

Application of cardinal radial basis interpolation operators to numerical solution of the Poisson equation

GIAMPIETRO ALLASIA – ALESSANDRA DE ROSSI

ABSTRACT: *We consider the application of a new scattered data approximation scheme to numerically solving the Dirichlet problem for the Poisson equation. This collocation method, which is mesh-free and substantially independent on the space dimension, makes use of interpolation operators with cardinal radial basis and differs from the well-known discretization approach introduced by E. J. Kansa in 1990 and then extensively developed, based on Hardy's multiquadrics or others radial basis functions. In our method the discretization matrix, whose dimension equals the number of internal points in the domain, is symmetric and strictly diagonally dominant, so that the discrete problem is well-posed and also well-conditioned, since the matrix condition number is small. Numerical experiments show that the performance of our method is comparable in many cases with that of Kansa's method; moreover, the former works well even if the number of collocation points is large.*

1 – Introduction

In early the 1990s E. J. KANSA [13], [14] proposed a method to solve hyperbolic, parabolic and elliptic partial differential equations using Hardy's multiquadric radial basis functions (MQ-RBFs). Then several authors extended Kansa's idea applying MQ-RBFs and other radial basis functions (RBFs) to the numerical solution of various types of PDEs (see recent developments in [16]).

KEY WORDS AND PHRASES: *Elliptic partial differential equations – Approximation scheme – Scattered data – Cardinal radial basis.*

A.M.S. CLASSIFICATION: 65N35 – 65N22 – 65D10 – 41A63

Two useful features of Kansa's method immediately appeared: it gives a truly mesh-free algorithm and its computational complexity does not increase consistently with spatial dimension. Mesh-free radial basis functions provide powerful discretization representations to simulate problems not merely in bidimensional domains, but in arbitrary n -dimensional domains with irregular boundaries, and they can be implemented wisely on massively parallel computers. There are also some well known disadvantages: MQ-RBFs (and in general RBFs) are globally supported and the discretization matrix is full and ill-conditioned. Several remedies have been proposed to circumvent the ill-conditioning, but this is yet a relevant and open problem in the resolution of PDEs by RBFs (see, e.g. [15]).

A seemingly new scheme for discretization of the Poisson equation and some other elliptic PDEs, which recalls Kansa's method but differs in some noteworthy aspects, has been proposed [4]. This collocation method, which is mesh-free and substantially independent on the space dimension, makes use of interpolation operators with cardinal radial basis (CRBIs). A suitable choice of the basis function type yields a discretization matrix, that is symmetric and strictly diagonally dominant, and has a dimension equal to the number of collocation points internal to the domain. So the discrete problem is well-posed and also well-conditioned, since the matrix condition number is small.

Numerical experiments show that the method considered gives solutions whose accuracy is in many cases comparable with that achieved by the MQ-RBF method. Furthermore, it appears clear that our method does not stop working well even if the set of collocation points is large.

2 – The CRBI Scheme

In order to investigate computational properties of our scattered data approximation scheme, we start outlining it and, for simplicity, focusing on the consideration of the Dirichlet problem for the Poisson equation.

Let $\Omega \subset \mathbf{R}^s$, ($s \geq 2$), be an open, simply connected point set, bounded by a piecewise regular hypersurface Σ , and let $h(x) \in C(\Omega \cup \Sigma)$ and $k(x) \in C(\Sigma)$ be given real functions with $x = (x_1, x_2, \dots, x_s)$. The Dirichlet problem for the Poisson equation is that of finding a real function $u(x)$ in the space $\mathcal{U} = \{u \in C^2(\Omega) \cap C(\Sigma)\}$ such that

$$(1) \quad -\Delta u(x) \equiv -\sum_{k=1}^s \frac{\partial^2 u(x)}{\partial x_k^2} = h(x), \quad x \in \Omega, \quad \text{and} \quad u(x) = k(x), \quad x \in \Sigma.$$

