Direct Solution of SPDEs in Hierarchical Tucker Format

(joint work with Jonas Ballani and Melanie Kluge)

Lars Grasedyck

RWTH AACHEN

Institut für Geometrie und Praktische Mathematik

http://www.igpm.rwth-aachen.de

OVERVIEW

Motivation

- PDEs with stochastic/many parameters

Overview

Motivation

- PDEs with stochastic/many parameters

$\mathcal{H}$-Tucker

- Multilevel / Hierarchical Rank
- Tree Structure

Overview

Motivation

- PDEs with stochastic/many parameters

$\mathcal{H}$-Tucker

- Multilevel / Hierarchical Rank
- Tree Structure

Black Box

- Rank Revealing Tensor Approximation
- Nestedness

Overview

Motivation

- PDEs with stochastic/many parameters

$\mathcal{H}$-Tucker

- Multilevel / Hierarchical Rank
- Tree Structure

Black Box

- Rank Revealing Tensor Approximation
- Nestedness

Application to SPDEs

Motivation

Model problem: 2nd order unif. ell. PDE

$$-\mathrm{div}\,\sigma(x)\,\nabla u(x) = f(x) \qquad x \in \Omega \subset \mathbb{R}^{2 \text{ or } 3}$$

Motivation

Model problem: 2nd order unif. ell. PDE with parameters

$$-\mathrm{div}\, \sigma(x, p)\, \nabla u(x, p) = f(x) \qquad x \in \Omega \subset \mathbb{R}^{2 \text{ or } 3}$$

Motivation

Model problem: 2nd order unif. ell. PDE with parameters

$$-\text{div}\, \sigma(x,p)\, \nabla u(x,p) = f(x,p) \qquad x \in \Omega \subset \mathbb{R}^{2 \text{ or } 3}$$

MOTIVATION

Model problem: 2nd order unif. ell. PDE with parameters

$$-\operatorname{div} \sigma(x,p)\, \nabla u(x,p) = f(x,p) \qquad x \in \Omega(p) \subset \mathbb{R}^{2 \text{ or } 3}$$

Parameter $p = (p_1, \ldots, p_d)$:

MOTIVATION

Model problem: 2nd order unif. ell. PDE with parameters

$$-\text{div}\,\sigma(x,p)\,\nabla u(x,p) = f(x,p) \qquad x \in \Omega(p) \subset \mathbb{R}^{2 \text{ or } 3}$$

Parameter $p = (p_1, \ldots, p_d)$:

- Could be uncertain and uniform $p_\mu \in (0,1)$

Motivation

Model problem: 2nd order unif. ell. PDE with parameters

$$-\mathrm{div}\,\sigma(x,p)\,\nabla u(x,p) = f(x,p) \qquad x \in \Omega(p) \subset \mathbb{R}^{2 \text{ or } 3}$$

Parameter $p = (p_1, \ldots, p_d)$:

- Could be uncertain and uniform $p_\mu \in (0,1)$
- Could be discrete $p_\mu \in \{1, \ldots, n_\mu\}$

Motivation

Model problem: 2nd order unif. ell. PDE with parameters

$$-\mathrm{div}\, \sigma(x,p)\, \nabla u(x,p) = f(x,p) \qquad x \in \Omega(p) \subset \mathbb{R}^{2 \text{ or } 3}$$

Parameter $p = (p_1, \ldots, p_d)$:

- Could be uncertain and uniform $p_\mu \in (0,1)$
- Could be discrete $p_\mu \in \{1, \ldots, n_\mu\}$
- Could be given by a probability distribution

$p_\mu =$	1	$\cdots$	n_μ
	$\rho_\mu(1)$	$\cdots$	$\rho_\mu(n_\mu)$

Motivation

Model problem: 2nd order unif. ell. PDE with parameters

$$-\mathrm{div}\, \sigma(x,p)\, \nabla u(x,p) = f(x,p) \qquad x \in \Omega(p) \subset \mathbb{R}^{2 \text{ or } 3}$$

Parameter $p = (p_1, \ldots, p_d)$:

- Could be uncertain and uniform $p_\mu \in (0,1)$
- Could be discrete $p_\mu \in \{1, \ldots, n_\mu\}$
- Could be given by a probability distribution

$p_\mu =$	1	$\cdots$	n_μ
	$\rho_\mu(1)$	$\cdots$	$\rho_\mu(n_\mu)$

- If p arbitrary random field $\rightarrow$ KL+PC expansion
  cf. Ghanem/Spanos,Babuska,Schwab,Matthies,Ernst,...
  Idea: from description of p obtain parameterization by **independent** p_j

