Preconditioning in collocation and interpolation with radial basis functions

Josephin Giacomini

Joint work with: Nadaniela Egidi, Pierluigi Maponi, Alessia Perticarini

Università degli Studi di Camerino

School of Science and Technology - Mathematics Division

josephin.giacomini@unicam.it

Due Giorni di Algebra Lineare Numerica e Applicazioni - February 15th, 2022

Overview

Overall objective:

To solve a physico-chemical problem with the RBFs approximation → *water percolation into a porous medium made of coffee powder.*

Specific objectives:

Large linear system solved by iterative methods:

- need of an efficient computation of the action of the interpolation matrix on a vector
- need of a preconditioning strategy

The interpolation problem

Let $N \in \mathbb{N}$ be the number of interpolation points. Let f be the function $f : \mathscr{D} \subset \mathbb{R}^2 \to \mathbb{R}$. Considering the *inverse multiquadric* RBFs $\{\varphi_j\}_{\{j=1,\ldots,N\}}$,

$$\varphi_j(\mathbf{x}) = \varphi(\|\mathbf{x} - \mathbf{x}_j\|) = \frac{1}{\sqrt{t^2 + \|\mathbf{x} - \mathbf{x}_j\|^2}},$$

where $\mathbf{x}, \mathbf{x}_j \in \mathscr{D}$, $j = 1, 2, \ldots, N$, $t \in \mathbb{R}$ and $\|\cdot\| = \|\cdot\|_2$, the interpolation system is

$$\sum_{j=1}^{N} \varphi(\|\mathbf{x}_i - \mathbf{x}_j\|) c_j = b_i, \; i = 1, 2, \ldots, N,$$

or, in matrix form,

$$A\mathbf{c} = \mathbf{b},$$

where $A \in \mathbb{R}^{N \times N}$, $A_{ij} = \varphi_j(\mathbf{x}_i)$, $\mathbf{c}, \mathbf{b} \in \mathbb{R}^N$, $b_i = f(\mathbf{x}_i)$.

Outline of the work

Fast matrix action

Iterative method → row-column product of A with tentative solutions: computational cost $\approx N^2$.

An efficient technique to compute the action of A on a vector.

Preconditioning

A is ill-conditioned (depending on the shape parameter t).

Stabilisation techniques:

- Perturbation of A
- MLS preconditioner[a]

[a] Fasshauer, G. E., & Zhang, J. G. (2009). *Preconditioning of radial basis function interpolation systems via accelerated iterated approximate moving least squares approximation*. In Progress on Meshless Methods.

Fast action of the matrix on a vector

A decomposition

A decomposition:

$$A = UDV,$$

where $U \in \mathbb{R}^{N\times s}$, $D \in \mathbb{R}^{s\times s}$, $D = \mathrm{diag}(d_1, d_2, \dots, d_s)$, $V \in \mathbb{R}^{s\times N}$, and $s < N$. Thus

$$A_{ij} = \sum_{l=1}^{s} U_{il} d_l V_{lj}, \quad i, j = 1, 2, \dots, N.$$

Inspired by *Green function of the Laplacian operator* in $\mathbb{R}^3$:

$$\Delta \mathscr{G}(\mathbf{x}; \mathbf{y}) = \delta(\mathbf{x} - \mathbf{y}),$$

$$\mathscr{G}(\mathbf{x}; \mathbf{y}) = \frac{1}{\|\mathbf{x} - \mathbf{y}\|}.$$

Considering $\overline{\mathbf{x}} = (x_1, x_2, t/2), \overline{\mathbf{y}} = (y_1, y_2, -t/2) \in \mathbb{R}^3$ and $\mathbf{x} = (x_1, x_2), \mathbf{y} = (y_1, y_2) \in \mathbb{R}^2$,

$$\mathscr{G}(\overline{\mathbf{x}}; \overline{\mathbf{y}}) = \frac{1}{\|\overline{\mathbf{x}} - \overline{\mathbf{y}}\|} = \frac{1}{\sqrt{\|\mathbf{x} - \mathbf{y}\|^2 + t^2}} = \varphi(\|\mathbf{x} - \mathbf{y}\|).$$

Fast action of the matrix on a vector

Spectral decomposition of $\mathscr{G}$

Let $\overline{\mathbf{x}} = (x_1, x_2, t/2) = (\rho_x, \theta_x, \omega_x), \overline{\mathbf{y}} = (y_1, y_2, -t/2) = (\rho_y, \theta_y, \omega_y)$.

$$\mathscr{G}(\overline{\mathbf{x}}; \overline{\mathbf{y}}) = \sum_{n=0}^{\infty} \sum_{m=0}^{n} \epsilon_m \frac{(n-m)!}{(n+m)!} P_n^m\left(\cos(\theta_y)\right) P_n^m\left(\cos(\theta_x)\right) \cdot$$

$$\cdot \cos\left(m(\omega_x - \omega_y)\right) \begin{cases} \rho_x^n / \rho_y^{n+1}, & \text{if } \rho_y > \rho_x \\ \rho_y^n / \rho_x^{n+1}, & \text{if } \rho_x > \rho_y \end{cases},$$

where ϵ_m is the Neumann factor, $\epsilon_0 = 1, \epsilon_m = 2, m > 0$, and P_n^m is the Legendre function of order m and degree n associated to the Legendre polynomial P_n of degree n[1].

