

# Иерархические матрицы и их применения (часть I)

Ричард Гржибовский  
Университет Саарланда, Германия  
richards@num.uni-sb.de

НГУ, 27.03.2013



- 1 Мотивация: пример безсеточной интерполяции
- 2 Постановка задачи аппроксимации
- 3 Сведения из линейной алгебры
  - матрицы с рангом  $r$
  - сингулярное разложение
  - норма матрицы
- 4 Аппроксимация матрицами с низким рангом
- 5 аппроксимация дегенерированными функциями
- 6 Перестановка (кластеризация)
- 7 Иерархическая кластеризация и дерево кластеров
- 8 Иерархические матрицы



**Дано:** Точки и значения функции

$$(x_i)_{i=1}^N \subset \mathbb{R}^n, \quad f(x_i) = f_i.$$

**Найти:** Интерполянт (или приближение)  $s(x)$  для  $f(x)$ .

**Решение:** *Интерполяция радиальными функциями (RBF-interpolation)*

Ищем  $s(x)$  в форме

$$s(x) = \sum_{i=1}^N a_i \varphi(\|x - x_i\|) + P_m(x),$$

где  $\varphi : [0, \infty) \rightarrow \mathbb{R}$  – базисная функция (базовая функция),  
 $P_m(x) = \sum_{k=1}^M c_k q_k(x)$  – многочлен порядка  $m - 1$ ,  
 $a_i, c_k$  – неизвестные коэффициенты.



Из условий интерполяции

$$\begin{cases} s(x_k) = f_k, & k = 1, \dots, N \\ \sum_{k=1}^N a_k q_j(x_k) = 0, & j = 1, \dots, M \end{cases}$$

получаем систему линейных уравнений

$$\begin{pmatrix} \Phi & Q \\ Q^T & 0 \end{pmatrix} \begin{pmatrix} a \\ c \end{pmatrix} = \begin{pmatrix} f \\ 0 \end{pmatrix}$$

где

$$\Phi_{ij} = \varphi(\|x_i - x_j\|), \quad i, j = 1 \dots N$$

$$Q_{kj} = q_j(x_k), \quad k = 1, \dots, N, \quad j = 1, \dots, M$$



Каким образом выбрать  $\varphi$  и  $m$ , чтобы система была однозначно разрешена?

- 1 Обобщённое преобразование Фурье  $\hat{\varphi}$  должно быть
  - 1 порядка  $m$ ,
  - 2 непрерывно на  $\mathbb{R}^n \setminus \{0\}$ ,
  - 3 неотрицательно,
  - 4 не обращаться в 0.
- 2 Множество точек  $x_1, \dots, x_N$  должно быть разрешимо относительно многочленов порядка  $(m - 1)$ .

## Замечания

В случае радиальных функций, решающую роль играет их преобразование Бесселя.

Требование 2 почти всегда выполнено.



## Примеры

### 1 Мультиквадрик (Multiquadrics):

$$\varphi(x) = (\|x\|^2 + \alpha^2)^\beta, \quad \beta > 0, \quad \beta \notin \mathbb{N}, \quad m = \lceil \beta \rceil$$

В частности:  $s(x) = \sum_{i=1}^N \sqrt{\|x - x_i\|^2 + \alpha^2} + c, \quad \alpha > 0.$

### 2 Степени:

$$\varphi(x) = \|x\|^\beta, \quad \beta > 0, \quad \beta \notin 2\mathbb{N}, \quad m = \lceil \beta/2 \rceil$$

### 3 Сплайны (Thin Plate Splines (TPS)):

$$\varphi(x) = \|x\|^{2\beta} \log \|x\|, \quad \beta \in \mathbb{N}, \quad m = \beta + 1$$

В частности:  $s(x) = \sum_{i=1}^N \|x - x_i\|^2 \log \|x - x_i\| + P_2(x)$



## Свойства множества данных

$$h_X = \min_i \min_{j \neq i} \|x_i - x_j\|$$

$$H_X = \max_i \min_{j \neq i} \|x_i - x_j\|$$

## Оценки точности интерполяции

Типичные оценки в  $L^2$  and  $L^\infty$

(в том числе и для производных):

$$\|f(x) - s(x)\| \leq C_2 H_X^m$$

## Свойства линейной системы

- Матрица  $\Phi$  является полной!
- Обусловленность системы достаточно плохая. Например для сплайнов (TPS)  $\text{cond}A \leq C_1 N \phi(\text{diam } X) h_X^{-2}$ .



Рассмотрим полные матрицы вида  $A = \{a_{ij}\}_{i,j=1}^N$

- 1  $a_{ij} = K(x_i, y_j)$  интерполяция, МГЭ Нюстрёма
- 2  $a_{ij} = \int_{T_i} K(x, y_j) \varphi_i(x) ds_x$  коллокационный МГЭ
- 3  $a_{ij} = \int_{T_i} \int_{T_j} K(x, y) \varphi_i(x) \varphi_j(y) ds_x ds_y$  МГЭ Галёркина

С каким количеством неизвестных справляются ЭВМ?

