Ensemble Grouping strategies for embedded Stochastic Collocation methods applied to anisotropic diffusion problems

SAND2016-3524C

Marta D'Elia, H.C. Edwards, J. Hu, E. Phipps, S. Rajamanickam

CCR
Center for Computing Research

March 31st 2016

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

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
- multilevel methods exploiting a hierarchy of physical and temporal discretizations
- methods that construct minimal or optimal uncertainty representation (compressed sensing, tensor methods)

INTRODUCTION

Unfortunately this is not enough

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

INTRODUCTION

Unfortunately this is 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*, submitted, 2015

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

- Introduction to Stochastic Collocation methods
- Summary of Embedded Ensemble Propagation
- Grouping strategies
- Case study: a highly anisotropic diffusion problem
- Numerical tests
- Current and 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

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

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

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

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

PROBLEM SETTING

Random parameter satisfying B and C:

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

Mercers'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

Mercers'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 $G_u(\boldsymbol{y})$

Examples:

- $G_u(\boldsymbol{y}) = \dfrac{1}{|D|} \displaystyle\int_D u(\boldsymbol{x}, \boldsymbol{y})\, d\boldsymbol{x}$
- $G_u(\boldsymbol{y}) = \max_{\boldsymbol{x} \in D} u(\boldsymbol{x}, \boldsymbol{y})$

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

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

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 $u_{h,M}(\boldsymbol{x}, \boldsymbol{y})$
- polynomial: interpolatory or based on a projection onto an orthonormal basis

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

Fully discrete approximation

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

the coefficients are determined by solving

$$\sum_{m=1}^{M} c_m(\boldsymbol{x})\psi_m(\boldsymbol{y}_{m'}) = u_h(\boldsymbol{x},\boldsymbol{y}_{m'}) \quad \forall\, m' = 1,\dots M$$

for **Lagrange** interpolation $c_m(\boldsymbol{x}) = u_h(\boldsymbol{x},\boldsymbol{y}_m) \quad \Rightarrow$

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

GENERALIZED SPARSE GRIDS

One-dimensional approximation

For $n = 1, \dots N$

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

1D interpolation operator: for $v \in C^0(\Gamma_n)$

$$\mathcal{I}_n^{m(l_n)}[v](y) = \sum_{k=1}^{m(l_n)} v(y_{n,k}^{l_n}) \psi_{n,k}^{l_n}(y), \quad l_n = 1, 2, \dots$$

$\psi_{n,k}^{l_n} \in \mathcal{P}_{m(l_n)-1}(\Gamma_n)$: Lagrange fundamental polynomials of degree $p_{l_n} = m(l_n) - 1$

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

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

1D difference operator: $\Delta_n^{m(l_n)} = \mathcal{I}_n^{m(l_n)} - \mathcal{I}_n^{m(l_n)-1} \quad \text{with} \quad \mathcal{I}_n^{m(0)} = 0$

N-dimensional difference operator: $\Delta^m = \bigotimes_{n=1}^{N} \Delta_n^{m(l_n)}$

(tensor product of the 1D operators)

L-th level generalized sparse grid operator: $\mathcal{J}_L^{m,g} = \sum_{\|\mathbf{l}\| \leq L} \Delta^m$

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

1D difference operator: $\Delta_n^{m(l_n)} = \mathcal{I}_n^{m(l_n)} - \mathcal{I}_n^{m(l_n)-1} \quad \text{with} \quad \mathcal{I}_n^{m(0)} = 0$

N-dimensional difference operator: $\Delta^m = \bigotimes_{n=1}^{N} \Delta_n^{m(l_n)}$

(tensor product of the 1D operators)

L-th level generalized sparse grid operator: $\mathscr{J}_L^{m,g} = \sum_{\|\mathbf{l}\| \leq L} \Delta^m$

Generalized sparse-grid approximation: $u_{h,L} = \mathscr{J}_L^{m,g}[u_h]$

GENERALIZED SPARSE GRIDS

PROS:

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

GENERALIZED SPARSE GRIDS

PROS:

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

CONS:

- perform well **only** when $u(\boldsymbol{x}, \boldsymbol{y})$ is extremely smooth wrt $\{y_n\}_{n=1}^N$
- fail to approximate solutions that have an irregular dependence

GENERALIZED SPARSE GRIDS

PROS:

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

CONS:

- perform well **only** when $u(\boldsymbol{x}, \boldsymbol{y})$ is extremely smooth wrt $\{y_n\}_{n=1}^N$
- 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

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*, submitted, 2015

A NEW STRATEGY

a few considerations:

- **advantage** of stochastic collocation methods: **fully nonintrusive**, i.e., they can be applied to a simulation code that numerically solves the SPDE with little or no modification.

- **problem:** in large-scale, high-performance scientific computing, the dominant cost (by far!) in collocation methods is solving the PDE system at each interpolation point

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

A NEW STRATEGY

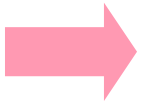
a few considerations:

- **advantage** of stochastic collocation methods: **fully nonintrusive**, i.e., they can be applied to a simulation code that numerically solves the SPDE with little or no modification.

- **problem:** in large-scale, high-performance scientific computing, the dominant cost (by far!) in collocation methods is solving the PDE system at each interpolation point

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

- **idea:** improve the performance of the method **"opening up the box"** and exploiting further 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 $\mathcal{L}(a)u = f$, where $\mathcal{L}$ is a linear elliptic operator
- discretization: for $\boldsymbol{y}_m$, $m = 1, \dots M_L$

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

J: number of spatial degrees of freedom

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

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

ENSEMBLE PROPAGATION in finite element simulations

Finite element discretization

- continuous problem $\mathcal{L}(a)u = f$, where $\mathcal{L}$ is a linear elliptic operator
- discretization: for $\boldsymbol{y}_m$, $m = 1, \dots M_L$

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

J: number of spatial degrees of freedom

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

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

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

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 the computation by only computing once per ensemble
  – reduction of memory usage by only storing them once per ensemble
  – reduction of memory traffic by only loading them once per ensemble

- Random memory accesses of sample-dependent quantities are 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

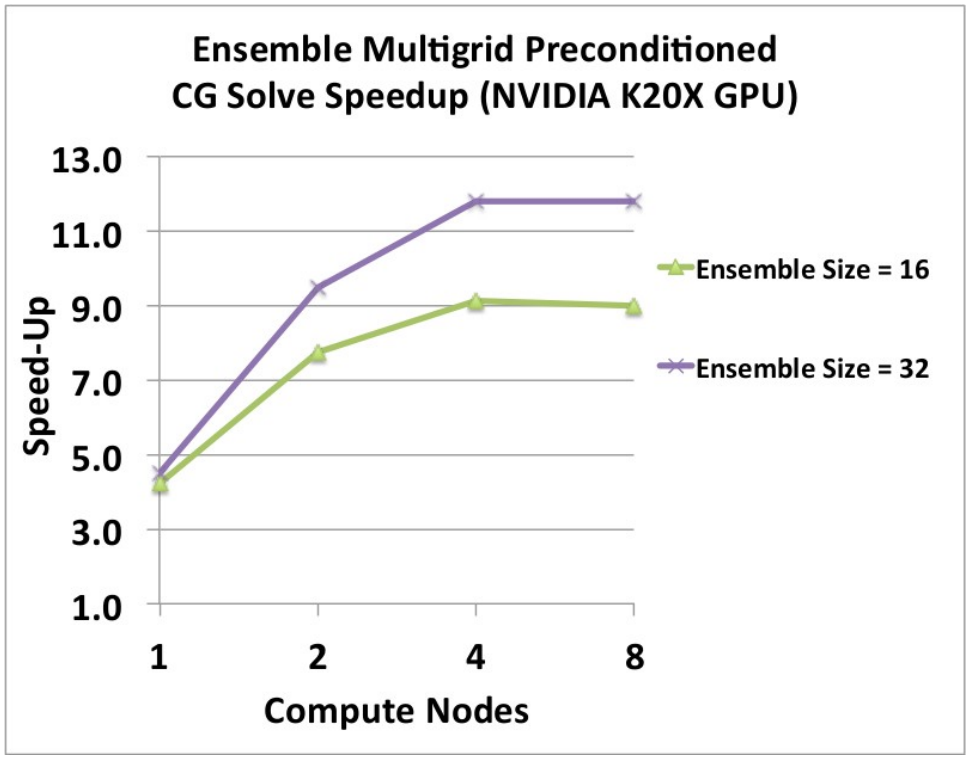
Consequences:

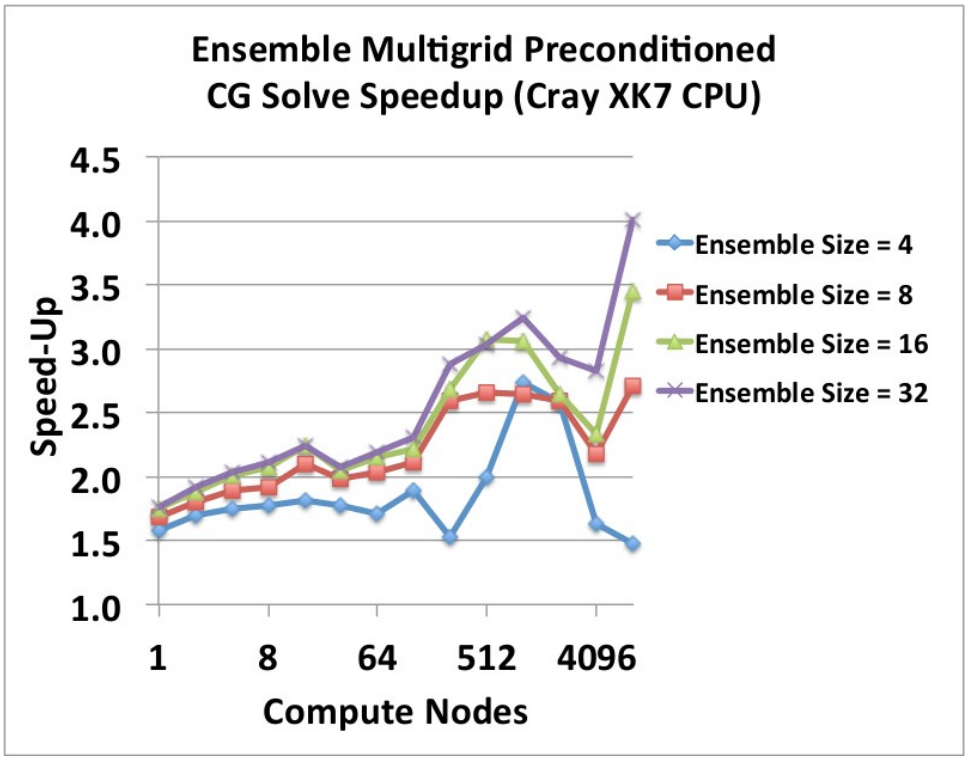
- Arithmetic on ensemble arrays can be naturally mapped to fine-grained vector parallelism present in most computer architectures today

- The number of distributed memory communication steps of sample-dependent information is reduced by a factor of S, with the size of each communication message increased by a factor of S

ENSEMBLE PROPAGATION: performance results

Results: SPEED-UP for the diffusion problem on GPU and CPU architectures for different ensemble size S





– Algebraic multigrid preconditioned CG solve
– Fixed finite element mesh: $64 \times 64 \times 64$ cells/compute node
  - 8 compute nodes on the GPU cluster
  - 4096 compute nodes with 16 processors per node on the Cray

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

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 always 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