To build up the discrete problem associated with (1), we consider a set of distinct points $S_N = \{\xi_i, i = 1, \dots, N\}$, in general arbitrarily distributed in the domain Ω , and a suitable family of cardinal basis functions $g_k \in C^2(\Omega) \cap C(\Sigma)$, ($k = 1, \dots, N$), or rather $g_k \in C^2(\Omega \cup \Sigma)$, such that

$$(2) \quad g_k(\xi_i) = \delta_{ki},$$

where δ_{ki} is the Kronecker delta. Each element F of the set $\mathcal{F}_N = \text{span}\{g_k, k = 1, \dots, N\}$, which is a linear space of dimension N and a subset of \mathcal{U} , is uniquely represented in the form

$$(3) \quad F(x) = \sum_{k=1}^N c_k g_k(x).$$

An approximate solution F of the problem (1) must satisfy the linear system

$$-\sum_{k=1}^N c_k \Delta g_k(\xi_i) = h(\xi_i), \quad \xi_i \in \Omega, \quad \text{and} \quad \sum_{k=1}^N c_k g_k(\xi_i) = k(\xi_i), \quad \xi_i \in \Sigma,$$

that is for (2)

$$(4) \quad -\sum_{k=1}^N c_k \Delta g_k(\xi_i) = h(\xi_i), \quad \xi_i \in \Omega, \quad \text{and} \quad c_i = k(\xi_i), \quad \xi_i \in \Sigma.$$

If we suppose for $k = 1, \dots, N$

$$w_k \in C^2(\Omega \cup \Sigma), \quad w_k(x) = \begin{cases} 0, & \text{for } x = \xi_i, i \neq k, \\ > 0, & \text{for } x = \xi_k, \end{cases}$$

then we can set

$$(5) \quad g_k(x) = \frac{w_k(x)}{\sum_{j=1}^N w_j(x)},$$

and these $g_k(x)$ can be interpreted as the basis functions in (3). Properties of the interpolation operator $F(x)$ are discussed in detail in [1], [2].

3 – The Basic Weight

The values of the second derivative of $F(x)$ at the nodes, which are needed in (4), depend from (5) on the choice of the weight $w_k(x)$. In a first approach, we choose a simple and classical weight setting for $k = 1, \dots, N$, and $\xi_i = (\xi_{i1}, \xi_{i2}, \dots, \xi_{is})$

$$(6) \quad w_k(x) = \prod_{\substack{i=1, \\ i \neq k}}^N d^2(x, \xi_i), \quad \text{with} \quad d^2(x, \xi_i) = \sum_{j=1}^s (x_j - \xi_{ij})^2.$$

Consequently, from (5) $g_k(x)$ takes the form

$$(7) \quad g_k(x) = \frac{w_k(x)}{\sum_{j=1}^N w_j(x)} = \frac{\prod_{\substack{i=1, \\ i \neq k}}^N d^2(x, \xi_i)}{\sum_{j=1}^N \prod_{\substack{i=1, \\ i \neq j}}^N d^2(x, \xi_i)}.$$

However, different choices are possible and worthy of careful consideration, in view of their numerical performance in applications.

As a result of rather tedious algebraic manipulations detailed in [4], we get for $p = 1, \dots, s$

$$(8) \quad \frac{\partial^2 F(\xi_i)}{\partial x_p^2} = \frac{1}{w_i(\xi_i)} \left(\sum_{\substack{m=1, \\ m \neq i}}^N c_m \frac{\partial^2 w_m(\xi_i)}{\partial x_p^2} - c_i \sum_{\substack{k=1, \\ k \neq i}}^N \frac{\partial^2 w_k(\xi_i)}{\partial x_p^2} \right),$$

where

$$(9) \quad w_m(\xi_i) = \prod_{\substack{j=1, \\ j \neq m}}^N d^2(\xi_i, \xi_j), \quad \text{if } m = i; \quad w_m(\xi_i) = 0, \quad \text{if } m \neq i;$$

and

$$(10) \quad \begin{aligned} \frac{\partial^2 w_m(\xi_i)}{\partial x_p^2} &= \sum_{\substack{k=1, \\ k \neq m}}^N \left[2 \prod_{\substack{j=1, \\ j \neq m, k}}^N d^2(\xi_i, \xi_j) + 4(\xi_{ip} - \xi_{kp}) \sum_{\substack{h=1, \\ h \neq m, k}}^N (\xi_{ip} - \xi_{hp}) \times \right. \\ &\quad \left. \times \prod_{\substack{j=1, \\ j \neq m, k, h}}^N d^2(\xi_i, \xi_j) \right], \quad \text{if } m = i; \\ \frac{\partial^2 w_m(\xi_i)}{\partial x_p^2} &= 2 \prod_{\substack{j=1, \\ j \neq m, i}}^N d^2(\xi_i, \xi_j), \quad \text{if } m \neq i. \end{aligned}$$