Проблема объёма памяти:  $\text{Mem}(A) = O(N^2)$

$N$	$10^3$	$5 \cdot 10^3$	$10 \cdot 10^3$	$20 \cdot 10^3$	$50 \cdot 10^3$
$\text{Mem}(A)$	7.6 MB	190.7 MB	763 MB	3 GB	19 GB

Проблема сложности вычисления:

Решая линейную систему итеративными методами, вычисление произведения полной матрицы на вектор имеет сложность  $O(N^2)$ !



# Основная идея: малоранговое представление

Пусть

$$\{x_i\}_{i=1}^m \subset X \subset \mathbb{R}^d, \quad \{y_j\}_{j=1}^n \subset Y \subset \mathbb{R}^d$$

два множества точек, и функция  $K : X \times Y \rightarrow \mathbb{R}$  имеет вид

$$K(x, y) = \sum_{k=1}^r u_k(x) v_k(y)$$

$u_k : X \rightarrow \mathbb{R}$  и  $v_k : Y \rightarrow \mathbb{R}$  для всех  $k = 1, \dots, r$ .

Тогда матрица  $A$  с элементами  $A_{ij} = K(x_i, y_j)$  имеет вид

$$A = UV^T \text{ где } U \in \mathbb{R}^{m \times r}, \quad V \in \mathbb{R}^{n \times r}.$$

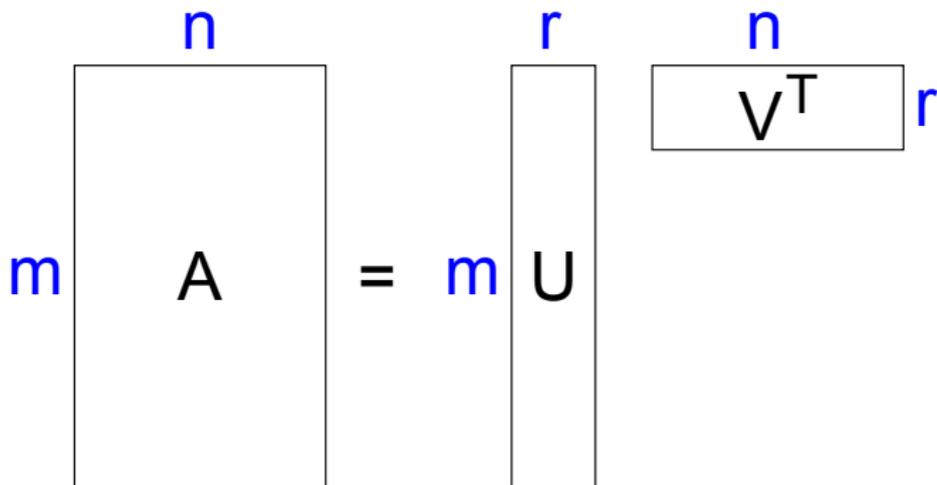
$$U_{ik} = u_k(x_i), \quad V_{jk} = v_k(y_j).$$

Важное свойство:  $r$  не зависит от  $m$  и  $n$ .



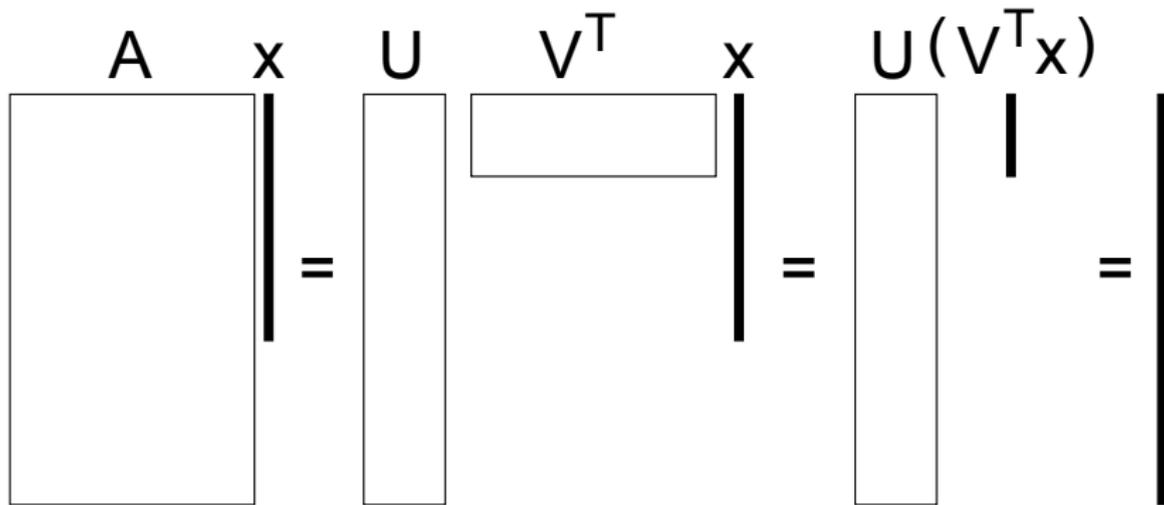
# Малоранговое представление $\rightarrow$ линейные затраты!

