Journal of Computational and Applied Mathematics 240 (2013) 163–173

Contents lists available at SciVerse ScienceDirect

Journal of Computational and Applied Mathematics

journal homepage: www.elsevier.com/locate/cam

Vector field approximation using radial basis functions

Daniel A. Cervantes Cabrera [a], Pedro González-Casanova [b], Christian Gout [c,*], L. Héctor Juárez [d], L. Rafael Reséndiz [d]

[a] Instituto de Matemáticas, UNAM, Ciudad Universitaria, México 04510, D.F., Mexico
[b] Instituto de Matemáticas, Ciudad Universitaria, México 04510, D.F., Mexico
[c] INSA Rouen, LMI, Av. de l'Université, BP 08, 76801 St Etienne du Rouvray cedex, France
[d] Departamento de Matemáticas, Universidad Autónoma Metropolitana-Iztapalapa, Av. San Rafael Atlixco 186, Col. Vicentina, México 09340, D.F., Mexico

ARTICLE INFO

Article history:
Received 10 February 2012
Received in revised form 20 June 2012

Keywords:
Vector field approximation
Radial basis functions
Runge phenomenon

ABSTRACT

In this article, we investigate the performance of RBF–PDE methods for approximating solenoidal fields. It is well known that global RBF collocation methods present a trade-off principle, which means that smoothness implies high convergence order plus ill-conditioning. On the other hand, local methods for solving this problem have recently appeared in the literature. In this paper, we perform a numerical investigation of the differences between RBF global and local methods, in order to investigate the possible advantage of using local methods for the approximation of vector fields. More precisely, we compare the local Hermite interpolation technique using inverse multiquadrics against the non-symmetric collocation method of Kansa.

1. Introduction

Radial basis function methods have proved to be highly effective for the solution of problems in the field of approximation theory and for the solution of PDE problems. This is due to, among other aspects, their capacity for handling complex geometries in higher dimensions for scattered node distributions, as well as the possibility of having spectral convergence. Due to the importance of approximating vector fields in subjects such as electromagnetism, meteorology, and scientific visualization among others, we explore the possibility of using these methods to solve such problems at great scales.

It is well known that, despite their success, there are major limitations to using radial basis function (RBF) methods to solve PDE problems, one of which is that they present high instability when the number of data is large and when the mesh parameter is large. This problem, known in the literature as the *uncertainty principle of Schaback* [1,2], means that when we have exponential convergence, the corresponding condition number of the Gram matrices increases in an exponential way. Another limitation, which has recently been incorporated into this discussion, is the Runge phenomenon. This important limitation is implied by the fact that Gaussian RBF interpolants in 1D converge to a polynomial as the shape parameter tends to zero (see [3–5] for more details).

We wish to mention the following methods linked to this problem for global RBFs.

Fornberg et al. [6,5] propose an expansion of the RBFs in terms of spherical harmonics, a method called QR-RBF, allowing them to get an approximation, in a stable way, of the solution for small values of the shape parameter and large numbers of nodes. However, this only applies to periodic domains, such as disks and spheres, where the Runge phenomenon does not appear [4]. On the other hand, for complex domains on the plane or in space, this algorithm has, to our knowledge, only been developed for non-trivial node distributions [5]. However, we stress that these techniques are currently under research.

* Corresponding author.
E-mail address: christian.gout@insa-rouen.fr (C. Gout).

0377-0427/$ – see front matter © 2012 Elsevier B.V. All rights reserved.
doi:10.1016/j.cam.2012.07.003

Platte and Driscoll [7] obtain node distributions making it possible to compute stable solutions for temporal problems in one dimension, or using tensor product techniques in two dimensions. They also remark that these distributions cannot be obtained for problems defined on general domains and suggest using minimum least square techniques for an overdetermined linear system built by using twice the number of collocation points relative to the PDE centers.

Elsewhere, Kee, Mao and Liu [8–11] use a similar approach for elasticity problems as well as inverse problems for the Laplace equation, by means of a local approximating scheme and node adaptive techniques. In this case, the overdetermination of the local systems is achieved by solving the differential equations and the boundary conditions over a set of additional nodes on the boundary.

Stevens et al. [12] use a local approximating scheme, based on Hermite symmetric interpolation over local domains applied to the partial differential operator of the problem, resulting in a sparse linear system of equations for the values of the solution at each center of the local domain.

In [13], the authors propose an alternative approach: they study a spline-based approximation of vector fields in the conservative case. In the modeling, they introduce a minimization problem on a Hilbert space: they introduce a regularized least square problem defined on a space of potentials (real-valued functions) to fit the vector field data set. For any $\epsilon > 0$, they consider the functional $\mathcal{J}_\epsilon$ defined as follows:

$$\mathcal{J}_\epsilon : \begin{cases} H^{m+1}(\Omega, \mathbb{R}) \to \mathbb{R} \\ v \mapsto \langle \rho(\nabla v) - u \rangle_N^2 + \epsilon |v|_{m+1,\Omega,\mathbb{R}}^2, \end{cases}$$

where $u = (u_1, \ldots, u_N)^T \in (\mathbb{R}^n)^N$ is the vector field data set and $|\cdot|_{m+1,\Omega,\mathbb{R}}$ is the semi-norm on the Sobolev space $H^{m+1}(\Omega, \mathbb{R})$. The authors establish the existence and uniqueness of the solution; they also give a convergence result in the introduced Sobolev space using norm equivalence and compactness arguments. We can also mention several other approaches used elsewhere to study this problem: finite element approximation (see [14]), PDE-based methods (see [15]), and spline and RBF approximations (see [16–22]).

Although we have emphasized the actual limitations of global RBF approximation for general domains, we stress that they are currently under intensive research and that several techniques, like domain decomposition methods [23] among others, are under investigation with a view to overcoming these limitations [24]. In this work, we propose local methods for getting a solution of the vector field approximation problem considered. Let us note that the resulting linear system of equations is sparse.

It is important to stress that the main advantage of the Stevens method is the following: by solving a set of small local systems of equations, a global sparse matrix is built. This makes it possible to solve large problems—unlike using Kansa's method, whose condition number can grow in an exponential way (for solving such problems). Moreover, the local systems do not increase their size as the number of nodes increases, which makes it possible to solve them using parallel computation. In Section 4 we investigate the running time of Stevens' algorithm for our problem.

We recall that compactly supported RBF methods (see [25]), which have been intensively studied, are a highly important alternative to these approaches and in particular to vector field interpolation, but here we focus on RBF spectral type convergence methods.