Note that

$$\frac{\partial^2 w_m(\xi_i)}{\partial x_p^2} = \frac{\partial^2 w_m(\xi_i)}{\partial x_1^2}, \quad p = 1, \dots, s.$$

From (4) and (8) we have that the condition to be satisfied by $F(x)$ at any node $\xi_i \in \Omega$ is given by

$$\frac{-1}{w_i(\xi_i)} \left\{ \sum_{\substack{m=1, \\ m \neq i}}^N c_m s \frac{\partial^2 w_m(\xi_i)}{\partial x_1^2} - c_i \sum_{\substack{k=1, \\ k \neq i}}^N s \frac{\partial^2 w_k(\xi_i)}{\partial x_1^2} \right\} = h(\xi_i),$$

whereas at any node $\xi_i \in \Sigma$ the Dirichlet condition yields

$$F(\xi_i) = c_i = k(\xi_i).$$

Hence, the system of linear equations $Ac = b$, obtained by discretization of the Dirichlet problem, is

$$(11) \quad \sum_{m=1}^N a_{im}c_m = b_i, \quad i = 1, \dots, N,$$

where

(a) for $\xi_i \in \Omega$: $b_i = h(\xi_i)$ and

$$(12) \quad a_{im} = \begin{cases} -s \frac{1}{w_i(\xi_i)} \frac{\partial^2 w_m(\xi_i)}{\partial x_1^2}, & \text{if } m \neq i, \\ s \frac{1}{w_i(\xi_i)} \sum_{\substack{k=1, \\ k \neq i}}^N \frac{\partial^2 w_k(\xi_i)}{\partial x_1^2}, & \text{if } m = i; \end{cases}$$

(b) for $\xi_i \in \Sigma$: $b_i = k(\xi_i)$ and $a_{im} = \delta_{im}$.

If A is nonsingular, the solution of the $N \times N$ system $Ac = b$ is a vector c whose components $c_i, (i = 1, \dots, N)$, approximate the quantities appearing in the expression (3) of the interpolation operator F .

4 – Properties of Discretization Scheme

If N_1 nodes belong to Ω and N_2 to Σ , with $N = N_1 + N_2$, it is convenient to order the equations of the system (11) such that the first equations correspond to the first N_1 nodes. Thus, taking into account the point (b) above, the system can be rewritten as

$$\sum_{m=1}^{N_1} a_{im}c_m + \sum_{n=N_1+1}^N a_{in}c_n = h(\xi_i), \quad \text{for } i = 1, \dots, N_1,$$

$$c_j = k(\xi_j), \quad \text{for } j = N_1 + 1, \dots, N,$$

which is equivalent to

$$\sum_{m=1}^{N_1} a_{im}c_m = h(\xi_i) - \sum_{n=N_1+1}^N a_{in}k(\xi_n), \quad \text{for } i = 1, \dots, N_1.$$

This relation shows that the initial $N \times N$ system $Ac = b$ can be reduced to an $N_1 \times N_1$ system $\tilde{A}\tilde{c} = \tilde{b}$, namely

$$(13) \quad \sum_{m=1}^{N_1} \tilde{a}_{im}\tilde{c}_m = \tilde{b}_i, \quad \text{for } i = 1, \dots, N_1,$$

where

$$(14) \quad \begin{aligned} \tilde{a}_{im} &= a_{im}, & \text{for } i, m &= 1, \dots, N_1, \\ \tilde{c}_m &= c_m, & \text{for } m &= 1, \dots, N_1, \\ \tilde{b}_i &= h(\xi_i) - \sum_{n=N_1+1}^N a_{in}k(\xi_n), & \text{for } i &= 1, \dots, N_1. \end{aligned}$$

We note that the dimension of the system depends only from the N_1 internal nodes. Nevertheless, the N_2 nodes on the boundary increase a little the computational effort to obtain the terms \tilde{b}_i in (14).