Motivation

From the solution

$$u(x,p) \qquad x \in \Omega, p \in [0,1]^d$$

we want to compute functionals of interest:

$$G(x) = \mathcal{G}[u(x,\cdot)] \qquad x \in \Omega$$

$$G = \mathcal{G}[u(\cdot,\cdot)]$$

Problem:

$$\Omega \times [0,1]^d \qquad \text{is huge}$$

Motivation

From the solution

$$u(x,p) \qquad x \in \Omega, p \in [0,1]^d$$

we want to compute functionals of interest:

$$G(x) = \mathcal{G}[u(x,\cdot)] \qquad x \in \Omega$$

$$G = \mathcal{G}[u(\cdot,\cdot)]$$

Problem:

$$\Omega \times [0,1]^d \qquad \text{is huge}$$

Approaches:

- Monte Carlo (MC), QMC, Multilevel MC
- Small parameter set $p \in P_{small}$, separate x and p
- Sparse representation of $u(\cdot,\cdot)$

Motivation

From the solution

$$u(x, p) \qquad x \in \Omega, p \in [0,1]^d$$

we want to compute functionals of interest:

$$G(x) = \mathcal{G}[u(x,\cdot)] \qquad x \in \Omega$$

$$G = \mathcal{G}[u(\cdot,\cdot)]$$

Problem:

$$\Omega \times [0,1]^d \qquad \text{is huge}$$

Approaches:

- Monte Carlo (MC), QMC, Multilevel MC
- Small parameter set $p \in P_{small}$, separate x and p
- Sparse representation of $u(\cdot,\cdot)$

Our approach:

Low Rank Structures

MOTIVATION

$$u(x, p) \qquad x \in \Omega, p \in [0,1]^d$$

Why low rank structures ?

- Good for analysis
- Powerful Linear Algebra Tools
- Simple + Successful

MOTIVATION

$$u(x, p) \qquad x \in \Omega, p \in [0,1]^d$$

Why low rank structures ?

- Good for analysis
- Powerful Linear Algebra Tools
- Simple + Successful

What is the rank k of $u(x, p)$?

$$u(x, p) = \sum_j v_j(x) w_j(p) \qquad \rightarrow \text{separation rank for } x \text{ and } p$$

MOTIVATION

$$u(x,p) \qquad x \in \Omega, p \in [0,1]^d$$

Why low rank structures ?

- Good for analysis
- Powerful Linear Algebra Tools
- Simple + Successful

What is the rank k of $u(x,p)$?

$$u(x,p) = \sum_j v_j(x) w_j(p) \qquad \to \text{separation rank for } x \text{ and } p$$

Can we extend this to more than one separation ?

Motivation

$$u(x,p) \qquad x \in \Omega, p \in [0,1]^d$$

Why low rank structures ?

- Good for analysis
- Powerful Linear Algebra Tools
- Simple + Successful

What is the rank k of $u(x,p)$?

$$u(x,p) = \sum_j v_j(x) w_j(p) \qquad \rightarrow \text{separation rank for } x \text{ and } p$$

Can we extend this to more than one separation ?

$$u(x,p) = \sum_j v_j(x) \underbrace{w_{j,1}(p_1) \cdots w_{j,d}(p_d)}_{\text{one-dimensional functions}}$$

Motivation

$$u(x,p) \qquad x \in \Omega, p \in [0,1]^d$$

Why low rank structures ?

- Good for analysis
- Powerful Linear Algebra Tools
- Simple + Successful

What is the rank k of $u(x,p)$?

$$u(x,p) = \sum_j v_j(x) w_j(p) \qquad \rightarrow \text{separation rank for } x \text{ and } p$$

Can we extend this to more than one separation ?

$$u(x,p) = \sum_j \underbrace{w_{j,1}(x,p_1) \cdots w_{j,d}(x,p_d)}_{\text{low-dimensional functions}}$$

$\mathcal{H}$-Tucker Format

Function $f(p_1, \ldots, p_d)$

Tensor $A_{i_1,\ldots,i_d} \in \mathbb{R}^{\mathcal{I}_1 \times \cdots \times \mathcal{I}_d}$