2. Vector field approximation: modeling

In this section, given an initial vector field u^0, which can be a prescribed function or a given set of values, we describe the mathematical formulation for approximating a vector field u as closely as possible to u^0, such that $u \cdot n = 0$ on Γ_N and $\nabla \cdot u = 0$.

We formulate the problem by means of a variational approach, which led to the solution of the corresponding Euler–Lagrange equations [26,27]. In what follows we introduce the energy functional and the corresponding Euler–Lagrange equations.

Let $\Omega \subset \mathbb{R}^d$ ($d = 2$ or 3) be an open, simply connected and bounded set with boundary $\partial\Omega = \Gamma_N \cup \Gamma_D$, where $\Gamma_N \neq \emptyset$ is the part of the boundary associated with the surface terrain (topography), $\Gamma_D \neq \emptyset$ is the rest of the boundary and $\mathbf{n}$ is an exterior normal vector to Ω (see Fig. 1).

So, according to the method proposed in [28]: given an initial field u^0, the exact divergence-free field is the result of minimizing the following functional:

$$\mathbf{L}(u, \lambda) = \frac{1}{2} \int_\Omega \{S(u - u^0) \cdot (u - u^0) + \lambda[\nabla u]\} dV,$$

where λ is a Lagrange multiplier and S is a diagonal matrix related to the scales of the components of the velocity, that is

$$S = \begin{pmatrix} \alpha_1^2 & 0 & 0 \\ 0 & \alpha_2^2 & 0 \\ 0 & 0 & \alpha_3^2 \end{pmatrix}$$

where the α_i are weight parameters called Gaussian precision moduli, related to the scales of the respective components of the velocity field.

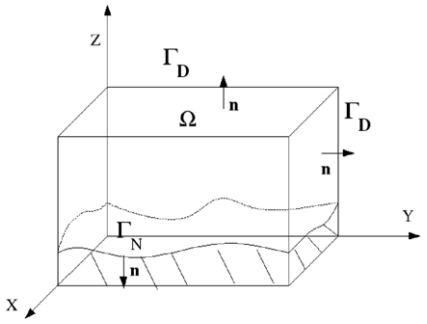

Fig. 1. The domain studied, Ω, with boundary conditions (Neumann and Dirichlet).

According to Ratto et al. [27] the Euler–Lagrange equations for the Lagrangian are

$$u = u^0 + S^{-1}\nabla\lambda, \tag{1}$$

$$\lambda u \cdot \mathbf{n} = 0. \tag{2}$$

Finally, in [29,26], the authors show that the Lagrange multipliers satisfy the following elliptic PDE:

$$-\nabla \cdot (S^{-1}\nabla\lambda) = \nabla \cdot u^0 \quad \text{in } \Omega, \tag{3}$$

$$\mathcal{B}_1\lambda = g_1 \quad \text{on } \Gamma_D, \tag{4}$$

$$\mathcal{B}_2\lambda = g_2 \quad \text{on } \Gamma_N, \tag{5}$$

where $\mathcal{B}_1$ and $\mathcal{B}_2$ are boundary conditions and g_1 and g_2 are given functions. Once λ is calculated, the approximated field u can be recovered from (1).

3. The RBF solution

In this section, we use the classical Kansa collocation method, with inverse multiquadric kernel $\phi(r) = (r^2 + c^2)^{-1/2}$, and we apply it in the reconstruction of a two-dimensional solenoidal vector field u.

3.1. The non-symmetric collocation method

Here, we get a radial basis approximation, so that we can obtain the adjusted vector field u. To do that, we use Kansa's approach [30] to solve numerically the PDE system (3)–(5) for the Lagrange multiplier λ and then we obtain u from (1).

In order to do that, let us introduce the sets of points $\Xi = \{\xi_i\}_{i=1}^N$ and $\chi = \{\mathbf{x}_i\}_{i=1}^M$ in $\mathbb{R}^d$ named *centers* and *collocation points* respectively (in this case we will take $\Xi = \chi$). Then, a radial approximation $\hat{\lambda}$ of the Lagrange multiplier λ could be defined as follows:

$$\hat{\lambda}(\mathbf{x}) = \sum_{j=1}^{N} \beta_j \phi(r_j) = \sum_{j=1}^{N} \beta_j \phi(\|\mathbf{x} - \xi_j\|), \tag{6}$$

where $\|\mathbf{x}\| = \sqrt{x_1^2 + \cdots + x_d^2}$, $\phi : \mathbb{R}^d \to \mathbb{R}$ is the radial basis function, and the set $\{\beta\}_{j=1}^N$ are the coefficients that we must determine.

To simplify the notation, we define the following differential operator:

$$\mathcal{L} := -\nabla \cdot (S^{-1}\nabla). \tag{7}$$

Now, after substituting (6) in (3)–(5) we get

$$\sum_{j=1}^{N} \beta_j \mathcal{L}\phi(\|\mathbf{x} - \xi_j\|)|_{\mathbf{x}=\mathbf{x}_i} = \nabla \cdot u^0(\mathbf{x}_i) \quad \mathbf{x}_i \in \Omega,\ i = 1 \ldots, N_I, \tag{8}$$

$$\sum_{j=1}^{N} \beta_j \mathcal{B}_1\phi(\|\mathbf{x}_i - \xi_j\|) = g_1(\mathbf{x}_i) \quad \mathbf{x}_i \in \Gamma_D,\ i = N_I + 1, \ldots, N_I + N_D, \tag{9}$$

$$\sum_{j=1}^{N} \beta_j \mathcal{B}_2\phi(\|\mathbf{x} - \xi_j\|)|_{\mathbf{x}-\mathbf{x}_i} = g_2(\mathbf{x}_i) \quad \mathbf{x}_i \in \Gamma_N,\ i = N_I + N_D + 1, \ldots, N, \tag{10}$$

Table 1
Inverse Gram matrix: SVD.