For $i, m = 1, \dots, N_1, i \neq m$,

$$(15) \quad \tilde{a}_{im} = \frac{-s}{w_i(\xi_i)} \frac{\partial^2 w_m(\xi_i)}{\partial x_1^2} = \frac{-s}{\prod_{\substack{j=1, \\ j \neq i}}^N d^2(\xi_i, \xi_j)} 2 \prod_{\substack{j=1, \\ j \neq m, i}}^N d^2(\xi_i, \xi_j) = \frac{-2s}{d^2(\xi_i, \xi_m)},$$

which shows the symmetry of \tilde{A} .

The matrix \tilde{A} is strictly diagonally dominant. In fact, the expression of the i th diagonal element \tilde{a}_{ii} of \tilde{A} can be rewritten by (9) and (10), in the form

$$(16) \quad \tilde{a}_{ii} = \frac{1}{\prod_{\substack{j=1, \\ j \neq i}}^N d^2(\xi_i, \xi_j)} \sum_{\substack{k=1, \\ k \neq i}}^N 2s \prod_{\substack{j=1, \\ j \neq k, i}}^N d^2(\xi_i, \xi_j) = 2s \sum_{\substack{k=1, \\ k \neq i}}^N \frac{1}{d^2(\xi_i, \xi_k)}.$$

On the other hand, the sum of entries in the i th row, omitting the diagonal term, is

$$(17) \quad \sum_{k=1, k \neq i}^{N_1} \tilde{a}_{ik} = -\frac{\sum_{k=1, k \neq i}^{N_1} 2s \prod_{j=1, j \neq k, i}^N d^2(\xi_i, \xi_j)}{\prod_{j=1, j \neq i}^N d^2(\xi_i, \xi_j)} = -2s \sum_{k=1, k \neq i}^{N_1} \frac{1}{d^2(\xi_i, \xi_k)}.$$

Comparing the quantities in (16) and (17), we have the inequalities

$$|\tilde{a}_{ii}| > \sum_{k=1, k \neq i}^{N_1} |\tilde{a}_{ik}| \quad \text{for all } i = 1, \dots, N_1,$$

because we sum N terms in (16) but only the first N_1 of them in (17).

Since a strictly diagonally dominant matrix is nonsingular, the system $\tilde{A}\tilde{c} = \tilde{b}$ has a unique solution.

5 – A More Localizing Weight

A crucial point in the considered scheme is the choice of the weight. Actually, the approximation accuracy obtained by using (6) is unsatisfactory, at least considering small domains, since the root mean square error (RMSE) and the maximum absolute error (MAE) are quite high in comparison with those arising from approximation with MQs. The reason is to be searched in the behavior of the basis function $g_k(x)$ in (7), which is not sufficiently localizing. In fact, $g_k(x)$ can be rewritten as

$$g_k(x) = \begin{cases} \frac{1/d^2(x, \xi_k)}{\sum_{j=1}^N 1/d^2(x, \xi_j)}, & \text{if } x \neq \xi_k, \\ 1, & \text{if } x = \xi_k, \end{cases}$$

that shows the connection between the behaviors of $g_k(x)$ and

$$\phi(d^2(x, \xi_k)) = 1/d^2(x, \xi_k).$$

Now, the latter is too sensitive to the effects of any node ξ_k relatively far from the interpolation point x and, in particular, it happens when the considered distances are less than one. From a practical viewpoint, it is easier to consider the function $\phi(t) = 1/t^2$ instead of $\phi(d^2(x, \xi_k))$.

As a matter of fact, if the number N of nodes is large in comparison with the domain extension, it may be deemed expedient to strongly localize the weight so that more distant points do not work. A possible solution would consist in introducing in (3) localizing factors with reduced compact supports, but in such a way as to conserve the continuity of the second derivatives of the weights. Otherwise, one could consider beside (6) other weights which are strongly decaying as distance increases. Up to now, we have mainly explored functions of the second type.