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

GROUPING STRATEGIES

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

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

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

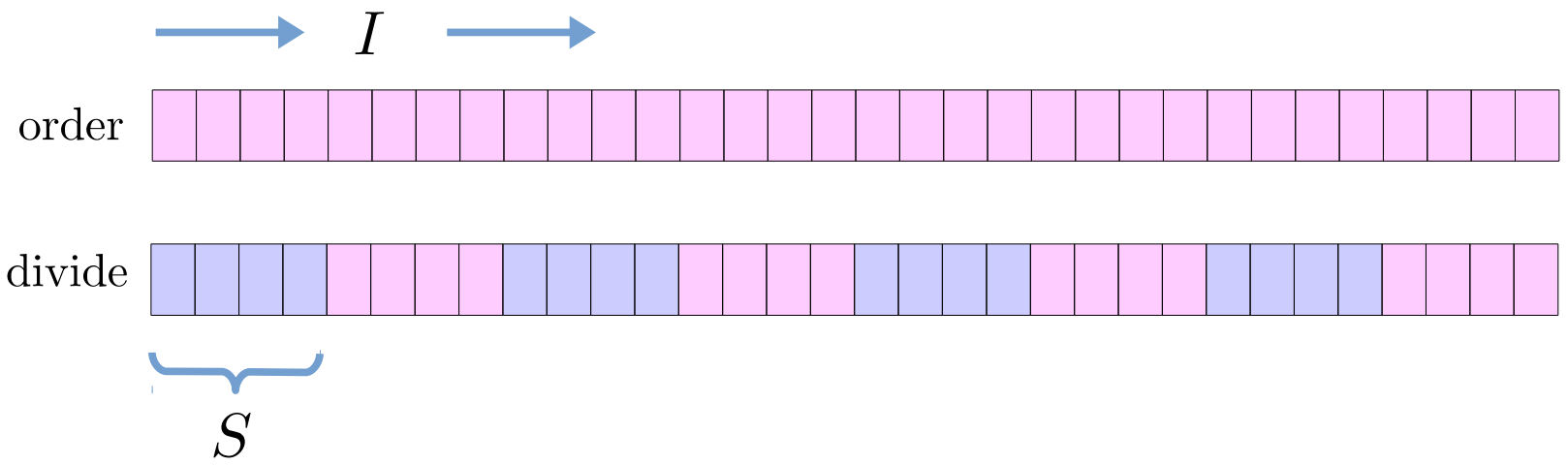
max of r over D: measure of the anisotropy associated with $\widetilde{\boldsymbol{y}}$

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

GROUPING STRATEGIES

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

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



GROUPING STRATEGIES

B. KL-BASED: the grouping depends on the effect that each random variable has on the uncertain parameter

Indicator: • when $\Gamma \subset \mathbb{R}^3$, $\widehat{I}(\widetilde{\boldsymbol{y}}) = \pm\,\widetilde{y}_1 + |\widetilde{y}_2| + |\widetilde{y}_3|$

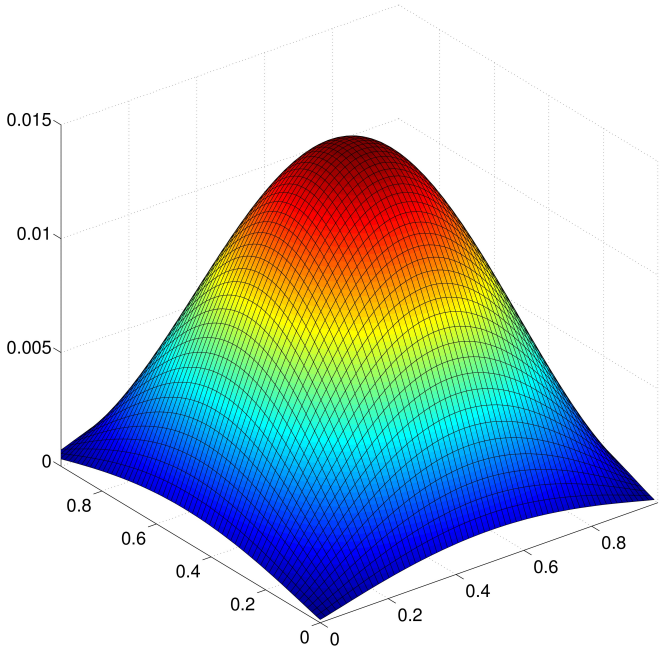
$$\text{recall that } \mathrm{a}(\boldsymbol{x},\widetilde{\boldsymbol{y}}) = a_{\min} + \widehat{a}\exp\left\{\sum_{n=1}^{N}\sqrt{\lambda_n}b_n(\boldsymbol{x})\widetilde{y}_n\right\}$$

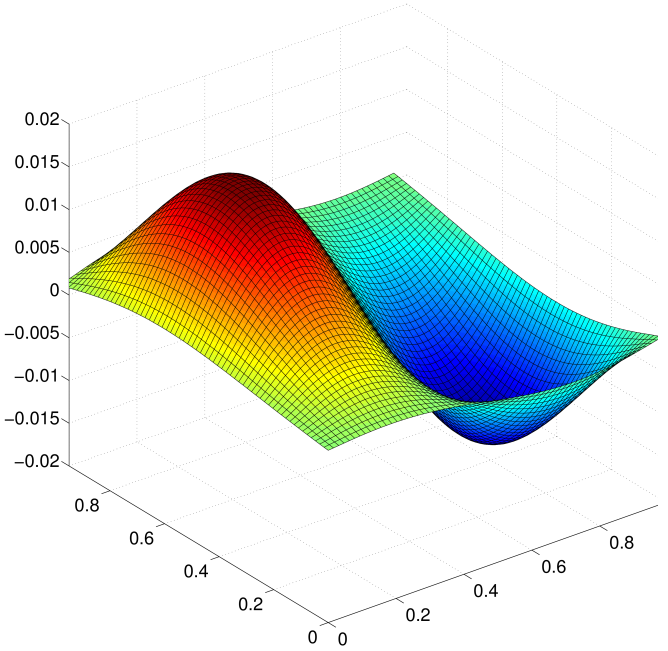
GROUPING STRATEGIES

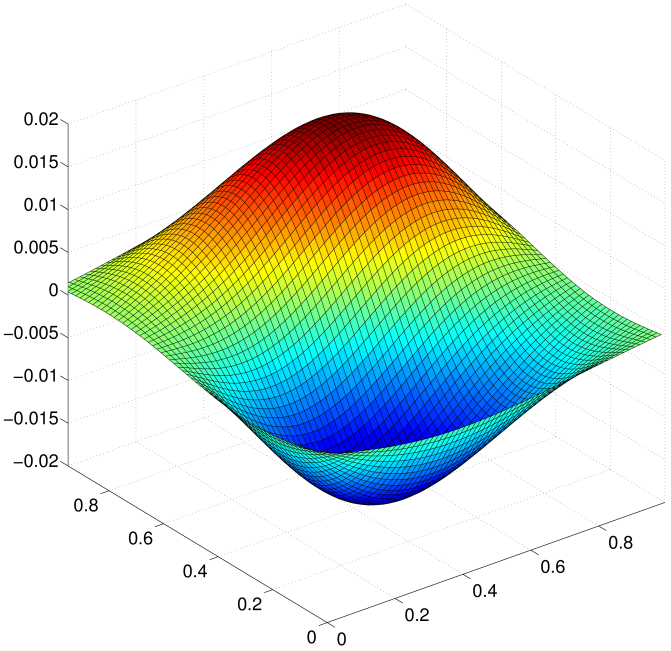
B. KL-BASED: the grouping depends on the effect that each random variable has on the uncertain parameter

Indicator: • when $\Gamma \subset \mathbb{R}^3$, $\widehat{I}(\widetilde{\boldsymbol{y}}) = \pm\,\widetilde{y}_1 + |\widetilde{y}_2| + |\widetilde{y}_3|$

$$\text{recall that } \mathrm{a}(\boldsymbol{x},\widetilde{\boldsymbol{y}}) = a_{\min} + \widehat{a}\exp\left\{\sum_{n=1}^{N}\sqrt{\lambda_n}b_n(\boldsymbol{x})\widetilde{y}_n\right\}$$







GROUPING STRATEGIES

B. KL-BASED: the grouping depends on the effect that each random variable has on the uncertain parameter

Indicator:

- when $\Gamma \subset \mathbb{R}^3$, $\widehat{I}(\widetilde{\boldsymbol{y}}) = \pm\,\widetilde{y}_1 + |\widetilde{y}_2| + |\widetilde{y}_3|$

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

- when $\Gamma \subset \mathbb{R}^N$, $N > 3$ $\quad \widehat{I}(\widetilde{\boldsymbol{y}}) = \pm\,\widetilde{y}_1 + \sum_{n=2}^{N}|\widetilde{y}_n|$

GROUPING STRATEGIES

B. KL-BASED: the grouping depends on the effect that each random variable has on the uncertain parameter

Indicator:

- when $\Gamma \subset \mathbb{R}^3$, $\widehat{I}(\widetilde{\boldsymbol{y}}) = \pm\,\widetilde{y}_1 + |\widetilde{y}_2| + |\widetilde{y}_3|$

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

- when $\Gamma \subset \mathbb{R}^N$, $N > 3$ $\quad \widehat{I}(\widetilde{\boldsymbol{y}}) = \pm\,\widetilde{y}_1 + \sum_{n=2}^{N}|\widetilde{y}_n|$

Advantage: $\widehat{I}$ requires **less computational effort** and **does not assume any knowledge** of the parameters or of the SPDE itself

GROUPING STRATEGIES

note: the most natural approach without using any notion of the SPDE

➡ **clustering** in the sample space based on the geometric location

even with sparse grids?

GROUPING STRATEGIES

note: the most natural approach without using any notion of the SPDE
➡ **clustering** in the sample space based on the geometric location

even with sparse grids?

C. HSFC-BASED: grouping based on the partition of the sample space given by the Hilbert space-filling curve (HSFC)

idea: the HSFC establishes a mapping between one-dimensional and two-/three-dimensional spaces: partition in 1D ➡ partition in the space

boxes in the partition are ordered ➡ samples are ordered on the basis of the box they belong to

GROUPING STRATEGIES

note: the most natural approach without using any notion of the SPDE
➡ **clustering** in the sample space based on the geometric location

even with sparse grids?

C. HSFC-BASED: grouping based on the partition of the sample space given by the Hilbert space-filling curve (HSFC)

idea: the HSFC establishes a mapping between one-dimensional and two-/three-dimensional spaces: partition in 1D ➡ partition in the space

boxes in the partition are ordered ➡ samples are ordered on the basis of the box they belong to

disadvantage: points that are close on the curve are close in the sample space

but points that are close in the space are not necessarily close in the curve (and in the ordering!)

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

PROBLEM SETTING

Domains: $D = [0, 1]^2$ and $\Gamma = [-17, 17]^3$ or $\Gamma = [-17, 17]^6$

Covariance functions:

A. Squared Exponential (Gaussian): $cov(\boldsymbol{x}, \boldsymbol{x}') = \sigma_0 \exp\left\{-\frac{\|\boldsymbol{x} - \boldsymbol{x}'\|^2}{2\delta^2}\right\}$;

B. Exponential: $cov(\boldsymbol{x}, \boldsymbol{x}') = \sigma_0 \exp\left\{-\frac{\|\boldsymbol{x} - \boldsymbol{x}'\|}{\delta}\right\}$;

C. γ-Exponential: $cov(\boldsymbol{x}, \boldsymbol{x}') = \sigma_0 \exp\left\{-\frac{\|\boldsymbol{x} - \boldsymbol{x}'\|^\gamma}{\delta^\gamma}\right\}$, $\gamma \in (0, 2]$;

D. Rational quadratic: $cov(\boldsymbol{x}, \boldsymbol{x}') = \left(1 + \frac{\|\boldsymbol{x} - \boldsymbol{x}'\|^2}{2\alpha\delta^2}\right)^{-\alpha}$, $\alpha > 0$;

δ: characteristic distance of the spatial domain

PROBLEM SETTING

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

Sparse Grid generation:

- **technique:** adaptive refinement, local piecewise linear basis
- **software:** TASMANIAN `http://tasmanian.ornl.gov`
  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`
  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

$$R_l = \frac{S \sum_{i=1}^{\overline{S}} \text{ITS}_{l,i}}{\sum_{k=1}^{m(l)} \text{its}_{l,k}}$$

total increase in work induced by the ensemble propagation with a level

$$\overline{R} = \frac{S \sum_{l=4}^{L} \sum_{i=1}^{\overline{S}} \text{ITS}_{l,i}}{\sum_{l=4}^{L} \sum_{k=1}^{m(l)} \text{its}_{l,k}}$$

total increase in work over all levels

$\text{ITS}_{l,i}$: the number of iterations required by the i-th ensamble at level l