$\mathcal{H}$-Tucker Format

Function $f(p_1, \ldots, p_d)$

Two variables

$$f(x, y) = \sum_j u_j(x) v_j(y)$$

Tensor $A_{i_1,\ldots,i_d} \in \mathbb{R}^{\mathcal{I}_1 \times \cdots \times \mathcal{I}_d}$

Two indices

$$A_{i_1,i_2} = \sum_j (u_j)_{i_1} (v_j)_{i_2}$$

$\mathcal{H}$-Tucker Format

Function $f(p_1, \ldots, p_d)$

Two variables

$$f(x, y) = \sum_j u_j(x) v_j(y)$$

Several variables

$$f(p_1, \ldots, p_d) = \sum_j u_j(p_1, \ldots, p_q) v_j(p_{q+1}, \ldots, p_d)$$

Tensor $A_{i_1,\ldots,i_d} \in \mathbb{R}^{\mathcal{I}_1 \times \cdots \times \mathcal{I}_d}$

Two indices

$$A_{i_1,i_2} = \sum_j (u_j)_{i_1} (v_j)_{i_2}$$

Several indices:

$$A_{i_1,\ldots,i_d} = \sum_j (u_j)_{i_1,\ldots,i_q} (v_j)_{i_{q+1},\ldots,i_d}$$

$\rightarrow$ Matricization + Low Rank

$\mathcal{H}$-Tucker Format

Function $f(p_1, \ldots, p_d)$

Tensor $A_{i_1,\ldots,i_d} \in \mathbb{R}^{\mathcal{I}_1 \times \cdots \times \mathcal{I}_d}$

$$f(p_1, \ldots, p_d) = \sum_j u_j(p_1, \ldots, p_q) v_j(p_{q+1}, \ldots, p_d)$$

$$A_{i_1,\ldots,i_d} = \sum_j (u_j)_{i_1,\ldots,i_q} (v_j)_{i_{q+1},\ldots,i_d}$$

$\mathcal{H}$-Tucker Format

Function $f(p_1, \ldots, p_d)$

Tensor $A_{i_1,\ldots,i_d} \in \mathbb{R}^{\mathcal{I}_1 \times \cdots \times \mathcal{I}_d}$

$$f(p_1, \ldots, p_d) = \sum_j u_j(p_1, \ldots, p_q) v_j(p_{q+1}, \ldots, p_d)$$

$$A_{i_1,\ldots,i_d} = \sum_j (u_j)_{i_1,\ldots,i_q} (v_j)_{i_{q+1},\ldots,i_d}$$

Several possiblities for splitting, e.g.

$$f(p_1, \ldots, p_d) = \sum_j u_j((p_r)_{r \in R}) v_j((p_c)_{c \in C})$$

$$A_{i_1,\ldots,i_d} = \sum_j (u_j)_{(i_r)_{r \in R}} (v_j)_{(i_c)_{c \in C}}$$

where $R \dot\cup C = \{1, \ldots, d\}$

$\mathcal{H}$-Tucker Format

Function $f(p_1,\ldots,p_d)$

Tensor $A_{i_1,\ldots,i_d} \in \mathbb{R}^{\mathcal{I}_1\times\cdots\times\mathcal{I}_d}$

$$f(p_1,\ldots,p_d) = \sum_j u_j(p_1,\ldots,p_q) v_j(p_{q+1},\ldots,p_d)$$

$$A_{i_1,\ldots,i_d} = \sum_j (u_j)_{i_1,\ldots,i_q} (v_j)_{i_{q+1},\ldots,i_d}$$

Several possiblities for splitting, e.g.

$$f(p_1,\ldots,p_d) = \sum_j u_j((p_r)_{r\in R}) v_j((p_c)_{c\in C})$$

$$A_{i_1,\ldots,i_d} = \sum_j (u_j)_{(i_r)_{r\in R}} (v_j)_{(i_c)_{c\in C}}$$

where $R \dot{\cup} C = \{1,\ldots,d\}$

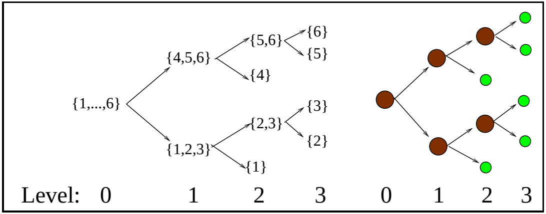
We need a system of splittings organized in a tree T:

$$t \in T : t \subset \{1,\ldots,d\}, \qquad Sons(t) = \{t_1,t_2\}, t_1 \dot{\cup} t_2 = t$$

