwards, J. Hu, E. Phipps, S. Rajamanickam,
Ensemble Grouping Strategies for Embedded Stochastic Collocation Methods Applied to Anisotropic Diffusion Problems, submitted, 2016

GROUPING STRATEGIES

Grouping: – **order** the samples according to increasing values of I

– **divide** the samples into groups of size S



GROUPING STRATEGIES

B. SURROGATE-BASED[2]: the grouping depends on a *sparse grid* surrogate for the number of iterations associated with a new sample

$G(\widetilde{\boldsymbol{y}})$: *exact* QOI

$\widehat{G}(\widetilde{\boldsymbol{y}})$: *predicted* QOI,

$\widehat{G}(\cdot)$: *surrogate* for QOI (sparse grid approximation)

$I(\widetilde{\boldsymbol{y}})$: *exact* #its

$\widehat{I}(\widetilde{\boldsymbol{y}})$: *predicted* #its

$\widehat{I}(\cdot)$: *surrogate* for #its

[2] MD, M. Ebeida, E. Phipps, A. Rushdi *Surrogate-based Ensemble Grouping Strategies for Embedded Stochastic Collocation Methods*, in preparation.

GROUPING STRATEGIES

B. SURROGATE-BASED[2]: the grouping depends on a *sparse grid* surrogate for the number of iterations associated with a new sample

$G(\tilde{\boldsymbol{y}})$: *exact* QOI — can be computed for the **current** and **past** levels

$\widehat{G}(\tilde{\boldsymbol{y}})$: *predicted* QOI,

$\widehat{G}(\cdot)$: *surrogate* for QOI (sparse grid approximation)

$I(\tilde{\boldsymbol{y}})$: *exact* #its — can be computed for the **current** and **past** levels

$\widehat{I}(\tilde{\boldsymbol{y}})$: *predicted* #its

$\widehat{I}(\cdot)$: *surrogate* for #its

[2] MD, M. Ebeida, E. Phipps, A. Rushdi *Surrogate-based Ensemble Grouping Strategies for Embedded Stochastic Collocation Methods*, in preparation.

GROUPING STRATEGIES

B. SURROGATE-BASED[2]: the grouping depends on a *sparse grid* surrogate for the number of iterations associated with a new sample

$G(\tilde{\boldsymbol{y}})$: *exact* QOI

can be computed for the **current** and **past** levels

$\widehat{G}(\tilde{\boldsymbol{y}})$: *predicted* QOI,

$\widehat{G}(\cdot)$: *surrogate* for QOI (sparse grid approximation)

$I(\tilde{\boldsymbol{y}})$: *exact* #its

can be computed for the **current** and **past** levels

$\widehat{I}(\tilde{\boldsymbol{y}})$: *predicted* #its

$\widehat{I}(\cdot)$: *surrogate* for #its

Advantage: does not require a lot of computational effort and **does not assume any knowledge** of the parameters or of the SPDE itself

[2] MD, M. Ebeida, E. Phipps, A. Rushdi *Surrogate-based Ensemble Grouping Strategies for Embedded Stochastic Collocation Methods*, in preparation.

B. SURROGATE-BASED

Algorithm

given $N_{\max}$ (sample budget), S (ensemble size) and τ (error tolerance)

A. **generate** $\mathcal{Y}_0$ (initial sample set)

B. **group** the samples in the order they are generated

C. **iterate** until we reach the budget or satisfy the tolerance

GROUPING STRATEGIES

B. SURROGATE-BASED

Algorithm

given $N_{\max}$ (sample budget), S (ensemble size) and τ (error tolerance)

A. **generate** $\mathcal{Y}_0$ (initial sample set)

B. **group** the samples in the order they are generated

C. **iterate** until we reach the budget or satisfy the tolerance

1. **solve** the PDEs and evaluate $G(\boldsymbol{y}_i)$ and $I(\boldsymbol{y}_i)$ $\forall \boldsymbol{y}_i \in \mathcal{Y}_l$ (current sample set)
2. **build the surrogates** $\widehat{G}$ and $\widehat{I}$
3. **determine** a candidate sample set for grid refinement

   IF *the conditions of the stopping criterion are not satisfied, use* $\widehat{G}$ *and* $\widehat{I}$ *to*

4. **select** $\mathcal{Y}_{l+1}$ (new sample set)
5. **group** the samples in ensembles

Numerical tests

MD, H.C. Edwards, J. Hu, E. Phipps, S. Rajamanickam, *Ensemble Grouping Strategies for Embedded Stochastic Collocation Methods Applied to Anisotropic Diffusion Problems*, submitted, 2016

MD, M. Ebeida, E. Phipps, A. Rushdi *Surrogate-based Ensemble Grouping Strategies for Embedded Stochastic Collocation Methods*, in preparation.