$\text{its}_{l,k}$: the number of iterations required by the k-th sample at level l

PROBLEM SETTING

Indicators of computational savings

$$R_l = \frac{S \sum_{i=1}^{\overline{S}} \text{ITS}_{l,i}}{\sum_{k=1}^{m(l)} \text{its}_{l,k}}$$

total increase in work induced by the ensemble propagation with a level

$$\overline{R} = \frac{S \sum_{l=4}^{L} \sum_{i=1}^{\overline{S}} \text{ITS}_{l,i}}{\sum_{l=4}^{L} \sum_{k=1}^{m(l)} \text{its}_{l,k}}$$

total increase in work over all levels

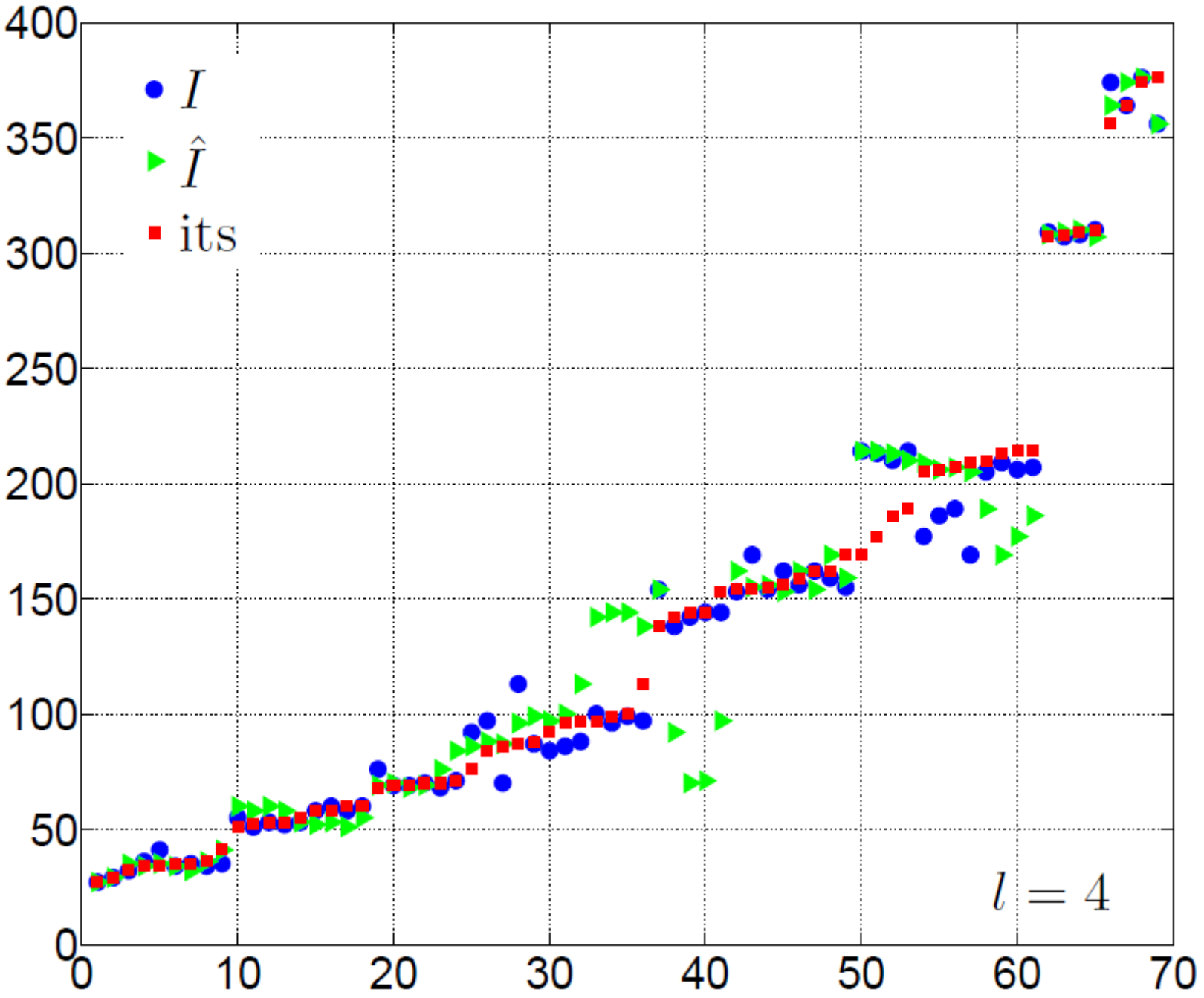
achieved speed-up:

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

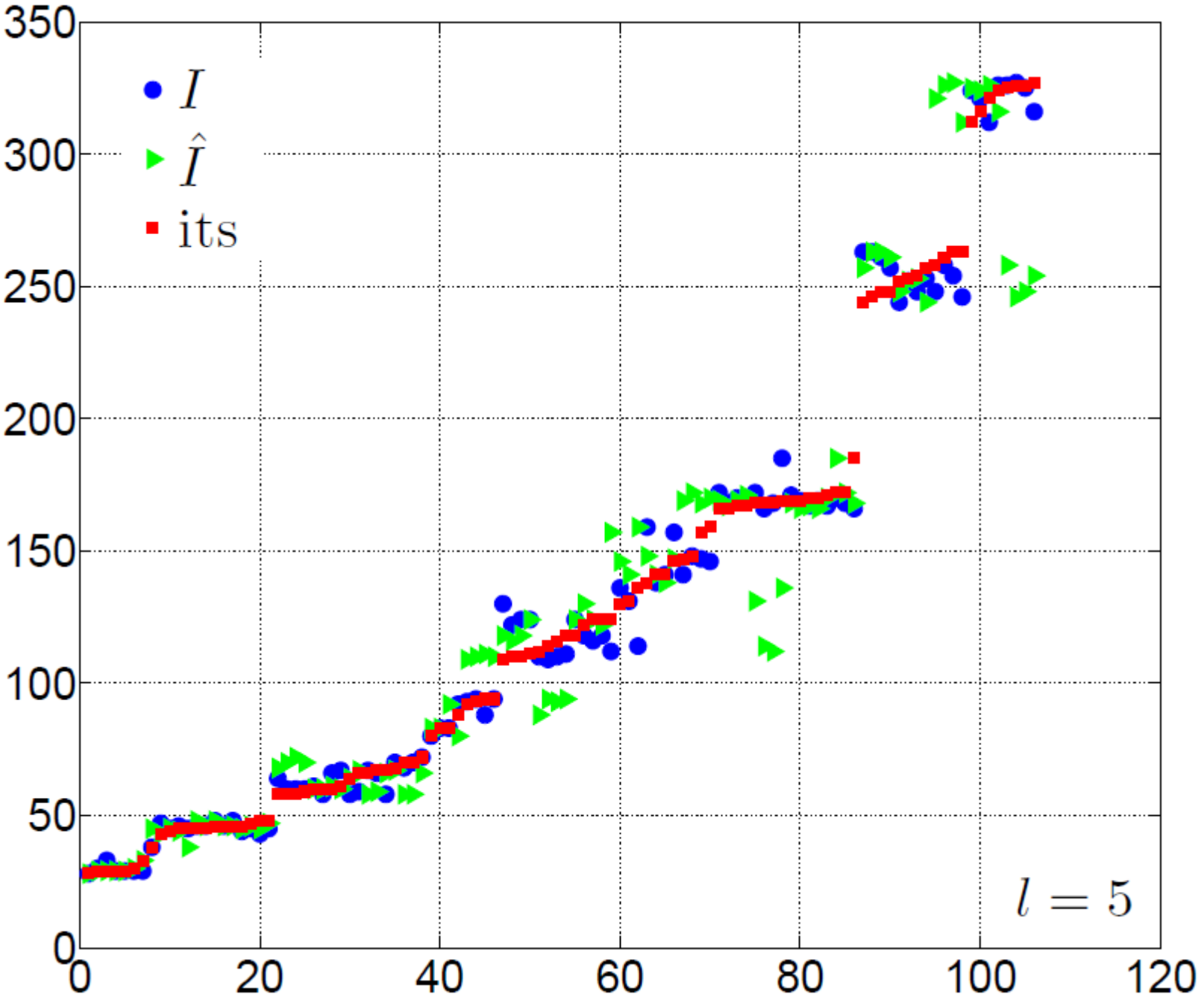
$\text{ITS}_{l,i}$: the number of iterations required by the i-th ensamble at level l

$\text{its}_{l,k}$: the number of iterations required by the k-th sample at level l