$\mathcal{H}$-Tucker Format

$A \in \mathbb{R}^{\mathcal{I}}, \quad \mathcal{I} = \mathcal{I}_1 \times \cdots \times \mathcal{I}_d$

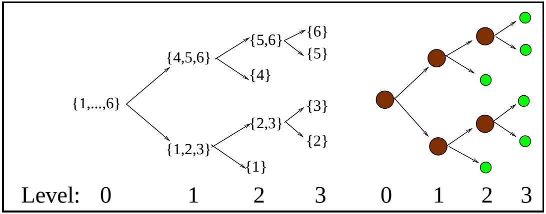
Dimension tree T



$\mathcal{H}$-Tucker Format

Dimension tree T

$A \in \mathbb{R}^{\mathcal{I}}, \quad \mathcal{I} = \mathcal{I}_1 \times \cdots \times \mathcal{I}_d$



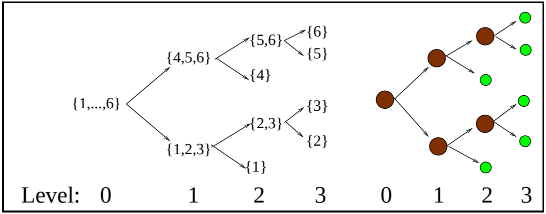
t-Matricization: $t \in T, \quad \mathcal{I}_t = \bigtimes_{\mu \in t} \mathcal{I}_\mu, \quad \mathcal{I}_{t'} = \bigtimes_{\mu \notin t} \mathcal{I}_\mu$

$$A \rightarrow A^{(t)} \in \mathbb{R}^{\mathcal{I}_t \times \mathcal{I}_{t'}} \quad t' := \{1, \ldots, d\} \setminus t$$

$\mathcal{H}$-Tucker Format

Dimension tree T

$A \in \mathbb{R}^{\mathcal{I}}, \quad \mathcal{I} = \mathcal{I}_1 \times \cdots \times \mathcal{I}_d$



***t*-Matricization:** $t \in T, \qquad \mathcal{I}_t = \bigtimes_{\mu \in t} \mathcal{I}_\mu, \qquad \mathcal{I}_{t'} = \bigtimes_{\mu \notin t} \mathcal{I}_\mu$

$$A \;\to\; A^{(t)} \in \mathbb{R}^{\mathcal{I}_t \times \mathcal{I}_{t'}} \qquad t' := \{1, \ldots, d\} \setminus t$$

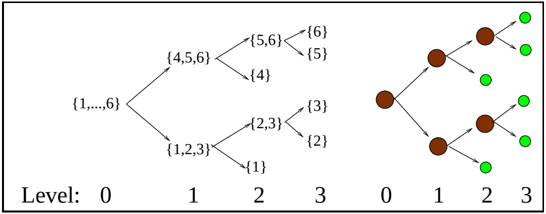
***t*-rank:**

$$\boxed{k_t := \operatorname{rank}(A^{(t)})}$$

$\mathcal{H}$-Tucker Format

Dimension tree T

$A \in \mathbb{R}^{\mathcal{I}}, \quad \mathcal{I} = \mathcal{I}_1 \times \cdots \times \mathcal{I}_d$



t-Matricization: $t \in T, \quad \mathcal{I}_t = \times_{\mu \in t} \mathcal{I}_\mu, \quad \mathcal{I}_{t'} = \times_{\mu \notin t} \mathcal{I}_\mu$

$$A \rightarrow A^{(t)} \in \mathbb{R}^{\mathcal{I}_t \times \mathcal{I}_{t'}} \quad t' := \{1, \ldots, d\} \setminus t$$

t-rank:

$$\boxed{k_t := \mathrm{rank}(A^{(t)})}$$

$\mathcal{H}$-Tucker tensor:

$$\mathcal{H}\text{-Tucker}(k, T) := \{A \in \mathbb{R}^{\mathcal{I}} \mid \forall t \in T : \ \mathrm{rank}(A^{(t)}) \leq k_t\}.$$

[Grasedyck: Hierarchical SVD of Tensors, SIMAX 31(2010)]

[Hackbusch/Kühn, Tyrtyshnikov/Oseledets, Khoromskij, Vidal, etc.]