$$\text{Mem}(A) = \text{Mem}(U) + \text{Mem}(V) = r(m+n),$$



# Малоранговое представление $\rightarrow$ линейные затраты!

$$\text{Compl}(Ax) = \text{Compl}(V^T x) + \text{Compl}(U(V^T x)) = r(m+n)$$



# Малоранговое представление $\rightarrow$ линейные затраты!

Если

$$A = UV^T \text{ где } U \in \mathbb{R}^{m \times r}, \quad V \in \mathbb{R}^{n \times r}.$$

то

$$\text{Mem}(A) = \text{Mem}(U) + \text{Mem}(V) = r(m+n),$$

а также

$$Ax = UV^T x = U(V^T x), \quad x \in \mathbb{R}^n,$$

$$\text{Compl}(Ax) = \text{Compl}(V^T x) + \text{Compl}(U(V^T x)) = r(m+n)$$



# Малоранговая аппроксимация дегенерированными функциями

Пусть  $K$  может быть представлена как

$$K(x, y) = \sum_{k=1}^r u_k(x) v_k(y) + R_{r(\varepsilon)}(x, y),$$

где остаточный член  $R_{r(\varepsilon)}$  экспоненциально стремится к нулю

$$\|R_r\| \leq Cq^{r^\alpha} \|K\|, \quad r \rightarrow \infty, \text{ и } C > 0, \alpha > 0, q > 0 \text{ постоянные.}$$

Тогда для достижения точности  $\varepsilon$  необходима аппроксимация ранга

$$r(\varepsilon) = O\left(\ln \frac{1}{\varepsilon}\right)^{1/\alpha}$$

На практике  $\varepsilon$  зависит от дискретизации  $n, m$ . Пусть

$$n \sim m, \quad \varepsilon = O\left(n^{-\beta}\right), \quad \beta > 0, \quad \text{тогда } \text{Mem}(A) = O\left(n(\ln n)^{1/\alpha}\right).$$



# Пример: ряд Тейлора

$$K(x, y) = \sum_{\alpha: |\alpha| \leq p} \frac{1}{\alpha!} \partial_x^\alpha K(x^*, y) (x - x^*)^\alpha + R_r(x, y).$$

где  $\alpha = (\alpha_1, \dots, \alpha_d)$  мульти-индекс,

$$\alpha! = \alpha_1! \cdot \dots \cdot \alpha_d!, \quad (x - x^*) = (x_1 - x_1^*)^{\alpha_1} \cdot \dots \cdot (x_d - x_d^*)^{\alpha_d},$$

и  $x^*$  фиксирован.

Количество слагаемых  $r$

$$r = \binom{p+d}{d} \Rightarrow r \sim p^d.$$



Для сходимости ряда Тейлора достаточно двух условий:

- 1 асимптотическая гладкость  $K$

$$\partial_x^\alpha \partial_y^\beta K(x, y) \leq C q! \gamma^q |x - y|^{-k} |K(x, y)|,$$

$$q = |\alpha| + |\beta|, C, \gamma \in \mathbb{R}.$$

- 2 разделённость множеств  $X$  и  $Y$

$$\min \{ \text{diam } X, \text{diam } Y \} < \eta \text{dist}(X, Y), \eta \in (0, 1).$$

$$\text{Тогда } \|R_r\| \leq C \frac{\gamma \sqrt{d} \eta}{1 - \gamma \sqrt{d} \eta} (\gamma \sqrt{d} \eta)^p \|K\|,$$

где  $\|K\| = \max_{x \in X} \max_{y \in Y} |K(x, y)|$ .

Таким образом  $r(\varepsilon) = O(\ln \varepsilon)^d$ .



Рассмотрим матрицу  $A$ ,

$$A_{ij} = K(x_i, y_j), \quad 1 \leq i \leq m, \quad 1 \leq j \leq n, \quad x_i \in X, \quad y_j \in Y,$$

где  $K$  асимптотически гладкая,  
множества  $X, Y$  хорошо разделены.

Существует малоранговая аппроксимация матрицы.

## Задача

Как найти эту аппроксимацию,  
используя только элементы  $A_{ij}$ ?

Ответ: сингулярное разложение.