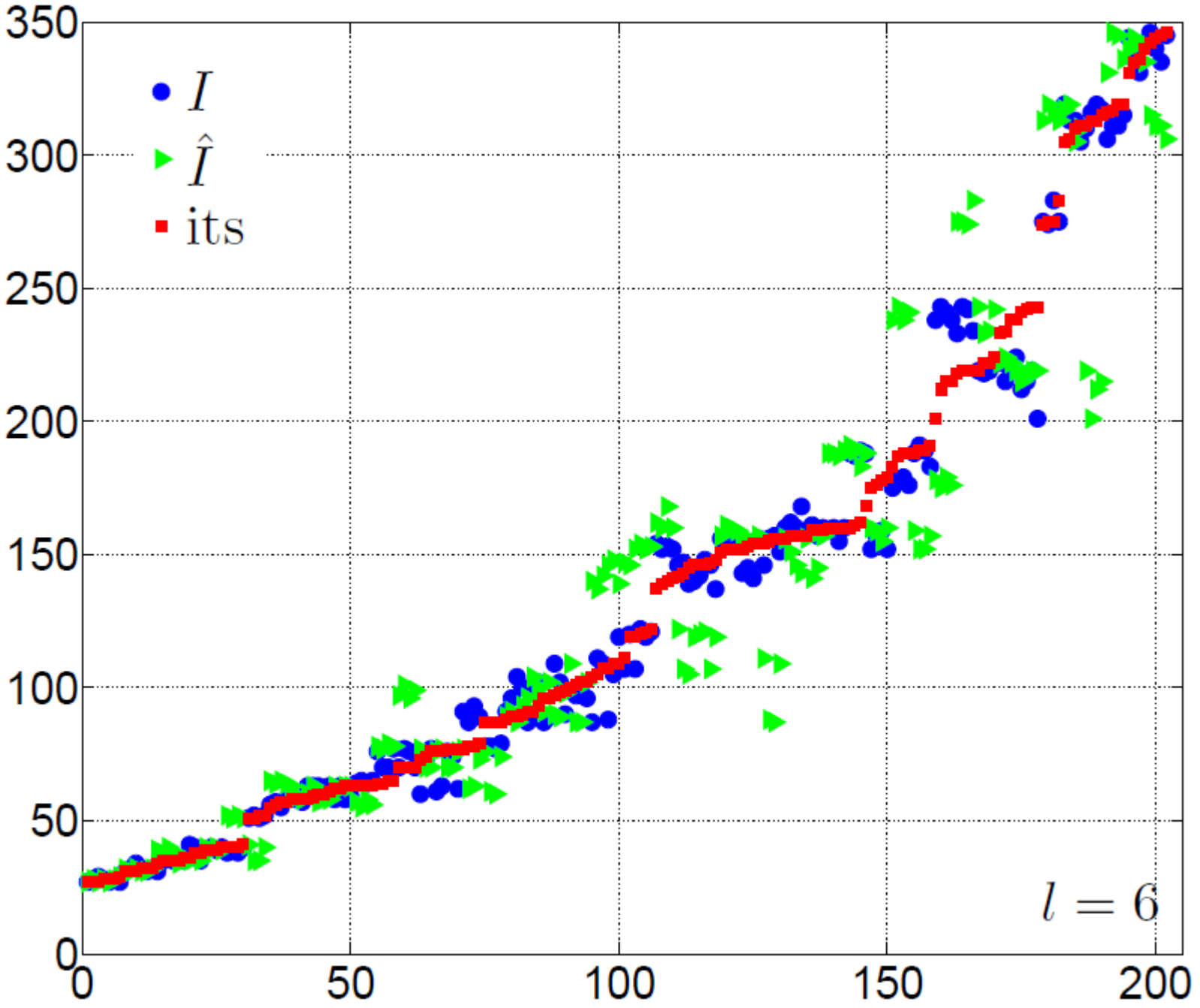
SQUARED EXPONENTIAL COVARIANCE



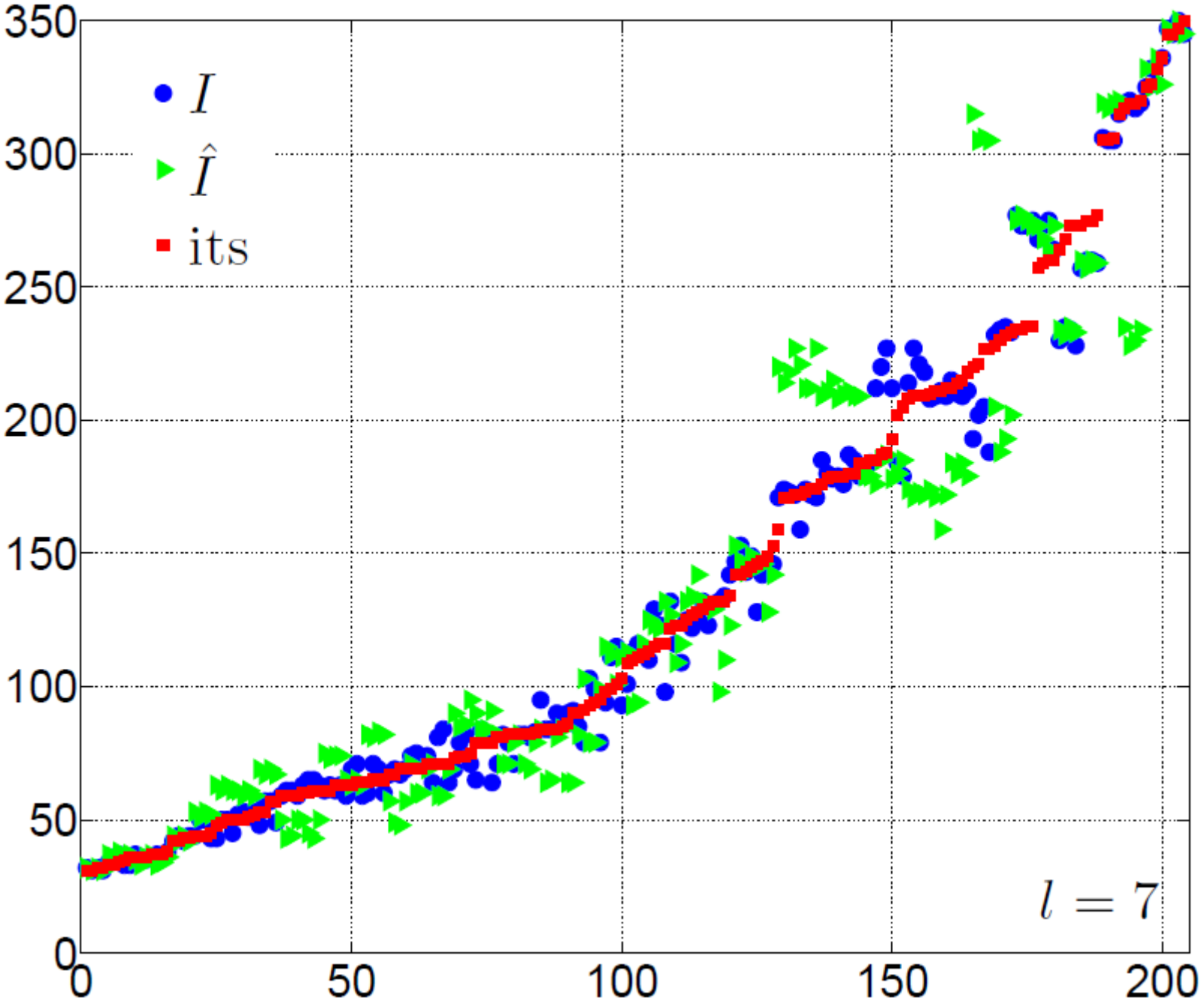
SQUARED EXPONENTIAL COVARIANCE



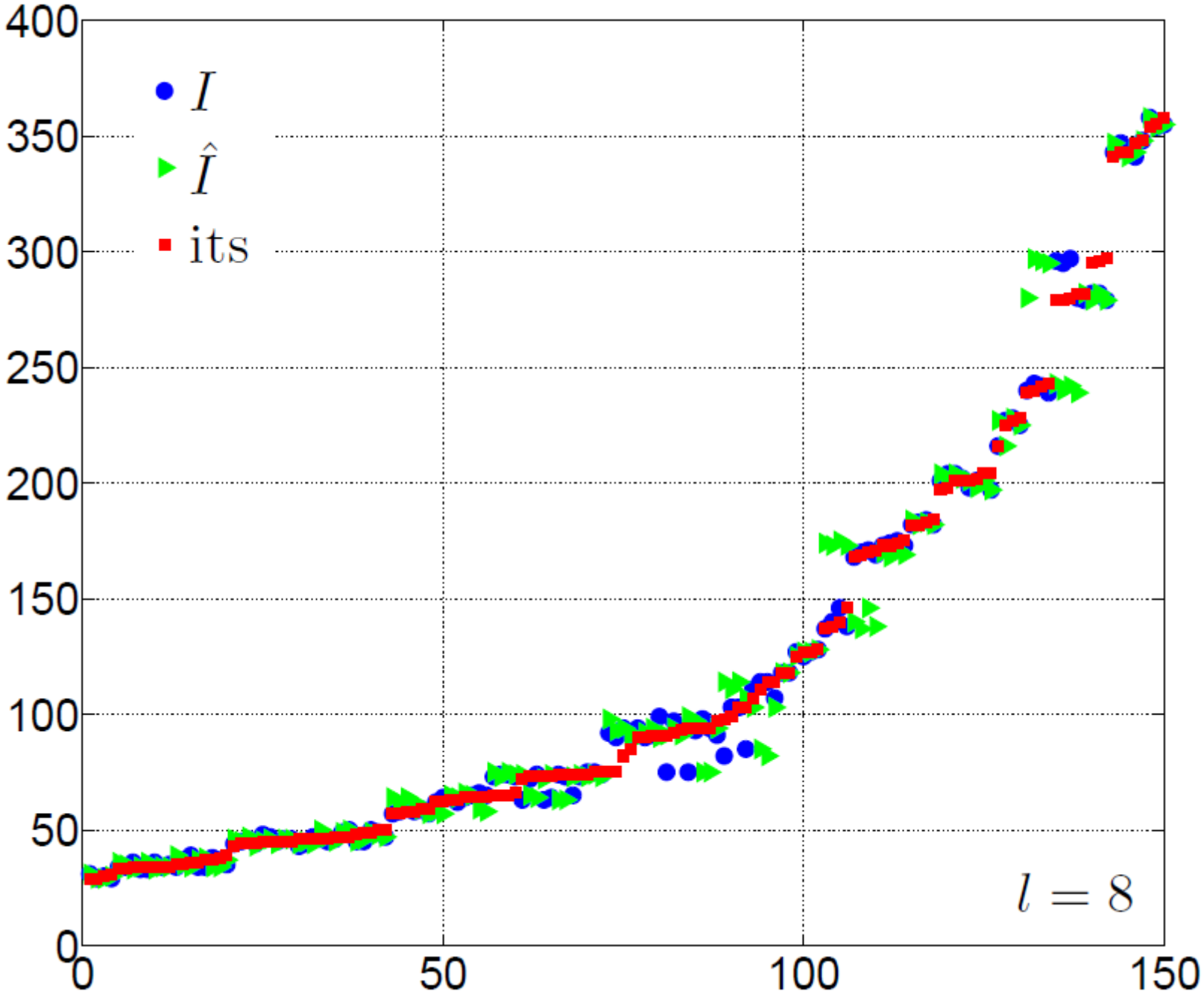
SQUARED EXPONENTIAL COVARIANCE



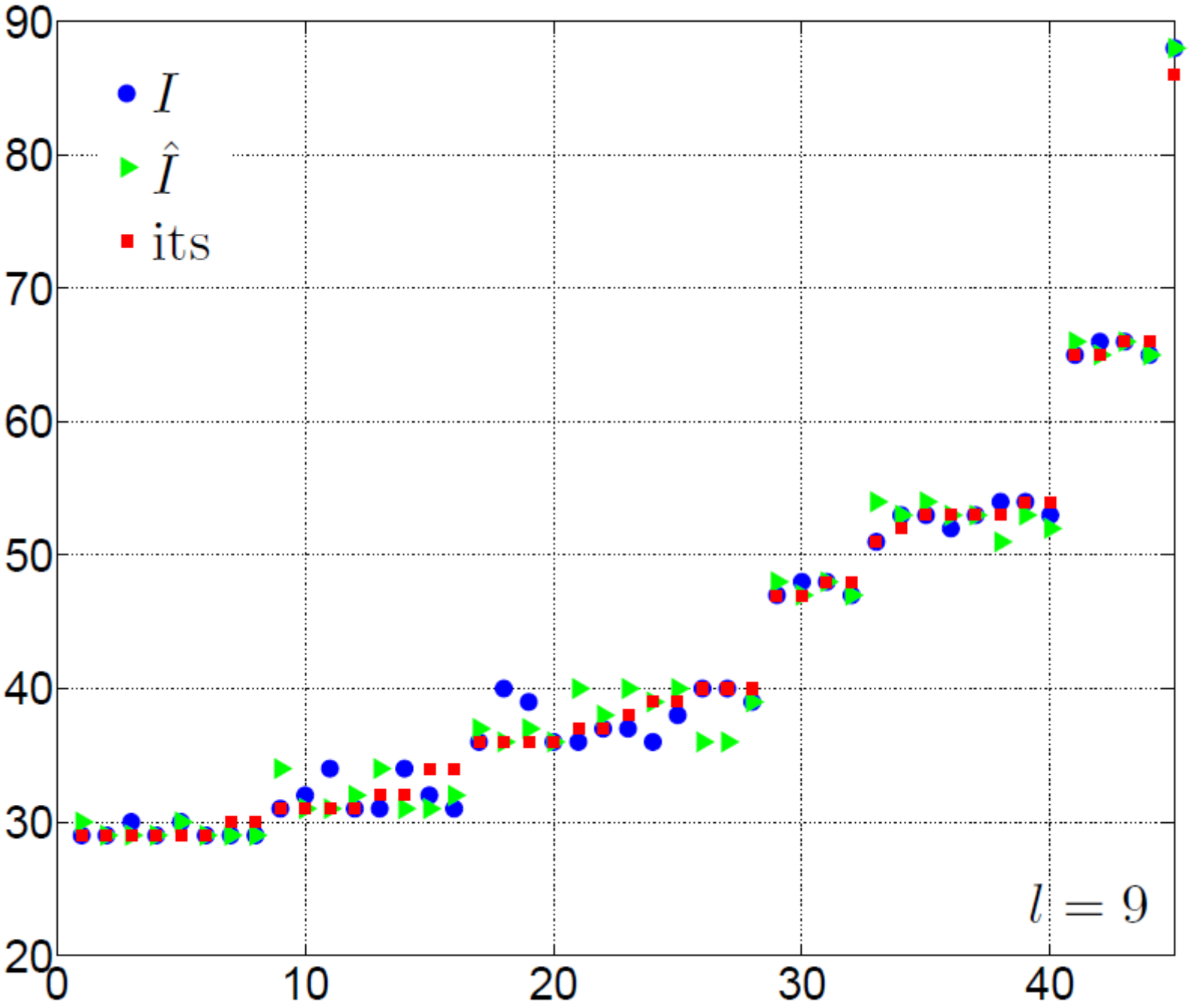
SQUARED EXPONENTIAL COVARIANCE



SQUARED EXPONENTIAL COVARIANCE

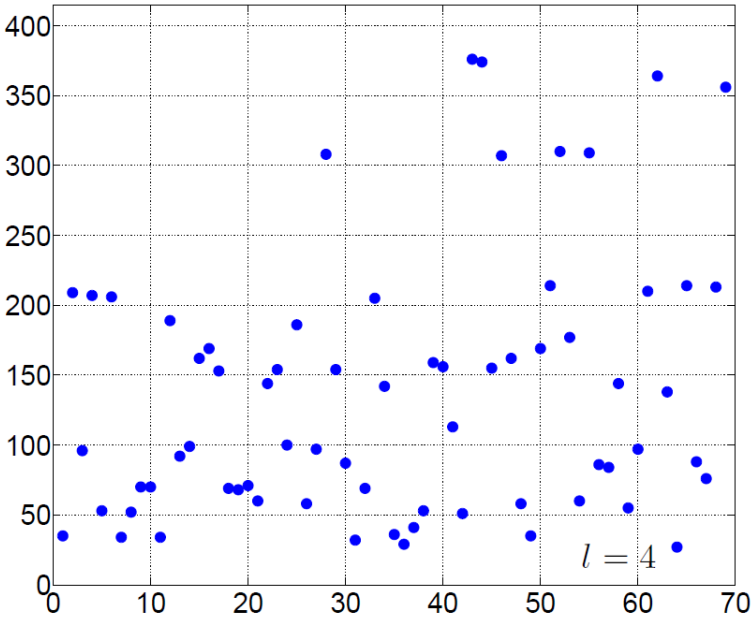


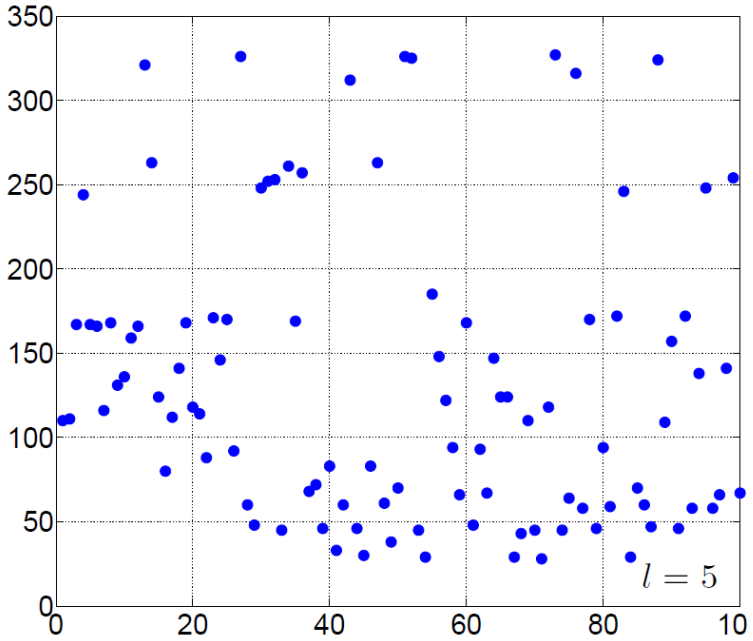
SQUARED EXPONENTIAL COVARIANCE