A weight to be considered to improve the scheme is

$$(18) \quad \hat{w}_m(x) = \prod_{\substack{j=1, \\ j \neq m}}^N \{\exp[\alpha d^2(x, \xi_j)] - 1\}, \quad \alpha \geq 1,$$

related to the function $\hat{\phi}(t) = 1/[\exp(\alpha t^2) - 1]$, whose localizing effect increases with α . The discretization scheme by using the weight (18) still enjoys all the properties discussed above. In particular, relations (12) (remembering (15) and (16)) become

$$a_{im} = \begin{cases} -\frac{1}{\hat{w}_i(\xi_i)} s \frac{\partial^2 \hat{w}_m(\xi_i)}{\partial x_1^2} = \frac{-2s\alpha}{d^2(\xi_i, \xi_m)}, & \text{if } m \neq i, \\ \frac{1}{\hat{w}_i(\xi_i)} \sum_{\substack{k=1, \\ k \neq i}}^N s \frac{\partial^2 \hat{w}_k(\xi_i)}{\partial x_1^2} = 2s\alpha \sum_{\substack{k=1, \\ k \neq i}}^N \frac{1}{d^2(\xi_i, \xi_k)}, & \text{if } m = i. \end{cases}$$

It must be noted that a drawback of using parameters in the weights is due to the requirement of determining their optimal values.

6 – Solving the Linear System

Since the system matrix \tilde{A} is strictly diagonally dominant, Gaussian elimination algorithm can be applied to solve the system (13) without row or column interchanges, and the computations are stable with respect to the growth of roundoff errors [[12], pp. 181-182]. Since all the reduced matrices $\tilde{A}^{(k)}$ given by the algorithm are symmetric, the amount of work for the decomposition is approximately halved, that is, $O(n^3/6)$ multiplications/divisions and as many additions/subtractions are required.

Gaussian elimination ensures the factorization $\tilde{A} = LU$, where L is a lower triangular matrix with ones on the main diagonal and U is an upper triangular matrix. Other factorization methods can also be considered as the factorization LDL^T , where L is lower triangular with ones on its diagonal and D is the diagonal matrix with $a_{11}, a_{22}, \dots, a_{nn}$ on its diagonal, and, since the matrix is positive definite, Choleski's factorization LL^T , where L has positive diagonal elements. They require computational efforts of the same order as Gaussian elimination, but the last is more easily handled in order to apply the iterative refinement [11], [9], [12].

The numerical stability allows to handle large systems and, in case of very large systems, Gaussian elimination can be efficiently performed in a parallel processing environment [10]. This feature is very important when the space dimension s is larger than two.

For the direct solution of linear systems we used subroutines given in LAPACK, because it represents a standard benchmark and enjoys an excellent documentation. Moreover, accompanying LAPACK is the set of lower-level operations called BLAS which has to be considered for implementation on both shared and local memory multiprocessors. Up to now, however, considering equations in two variables, we have done only serial computations.

7 – Smoothing Resulting Surfaces

After solving the system $\tilde{A}\tilde{c} = \tilde{b}$ and obtaining the values of all the coefficients c_m , ($m = 1, \dots, N$), in the expression of the approximation operator $F(x)$, one can proceed to obtain those values of $F(x)$ which are of interest or, rather, to give an approximate representation (possibly graphical) of the solution of the Dirichlet problem.

A computational problem arises from the very definition of $F(x)$ which is from (3) and (7)

$$(19) \quad F(x) = \sum_{k=1}^N c_k g_k(x) = \sum_{k=1}^N c_k \frac{\prod_{\substack{i=1, \\ i \neq k}}^N d^2(x, \xi_i)}{\sum_{j=1}^n \prod_{\substack{i=1, \\ i \neq j}}^N d^2(x, \xi_j)}.$$

The question is if the product form (19) of the operator achieves more numerical stability than the equivalent barycentric form

$$(20) \quad F(x) = \begin{cases} \frac{\sum_{k=1}^N c_k \frac{1/d^2(x, \xi_k)}{N}}{\sum_{j=1}^N 1/d^2(x, \xi_j)}, & \text{if } x \neq \xi_k, \\ c_k, & \text{if } x = \xi_k, \end{cases}$$

or viceversa. SCHNEIDER and WERNER [18] at first observe that the operator in one dimension may be considered for $p = 2$ a rational Hermite interpolation, i.e., the derivative is approximated (rather arbitrarily) by 0 at the data sites. Then they state that the barycentric form offers the advantage of a remarkable numerical stability: even in the presence of rounding errors, which may occur during the computation of c_k , the interpolation property is maintained.