Some Properties of $\mathcal{H}$-Tucker

$$\boxed{A \in \mathcal{H}\text{-Tucker}(k, T),\ k_t := \mathrm{rank}(A^{(t)})}$$

1. Can store A with $dk^3 + dnk$ data, $k \geq k_t$, $n \geq \#\mathcal{I}_j$

[Grasedyck: Hierarchical SVD of Tensors, SIMAX 31(2010)]

SOME PROPERTIES OF $\mathcal{H}$-TUCKER

$$A \in \mathcal{H}\text{-Tucker}(k, T),\ k_t := \mathrm{rank}(A^{(t)})$$

1. Can store A with $dk^3 + dnk$ data, $k \geq k_t$, $n \geq \#\mathcal{I}_j$
2. Can find quasi-best approximation $\tilde{A} \in \mathcal{H}\text{-Tucker}(\tilde{k}, T)$:

$$\left\|A - \tilde{A}\right\| \leq \sqrt{2d-3} \inf_{A' \in \mathcal{H}\text{-Tucker}(\tilde{k}, T)} \left\|A - A'\right\|$$

in complexity $\mathcal{O}(dk^4 + dnk^2)$

[Grasedyck: Hierarchical SVD of Tensors, SIMAX 31(2010)]

SOME PROPERTIES OF $\mathcal{H}$-TUCKER

$$\boxed{A \in \mathcal{H}\text{-Tucker}(k, T),\ k_t := \operatorname{rank}(A^{(t)})}$$

1. Can store A with $dk^3 + dnk$ data, $k \geq k_t$, $n \geq \#\mathcal{I}_j$
2. Can find quasi-best approximation $\tilde{A} \in \mathcal{H}\text{-Tucker}(\tilde{k}, T)$:

$$\left\| A - \tilde{A} \right\| \leq \sqrt{2d - 3} \inf_{A' \in \mathcal{H}\text{-Tucker}(\tilde{k}, T)} \left\| A - A' \right\|$$

   in complexity $\mathcal{O}(dk^4 + dnk^2)$
3. Can evaluate entry of A in $\mathcal{O}(dk^3)$
4. ... several things very fast

[Grasedyck: Hierarchical SVD of Tensors, SIMAX 31(2010)]

Arthmetics of $\mathcal{H}$-Tucker Tensors

Truncation (=finding good low rank approximation) accuracy:

$$\left\| A - \underbrace{A_{\mathcal{H}\text{-Tucker}}}_{\text{computed}} \right\| \leq \sqrt{2d-3} \left\| A - \underbrace{A_{\mathcal{H}\text{-Tucker-best}}}_{\text{best in } \mathcal{H}\text{-format}} \right\|$$

ARTHMETICS OF $\mathcal{H}$-TUCKER TENSORS

Truncation (=finding good low rank approximation) accuracy:

$$\Big\| A - \underbrace{A_{\mathcal{H}\text{-Tucker}}}_{\text{computed}} \Big\| \leq \sqrt{2d-3} \Big\| A - \underbrace{A_{\mathcal{H}\text{-Tucker-best}}}_{\text{best in } \mathcal{H}\text{-format}} \Big\|$$

Truncation: Cut off small singular values/vectors in each $A^{(t)}$.

ARTHMETICS OF $\mathcal{H}$-TUCKER TENSORS

Truncation (=finding good low rank approximation) accuracy:

$$\Big\| A - \underbrace{A_{\mathcal{H}\text{-Tucker}}}_{\text{computed}} \Big\| \le \sqrt{2d-3} \Big\| A - \underbrace{A_{\mathcal{H}\text{-Tucker-best}}}_{\text{best in } \mathcal{H}\text{-format}} \Big\|$$

Truncation: Cut off small singular values/vectors in each $A^{(t)}$.
Example ($n_\mu = 100$ for all modes, random entries)

Input: $\mathcal{H}$-rank k $\rightarrow$ Output: $\mathcal{H}$-rank $\tilde{k} \le k$

k	d=	10	100	1,000	
25	size	1.15	13.59	138.05	MB
25	time	0.24	1.97	19.39	Sec.
50	size	8.03	97.29	989.93	MB
50	time	2.60	30.20	306.02	Sec.
100	size	68.74	755.39	-	MB
100	time	57.05	685.98	-	Sec.