SQUARED EXPONENTIAL COVARIANCE

HSFC "grouping" – levels 4 and 5



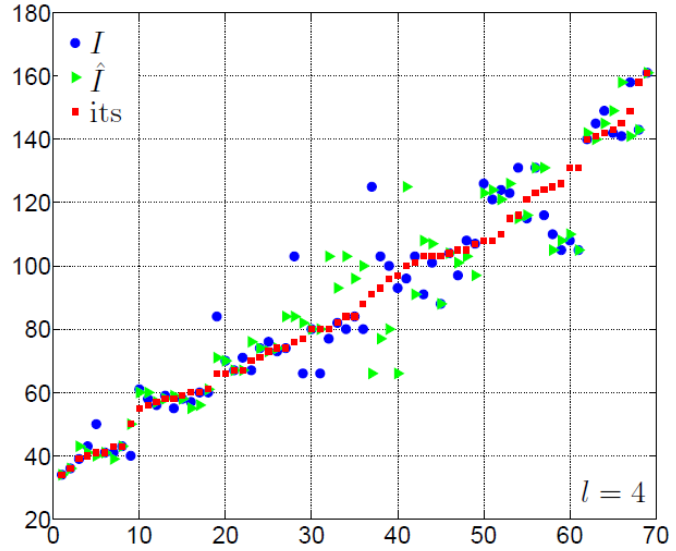


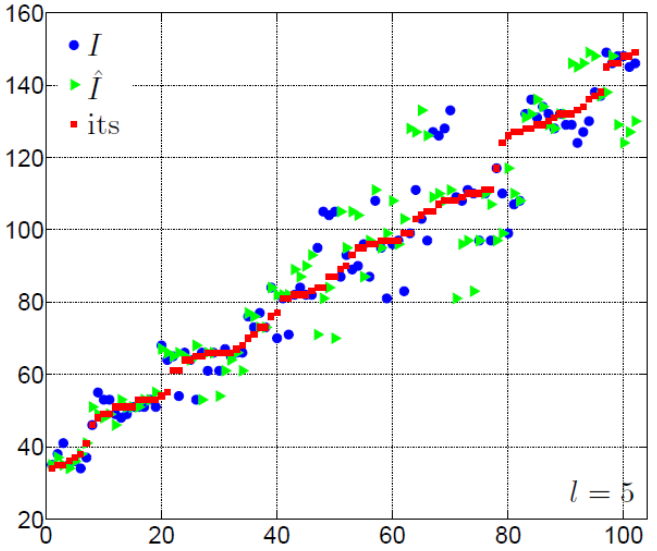
SQUARED EXPONENTIAL COVARIANCE

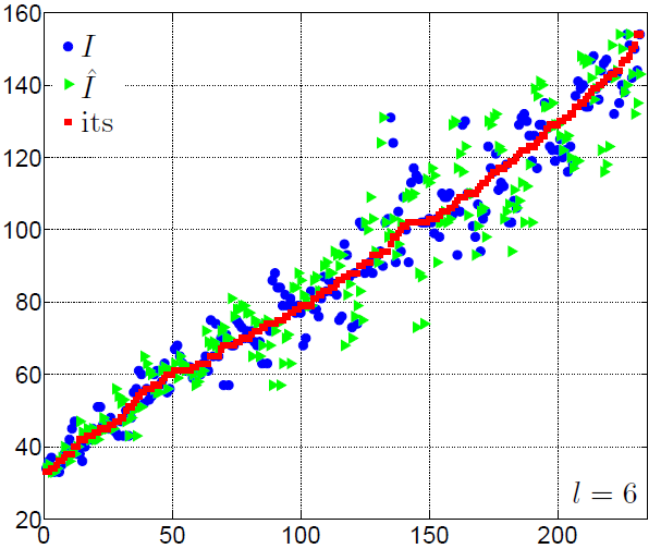
N	S	parameter	KL	No ordering
3	8	1.374	1.508	1.793
3	16	1.469	1.739	2.197
3	32	1.652	1.989	2.852

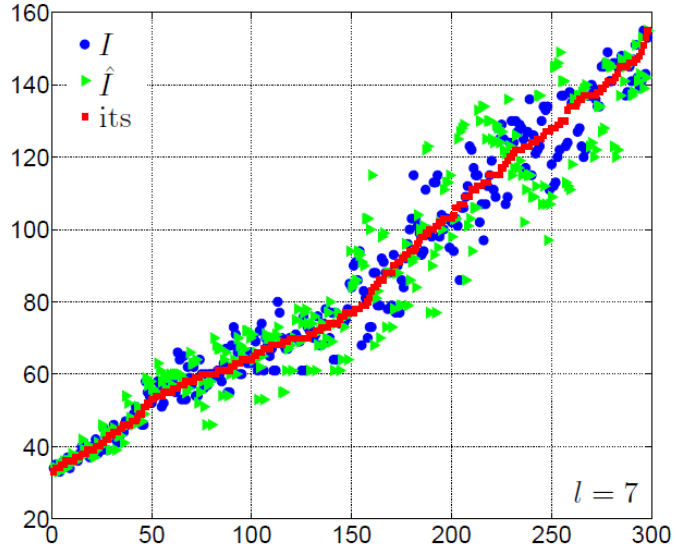
N	S	parameter	KL	No ordering
6	8	1.868	2.488	2.620
6	16	1.944	2.836	3.064
6	32	2.040	3.105	3.572

