

Hierarchical Randomized Low-Rank Approximations (LRA) - Applications to covariance kernel matrices -SIAM LA 2015, Atlanta, GA

**<u>P. Blanchard</u><sup>1</sup>**, O. Coulaud<sup>1</sup>, E. Darve<sup>2</sup>, B. Bramas<sup>1</sup>

[1]: HiePACS team, Inria Bordeaux, FRANCE | [2]: Stanford Mechanical Engineering, Stanford, CA

October 29, 2015

#### Introduction

### Overview

#### Background: Low-Rank Approximation (LRA) of matrices.

- ► Efficient decomposition, square root, products, inverse...
- ► Fast Multipole Methods (FMM) → <u>ScalFMM</u>
- Randomized LRA  $\rightarrow$  <u>FMR</u>



#### Introduction

### Overview

#### Background: Low-Rank Approximation (LRA) of matrices.

- ► Efficient decomposition, square root, products, inverse...
- ► Fast Multipole Methods (FMM) → <u>ScalFMM</u>
- Randomized LRA  $\rightarrow$  <u>FMR</u>

#### Aim: Combining

- ▶ a  $\mathcal{H}^2$ -MVP method
  - ▶ new variant of Black-Box FMM, see [Fong and Darve, 2009].
  - optimized for smooth kernels and heterogenous grids.
- a Randomized LRA
  - ► randomized SVD, see [Halko et al., 2011].
  - MVPs dominate the  $\mathcal{O}(N^2)$  overall cost.
- ... results in an  $\mathcal{O}(N)$  LRA.



#### Covariance matrices

Consider a  $N \times N$  covariance (kernel) matrix C, i.e.

$$C_{ij}=k(r^{ij})$$
, where  $r^{ij}=|\mathbf{x}^i-\mathbf{x}^j|$  and  $\mathbf{x}^i\in\mathbb{R}^3, orall i,j=1\dots N$ 

with k a correlation function, e.g.  $\mathit{Mat\'ern}$  functions  $k_\nu$ 

Exponential:	$k_{1/2}(r)$	$= e^{-r/\ell}$
Gaussian:	$k_{\infty}(r)$	$= e^{-r^2/2\ell^2}$

where  $\ell$  is the (correlation) length scale.

Applications: Large covariance matrices arising from

- Geostatistics: Gaussian Random Fields  $\rightarrow N = \mathcal{O}(10^{5-6})$
- MultiDimensional Scaling (MDS)  $\rightarrow N = \mathcal{O}(10^5)$
- Data Assimilation (DA)  $\rightarrow N = \mathcal{O}(10^{5+})$

### Outline

#### UFMM: A new efficient $\mathcal{H}^2$ -MVP method

Fundamentals & Terminology Fast Lagrange interpolation using FFT Comparative analysis

#### Randomized Low-Rank Approximations (LRA)

A standard Randomized LRA algorithm Randomized SVD powered by  $\mathcal{H}^2\text{-}\mathsf{MVPs}$  Comparative analysis

#### Scientific applications



# 1 The Uniform FMM (UFMM) a new efficient $\mathcal{H}^2$ -MVP method

naía

PIERRE BLANCHARD - FFT-ACCELERATED FMM

October 29, 2015- 5

UFMM: A new efficient  $\mathcal{H}^2$ -MVP method Fundamentals & Terminology

#### A Hierarchical MVP Algorithm: the FMM

<u>Goal</u>: Compute  $q_i = \sum_{j \leq N} K_{ij} \omega_j$  for all  $i \leq N$  in  $\mathcal{O}(N)$ , with  $K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$  and particle  $P_i(\mathbf{x}_i, w_i, q_i)$  located at  $\mathbf{x}_i$ .