ARTHMETICS OF $\mathcal{H}$-TUCKER TENSORS

Truncation (=finding good low rank approximation) accuracy:

$$\Big\| A - \underbrace{A_{\mathcal{H}\text{-Tucker}}}_{\text{computed}} \Big\| \leq \sqrt{2d-3} \Big\| A - \underbrace{A_{\mathcal{H}\text{-Tucker-best}}}_{\text{best in } \mathcal{H}\text{-format}} \Big\|$$

Truncation: Cut off small singular values/vectors in each $A^{(t)}$.
Example ($n_\mu = 20$ for all modes, random entries)

Input: $\mathcal{H}$-rank k $\rightarrow$ **Output: $\mathcal{H}$-rank $\tilde{k} \leq k$**

k	d=	10,000	100,000	1,000,000	
2	size	3.7	36.6	366.2	MB
2	time	0.2	1.6	15.4	Sec.
5	size	17.2	171.7	-	MB
5	time	0.8	7.6	-	Sec.
10	size	91.5	915.5	-	MB
10	time	5.8	55.9	-	Sec.

Black Box Approximation

Question:

How do we find $u(x, p)$ in this format ?

Black Box Approximation

Question:

How do we find $u(x, p)$ in this format ?

Request:

- Direct black box approximation in $\mathcal{H}$-Tucker
- Use only few samples $u(\cdot, p^j)$), $j = 1, \ldots, M$
- M deterministic 'classic' solves for fixed parameters
- Can allow nonlinearities etc.

Black Box Approximation

Question:

How do we find $u(x, p)$ in this format ?

Request:

- Direct black box approximation in $\mathcal{H}$-Tucker
- Use only few samples $u(\cdot, p^j))$, $j = 1, \ldots, M$
- M deterministic 'classic' solves for fixed parameters
- Can allow nonlinearities etc.
- similar to quasi-MC, non-intrusive

Black Box Approximation

Question:

How do we find $u(x, p)$ in this format ?

Request:

- Direct black box approximation in $\mathcal{H}$-Tucker
- Use only few samples $u(\cdot, p^j)), j = 1, \ldots, M$
- M deterministic 'classic' solves for fixed parameters
- Can allow nonlinearities etc.
- similar to quasi-MC, non-intrusive
- **Samples taken adaptively**

Black Box Approximation

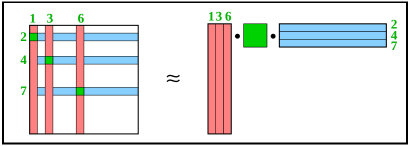
Question:

How do we find $u(x,p)$ in this format ?

Request:

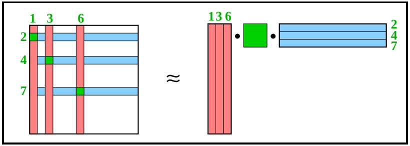
- Direct black box approximation in $\mathcal{H}$-Tucker
- Use only few samples $u(\cdot,p^j)), j = 1,\dots,M$
- M deterministic 'classic' solves for fixed parameters
- Can allow nonlinearities etc.
- similar to quasi-MC, non-intrusive
- **Samples taken adaptively**
- Require approximate separability $p_1,\dots,p_q \leftrightarrow p_{q+1},\dots,p_d$

Skeleton/Cross Approximation



Matrix approximation

Skeleton/Cross Approximation



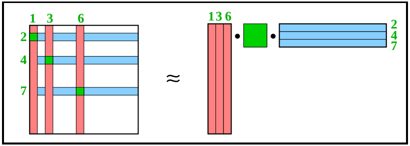
Matrix approximation

$$A \approx A|_{\mathcal{I}_t \times \mathcal{Q}} \cdot S \cdot A|_{\mathcal{P} \times \mathcal{I}_{t'}}$$

Transfertensor

$$S = A|^{-1}_{\mathcal{P} \times \mathcal{Q}}$$

SKELETON/CROSS APPROXIMATION



Matrix approximation

$$A \approx A|_{\mathcal{I}_t \times \mathcal{Q}} \cdot S \cdot A|_{\mathcal{P} \times \mathcal{I}_{t'}}$$

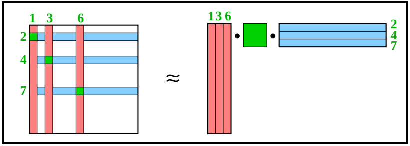
Transfertensor

$$S = A|_{\mathcal{P} \times \mathcal{Q}}^{-1}$$

Properties:

- Exact recovery in case of low rank if $A|_{\mathcal{P} \times \mathcal{Q}}$ invertible

Skeleton/Cross Approximation



Matrix approximation

$$A \approx A|_{\mathcal{I}_t \times \mathcal{Q}} \cdot S \cdot A|_{\mathcal{P} \times \mathcal{I}_{t'}}$$