Пусть  $A = U\Sigma V^T$  сингулярное разложение матрицы  $A$ , и

$$A_k = U\Sigma_k V^T, \Sigma_k = \text{diag}(\sigma_1, \dots, \sigma_k, 0, \dots, 0).$$

Тогда для любой унитарно-инвариантной нормы

$$\min_{B: \text{rank} B = k} \|A - B\| = \|A - A_k\| = \|\Sigma - \Sigma_k\|.$$

Таким образом

$$\|A - A_k\|_2 = \sigma_{k+1}, \quad \|A - A_k\|_F = \sqrt{\sum_{s>k} \sigma_s^2}$$



## Как найти малоранговую аппроксимацию?

- 1 Сингулярное разложение  $A = U\Sigma V^T$ , выбрать  $A_k$ , где  $\sigma_{k+1} < \varepsilon\sigma_1$ .
  - ⊕  $\text{Mem}(A_k) = O(n \log n)$ .
  - ⊕  $A_k$  самая лучшая аппроксимация ранга  $k$ !
  - ⊖  $\text{Comp}(A_k) = O(n^3)$ .
- 2 Адаптивная крестовая аппроксимация (Adaptive Cross Approximation (ACA))
  - ⊕  $\text{Mem}(A_k) = O(n \log^\alpha n)$ .
  - ⊕  $\text{Comp}(A_k) = O(n \log^\alpha n)$ .



Пусть  $A \in \mathbb{R}^{m \times n}$ .

1. Инициализация

$$R_0 = A, \quad S_0 = 0.$$

2. Для  $i = 0, 1, 2, \dots$ , пока  $\|R_i\|_F > \varepsilon \|A\|_F$

2.1. пивотный элемент

$$(k_{i+1}, \ell_{i+1}) = \text{ArgMax } |(R_i)_{kl}|,$$

2.2. постоянная нормировки

$$\gamma_{i+1} = ((R_i)_{k_{i+1}\ell_{i+1}})^{-1},$$

2.3. новые вектора

$$u_{i+1} = \gamma_{i+1} R_i e_{\ell_{i+1}}, \quad v_{i+1} = R_i^\top e_{k_{i+1}},$$

2.4. новая невязка

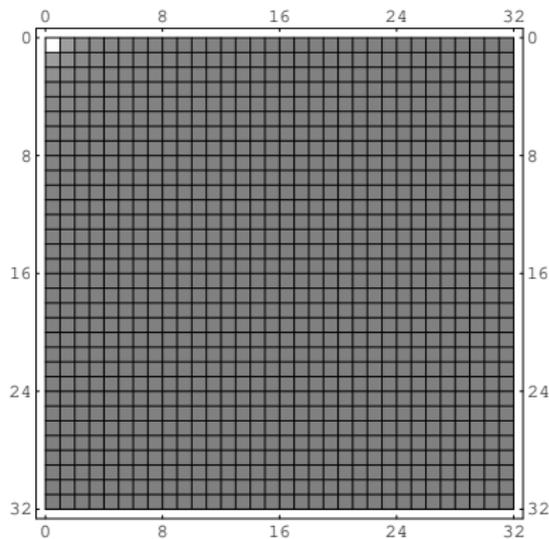
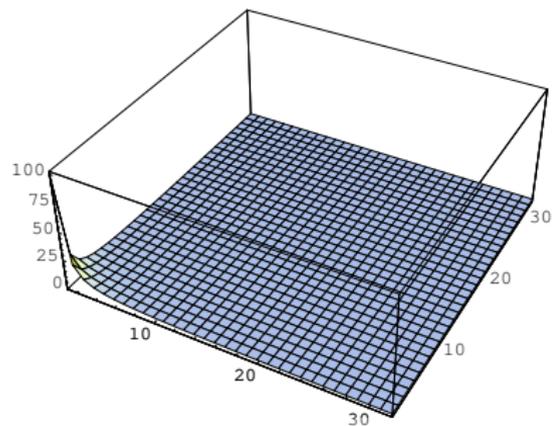
$$R_{i+1} = R_i - u_{i+1} v_{i+1}^\top,$$

2.5. новая аппроксимация

$$S_{i+1} = S_i + u_{i+1} v_{i+1}^\top.$$



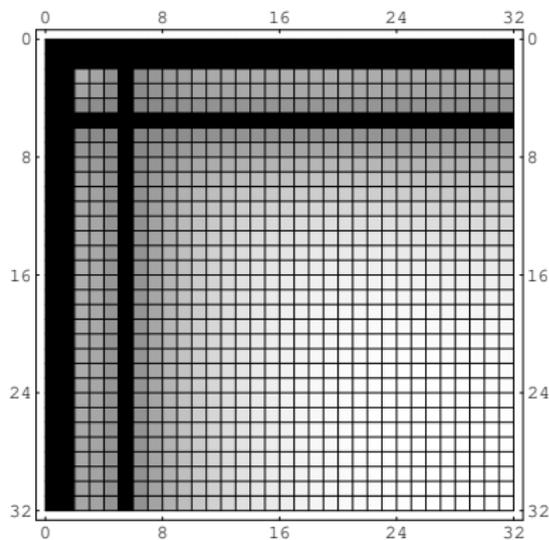
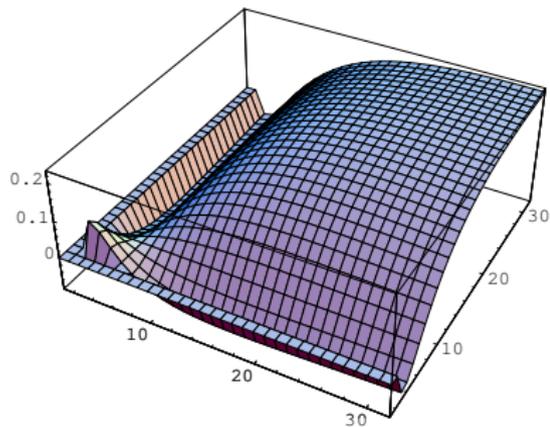
$$A_{ij} = 1 / (0.01 + i + j), \quad m = n = 32$$