Figure : Hierarchical tree structure (1D) and associated FMM steps.

nnía

### Interpolation based FMM

Kernel independent polynomial expansion of order p:

$$k(\mathbf{x}, \mathbf{y}) \approx \sum_{|\boldsymbol{\alpha}| \le p} S(\bar{\mathbf{x}}_{\boldsymbol{\alpha}}, \mathbf{x}) \sum_{|\boldsymbol{\beta}| \le p} k(\bar{\mathbf{x}}_{\boldsymbol{\alpha}}, \bar{\mathbf{y}}_{\boldsymbol{\beta}}) S(\bar{\mathbf{y}}_{\boldsymbol{\beta}}, \mathbf{y})$$

Chebyshev grid:

Equispaced grid:



 $\rightarrow$  Black-Box FMM (bbfmm) [Fong and Darve, 2009]

 $\bar{\mathbf{x}}_{\alpha}$ 

 $\rightarrow$  Uniform FMM (ufmm) [Blanchard et al., 2015] (preprint)

NB: Lagrange interpolation unstable as p grows. However, high accuracy are often not required and instabilities can always be regularized.



## Fast Lagrange interpolation using FFT

M2L operators  $\bar{K}_{\alpha\beta} = k(\bar{\mathbf{x}}_{\alpha}, \bar{\mathbf{y}}_{\beta})$  are *Toeplitz* since

$$\blacktriangleright k(\mathbf{x}, \mathbf{y}) = k(|\mathbf{x} - \mathbf{y}|)$$

• Grid  $(\bar{\mathbf{x}}_{\boldsymbol{\alpha}}, \bar{\mathbf{y}}_{\boldsymbol{\beta}})$  is equispaced

M2L optimization based on FFT in 1D:

• Write M2L in convolution form  $\Rightarrow$  Storage in  $\mathcal{O}(p)$ 

Let 
$$\bar{K} = \begin{bmatrix} a & b & c \\ d & a & b \\ e & d & a \end{bmatrix} \in \mathbb{R}^{p \times p}$$
,  $E(\bar{K}) = \begin{bmatrix} e & d \\ \bar{K} & c & e \\ b & c \\ c & e & d & a \\ b & c & e & d & a \end{bmatrix} \in \mathbb{R}^{2p-1 \times 2p-1}$ 



## Fast Lagrange interpolation using FFT

M2L operators  $\bar{K}_{\alpha\beta} = k(\bar{\mathbf{x}}_{\alpha}, \bar{\mathbf{y}}_{\beta})$  are *Toeplitz* since

$$\blacktriangleright k(\mathbf{x}, \mathbf{y}) = k(|\mathbf{x} - \mathbf{y}|)$$

• Grid  $(\bar{\mathbf{x}}_{\boldsymbol{\alpha}}, \bar{\mathbf{y}}_{\boldsymbol{\beta}})$  is equispaced

M2L optimization based on FFT in 1D:

• Write M2L in convolution form  $\Rightarrow$  Storage in  $\mathcal{O}(p)$ 

Let 
$$\bar{K} = \begin{bmatrix} a & b & c \\ d & a & b \\ e & d & a \end{bmatrix} \in \mathbb{R}^{p \times p}, E(\bar{K}) = \begin{bmatrix} a & b & c & e & d \\ d & a & b & c & e \\ e & d & a & b & c \\ c & e & d & a & b \\ b & c & e & d & a \end{bmatrix}$$

$$\in \mathbb{R}^{2p-1 \times 2p-1}$$



## Fast Lagrange interpolation using FFT

M2L operators  $\bar{K}_{\alpha\beta} = k(\bar{\mathbf{x}}_{\alpha}, \bar{\mathbf{y}}_{\beta})$  are *Toeplitz* since

$$\blacktriangleright \ k(\mathbf{x}, \mathbf{y}) = k(|\mathbf{x} - \mathbf{y}|)$$

• Grid  $(\bar{\mathbf{x}}_{\boldsymbol{\alpha}}, \bar{\mathbf{y}}_{\boldsymbol{\beta}})$  is equispaced

M2L optimization based on FFT in 1D:

• Write M2L in convolution form  $\Rightarrow$  Storage in  $\mathcal{O}(p)$ 

Let 
$$\bar{K} = \begin{bmatrix} a & b & c \\ d & a & b \\ e & d & a \end{bmatrix} \in \mathbb{R}^{p \times p}$$
,  $E(\bar{K}) = \begin{bmatrix} \bar{R} \\ d & a & b & c \\ e & d & a & b \\ c & e & d & a \\ b & c & e & d & a \end{bmatrix} \in \mathbb{R}^{2p-1 \times 2p-1}$ 

▶ Build in  $\mathcal{O}(p \log p) / \text{Apply in } \mathcal{O}(p)$  flops in Fourier domain:

$$\bar{K}X = FFT^{-1}(FFT(\bar{R}) \odot FFT(X))$$

# UFMM vs. BBFMM: [Accuracy, Memory] = f(p)

Similar accuracies, while UFMM significantly reduces storage.