As a matter of fact, in the practice of numerical calculation, the barycentric formula is definitely preferred (see more considerations in [1]). Obviously suitable tricks must be adopted to control the growth of rounding and truncation errors when x is very close to ξ_k .

In the graphical representation of $F(x)$ a drawback is given by the appearance of flat spots at the data points, since the partial derivatives of $F(x)$ vanish there (see [3]).

To avoid this generally undesirable property, we construct local approximants $Q_k(x)$ to $F(x)$ at ξ_k , obtained by means of the moving weighted least-squares method using weight functions with reduced compact support. These local approximations are to be used instead of the coefficients c_k in order to get a smoother surface. So the operator $F(x)$ is expressed as a convex combination of the local approximants

$$F(x) = \sum_{k=1}^N Q_k(x) \frac{w_k(x)}{\sum_{j=1}^N w_j(x)}.$$

Best performance is achieved by using for every node ξ_k a paraboloid $Q_k(x)$ which interpolates at the node.

8 – Boundary Effects

A common feature in all RBF approximations is how relatively inaccurate they are at boundaries. This accuracy degradation near boundaries in many cases severely limits the utility of methods based on RBFs. Actually, large boundary-induced errors of this type will contaminate less or more the solution everywhere across the domain [8]. In applying MQ-RBFs to the solution of PDEs the residual error is typically largest by one or two orders near the boundary compared to the residual error in the domain far from the boundary. An improvement has been proposed which consists in adding an additional set of nodes, lying inside or outside of the domain, and correspondingly in adding an additional set of collocation equations obtained via collocation of the PDE on the boundary [7]. This PDE collocation on the boundary reduces dramatically the residual.

Our scheme allows to increase considerably the number of nodes collocated on the boundary, and this is done in a simple way with a very reduced computational cost. This feature could be particularly useful to control the difficulties possibly arising near the domain boundary.

9 – Numerical Results

In the following test examples we restrict ourselves to two dimensional Poisson and Laplace problems whose analytic solution are available. In all cases we use both MQ-RBF and CRBI approximations of the unknown function u . In particular CRBI method is applied both with the basic weight (6) and the localizing exponential weight (18).

EXAMPLE 1. We consider the Poisson problem studied in [6]

$$\begin{aligned}\Delta u(x, y) &= f(x, y), & (x, y) \in \Omega, \\ u(x, y) &= g(x, y), & (x, y) \in \Sigma,\end{aligned}$$

where $\Omega \cup \Sigma = [1, 2] \times [1, 2]$, the inhomogeneous term $f(x, y)$ is given by

$$\begin{aligned}f(x, y) &= -\frac{751\pi^2}{144} \sin \frac{\pi x}{6} \sin \frac{7\pi x}{4} \sin \frac{3\pi y}{4} \sin \frac{5\pi y}{4} + \frac{7\pi^2}{12} \cos \frac{\pi x}{6} \cos \frac{7\pi x}{4} \times \\ &\times \sin \frac{3\pi y}{4} \sin \frac{5\pi y}{4} + \frac{15\pi^2}{8} \sin \frac{\pi x}{6} \sin \frac{7\pi x}{4} \cos \frac{3\pi y}{4} \cos \frac{5\pi y}{4},\end{aligned}$$

and

$$g(x, y) = \sin \frac{\pi x}{6} \sin \frac{7\pi x}{4} \sin \frac{3\pi y}{4} \sin \frac{5\pi y}{4}.$$

The exact solution $u(x, y)$ to this problem coincides with the boundary condition $g(x, y)$.