PROBLEM SETTING

Domains: $D = [0,\, 1]^3 \times \Gamma = [-1,\, 1]^6$

Covariance function: exponential: $cov(\boldsymbol{x}, \boldsymbol{x}') = \sigma_0 \exp\left\{-\frac{\|\boldsymbol{x} - \boldsymbol{x}'\|}{\delta}\right\};$

PROBLEM SETTING

Quantity of Interest: $\|\mathbf{u}\|_2$

Sparse Grid generation:

- **technique:** adaptive refinement, local piecewise linear basis
- **software:** TASMANIAN `http://tasmanian.ornl.gov`, by M. Stoyanov
  robust libraries for high dimensional integration and interpolation

PROBLEM SETTING

Quantity of Interest: $\|\mathbf{u}\|_2$

Sparse Grid generation:

– **technique:** adaptive refinement, local piecewise linear basis

– **software:** TASMANIAN `http://tasmanian.ornl.gov`, by M. Stoyanov
robust libraries for high dimensional integration and interpolation

Solver:

– **FE assembling:** Intrelab, Matlab interface of the Trilinos package Intrepid

– **FE linear solver:** ML (Matlab interface), AMG preconditioned CG

PROBLEM SETTING

Indicators of computational savings

$$R = \frac{S \sum_{k=1}^{K} \mathbf{I}_k}{\sum_{k=1}^{K} \sum_{i=1}^{S} I(\boldsymbol{y}_{k,i})}$$

total increase in work over all levels

$\mathbf{I}_k$: # its for the kth

$I(\boldsymbol{y}_{k,i})$: # its for ith sample in the kth ensemble

PROBLEM SETTING

Indicators of computational savings

$$R = \frac{S \sum_{k=1}^{K} \mathbf{I}_k}{\sum_{k=1}^{K} \sum_{i=1}^{S} I(\boldsymbol{y}_{k,i})}$$ total increase in work over all levels