Изначальная невязка



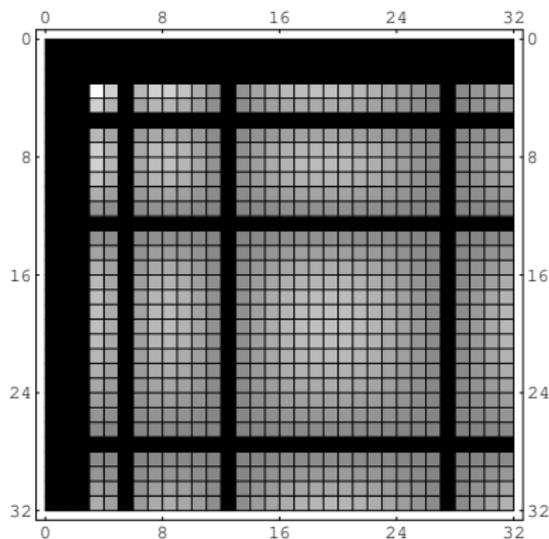
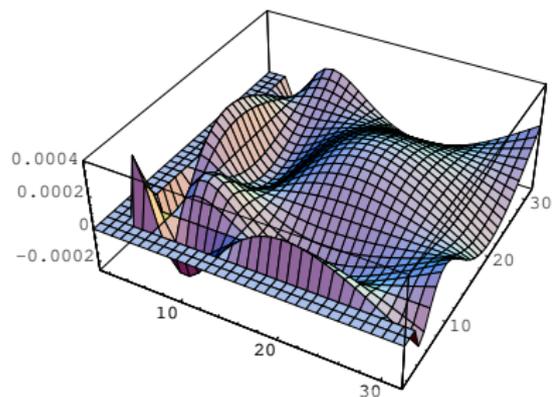
$$A_{ij} = 1 / (0.01 + i + j), \quad m = n = 32$$



невязка  $R_3$



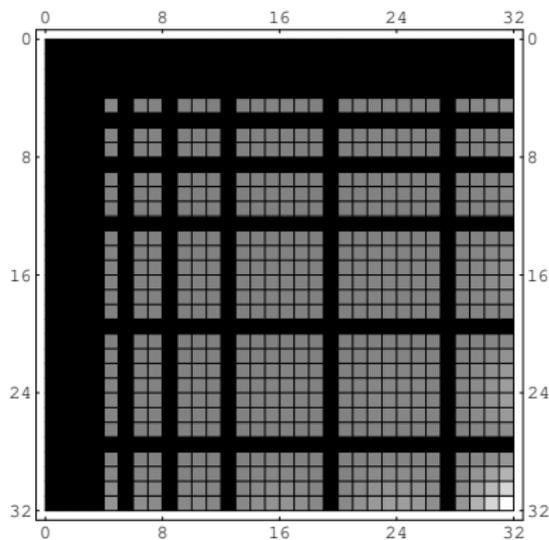
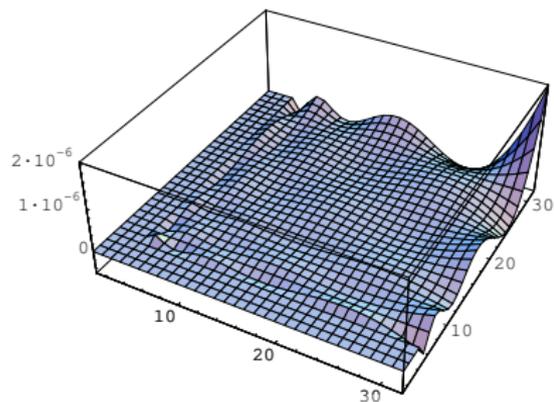
$$A_{ij} = 1 / (0.01 + i + j), \quad m = n = 32$$