For this test we selected various uniform distributions of collocation points in the domain $[1, 2] \times [1, 2]$. Fig. 1 shows a uniform distribution on a grid of 81 collocation points. We solved the above problem using the MQ-RBF method with a shape parameter $c = 1$ and the CRBI method. The resulting algebraic systems were solved using Gauss elimination. Due to uncertainty of how to choose the values of the parameters for RBFs and CRBIs, we do not looked for

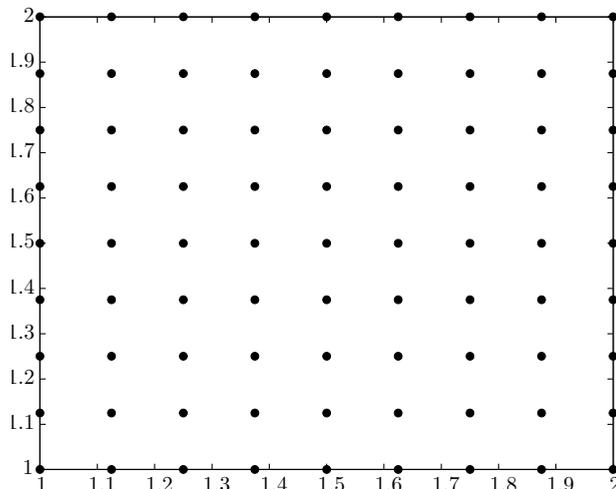


Fig. 1: Uniform distribution of 81 collocation points.

their optimal values. Simply, in this preliminary investigation, we set $c = 1$ as in [17] and α roughly proportional to the number of collocation points.

In Table 1 are listed the dimensions and the approximate condition numbers of discretization matrices. The subscripts K, C and Ce refer to Kansa's scheme and to CRBI based schemes with the two different weights, respectively.

TABLE 1: *Matrix dimensions (MD) and condition numbers (CN) for some uniform distributions.*

N	6 × 6	8 × 8	10 × 10	12 × 12
MD _K	36	64	100	144
CN _K	9.6026e07	1.8021e11	2.8824e14	5.7559e17
MD _C	16	36	64	100
CN _C	3.2483e00	4.6815e00	6.1136e00	7.5405e00
MD _{Ce}	16	36	64	100
CN _{Ce}	8.2780e00	1.5853e01	2.8020e01	4.2069e01

Table 1 shows the well-known disadvantage of ill-conditioning of the discretization matrices arising from the MQ-RBFs method and, on the contrary, how the matrices for the CRBI method have much smaller condition numbers. Moreover, as already pointed-out, the matrix dimension in CRBI approach is equal to the number of internal collocation points in the domain.

TABLE 2: *Root mean square errors (RMSE) and maximum absolute errors (MAE) for some uniform distributions.*

N	6 × 6	9 × 9	11 × 11	16 × 16	21 × 21
RMSE _K	5.4416e−3	2.8910e−4	4.4902e−5	9.3432e−7	1.7773e−5
MAE _K	1.2663e−2	8.8215e−4	1.6874e−4	2.9001e−6	5.2733e−5
RMSE _C	8.5317e−2	1.1795e−1	1.3046e−1	1.5005e−1	1.6212e−1
MAE _C	3.3857e−1	4.5457e−1	4.7875e−1	5.1510e−1	5.4705e−1
RMSE _{Ce}	6.8051e−3	2.8701e−3	1.8761e−3	8.5708e−4	4.9346e−4
MAE _{Ce}	1.7341e−2	7.1686e−3	4.8149e−3	2.0704e−3	1.3317e−3

In order to test accuracy, we increased the number of collocation points considering several uniform grids in the domain. In Table 2 we list some of the results obtained. The root mean square errors and the absolute maximum errors, computed on the set of the collocation points, are better for the MQ-RBF method, but they begin to make worse as the number of collocation points increases starting from the 41×41 grid. For example, considering the uniform 51×51 grid (2601 collocation points) the RMSE goes down to $2.5025e-3$ and the MAE to $5.7033e-3$.