c	N	$\kappa(A)$	$\nabla \cdot \hat{u}$	L^2 error
1	20	2.5e+10	2.3e−06	6.6e−01
1	60	1.1e+15	6.6e−06	7.0e−01
1	200	3.0e+23	2.0e−06	1.8e−03
0.1	20	3.1e+19	2.4e−06	4.9e−01
0.1	60	2.8e+23	1.7e−06	2.2e−05
0.1	200	4.1e+22	1.6e−06	3.7e−06
0.01	20	2.3e+19	1.7e−06	3.0e−04
0.01	60	2.7e+20	1.8e−06	2.8e−06
0.01	200	9.2e+20	1.1e−06	1.7e−06

where N_I, N_D and N_N are the numbers of points in Ω, Γ_D and Γ_N respectively ($N = N_I + N_D + N_N$). Eqs. (8)–(10) imply the algebraic system

$$A\beta = b \tag{11}$$

where the Gram matrix

$$A = \begin{bmatrix} \widetilde{A}_{\mathcal{L}} \\ \widetilde{A}_{\mathcal{B}_1} \\ \widetilde{A}_{\mathcal{B}_2} \end{bmatrix}$$

is defined as follows: for $j = 1 \ldots, N$,

$$(\widetilde{A}_{\mathcal{L}})_{i,j} = \mathcal{L}\phi(\|\mathbf{x} - \xi_j\|)|_{\mathbf{x}=x_i} \quad \text{for } i = 1, \ldots, N_I, \tag{12}$$

$$(\widetilde{A}_{\mathcal{B}_1})_{i,j} = \mathcal{B}_1\phi(\|\mathbf{x}_i - \xi_j\|), \quad \text{for } i = N_I + 1, \ldots, N_I + N_D,$$

$$(\widetilde{A}_{\mathcal{B}_2})_{i,j} = \mathcal{B}_2\phi(\|\mathbf{x} - \xi\|)|_{x=x_i} \quad \text{for } i = N_I + N_D + 1, \ldots, N. \tag{13}$$

To illustrate this approach, we now give an example which was studied in [26].

Example 1. We consider the two-dimensional solenoidal vector field $\mathbf{u}(x, z) = (x, -z)$ defined in $\Omega = (1, 2) \times (0, 1)$. Assuming $u^0(x, z) = (x, 0)$ as an initial horizontal wind field and $\alpha_1 = 1, \alpha_3 = 0.001$, we consider the following boundary conditions in (4) and (5):

$$\mathcal{B}_1\lambda = \lambda = g_1 = 0 \quad \text{on } \Gamma_D,$$
$$\mathcal{B}_2\lambda = -S^{-1}\nabla(\lambda) \cdot \mathbf{n} = g_2 = u^0 \cdot \mathbf{n} \quad \text{on } \Gamma_N.$$

In Table 1, we show the numerical results obtained with an inverse multiquadric as the radial basis function in (6), with different numbers of nodes and shape parameters.

Note that the condition number grows as the number of nodes increases or the shape parameter decreases, in agreement with the uncertainty principle of Schaback [1,2]. In order to reduce the influence of ill-conditioning in the inversion of the Gram matrix, the truncated SVD decomposition can be applied to obtain a stable solution by means of a change of basis [31]. The results obtained using this decomposition are displayed in Table 1. Note that the L^2 error, for the SVD case, is good enough and that the values of the errors are reasonable when N grows or the shape parameter decreases.

It is important to note that although the truncated SVD method improves our results, this algorithm cannot be applied in general to large scale problems, because the technique that eliminates the singular values simultaneously removes their respective singular eigenvectors, degrading the basis of the space. Moreover, it should be noted that the computation of this decomposition is of order $O(2N^3)$ for square matrices, and due to this, some alternatives to these techniques, like domain decomposition methods and techniques based on local approximations, have been developed.

In the following tables we use the following notation: N is the number of nodes, $\kappa(A)$ denotes the condition number of the matrix A, $\nabla \cdot \hat{u}$ is the divergence of the approximated field $\hat{u}$ and the L^2 error is given by $\frac{\|u-\hat{u}\|_2}{\|\hat{u}\|_2}$.

3.2. The Runge phenomenon

In the previous section, we noted that once the condition number is controlled, it is possible to improve the relative error when the shape parameter tends to zero, so we can expect further reducing (bounded by the machine epsilon) of this parameter; thus it might be possible to obtain lower relative errors. But as shown in Table 2, this trend is preserved only for values greater than or equal to $c = 10^{-4}$ (getting a relative error of order $O(10^{-7})$), since for lower values, the relative error increases. This behavior, which has been reported in [32,4], is due to the Runge phenomenon which is well known in polynomial interpolation for equidistant partitions.

The existence of the Runge phenomenon for RBF approximations was first proved in [32]; the authors prove that Gaussian kernel interpolants, for equispaced nodes in one dimension, converge to a polynomial when the shape parameter tends to zero or the number of nodes increases.

Table 2
Runge phenomenon for the Kansa scheme.

c	N	$\kappa(A)$	$\nabla \cdot \hat{u}$	L^2 error
1	200	3.015e+23	2.0e−06	1.8e−03
0.1	200	4.1e+22	1.6e−06	3.7e−06
10^{-2}	200	9.2e+20	1.1e−06	1.7e−06
10^{-3}	200	1.6e+19	1.1e−06	2.2e−06
10^{-4}	200	4.5e+20	2.9e−07	6.2e−07
10^{-5}	200	4.0e+22	−3.5e−04	4.2e−04
10^{-6}	200	6.3e+24	8.3e−03	7.0e−01
10^{-7}	200	2.9e+26	8.7e−02	7.0e−01

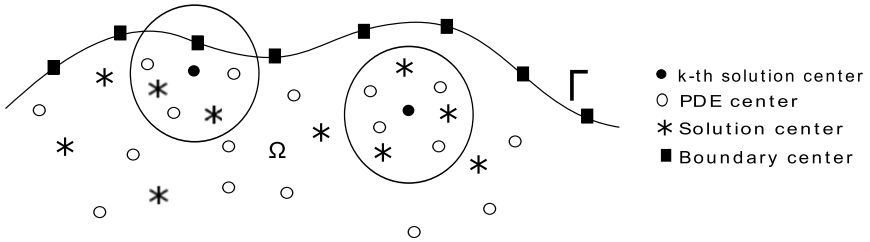

Fig. 2. Centers and local subdomains for the LHI method.

4. The local Hermitian interpolation method

Due to the limitations observed for global collocation methods, i.e., the uncertainty principle of Schaback and the Runge phenomenon, in this section we propose to apply the local algorithm developed in [12] in the reconstruction of solenoidal fields. We aim to show that unlike for Kansa's technique, the condition number of the global matrix corresponding to the local Hermitian interpolation (LHI) method does not increase in an exponential way as the number of nodes and/or the shape parameter increases. In what follows we formulate the LHI method for the vector field problem and discuss whether or not it is capable of attaining an exponential order of convergence. That is, we want to study whether it is competitive with the global collocation method of Kansa. We shall do this by means of different numerical examples.