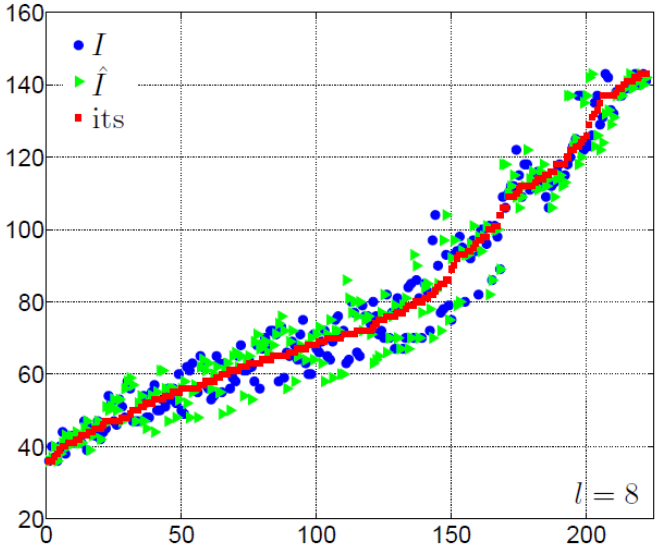
EXPONENTIAL COVARIANCE

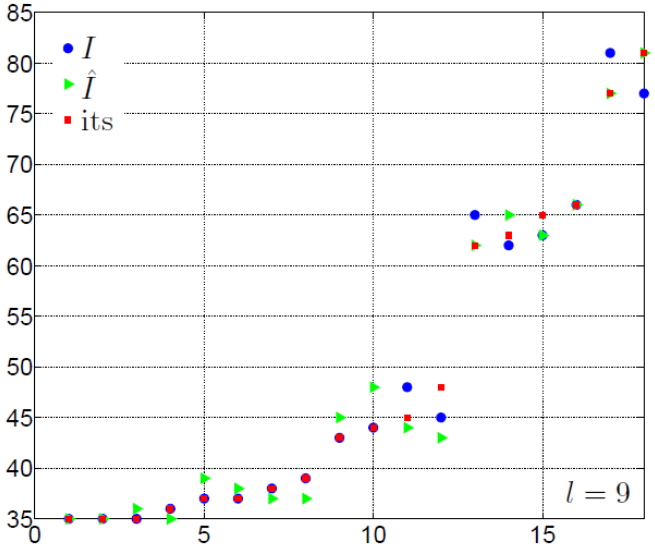




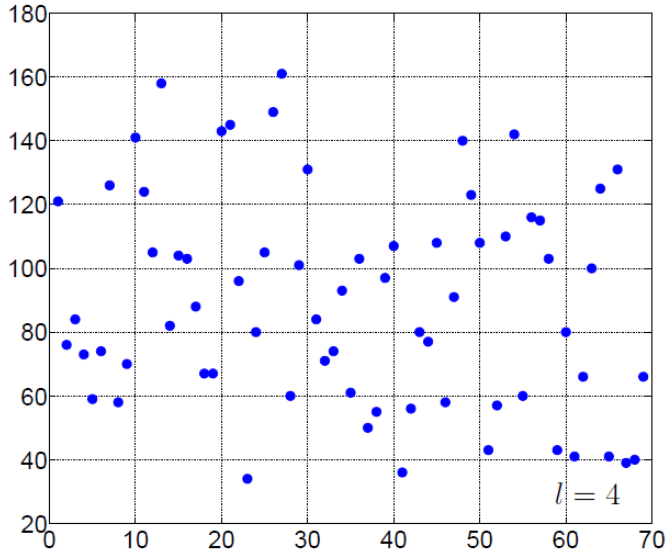


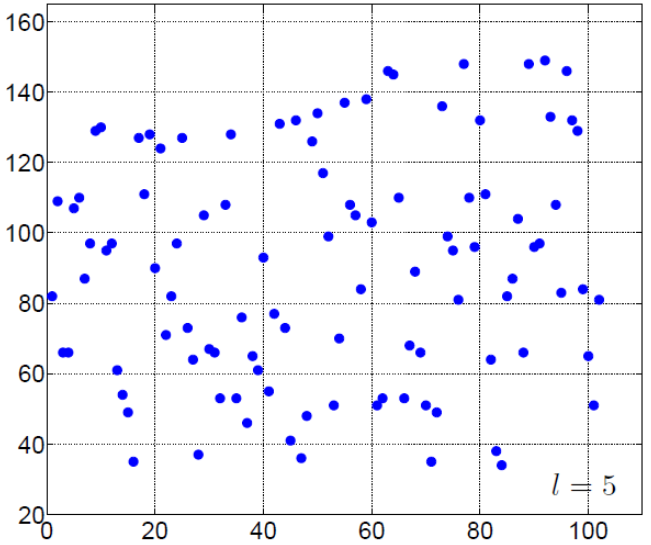


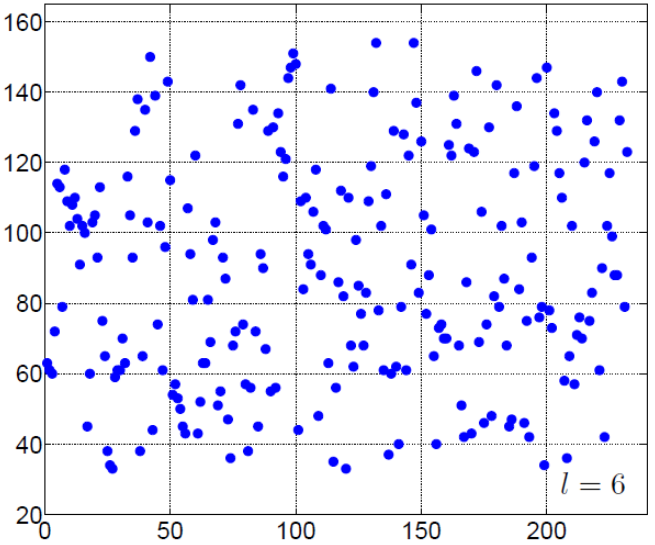


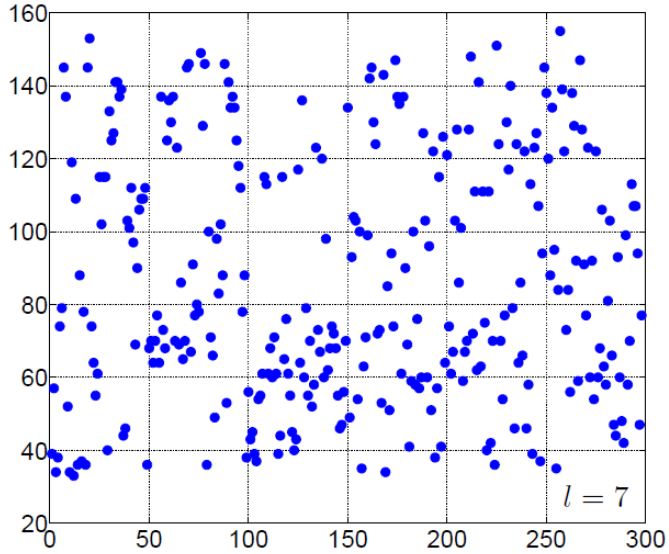


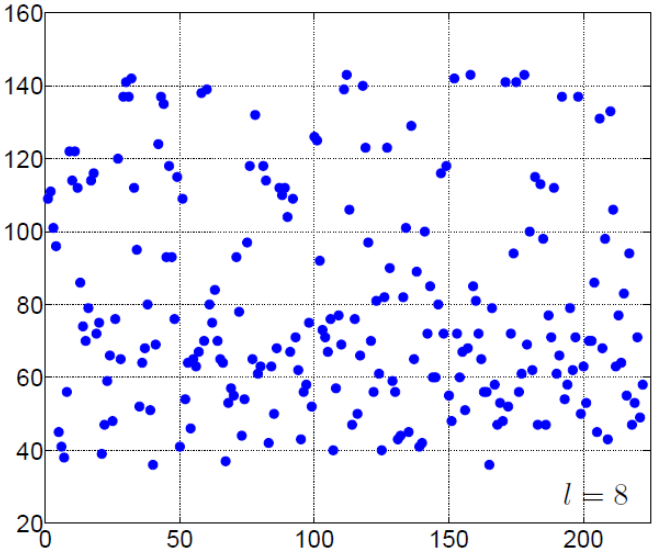
EXPONENTIAL COVARIANCE

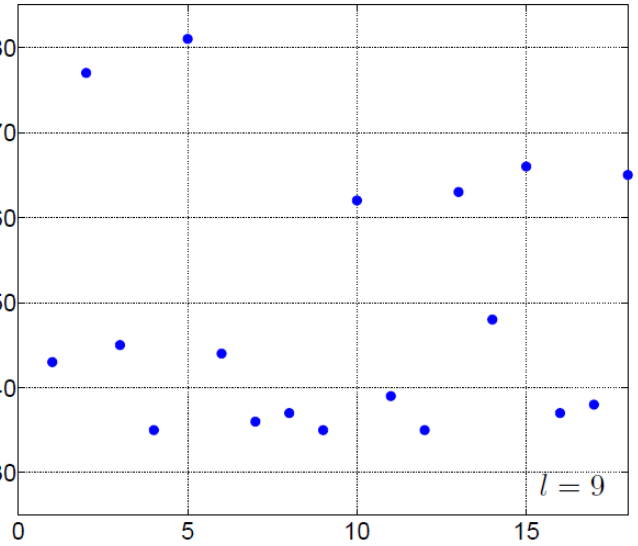










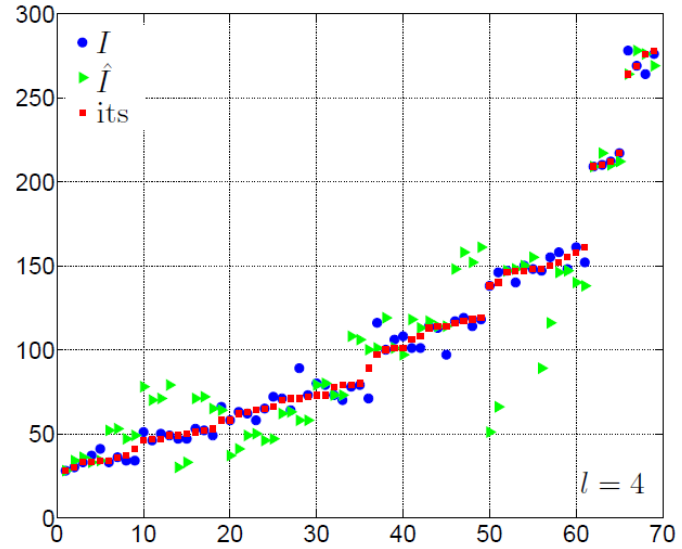


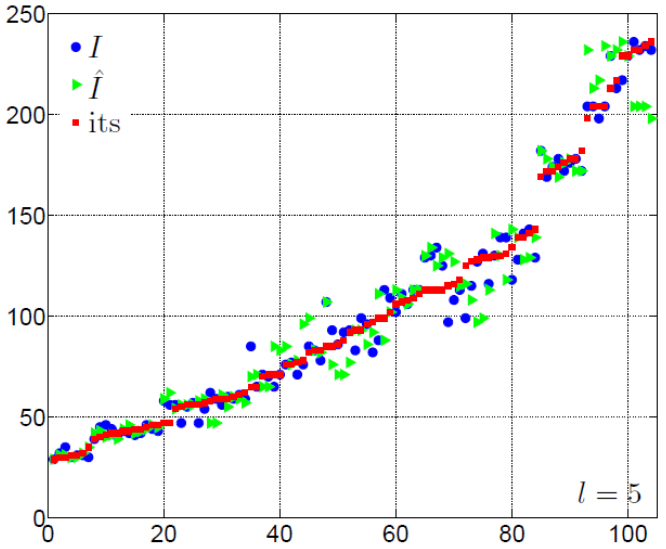
EXPONENTIAL COVARIANCE

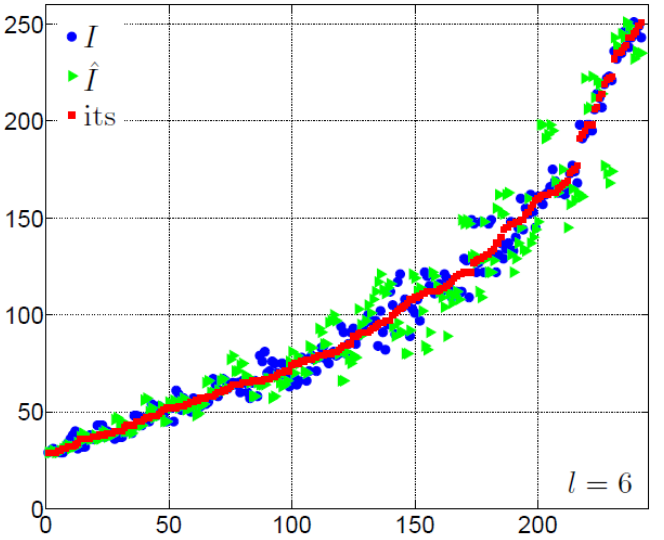
N	S	parameter	KL	No ordering
3	8	1.274	1.325	1.448
3	16	1.337	1.430	1.673
3	32	1.427	1.567	1.847

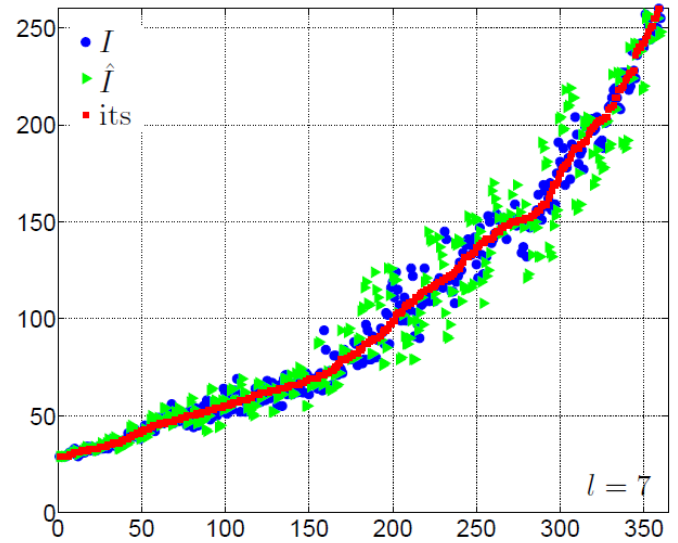
N	S	parameter	KL	No ordering
6	8	1.630	2.021	2.116
6	16	1.702	2.241	2.449
6	32	1.794	2.412	2.837