невязка  $R_6$



$$A_{ij} = 1 / (0.01 + i + j), \quad m = n = 32$$



невязка  $R_9$



$$A_{ij} = 1 / (0.01 + i + j), \quad m = n = 32$$

Step	Pivot row	Pivot column	Pivot value	Relative error
1	1	1	$1.00 \cdot 10^{+2}$	$3.43 \cdot 10^{-1}$
2	2	2	$7.91 \cdot 10^{+0}$	$1.62 \cdot 10^{-1}$
3	6	6	$1.10 \cdot 10^{+0}$	$3.66 \cdot 10^{-2}$
4	28	28	$2.25 \cdot 10^{-1}$	$2.26 \cdot 10^{-3}$
5	3	3	$6.10 \cdot 10^{-2}$	$8.40 \cdot 10^{-4}$
6	13	13	$9.87 \cdot 10^{-3}$	$2.28 \cdot 10^{-5}$
7	4	4	$3.91 \cdot 10^{-4}$	$8.85 \cdot 10^{-6}$
8	20	20	$1.02 \cdot 10^{-4}$	$2.69 \cdot 10^{-7}$
9	9	9	$6.32 \cdot 10^{-6}$	$3.30 \cdot 10^{-8}$
10	32	32	$1.97 \cdot 10^{-6}$	$1.13 \cdot 10^{-9}$

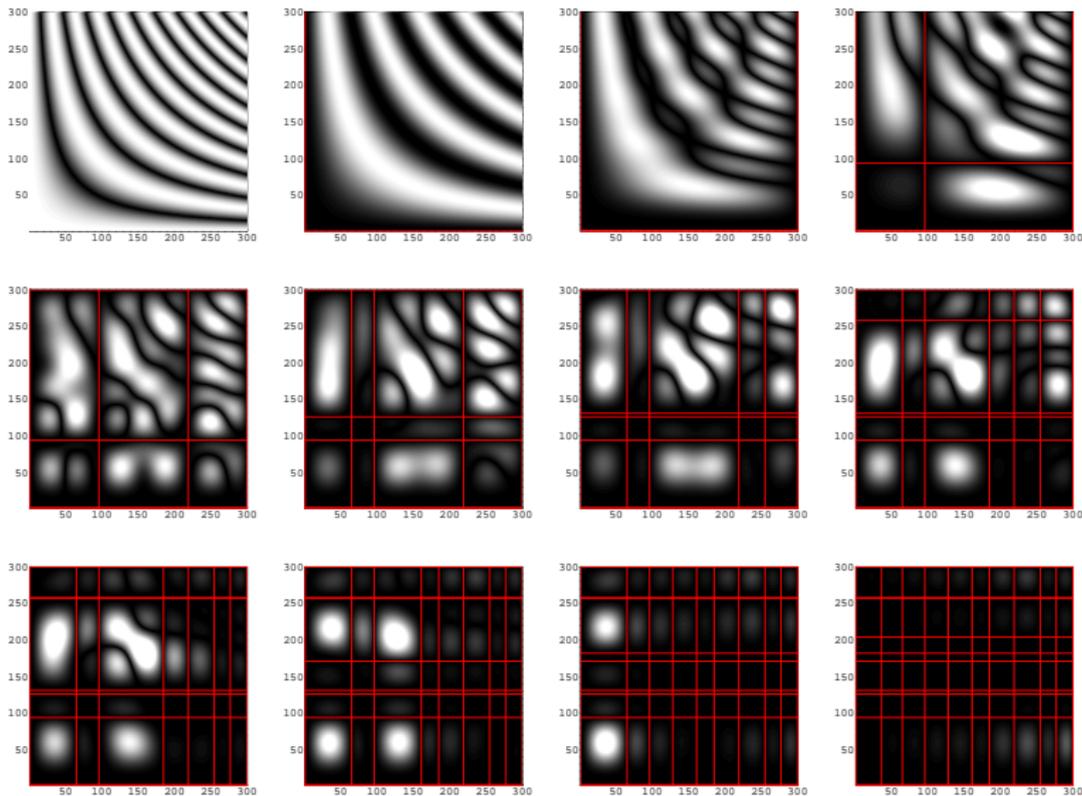


В более практичной версии алгоритма пивотный элементы выбираются среди уже вычисленных элементов матрицы.

Такая версия называется крестовая аппроксимация с частичным пивотированием (partially pivoted ACA algorithm).



# Пример



## Свойства АСА

- 1 АСА сходится для
  - 1 матриц интерполяции/МГЭ Ньюстрёма (Bebendorf '00)
  - 2 матриц МГЭ коллокации (Bebendorf, Rjasanow '03)
  - 3 матриц МГЭ Галёркина (Bebendorf, GRZ '05)
- 2 в АСА вычисляются далеко не все элементы матрицы!
- 3 АСА алгебраический



-  M. Bebendorf.  
Approximation of boundary element matrices..  
*Numerische Mathematik*, 86 (2000), no. 4, 565–589.
-  M. Bebendorf, S. Rjasanow.  
Adaptive low-rank approximation of collocation matrices.  
*Computing*, 70 (2003), no. 1, 1–24.
-  M. Bebendorf, R. Grzhibovskis.  
Accelerating Galerkin BEM for linear elasticity using adaptive cross approximation.  
*Mathematical Methods in the Applied Sciences*, 29 (2006), no. 14, 1721–1747.