In this approach, the analytic solution u of a PDE system

$$\mathcal{L}u(x) = f(x) \in \Omega, \tag{14}$$
$$\mathcal{B}u(x) = g(x) \in \partial\Omega, \tag{15}$$

is approximated, in a set of nodes $\Omega_{sc} = \{x_1, \ldots, x_{N_{sc}}\} \subset \Omega$ (called solution centers), by means of N_{sc} local subsystems defined for each x_i as follows:

$$u(x_i) = h_i \quad x_i \in \Omega_{sc}, \tag{16}$$
$$\mathcal{L}u(x) = f(x) \quad x \in \Omega_{pdec}, \tag{17}$$
$$\mathcal{B}u(x) = g(x) \quad x \in \partial\Omega_{fc}, \tag{18}$$

where the h_i are the unknown parameters, $\Omega_{pdec} = \{x_1, \ldots, x_{N_{pdec}}\} \subset \Omega$ is a set of interior nodes related to the differential operator $\mathcal{L}$ and $\partial\Omega_{fc} = \{x_1, \ldots, x_{N_{fc}}\} \subset \partial\Omega$, the boundary nodes.

This procedure generates a set of local linear systems given by

$$A^{(k)}\beta^{(k)} = d^{(k)} \quad k = 1, \ldots, N_{sc}, \tag{19}$$

which are obtained by substituting the Stevens radial ansatz given by

$$\hat{u}^{(k)}(\mathbf{x}) = \sum_{j=1}^{N_{sc}} \beta_j^{(k)} \phi_j(r) + \sum_{j=N_{sc}+1}^{N_{sc}+N_{fc}} \beta_j^{(k)} \mathcal{B}^{\xi}\phi_j(r) + \sum_{j=N_{sc}+N_{fc}+1}^{N_{sc}+N_{fc}+N_{pdec}} \beta_j^{(k)} \mathcal{L}^{\xi}\phi_j(r) + p_k^m \tag{20}$$

in (16)–(17), for the local domains displayed (in circles) in Fig. 2. In (20), k is the local system index, N_{sc} the number of solution centers, N_{pdec} the number of PDE centers in the local system and N_{fc} the number of boundary centers in the local system. Moreover, $\mathcal{L}^{\xi}\phi_j(r) := \mathcal{L}\phi(\|\mathbf{x} - \xi\|)|_{\xi=\xi_j}$ and $\mathcal{B}^{\xi}\phi_j(r) := \mathcal{B}\phi(\|\mathbf{x} - \xi\|)|_{\xi=\xi_j}$, $\phi(r)$ the inverse multiquadric and p_k^m a polynomial in $\mathbb{R}^d$ and of degree m, which is defined by the null space of (14) and (15), i.e. for $d = 2$ and $m = 2$, the polynomial is given by $p(x, y) = a_0 + a_1x + a_2y + a_3xy + a_4x^2 + a_5y^2$.

By using (19), we obtain the following Gram matrix and right hand vector:

$$A^{(k)} = \begin{bmatrix} \Phi_{ij} & \mathcal{L}^{\xi}[\Phi_{ij}] & \mathcal{B}^{\xi}[\Phi_{ij}] & P_{ij} \\ \mathcal{L}^{\mathbf{x}}[\Phi_{ij}] & \mathcal{L}^{\mathbf{x}}\mathcal{L}^{\xi}[\Phi_{ij}] & \mathcal{L}^{\mathbf{x}}\mathcal{B}^{\xi}[\Phi_{ij}] & \mathcal{L}^{\mathbf{x}}[P_{ij}] \\ \Phi_{ij} & \mathcal{L}^{\xi}[\Phi_{ij}] & \mathcal{B}^{\xi}[\Phi_{ij}] & P_{ij} \\ \mathcal{B}^{\mathbf{x}}[\Phi_{ij}] & \mathcal{B}^{\mathbf{x}}\mathcal{L}^{\xi}[\Phi_{ij}] & \mathcal{B}^{\mathbf{x}}\mathcal{B}^{\xi}[\Phi_{ij}] & \mathcal{B}^{\mathbf{x}}[P_{ij}] \end{bmatrix} \quad \text{and} \quad d^{(k)} = \begin{bmatrix} h_i \\ f_i \\ 0 \\ g_i \end{bmatrix}.$$

Thus, each approximation $\hat{u}^{(k)}(\mathbf{x})$ has the form

$$\hat{u}^{(k)}(\mathbf{x}) = H(\mathbf{x})\beta^{(k)}(h_1, \ldots, h_{N_s}), \tag{21}$$

where

$$H^{(k)}(\mathbf{x}) = \left[\phi(\|\mathbf{x}-\xi\|)\ \mathcal{L}^{\xi}\phi(\|\mathbf{x}-\xi\|)\ \phi(\|\mathbf{x}-\xi\|)\ \mathcal{B}^{\xi}\phi(x-\xi)\right].$$

Finally, the local systems (19) are coupled in a global sparse linear system whose unknowns are the h_i values. This coupling is performed by applying the differential operator $\mathcal{L}$ to each local interpolator (21) and evaluating it at the corresponding solution center, i.e.

$$\mathcal{L}[\hat{u}^{(k)}(\mathbf{x})]|_{\mathbf{x}=x_i} = f(x_i) \tag{22}$$

which is the discretization of (17). Once the values of the unknowns are obtained, the approximation of u at each solution center (see (16)) can be determined.

In our problem, i.e. the approximation of vector fields, as has been mentioned, it is necessary to calculate an approximation of the Lagrange multiplier λ by solving the corresponding Euler–Lagrange equations and by using Eq. (1).

It should be noted, however, that in this case it is necessary to approximate the gradient of λ, so the Stevens algorithm cannot be applied directly. This is because in the other case they approximate the solution, whereas in our case we need to estimate the components of the gradient of the Lagrange multiplier.

To achieve this result, we need to change the original Stevens ansatz (20), by incorporating the partial derivatives in x and y in the following way:

$$\begin{aligned} \hat{\lambda}_x^{(k)}(\mathbf{x}) \ = \ & \sum_{j=1}^{N_{sc}} \beta_j^{(k)} \frac{\partial \phi_j(r)}{\partial x} + \sum_{j=N_{sc}+1}^{N_{sc}+N_{\mathcal{B}_1}} \beta_j^{(k)} \mathcal{B}_1^{\xi}\phi(r) + \sum_{j=N_{sc}+N_{\mathcal{B}_1}+1}^{N_{sc}+N_{\mathcal{B}_1}+N_{\mathcal{B}_2}} \beta_j^{(k)} \mathcal{B}_2^{\xi}\phi(r) \\ & + \sum_{j=N_{sc}+N_{fc}+1}^{N_{sc}+N_{fc}+N_{pdec}} \beta_j^{(k)} \mathcal{L}^{\xi}\phi_j(r) + p^m_{\mathbf{k},\mathbf{x}}(x) \end{aligned} \tag{23}$$

$$\begin{aligned} \hat{\lambda}_y^{(k)}(\mathbf{x}) \ = \ & \sum_{j=1}^{N_{sc}} \beta_j^{(k)} \frac{\partial \phi_j(r)}{\partial y} + \sum_{j=N_{sc}+1}^{N_{sc}+N_{\mathcal{B}_1}} \beta_j^{(k)} \mathcal{B}_1^{\xi}\phi(r) + \sum_{j=N_{sc}+1}^{N_{sc}+N_{\mathcal{B}_2}} \beta_j^{(k)} \mathcal{B}_2^{\xi}\phi(r) \\ & + \sum_{j=N_{sc}+N_{fc}+1}^{N_{sc}+N_{fc}+N_{pdec}} \beta_j^{(k)} \mathcal{L}^{\xi}\phi_j(r) + p^m_{\mathbf{k},\mathbf{y}}(x). \end{aligned} \tag{24}$$