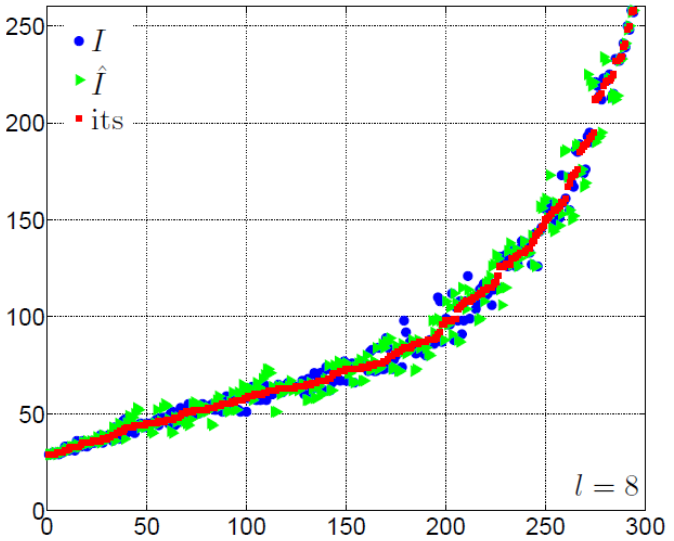
γ-EXPONENTIAL COVARIANCE

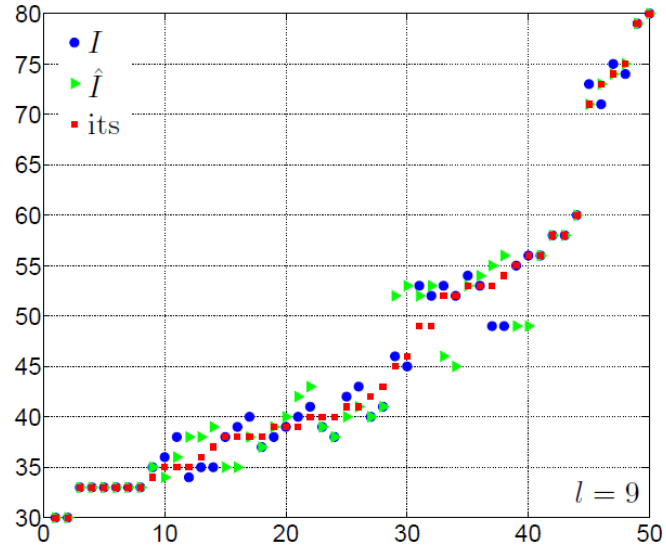




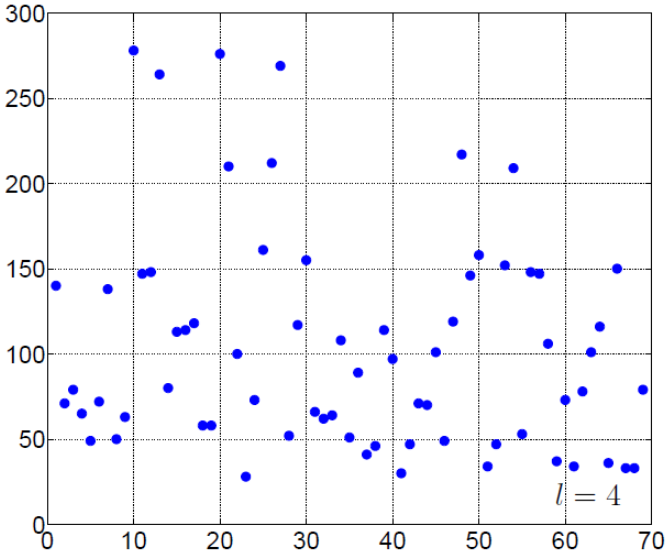


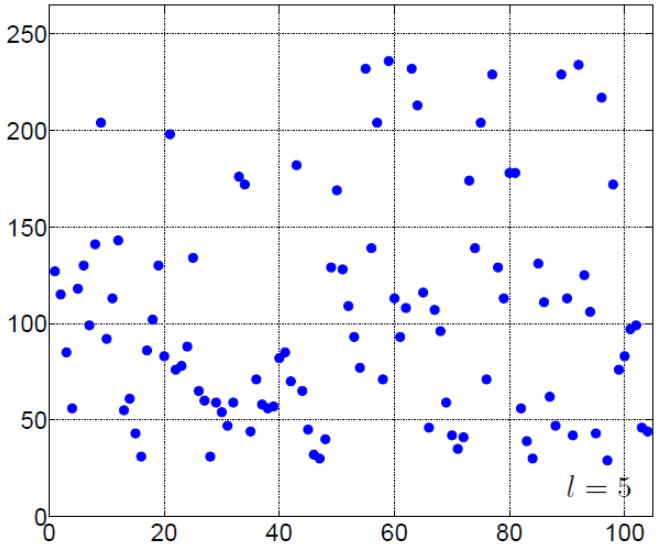


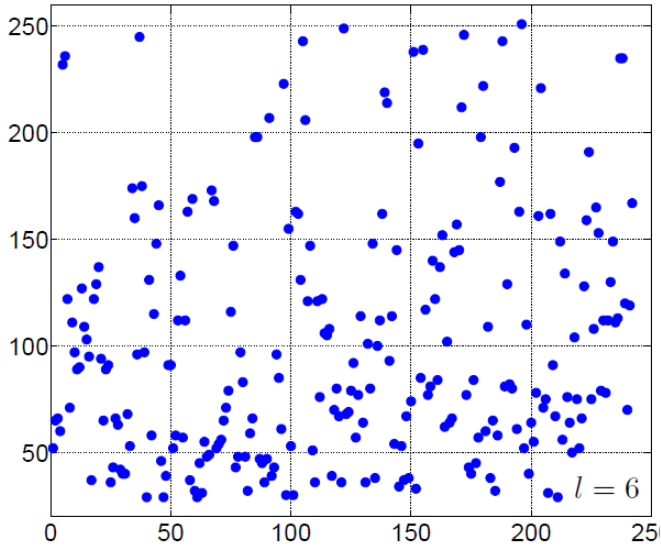


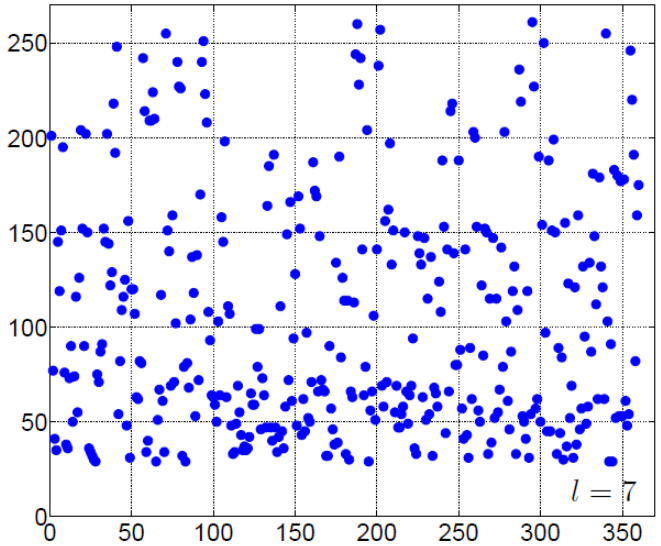


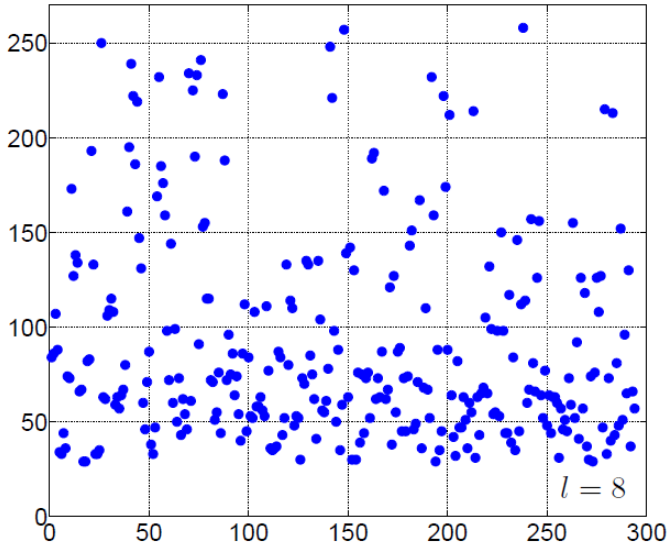
γ-EXPONENTIAL COVARIANCE

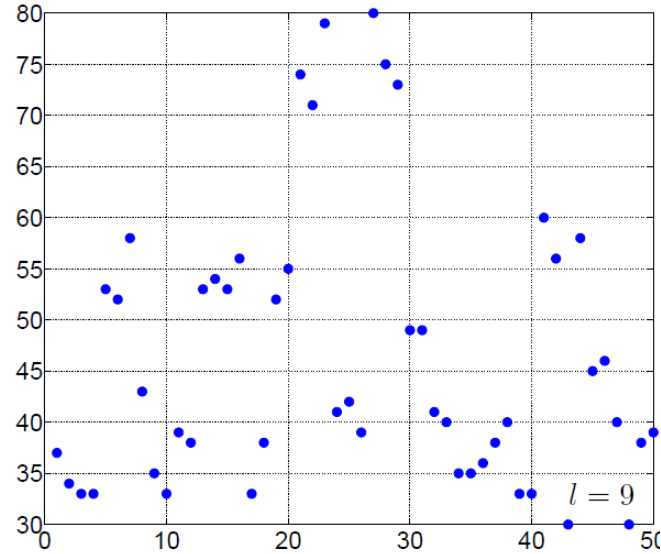










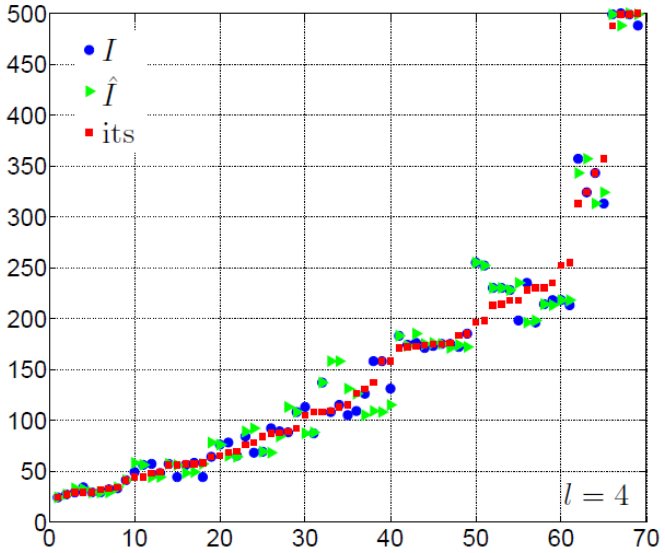


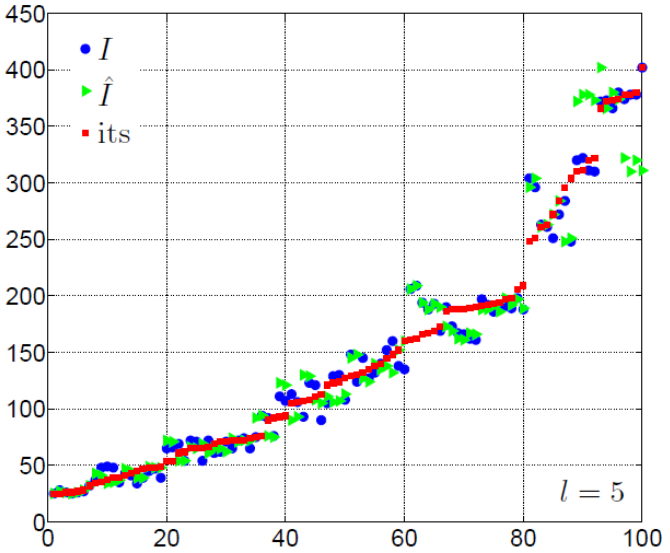
γ-EXPONENTIAL COVARIANCE

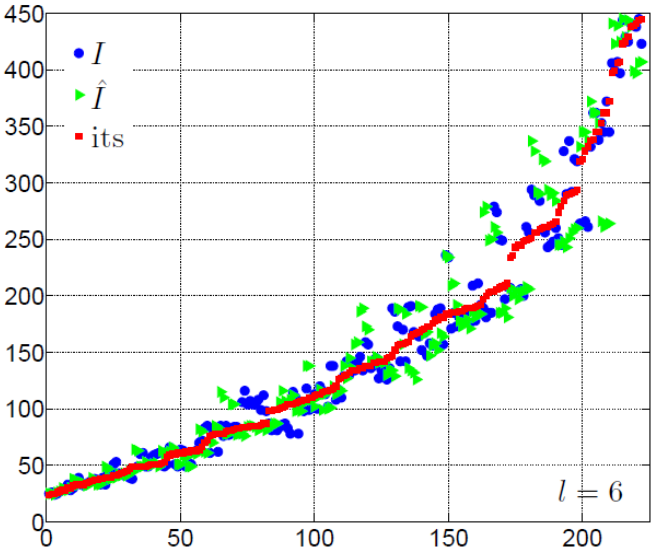
N	S	parameter	KL	No ordering
3	8	1.217	1.324	1.503
3	16	1.272	1.467	1.794
3	32	1.384	1.627	2.223