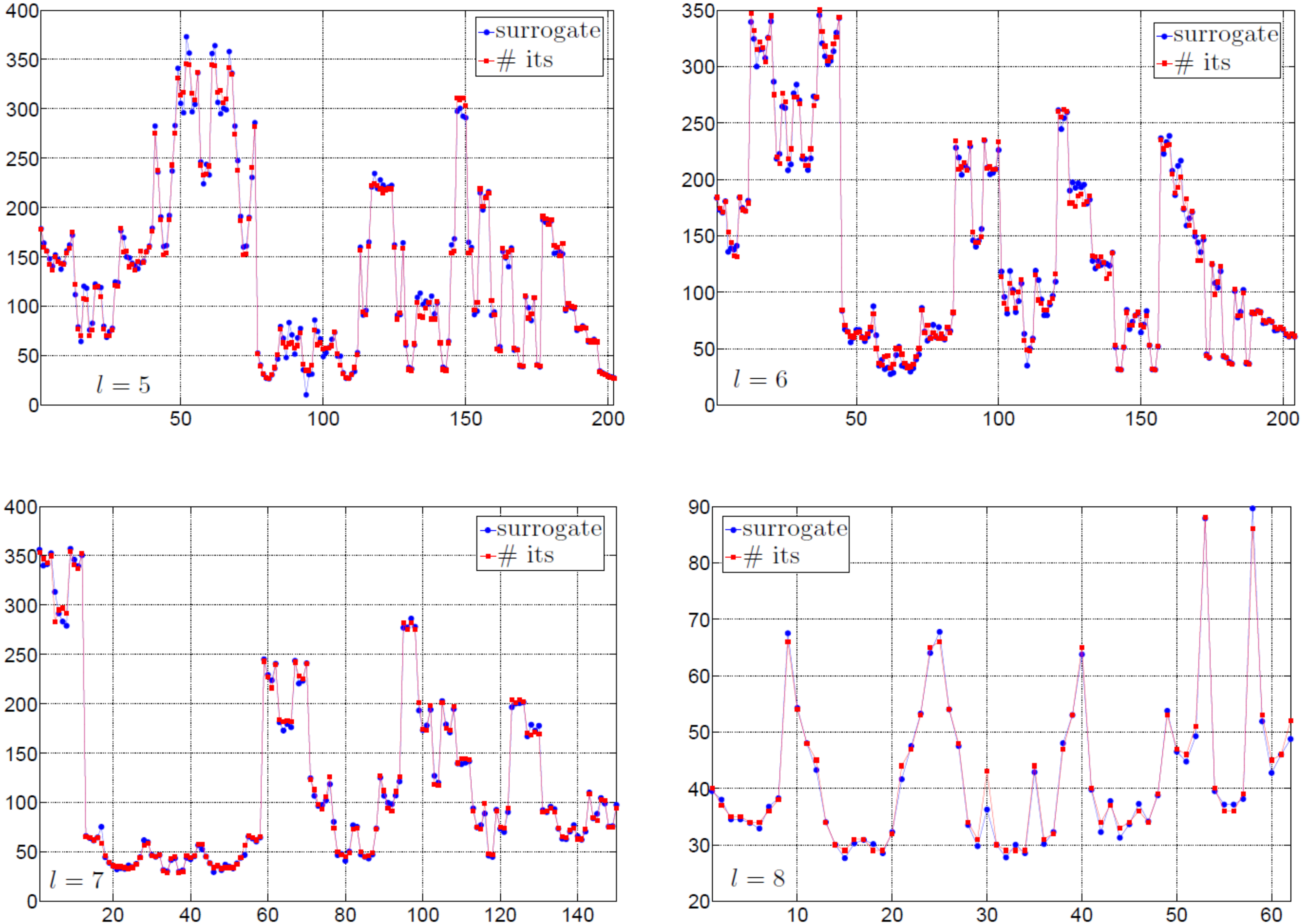
$\mathbf{I}_k$: # its for the kth

$I(\boldsymbol{y}_{k,i})$: # its for ith sample in the kth ensemble

achieved speed-up:

$$\frac{\text{speed-up}(\text{ensemble prop})}{R}$$

SQUARED EXPONENTIAL COVARIANCE



SQUARED EXPONENTIAL COVARIANCE – *R*

N	S	parameter	exact its	surrogate its	No ordering
3	8	1.445	1.447	1.543	1.791
3	16	1.580	1.589	1.691	2.146
3	32	1.895	1.912	2.044	2.806

N	S	parameter	exact its	surrogate its	No ordering
6	8	1.991	2.012	2.185	2.630
6	16	2.230	2.198	2.421	3.071
6	32	2.403	2.433	2.780	3.604

initial level in the adaptive grid generation: $l = 4$

SQUARED EXPONENTIAL COVARIANCE – *R*

N	S	parameter	exact its	surrogate its	No ordering
3	8	1.445	1.447	1.543	1.791
3	16	1.580	1.589	1.691	2.146
3	32	1.895	1.912	2.044	2.806

N	S	parameter	exact its	surrogate its	No ordering
6	8	1.991	2.012	2.185	2.630
6	16	2.230	2.198	2.421	3.071
6	32	2.403	2.433	2.780	3.604