Kernel: k(x, y) = 1/|x - y| - Distrib.: Cube (volumic) #particles: N = 20k - #levels: h = 5



NB: See [Messner et al., 2012] for M2L optimized bbfmm (comp. and sym.).

PIERRE BLANCHARD - Hierarchical Randomized LRA

## UFMM vs. BBFMM: [Time] = f(p)

UFMM substantially faster than generic variants of BBFMM.

Kernel: k(x, y) = 1/|x - y| - Distrib.: Cube (volumic) #particles: N = 20k - #levels: h = 5



NB: See [Messner et al., 2012] for M2L optimized bbfmm (comp. and sym.).

PIERRE BLANCHARD - Hierarchical Randomized LRA

# UFMM: [Time] = f(N)

Smooth UFMM: Approximate nearfield for smooth k, i.e. no P2P.

Kernel: Gaussian  $k(x, y) = k_{\infty}(x, y)$  ( $\ell = 0.5$ ) - Distrib.: Cube Accuracy: Choose  $p / \varepsilon_{FMM} < 10^{-3}$ 





# UFMM: [Time] = f(N)

Smooth UFMM: Approximate nearfield for smooth k, i.e. no P2P.

Kernel: Gaussian  $k(x, y) = k_{\infty}(x, y)$  ( $\ell = 0.5$ ) - Distrib.: Cube Accuracy: Choose  $p \ / \ \varepsilon_{FMM} < 10^{-3}$ 





# 2 Randomized Low-Rank Approximations

Standard & hierarchical variants.

nín

October 29, 2015

### Randomized SVD

<u>Reminder</u>: Covariance kernel matrix  $C_{ij} = k(|\mathbf{x}^i - \mathbf{x}^j|), \ \mathbf{x}^{i,j} \in \mathbb{R}^3$ .

Algorithm: [Halko et al., 2011] Overall complexity  $\mathcal{O}(\mathbf{r} \times \mathbf{N}^2)$ 

Step I - Build  $Q \in \mathbb{R}^{N \times r} / Q \approx range(C)$  using random projection.

- 1) Draw  $X \in \mathbb{R}^{N \times r}$ , *i.e.* r random vectors.
- 2) Compute  $Y \in \mathbb{R}^{N \times r}$ :  $Y = \mathbf{CX}$ .  $\mathcal{O}(\mathbf{r} \times \mathbf{N^2})$  $\mathcal{O}(r^2 \times N)$
- 3) Perform QR decomposition of Y: QR = Y

Step II - Approximate 
$$C^{1/2}$$
 (or decompose  $C$ ) using SVD of  $Q^t CQ$ .  
1) Build  $B = Q^t \mathbb{C} \mathbb{Q} \in \mathbb{R}^{r \times r}$ .  
2) Compute  $B^{1/2} = U_B \Sigma_B^{1/2}$  using SVD  $(B = U_B \Sigma_B U'_B)$ .  
3) Form  $C^{1/2} = QB^{1/2} \in \mathbb{R}^{N \times r}$   
 $\mathcal{O}(N \times r^2)$ 

*NB:*  $C \approx Q (Q^{*}CQ) Q^{*}$  (low-rank representation)

# Randomized SVD powered by $\mathcal{H}^2$ -MVPs

<u>Reminder</u>: Covariance kernel matrix  $C_{ij} = k(|\mathbf{x}^i - \mathbf{x}^j|)$ ,  $\mathbf{x}^{i,j} \in \mathbb{R}^3$ .

Algorithm: [Halko et al., 2011] Overall complexity  $\mathcal{O}(\mathbf{r}^2 \times \mathbf{N})$ 

Step I - Build  $Q \in \mathbb{R}^{N \times r} / Q \approx range(C)$  using random projection.