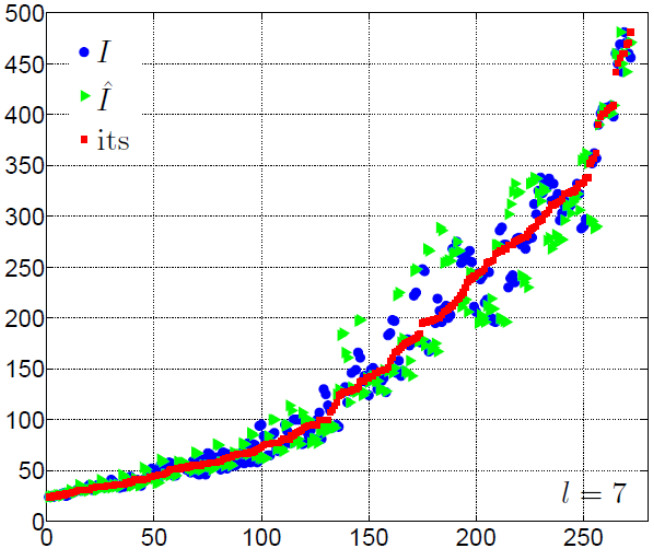
N	S	parameter	KL	No ordering
6	8	1.770	2.254	2.380
6	16	1.863	2.571	2.815
6	32	1.999	2.837	3.375

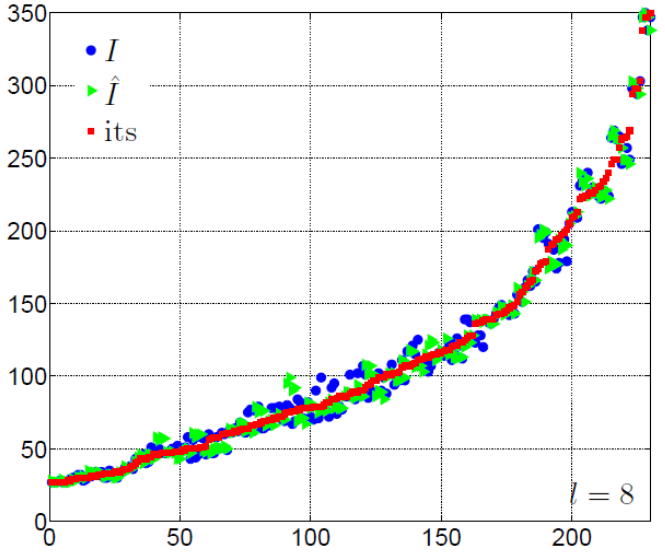
RATIONAL QUADRATIC COVARIANCE

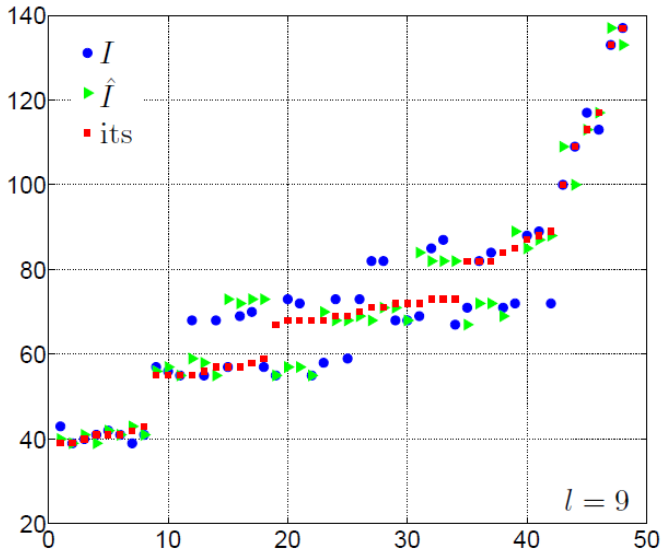




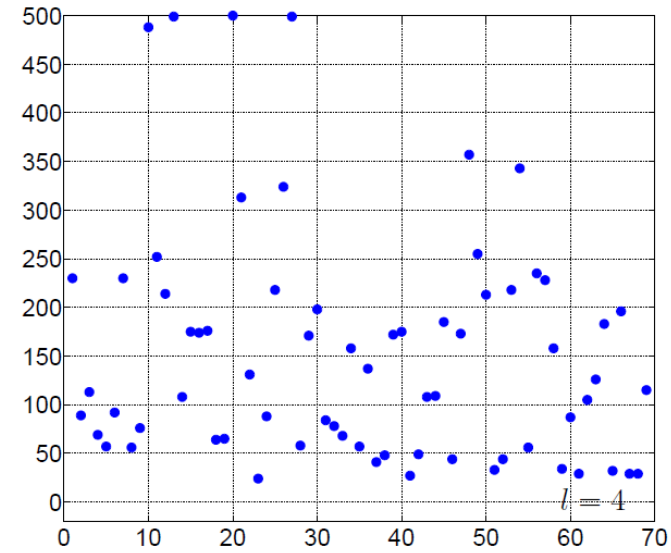


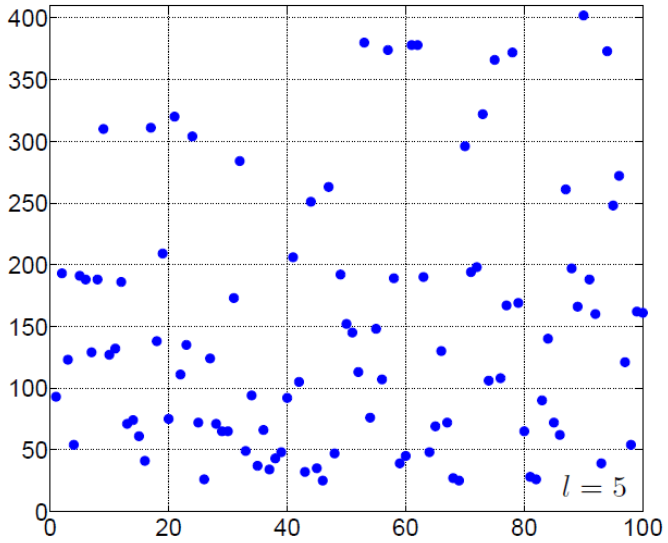


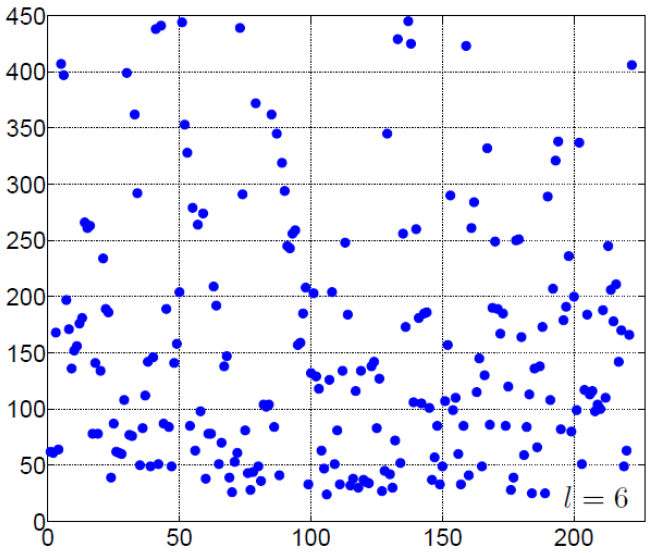


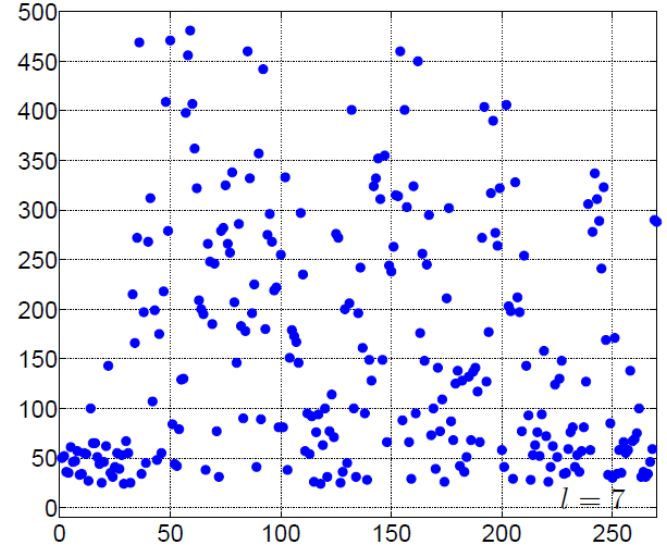


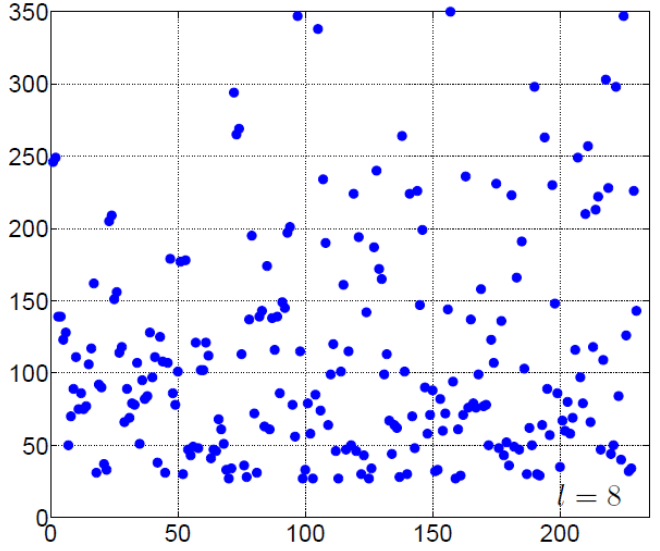
RATIONAL QUADRATIC COVARIANCE

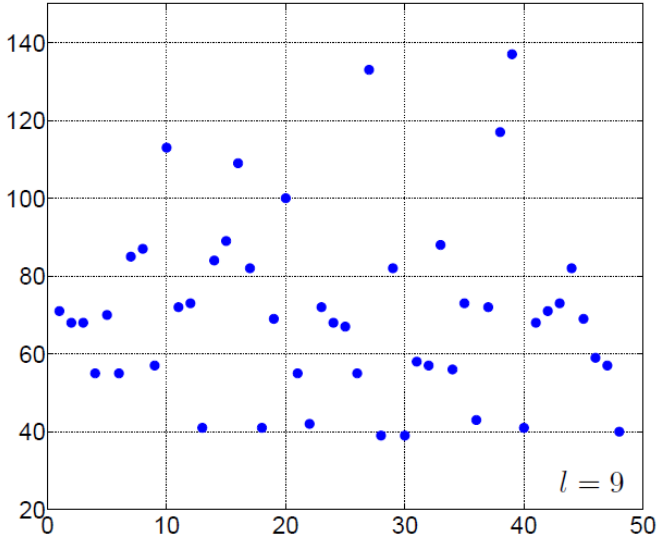












RATIONAL QUADRATIC COVARIANCE

N	S	parameter	KL	No ordering
3	8	1.348	1.499	1.785
3	16	1.436	1.709	2.198
3	32	1.601	1.925	2.633

N	S	parameter	KL	No ordering
6	8	1.773	2.219	2.354
6	16	1.848	2.415	2.713
6	32	1.923	2.611	3.079

LOTS OF THINGS TO DO!

note 1: we introduced indicators that induce an ordering as close as possible to the one associated with the number of iterations

idea: within the adaptive algorithm use quantities computed at the previous level to predict the number of iterations for the new samples

➡ design of surrogates for the number of iterations

LOTS OF THINGS TO DO!

note 1: we introduced indicators that induce an ordering as close as possible to the one associated with the number of iterations

idea: within the adaptive algorithm use quantities computed at the previous level to predict the number of iterations for the new samples

➡ design of surrogates for the number of iterations

note 2: points in the sparse grid can be represented in a tree structure

idea: 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

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