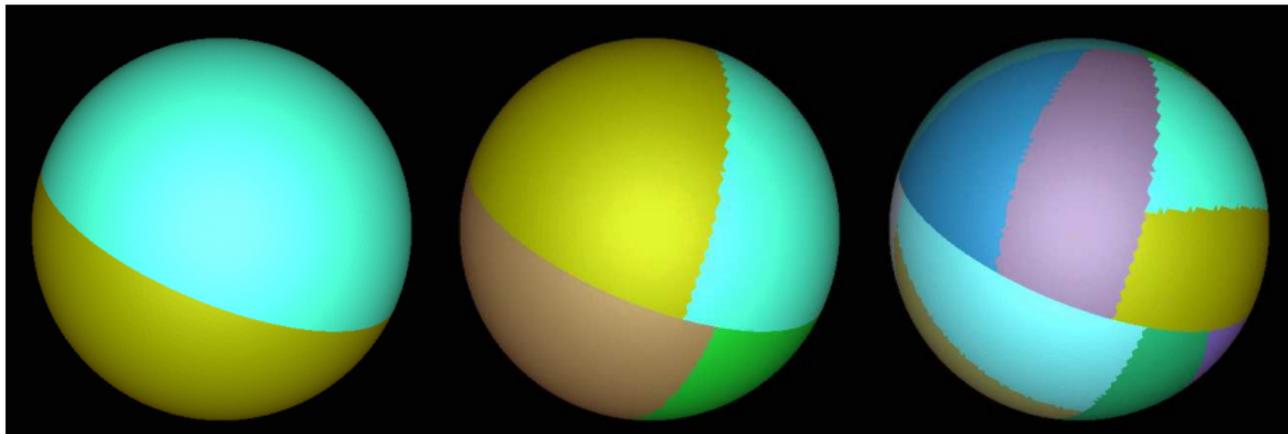


## Сингулярность при $x \rightarrow y$

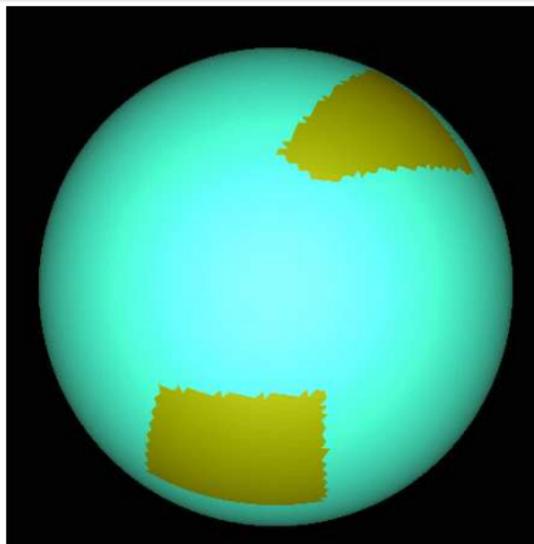
$$K(x, y) = \frac{1}{4\pi} \frac{1}{|x - y|}, \quad x, y \in \mathbb{R}^3.$$

- Конструкция дерева кластеров
- Пары кластеров
- Иерархическая матрица
- АСА-аппроксимация отдельных блоков



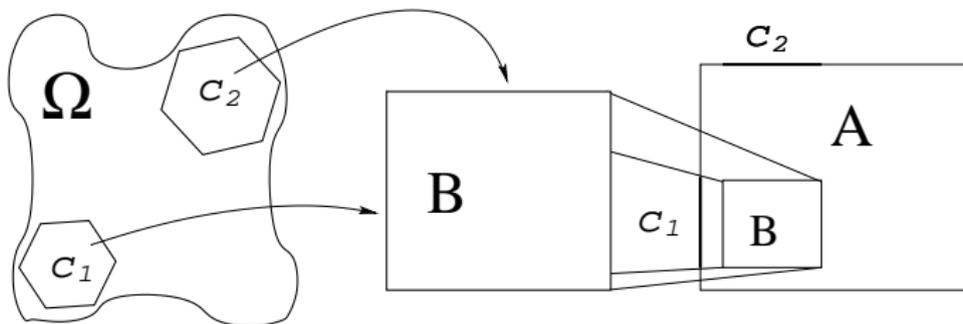


$$\min(\text{diam}(C_1), \text{diam}(C_2)) \leq \eta \text{dist}(C_1, C_2)$$

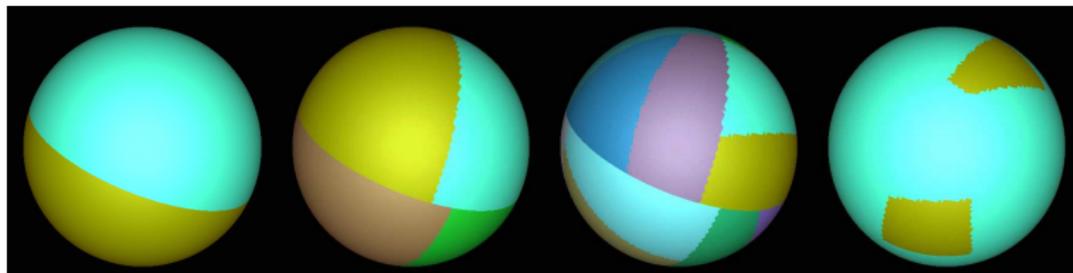


# Кластеризация

Можно перенумеровать точки/элементы так, чтобы все блоки в матрице были либо, допустимыми либо маленькими.