Here $N_{\mathcal{B}_1}$ is the number of boundary nodes in Γ_D and $N_{\mathcal{B}_2}$ is the number of boundary nodes in Γ_N, and $N_{fc} = N_{\mathcal{B}_1} + N_{\mathcal{B}_2}$. The corresponding local systems are thus given by

$$\begin{cases} \left.\dfrac{\partial}{\partial x}\lambda(x)\right|_{x=x_i} = h_i' & x_i \in \Omega_{sc} \\ \mathcal{L}\lambda(x) = f(x) & x \in \Omega_{pdec} \\ \mathcal{B}_1\lambda(x) = g_1(x) & x \in \Gamma_D \\ \mathcal{B}_2\lambda(x) = g_2(x) & x \in \Gamma_N \end{cases} \tag{25}$$

$$\begin{cases} \left.\dfrac{\partial}{\partial y}\lambda(x)\right|_{x=x_i} = h_i'' & x_i \in \Omega_{sc} \\ \mathcal{L}\lambda(x) = f(x) & x \in \Omega_{pdec} \\ \mathcal{B}_1\lambda(x) = g_1(x) & x \in \Gamma_D \\ \mathcal{B}_2\lambda(x) = g_2(x) & x \in \Gamma_N \end{cases} \tag{26}$$

with $\mathcal{L}$ defined in (7), and $\mathcal{B}_1$ and $\mathcal{B}_2$ the boundary conditions, so we can calculate $\nabla\lambda = (\lambda_x(x), \lambda_y(x))$. Note that $p^m_{\mathbf{k},\mathbf{x}}$ and $p^m_{\mathbf{k},\mathbf{y}}$ for our case are defined by

$$p(x,y)^2_{\mathbf{k},\mathbf{x}} = a_1 x + a_2\frac{x^2}{2} + a_3 xy + a_4\frac{x^2 y}{2} + a_5\frac{x^3}{3} + a_6 xy^2 \tag{27}$$

$$p(x,y)^2_{\mathbf{k},\mathbf{y}} = a_1 y + a_2 xy + a_3 xy + a_3\frac{y^2}{2} + a_5\frac{xy^2}{3} + a_6\frac{y^3}{3}. \tag{28}$$

Table 3
Inverse Gram matrix: SVD.

c	N	$\kappa(A)$	L^2 error	CPU
0.1	60	2.8e+23	2.2e−05	0.00181
0.1	200	4.1e+22	3.7e−06	0.032
0.1	400	1.4e+24	1.4e−06	0.1873
0.01	60	2.7e+20	2.8e−06	0.00175
0.01	200	9.2e+20	1.7e−06	0.0273
0.01	400	4.8e+25	5.7e−06	0.1821

Table 4
Inverse global matrix: LHI.

$\kappa(A)$	L^2 error	CPU
2.8e+13	2.4e−06	0.000210
2.1e+14	9.99e−03	0.00556
1.26e+14	6.85e−03	0.0171
8.1e+13	4.4e−05	0.000204
1.3e+14	2.4e−04	0.00164
2.41+14	2.5e−03	0.00663

This is, in our opinion, a significant modification, which yields an effective methodology for computing numerical solutions of the problem, according to the results that we will show below. We conjecture that this type of modification may be useful for approximating, not only the gradient, but also other kinds of differential operators like the rotational ones, when solving partial differential equations.

In order to illustrate the effectiveness of the LHI method, we display in Tables 3 and 4 the CPU times for Kansa's collocation method using SVD and the LHI method of Stevens et al. [12] for Example 1.

We can appreciate from these tables that the CPU time (with an Intel core 2, 2.13 GHz) of the LHI technique decreases by at least one order of magnitude with respect to the SVD–Kansa method. Note also that for the LHI method, a quadtree algorithm of numerical complexity $O(n\log(n))$, plus the inversion of the local matrices corresponding to each solution center, must be taken into account.

We denote as A_x and A_y the global Gram matrices resulting from applying the differential operator $\mathcal{L}$ to each ansatz (23) and (24), and evaluating them at the solution centers in agreement with Eq. (22). Simultaneously, A_x^{Loc} and A_y^{Loc} are the matrices which correspond to the local systems arising from substituting (23) and (24) in Eqs. (25) and (26) respectively.

In what follows, we shall consider first the case where the boundary conditions are of Dirichlet type and then the case where Neumann boundary conditions are taken into account. We aim to investigate whether the LHI scheme is capable of attaining exponential orders of convergence, with low condition numbers for the global matrices, and whether it can improve the performance of the global collocation method of Kansa. We also analyze the limitations of this method.

For this purpose, for the following numerical experiments, we display, in the first two columns of the following tables, the condition number and the L^2 error of the global matrices. Our intention is to examine whether exponential convergence can be reached and a low condition number can be attained simultaneously. On the other hand we display, in columns 4 and 5, the condition numbers of the local systems, to analyze whether these values influence the size of the condition number of the global matrix and the L^2 error.

Example 2 (*Approximation with Dirichlet Boundary Conditions*)**.** In this example we solve the same problem as in Example 1, but in this case we take Dirichlet boundary conditions, i.e.

$$\mathcal{B}_1\lambda(\mathbf{x}) = \lambda(\mathbf{x}) = g_1(x)$$
$$\mathcal{B}_2\lambda(\mathbf{x}) = \lambda(\mathbf{x}) = g_2(x)$$

with $g_1(1,z) = g1(2,z) = \frac{\alpha_3^2}{2}(1-z^2)$, $g_1(x,1) = 0$ and $g_2(x,0) = \frac{\alpha_3^2}{2}$. The results are shown in Tables 5, 6 and Fig. 3.