- 1) Draw  $X \in \mathbb{R}^{N \times r}$ , *i.e.* r random vectors.
- 2) Compute  $Y \in \mathbb{R}^{N \times r}$ :  $Y = \mathbf{FMM}(\mathbf{CX})$ .  $\mathcal{O}(\mathbf{r} \times \mathbf{N})$
- 3) Perform QR decomposition of Y: QR = Y  $\mathcal{O}(r^2 \times N)$

#### Step II - Approximate $C^{1/2}$ (or decompose C) using SVD of $Q^t CQ$ . 1) Build $B = Q^t \mathbf{FMM}(\mathbf{CQ}) \in \mathbb{R}^{r \times r}$ . 2) Compute $B^{1/2} = U_B \Sigma_B^{1/2}$ using SVD $(B = U_B \Sigma_B U'_B)$ . 3) Form $C^{1/2} = QB^{1/2} \in \mathbb{R}^{N \times r}$ . $\mathcal{O}(N \times r^2)$

NB:  $C \approx Q \left( Q^t C Q \right) Q^t$  (low-rank representation)

## Fixed accuracy ( $\varepsilon$ ) vs fixed rank (r) RandSVD.

RandSVD powered by dense BLAS::gemm() or  $\mathcal{H}^2$ -MMPs

Correlation: Gaussian  $\ell=0.5$  – Distrib.: Unit Sphere

Accuracy: Choose  $p \ / \ \varepsilon_{FMM} < \varepsilon / 10 = 10^{-3}$ 





# **4** Scientific Applications Ongoing works & Perspectives

nnia

PIERRE BLANCHARD - Randomized LRA

October 29, 2015

Scientific applications

### Geostatistics: Gaussian Random Fields simulation

<u>Goal</u>: Generate  $n_{real}$  realizations  $Y \sim \mathcal{N}(0, C)$ 

<u>Method</u>: Form  $Y = C^{1/2}X$  with  $X \sim \mathcal{N}(0, I_d)$  (white noise)

 $\begin{array}{l} \mbox{Correlation: Gaussian } \ell = 0.25 - \mbox{Distrib.: Unit Sphere } N = 72k \\ \mbox{RandSVD: } \varepsilon = 10^{-2} - \mbox{UFMM: } \varepsilon_{FMM} < \varepsilon/10 \end{array}$ 



$n_{real}$	$\ \tilde{C} - C\ _{2,rel}$	$\ \tilde{C} - C\ _{\infty, rel}$
$1 \cdot 10^{3}$	$2.48 \cdot 10^{-1}$	$1.47 \cdot 10^{-1}$
$1\cdot 10^4$	$7.88 \cdot 10^{-2}$	$4.53\cdot10^{-2}$
$1 \cdot 10^{5}$	$2.63 \cdot 10^{-2}$	$1.57 \cdot 10^{-2}$
$1\cdot 10^6$	$1.02 \cdot 10^{-2}$	$8.53\cdot10^{-3}$

$$\tilde{C} = \frac{1}{n_{real}} \sum_{i=1}^{n_{real}} (Y_i - \mathbb{E}(Y_i))(Y_i - \mathbb{E}(Y_i))^t$$

 $r = 210, r/N = 0.3\%, t_{BLAS} = 15$ min,  $t_{UFMM} = 1$ min,  $t_{SUFMM} = 25$ s

Scientific applications

# Ongoing works & Perspectives

#### Further developments:

- Use parallel FMM (ScalFMM)
- Covariance known algebraically, i.e. general H<sup>2</sup>-matrices

#### MultiDimensional Scaling (MDS):

- classification of tree species
- Reconstruct grid given a distance matrix (i.e. build C<sup>1/2</sup>)
- RandLRA lead to promising results
- Benefit from clustering of data

#### Data Assimilation (DA):

- Low-rank Ensemble Kalman Filters
- Preconditioned variationnal DA





#### Biblio



Blanchard, P., Darve, E., and Coulaud, O. (2015). Fast hierarchical algorithms for generating gaussian random fields. (preprint).



Fong, W. and Darve, E. (2009). The black-box fast multipole method. Journal of Computational Physics, 228(23):8712–8725.



Halko, N., Martinsson, P.-G., and Tropp, J. A. (2011). Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM review*, 53(2):217–288.



Messner, M., Bramas, B., Coulaud, O., and Darve, E. (2012). Optimized m2l kernels for the chebyshev interpolation based fast multipole method.

*CoRR*, abs/1210.7292.