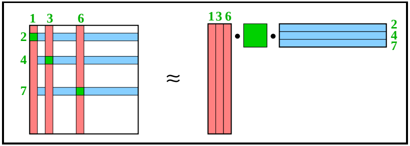
Transfertensor

$$S = A|_{\mathcal{P} \times \mathcal{Q}}^{-1}$$

Properties:

- Exact recovery in case of low rank if $A|_{\mathcal{P} \times \mathcal{Q}}$ invertible
- Adaptive with error estimate
- Heuristic, need to find suitable $\mathcal{P}, \mathcal{Q}$

Skeleton/Cross Approximation



Matrix approximation

$$A \approx A|_{\mathcal{I}_t \times \mathcal{Q}} \cdot S \cdot A|_{\mathcal{P} \times \mathcal{I}_{t'}}$$

Transfertensor

$$S = A|_{\mathcal{P} \times \mathcal{Q}}^{-1}$$

Properties:

- Exact recovery in case of low rank if $A|_{\mathcal{P} \times \mathcal{Q}}$ invertible
- Adaptive with error estimate
- Heuristic, need to find suitable $\mathcal{P}, \mathcal{Q}$
- [Goreinov,Tyrtyshnikov,Zamarashkin]:Maximal Volume

GREEDY PIVOT SEARCH

[Espig/Grasedyck/Hackbusch, Constr.Approx.30, 2009]

$$A \approx A|_{\mathcal{I}_t \times \mathcal{Q}} \cdot S \cdot A|_{\mathcal{P} \times \mathcal{I}_{t'}}, \qquad S = A|_{\mathcal{P} \times \mathcal{Q}}^{-1}$$

Greedy search along fiber-crosses
Properties:

- Evaluate $G(u(\cdot, p_1, \ldots, p_d))$ for all $p_\mu \in \mathcal{I}_\mu$, keep $|\max|$
- Complexity $\mathcal{O}(dnk^2)$
- Get $\mathcal{P}, \mathcal{Q}$
- Do not construct $A|_{\mathcal{I}_t \times \mathcal{Q}}, A|_{\mathcal{P} \times \mathcal{I}_{t'}}$!
- Only the $k \times k$ matrix S is needed

Black Box Approximation in $\mathcal{H}$-Tucker

From S for each $t \in T$ we reconstruct A

Black Box Approximation in $\mathcal{H}$-Tucker

From S for each $t \in T$ we reconstruct A

- complexity $\mathcal{O}(d \log(d) nk^2 + dk^4)$

[Ballani/Grasedyck/Kluge, to appear in Lin.Alg.Appl.]

related approach: Oseledets/Tyrtyshnikov,Savostyanov

Black Box Approximation in $\mathcal{H}$-Tucker

From S for each $t \in T$ we reconstruct A

- complexity $\mathcal{O}(d \log(d) n k^2 + d k^4)$
- nasty error estimate possible
- exact low rank recovery if pivots $\neq 0$

[Ballani/Grasedyck/Kluge, to appear in Lin.Alg.Appl.]
related approach: Oseledets/Tyrtyshnikov,Savostyanov

Black Box Approximation in $\mathcal{H}$-Tucker

From S for each $t \in T$ we reconstruct A

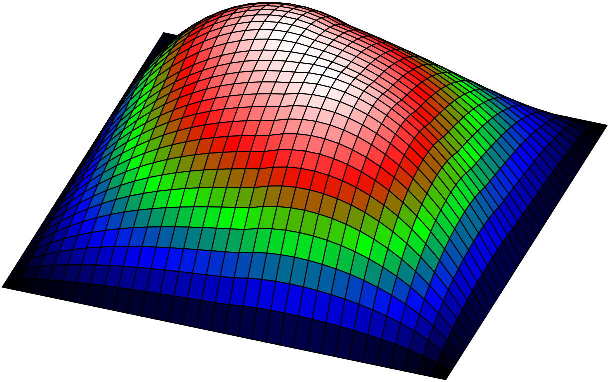
- complexity $\mathcal{O}(d \log(d) nk^2 + dk^4)$
- nasty error estimate possible
- exact low rank recovery if pivots $\neq 0$
- requires $\mathcal{O}(d \log(d) nk^2 + dk^3)$ samples

[Ballani/Grasedyck/Kluge, to appear in Lin.Alg.Appl.]
related approach: Oseledets/Tyrtyshnikov,Savostyanov