Example 3 (*Approximation with Dirichlet–Neumann Boundary Conditions*)**.** In this example we solve the same problem as in Example 1 but this time with the same Dirichlet–Neumann boundary conditions, i.e.

$$\mathcal{B}_1\lambda = \lambda = g_1 = 0 \quad \text{on } \Gamma_D$$
$$\mathcal{B}_2\lambda = -S^{-1}\nabla(\lambda)\cdot\mathbf{n} = g_2 = u^0\cdot\mathbf{n} \quad \text{on } \Gamma_N.$$

The results are shown in Tables 7, 8 and Fig. 4.

In Tables 5–8 only the condition number of A_y is shown while A_x is omitted; the reason is that the two values have very similar numerical behavior.

In Tables 5 and 6 we display, for Dirichlet boundary conditions, the numerical results for two different values of the shape parameter, namely for $c = 0.1$ and $c = 0.01$. It can be observed that the condition number of the global matrix A_y increases

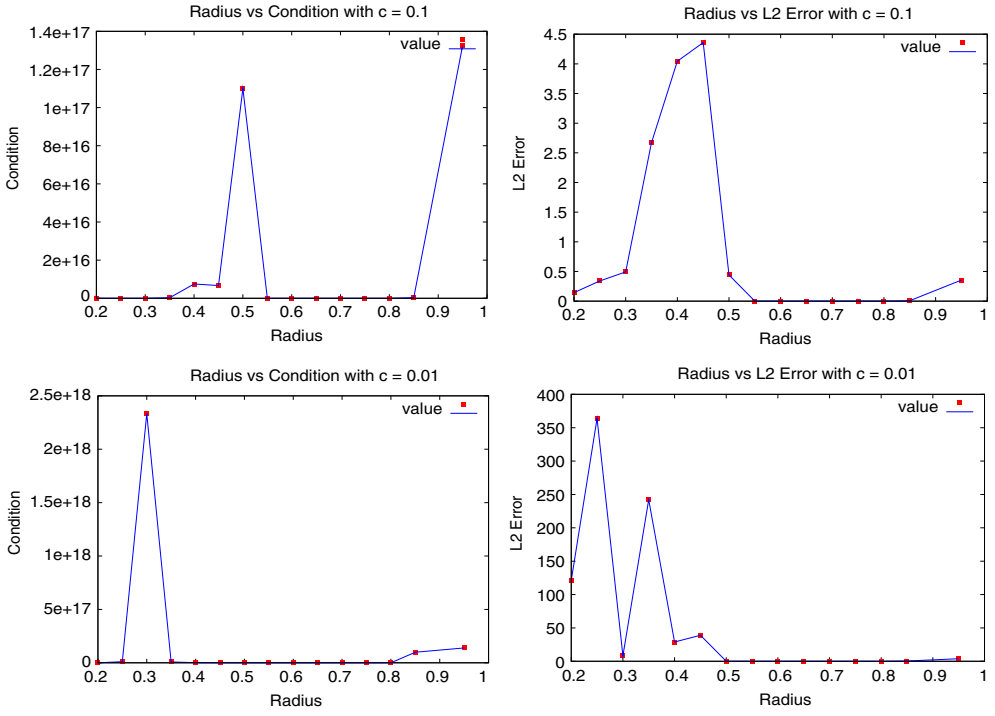

Fig. 3. Upper: radius versus condition number and radius versus L^2 error for $c = 0.1$ (Table 5). Lower: radius versus condition number and radius versus L^2 error for $c = 0.01$ (Table 6).

Table 5
LHI approximation with Dirichlet boundary conditions, number of interior nodes = 100 (50 solution centers and 50 PDE centers) and 20 boundary nodes, with shape parameter $c = 0.1$.

Radius	Global $\kappa(A_y)$	$\nabla \cdot \hat{u}$	L^2 error	Max local $\kappa(A_y^{\text{Loc}})$	Min local $\kappa(A_y^{\text{Loc}})$
0.2	3.8e+06	7.8e+00	1.4e−01	3.4e+27	1.2e+12
0.3	1.3e+12	4.1e−02	4.9e−01	2.9e+28	1.1e+17
0.4	7.4e+15	1.5e−01	4.0e+00	1.7e+29	1.0e+18
0.5	1.1e+17	3.9e−03	4.4e−01	1.1e+30	1.0e+27
0.6	1.9e+12	3.1e−03	2.1e−06	2.2e+30	1.6e+27
0.7	5.9e+12	4.1e−03	4.2e−06	5.6e+30	1.6e+27
0.8	2.7e+11	4.0e−03	3.7e−06	1.0e+30	2.6e+27
0.9	1.3e+17	3.4e−03	3.5e−01	3.5e+30	4.7e+27

Table 6
LHI approximation with Dirichlet boundary conditions, number of interior nodes = 100 (50 solution centers and 50 PDE centers) and 20 boundary nodes, with shape parameter $c = 0.01$.

Radius	Global $\kappa(A_y)$	$\nabla \cdot \hat{u}$	L^2 error	Max local $\kappa(A_y^{\text{Loc}})$	Min local $\kappa(A_y^{\text{Loc}})$
0.2	7.6e+08	8.5e+02	1.2e+02	1.5e+27	2.5e+16
0.3	2.3e+18	6.8e+02	8.2e+00	1.8e+27	9.1e+19
0.4	1.1e+15	4.7e+01	2.8e+01	2.9e+28	3.6e+20
0.5	3.7e+12	3.0e−02	1.0e−01	3.7e+27	1.7e+25
0.6	1.4e+13	3.7e−03	3.1e−02	2.3e+27	3.0e+25
0.7	1.4e+12	2.8e−02	3.1e−04	1.6e+27	5.3e+25
0.8	3.3e+10	9.0e−02	1.3e−05	1.0e+28	4.0e+25
0.9	1.3e+17	7.9e−01	3.8e+00	1.4e+27	7.2e+25

with the number of nodes contained in the circle. This happens within an interval and then the condition number change slows down within a region around the value 0.6. In this region it can be observed that both the condition number and the error decrease. In fact, the error is of the order of 10^{-6}, which is of the same order as that obtained with the global method of Kansa. These results are displayed in Fig. 3. Also, it can be appreciated that the numerical approximation of the divergence is reduced within this region and its value remains constant, having an order of 10^{-3}.

It can also be observed that there is no direct relation between the maximum ("Max local $\kappa(A_y^{\text{Loc}})$") and minimum ("Min local $\kappa(A_y^{\text{Loc}})$") values of the condition numbers of the local matrices A_y^{Loc}.

Table 7
LHI approximation with Dirichlet–Neumann boundary conditions, number of interior nodes = 200 (100 solution centers and 100 PDE centers) and 40 boundary nodes, with shape parameter $c = 0.1$.