initial level in the adaptive grid generation: $l = 4$

note: at the initial level no ordering performed

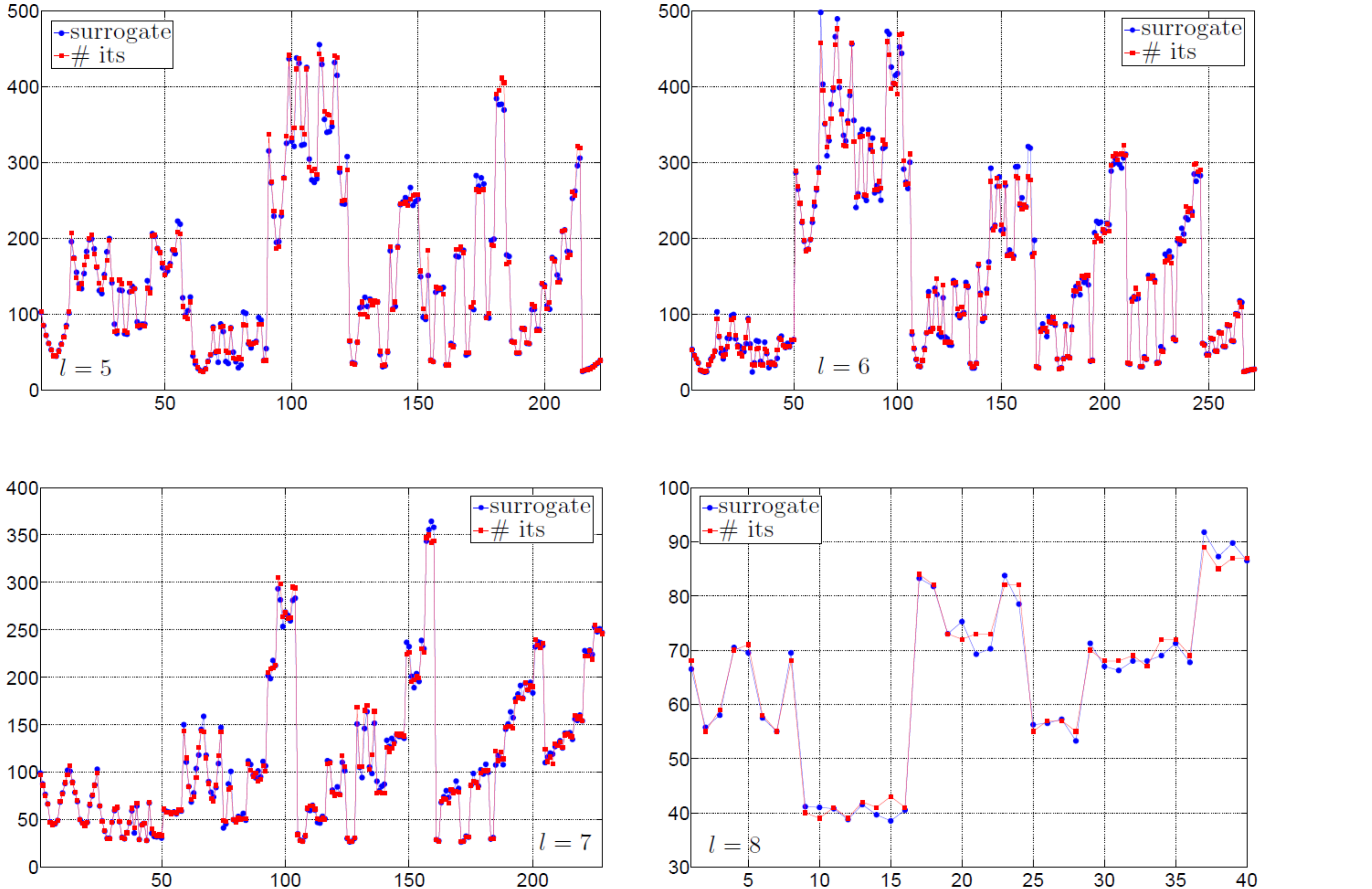
➡ loss of efficiency ➡ **idea:** start from level $l = 1$

SQUARED EXPONENTIAL COVARIANCE – *R*

...idea: start from level $l = 1$

N	S	exact its	surrogate its	No ordering
3	8	1.509	1.521	1.813
3	16	1.622	1.640	2.177
3	32	2.014	2.019	2.809

RATIONAL QUADRATIC COVARIANCE



RATIONAL QUADRATIC COVARIANCE – *R*

N	S	parameter	exact its	surrogate its	No ordering
3	8	1.348	1.350	1.485	1.752
3	16	1.436	1.473	1.671	2.203
3	32	1.721	1.738	1.966	2.734

N	S	parameter	exact its	surrogate its	No ordering
6	8	1.827	1.847	2.011	2.366
6	16	1.891	1.990	2.230	2.762
6	32	2.200	2.177	2.483	3.173

initial level in the adaptive grid generation: $l = 4$

Future work

LOOKING FOR(WARD TO) NEW INDICATORS

new idea: points in the sparse grid can be represented in a tree structure

➡ we expect children of the same parent to generate similar uncertain parameters

➡ keep track of the family history and group together samples with the same ancestors

Thank you

Sandia
National
Laboratories

Hierarchical basis

From M. Gunzburger, C. Webster, G. Zhang. *Stochastic finite element methods for partial differential equations with random input data.* Acta Numerica (2014), pp. 521650, 2014

Piecewise linear hierarchical basis

Piecewise linear nodal basis