For points outside the convergence radius, we use some ideas of the Rokhlin's translation technique[2]. In fact, let $\mathbf{z} \in \mathbb{R}^3$, then

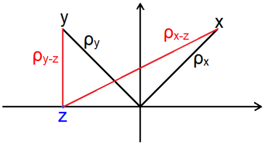
$$\mathscr{G}(\mathbf{x}; \mathbf{y}) = \frac{1}{\|\mathbf{x} - \mathbf{y}\|} = \frac{1}{\|(\mathbf{x} - \mathbf{z}) - (\mathbf{y} - \mathbf{z})\|} = \mathscr{G}(\mathbf{x} - \mathbf{z}; \mathbf{y} - \mathbf{z}).$$

[1] Morse P. M., Feshbach H., (1953). *Methods of theoretical Physics*, p. 1274.

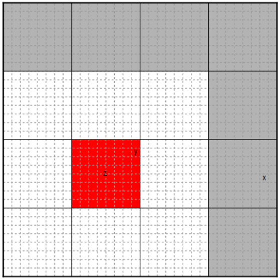
[2] Greengard, L., Rokhlin, V., (1987), *A fast algorithm for particle simulations*. In Journal of Computational Physics.

Fast action of the matrix on a vector

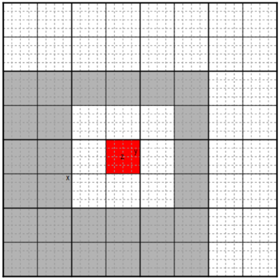
The translation technique



The **translation technique** on a square domain:



(a) Level 1



(b) Level 2

Results on a regular grid

N	Standard	Translation ($L = 5$)			Translation ($L = 10$)		
	T [s]	T [s]	err	Lev	T [s]	err	Lev
4096	7.59(-1)	1.36(-1)	1.23(-4)	1	2.40(-1)	8.26(-10)	1
16384	1.07(1)	1.61(0)	4.88(-4)	2	2.93(0)	2.52(-9)	2
65536	1.82(2)	2.16(1)	1.88(-3)	3	3.71(1)	1.04(-8)	3
262144	-	7.49(1)	-	4	2.38(2)	-	4

Table: Execution times T and error err varying the number of interpolation points N.

Perturbation of A

Right preconditioner: $AP^{-1}\mathbf{c} = \mathbf{b}$.

Preconditioning system:

$$P\tilde{\mathbf{x}} = \tilde{\mathbf{b}},$$

where $P = A + E$ perturbed matrix.

$$(A+E)\left(\mathbf{x}_0 + \sum_{k=1}^{K} \mathbf{x}_k\right) = \tilde{\mathbf{b}},$$

Procedure to solve the preconditioning system:

1. Solve $(A + \lambda I)\mathbf{x}_0 = \tilde{\mathbf{b}}$
2. Compute the residual r_1
3. for k=1,2,...,K
   i. Solve $(A + \lambda I)\mathbf{x}_k = r_k - E\mathbf{x}_{k-1}$
   ii. Compute the residual r_{k+1}
4. $\tilde{\mathbf{x}} = \mathbf{x}_0 + \sum_{k=1}^{K} \mathbf{x}_k$

MLS Preconditioner

Right preconditioner: $AP^{-1}\mathbf{c} = \mathbf{b}$.

Preconditioning system:

$$P\tilde{\mathbf{x}} = \tilde{\mathbf{b}} \quad \Rightarrow \quad \tilde{\mathbf{x}} = P^{-1}\tilde{\mathbf{b}},$$

where

$$P^{-1} = \sum_{k=0}^{K} (I - A)^k \approx A^{-1} \quad \text{for K sufficiently large.}$$

Preliminary results on the perturbed matrix

Perturbed matrix P:

$$P_{ij} = \varphi(\|(\mathbf{x}_i + \mathbf{p}_i) - (\mathbf{x}_j + \mathbf{p}_j)\|),$$

where $\mathbf{p}_i = (p_{i1}, p_{i2}), i = 1, \ldots, N$ is the random perturbation vector with $|p_{i1}| \leq h_1 p$, $|p_{i2}| \leq h_2 p$ with h_1, h_2 the two-dimensional grid steps and p a scaling factor.

Test case: $f(\mathbf{x}) = \exp\left(-\|\mathbf{x}\|^2\right)$

$t = 1$ (shape parameter of IM-RBF), $N = 256$, $\mathrm{cond}(A) = 10^{20}$.

p,K	iterations	residual	E_r
1,0	2	9.7(-7)	2.6(-5)
1(-8),4	2	2.0(-7)	3.7(-6)

Table: BicG-Stab convergence evaluation.

$$E_r = \max_{i=1,\ldots,N} \left| \frac{f(\mathbf{y}_i) - (B\mathbf{c})_i}{f(\mathbf{y}_i)} \right|,$$

where $B_{ij} = \varphi(\|\mathbf{y}_i - \mathbf{x}_j\|)$, $\{\mathbf{y}_i\}_{i=1,\ldots,N}$ evaluation points.

Preliminary results on MLS

Test case: $f(\mathbf{x}) = \exp\left(-\|\mathbf{x}\|^2\right)$,

$t = 1$ (shape parameter of IM-RBF), $N = 256$, $\text{cond}(A) = 10^{20}$.

K	iterations	residual	E_r
3	3607	1.2(-8)	5.3(-5)
4	2812	1.4(-8)	2.6(-5)
5	2395	1.1(-8)	3.6(-5)
10	2073	1.4(-8)	6.9(-5)
12	1967	1.6(-8)	9.0(-5)

Table: BicG-Stab convergence evaluation.

Further developments

- Refining of the preconditioning strategies and comparison with classical preconditioners.
- Composition of the various blocks in order to obtain an effective solving method for the interpolation problem.
- Generalisation of the whole method to a generic interpolation point set and different types of RBFs.
- Extension and adaptation of the method to discretisation schemes for differential problems.