Radius	Global $\kappa(A_y)$	$\nabla\hat{u}$	L^2 error	Max local $\kappa(A_y^{\text{Loc}})$	Min local $\kappa(A_y^{\text{Loc}})$
0.2	4.6e+10	4.3e+00	3.7e+00	1.2e+22	3.8e+18
0.3	1.0e+16	1.9e−01	5.4e+00	3.7e+28	3.3e+19
0.4	3.6e+20	3.1e−01	5.6e−01	1.6e+29	4.9e+19
0.5	3.1e+16	6.2e−01	5.4e−01	4.1e+29	8.0e+19
0.6	6.8e+14	8.2e−03	1.0e−05	4.7e+30	8.1e+26
0.7	3.0e+14	6.3e−03	7.7e−03	2.8e+30	1.3e+27
0.8	1.6e+15	4.4e−03	1.2e−01	5.1e+29	3.7e+27
0.9	2.9e+20	4.4e−01	3.1e−01	7.6e+29	4.4e+27

Table 8
LHI approximation with Dirichlet–Neumann boundary conditions; number of interior nodes = 200 (100 solution centers and 100 PDE centers) and 40 boundary nodes, with shape parameter $c = 0.01$.

Radius	Global $\kappa(A_y)$	$\nabla\hat{u}$	L^2 error	Max local $\kappa(A_y^{\text{Loc}})$	Min local $\kappa(A_y^{\text{Loc}})$
0.2	1.1e+14	1.5e+02	6.7e+01	1.7e+28	3.9e+16
0.3	7.1e+21	2.8e−01	5.7e−01	2.0e+27	8.7e+19
0.4	2.9e+21	5.9e−01	5.3e−01	1.1e+27	3.9e+20
0.5	6.8e+17	1.4e−01	6.4e−01	5.5e+27	1.1e+21
0.6	2.4e+14	3.6e−03	4.5e−03	1.8e+28	1.3e+25
0.7	2.3e+15	1.7e−02	5.2e−03	3.3e+28	2.4e+25
0.8	3.8e+12	3.4e−02	1.1e−01	1.0e+28	5.7e+25
0.9	2.7e+22	4.3e−01	4.1e−01	2.7e+28	5.3e+25

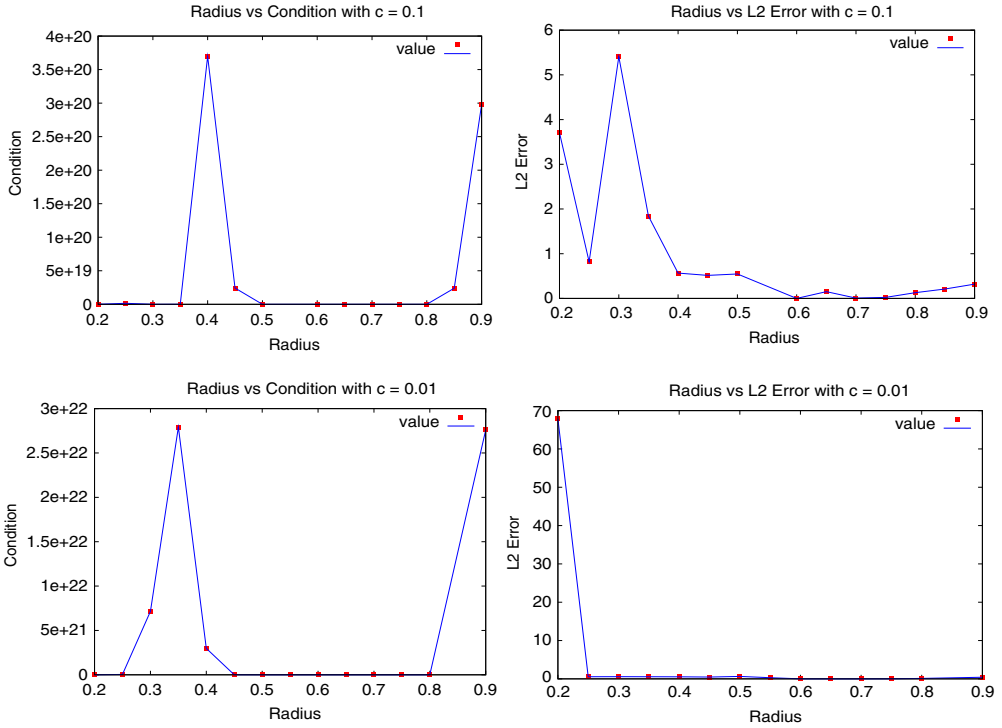

Fig. 4. Upper: radius versus condition number and radius versus L^2 error for $c = 0.1$ (Table 7). Lower: radius versus condition number and radius versus L^2 error for $c = 0.01$ (Table 8).

In Tables 5 and 6 we display the numerical results for when the boundary conditions are of Dirichlet–Neumann type. In this case it can be observed that the behavior is similar to that for the Dirichlet case; the main difference is that the error is of the order of 10^{-5} instead of 10^{-6}. These results are displayed in Fig. 4.

We stress that in both cases, for Dirichlet and Dirichlet–Neumann boundary conditions, there is a region for the value for the radius, i.e. for the number of local nodes, where the approximation is nearly as good as for the global collocation method, i.e. of spectral type.

5. Conclusions

In this work we have studied numerically the approximation of solenoidal vector fields. Our main objective was to compare global radial basis function techniques, of spectral type, with local methods for inverse multiquadric kernels. Our numerical results indicate that unlike global methods, local techniques and in particular the LHI method may solve problems of great scale since the global system of equations has a global sparse matrix with a corresponding low condition number.

The ansatz of the LHI is flexible and it is possible to incorporate differential operators that depend on the problem studied. However, in our case, this condition was not strong enough and it was necessary to modify the ansatz itself in order to approximate the gradient of the Lagrange multipliers. This modification of the ansatz is a significant innovation and confirms that this technique produces good results for problems like the one considered here.

We point out that in the case of Dirichlet boundary conditions, local methods achieve an error equivalent to the error obtained by the global approximation collocation techniques for this problem. The numerical experiments for vector fields were done for Cartesian meshes as well as for random data (Halton nodes), and verified that LHI scheme produced similar errors in both cases. This LHI local method is very much adequate for parallel computing because the computation of the inverse of each local matrix, corresponding to each solution center, can be performed independently with multiple processors.

On the other hand, we observed that for Dirichlet–Neumann boundary conditions the LHI method produces results which are less accurate than those from global collocation schemes. This agrees both with the numerical experiments performed in [12] and the results for least square methods reported in [8]. In both cases the authors found that Neumann boundary conditions produce instabilities with local methods. This behavior is a subject of current research by the scientific community.