Обычно эта перенумерация производится иерархическим способом.



Комбинируя адаптивную крестовую аппроксимацию с иерархической кластеризацией создаётся блочно-малоранговое приближение к полной матрице.

## Полная матрица



Построение:  $O(N^2)$

Хранение:  $O(N^2)$

Пример:

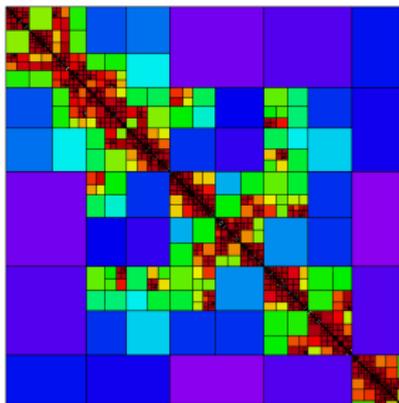
$$N = 3 \cdot 10^4,$$

выч.  $900 \cdot 10^6$  элем. (450 sec.),  
требуется 7 GB.



Комбинируя адаптивную крестовую аппроксимацию с иерархической кластеризацией создаётся блочно-малоранговое приближение к полной матрице.

## H-matrix + (F)ACA



Построение:  $O(N^2)$

Хранение:  $O(N \log^2 N)$

Пример:

$$N = 3 \cdot 10^4,$$

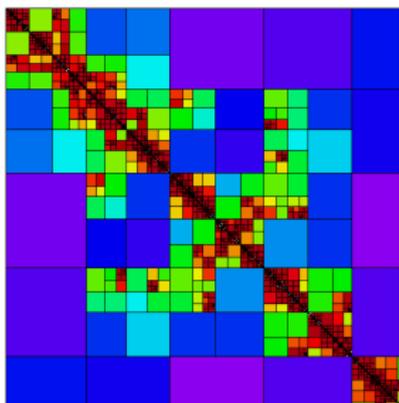
выч.  $900 \cdot 10^6$  элем. (450 sec.),

требуется 240 МВ.



Комбинируя адаптивную крестовую аппроксимацию с иерархической кластеризацией создаётся блочно-малоранговое приближение к полной матрице.

## H-matrix + ACA



Построение:  $O(N \log^2 N)$

Хранение:  $O(N \log^2 N)$

Пример:

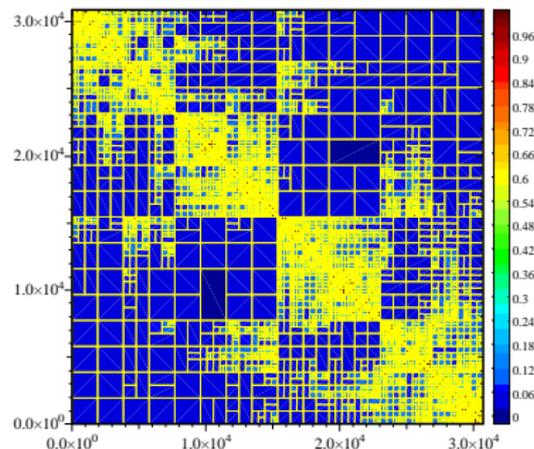
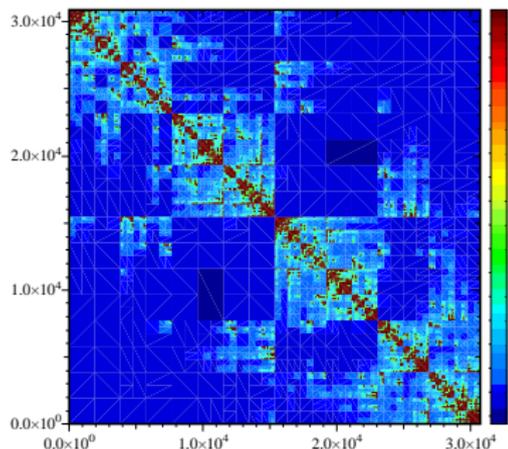
$$N = 3 \cdot 10^4,$$

выч.  $30 \cdot 10^6$  элем. (15 сек.),

требуется 240 МВ.



Матрица  $\Phi$ ,  $N = 30727$ ,  $\varphi(r) = r^2 \ln r$   
 аппроксимант  $\tilde{\Phi}$ , точность  $\varepsilon = 10^{-10}$



$\text{Mem}(\tilde{\Phi}) = 427.2\text{Mb}$

Степень сжатия  $\text{Mem}(\tilde{\Phi})/\text{Mem}(\Phi) = 0.059$ .