Numerical Examples

Example solution of our model problem, $1 \le \sigma(x) \le 2$

$$-\mathrm{div}\, \sigma(x)\, \nabla u(x) = 1 \qquad x \in [-1,1] \times [-1,1]$$



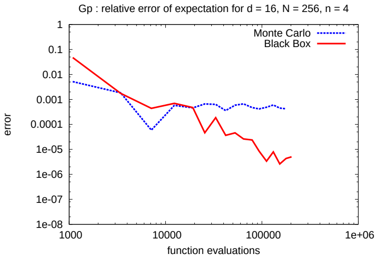
Numerical Examples

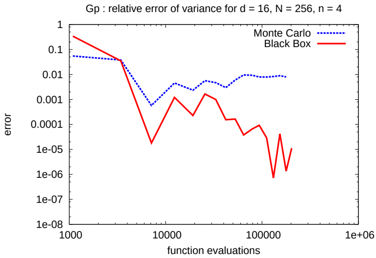
Effective ranks k_{eff} for $\mathcal{G}_p$ with $n = 4$ and $d = 16$

	$N = 16$			$N = 64$		
ε_{BB}	$k_{\text{eff}}^{(\text{BB})}$	k_{eff}	N_{eval}	$k_{\text{eff}}^{(\text{BB})}$	k_{eff}	N_{eval}
2^{-4}	3.9	1.0	22 K	3.9	1.0	22 K
2^{-6}	4.7	1.0	30 K	4.8	1.0	31 K
2^{-8}	5.6	1.3	42 K	5.7	1.3	42 K
2^{-10}	6.5	2.3	57 K	6.5	2.3	55 K
2^{-12}	7.7	2.9	82 K	7.8	2.9	83 K
2^{-14}	10.2	4.0	147 K	10.1	4.0	145 K
2^{-16}	12.8	5.5	250 K	13.1	5.5	265 K
2^{-18}	16.8	7.4	478 K	16.1	7.6	426 K
2^{-20}	22.1	9.7	919 K	22.0	9.9	908 K

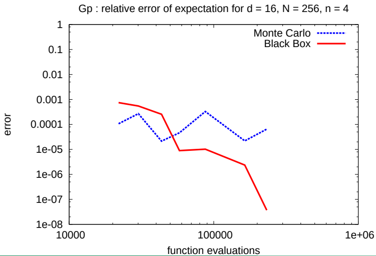
NUMERICAL EXAMPLES: EXP AND VAR OF G_p

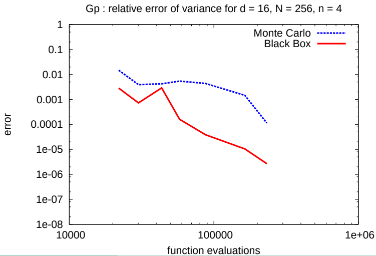
Fixed rank $k = 1, 2, \ldots$ identical in every node





Adaptively determined rank k independently in every node





CONCLUSION

CONCLUSION

1. Data sparse $\mathcal{H}$-Tucker model

CONCLUSION

1. Data sparse $\mathcal{H}$-Tucker model
2. Storage complexity $dnk + dk^3$

CONCLUSION

1. Data sparse $\mathcal{H}$-Tucker model
2. Storage complexity $dnk + dk^3$
3. Black Box approx. possible by fiber cross approximation
   number of samples needed: $\mathcal{O}(d \log(d) nk^2 + dk^3)$

CONCLUSION

1. Data sparse $\mathcal{H}$-Tucker model
2. Storage complexity $dnk + dk^3$
3. Black Box approx. possible by fiber cross approximation
   number of samples needed: $\mathcal{O}(d \log(d) nk^2 + dk^3)$
4. Direct approx. of functionals of SPDE-solutions
   - allows many parameters / stochastic variables
   - need only standard deterministic solvers
   - allows non-linearity, random boundaries/rhs

Reading List

L. Grasedyck:
Hierarchical SVD of Tensors
SIMAX 31 (2010)

J. Ballani, L. Grasedyck, M. Kluge:
Black Box Apx of Tensors in $\mathcal{H}$-Tucker
to appear in Lin.Alg.Appl.

J. Ballani, L. Grasedyck, M. Kluge:
Black Box Solution of SPDEs in $\mathcal{H}$-Tucker
in preparation

http://www.igpm.rwth-aachen.de/grasedyck