The numerical results obtained in this work showed that for the LHI method there is a "trade-off" principle similar to the one that exists for the compact support RBF method developed in [25]. More precisely, when the support of the local basis functions increases, the error of the approximated solution decreases and the condition number increases. On the other hand, if the support is small, the condition number is low but the error increases. Moreover, as is displayed in Figs. 3 and 4, there is a region where the sizes of the supports of the local functions presents small—in fact spectral—errors, and this is true for different values of the shape parameter c. Simultaneously, in this same region it can be observed (see Fig. 3) that the condition number of the global matrices reaches minimum values.

Although the maximum and minimum condition numbers of the local basis functions are high, our results indicate that the increases of the error and the condition number of the global matrix do not depend on these values. The study of this behavior is currently a topic of our research.

References

[1] R. Schaback, Error estimates and condition numbers for radial basis functions interpolation, Adv. Comput. Math. 3 (1995) 251–264.
[2] R. Schaback, Multivariate interpolation and approximation by translates of a basis function, in: Approximation Theory, World Scientific Publishing, Singapore, 1995, pp. 491–514.
[3] B. Fornberg, G. Wright, E. Larsson, Some observations regarding interpolants in the limit of flat radial basis functions, Comput. Math. Appl. 47 (2004) 37–55.
[4] B. Fornberg, J. Zuev, The Runge phenomenon and spatially variable shape parameters in RBF interpolation, Comput. Math. Appl. 54 (2007) 379–398.
[5] B. Fornberg, E. Larsson, N. Flyer, Stable computations with Gaussian radial basis functions, SIAM J. Sci. Comput. 33 (2011) 869–892.
[6] B. Fornberg, C. Piret, A stable algorithm for flat radial basis functions on a sphere, SIAM J. Sci. Comput. 30 (2007) 60–80.
[7] R. Platte, T. Driscoll, Eigenvalue stability of radial basis function discretizations for time-dependent problems, Comput. Math. Appl. 51 (2006) 1251–1268.
[8] B. Kee, G.R. Liu, C. Lu, A least-square radial point collocation method for adaptive analysis in linear elasticity, Eng. Anal. Bound. Elem. 32 (2008) 440–460.
[9] G. Liu, B. Kee, Lu Chun, A stabilized least-squares radial point collocation method (LS-RPCM) for adaptive analysis, Eng. Anal. Bound. Elem. 32 (2008) 440–460.
[10] X.-Z. Mao, Z. Li, Least-square-based radial basis collocation method for solving inverse problems of Laplace equation from noisy data, Internat. J. Numer. Methods Engrg. 84 (2010) 1–26.
[11] Xiong Zhang, Xiao-Hu Liu, Kang-Zu Song, Ming-Wang Lu, Least-squares collocation meshless method, Internat. J. Numer. Methods Engrg. 51 (2001) 1089–1100.
[12] D. Stevens, H. Power, M. Lees, H. Morvan, A local hermitian RBF meshless numerical method for the solution of multi-zone problems, Numer. Methods Partial Differential Equations 27 (5) (2011) 1201–1230.
[13] C. Le Guyader, D. Apprato, C. Gout, Spline approximation of gradient fields: applications to wind velocity fields, in revision, MATCOM, 2012.
[14] A.S. Dzhabrailov, Y.V. Klochkov, S.S. Marchenko, A.P. Nikolaev, The finite element approximation of vector fields in curvilinear coordinates, Russian Aeronaut. (Iz VUZ) 50 (2) (2007) 115–120.
[15] L. Amodei, M.N. Benbourhim, A vector spline approximation, J. Approx. Theory 67 (1) (1991) 51–79.
[16] G. Awanou, M.J. Lai, Trivariate spline approximations of 3D Navier–Stokes equations, Math. Comp. 74 (2005) 585–601.
[17] F. Dodu, C. Rabut, Vectorial interpolation using radial-basis-like functions, Comput. Math. Appl. 43 (3–5) (2002) 393–411.
[18] M.N. Benbourhim, A. Bouhamidi, Approximation of vector fields by thin plate splines with tension, J. Approx. Theory 136 (2005) 198–229.
[19] M.N. Benbourhim, A. Bouhamidi, Pseudo-polyharmonic vectorial approximation for div–curl and elastic semi-norms, Numer. Math. 109 (3) (2008) 333–364.
[20] S. Ettl-Lowitzsch, J. Kaminski, M.C. Knauer, G. Häusler, Shape reconstruction from gradient data, Appl. Opt. 47 (12) (2008) 2091–20971.
[21] S. Lowitzsch, Approximation and interpolation employing divergence-free radial basis functions with applications, in: Dissertation, Texas A&M University, 2002.
[22] S. Lowitzsch, Error estimate for matrix-valued radial basis functions interpolation, J. Approx. Theory 137 (2) (2005) 238–249.
[23] P. González-Casanova, J. Antonio Muñoz-Gómez, G. Rodríguez-Gómez, Node adaptive domain decomposition method by radial basis functions, Numer. Methods Partial Differential Equations 25 (6) (2009) 1482–1501.

[24] John P. Boyd, Six strategies for defeating the Runge Phenomenon in Gaussian radial basis functions on a finite interval, Comput. Math. Appl. 60 (12) (2010) 3108–3122.
[25] Holger Wendland, Multiscale analysis in Sobolev spaces on bounded domains, Numer. Math. 116 (2010) 493–517.
[26] C. Flores, H. Juárez, M. Nuñez, M. Sandoval, Algorithms for vector field generation in mass consistent models, Numer. Methods Partial Differential Equations 26 (2010) 826–842.
[27] C.F. Ratto, R. Festa, C. Romeo, O.A. Frumento, M. Galluzzi, Mass-consistent models for wind fields over complex terrain: the state of the art, Environmental Software 9 (4) (1994) 247–268.
[28] Y. Sasaki, An objective analysis based on the variational method, J. Meteorol. Soc. Japan 36 (1958) 77–88.
[29] V. Girault, P.A. Raviart, Finite Element Methods for the Navier–Stokes Equations: Theory and Algorithms, Springer-Verlag, Berlin, 1986.
[30] E.J. Kansa, Multiquadrics—a scattered data approximation scheme with applications to computational fluid dynamics II, Comput. Math. Appl. 19 (1990) 147–161.
[31] R. Schaback, Multivariate interpolation by polynomials and radial basis functions, Constr. Approx. 21 (3) (2005) 293–317.
[32] R. Platte, T. Driscoll, Polynomials and potential theory for Gaussian radial basis functions interpolation, SIAM J. Numer. Anal. 43 (2005) 750–766.