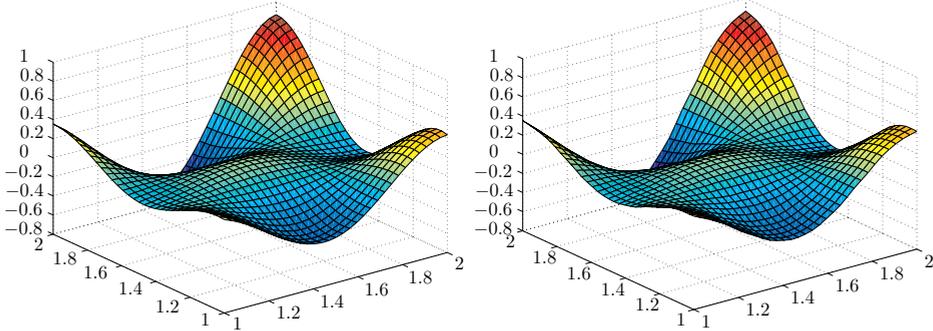


Fig. 2: The numerical solutions on the 33×33 grid obtained with MQ-RBF method and CRBI method with localizing weight using 81 collocation points.

Concerning the CRBI method, we observe that in this test example it is not accurate as MQ-RBF method, but it slowly improves when we use the exponential localizing weight. Instead, the accuracy of the numerical solution of CRBI with the basic weight is not satisfying. We must say that for this test problem is not necessary to consider a large set of collocation points (as show the plots in Fig. 2 and Fig. 3) but these remarks are important if we will consider a problem which require a large number of points to be numerically solved.

Fig. 2 shows the approximations obtained with the MQ-RBF method and the CRBI method with the localizing weight. We used 81 collocation points and the approximations are computed on a 33×33 grid. Considering the accuracy of the numerical solutions, the plot of the exact solution is not given. In Fig. 3 are

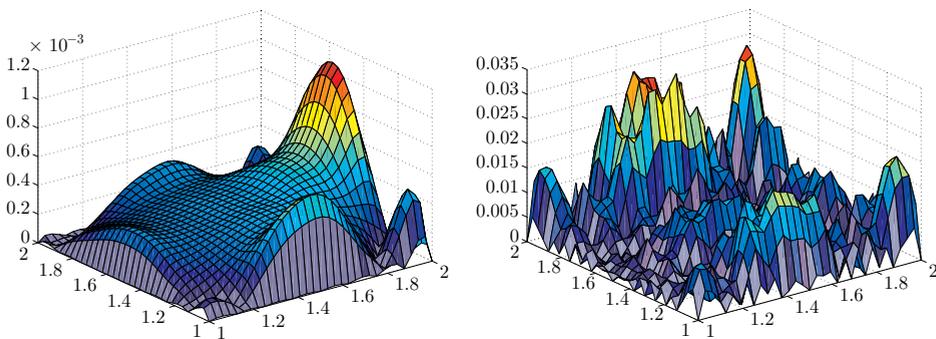


Fig. 3: The absolute errors computed on the 33×33 grid obtained with MQ-RBF method and CRBI method with localizing weight.

plotted the absolute errors in the two cases. Comparing the plots, we observe that the error is more uniformly distributed in the domain when CRBI method is used, while the error is more localized on the boundary for MQ-RBF method.

EXAMPLE 2. Let us consider the Laplace equation which models the steady state temperature distribution in a thin plate [5]

$$\Delta u(x, y) = 0, \quad (x, y) \in (0, 1) \times (0, 1)$$

with the Dirichlet boundary conditions

$$u(x, 0) = 1, \quad u(x, 1) = 0, \quad u(0, y) = 0, \quad u(1, y) = 0.$$

The analytic solution of this problem is given by

$$u(x, y) = \left(\frac{4}{\pi}\right) \sum_{i=0}^{\infty} \left\{ \left[\frac{1}{2i+1} \right] \sin[(2i+1)\pi x] \times \right. \\ \left. \times \sinh[(1-y)(2i+1)\pi] \cdot \cosh[(2i+1)\pi] \right\}.$$

For this test we selected various gridded and scattered data sets in the domain $[0, 1] \times [0, 1]$. Some of the data sets used are plotted in Fig. 4 and Fig. 5. The scattered data sets were generated randomly, selecting some of the points on the boundary.

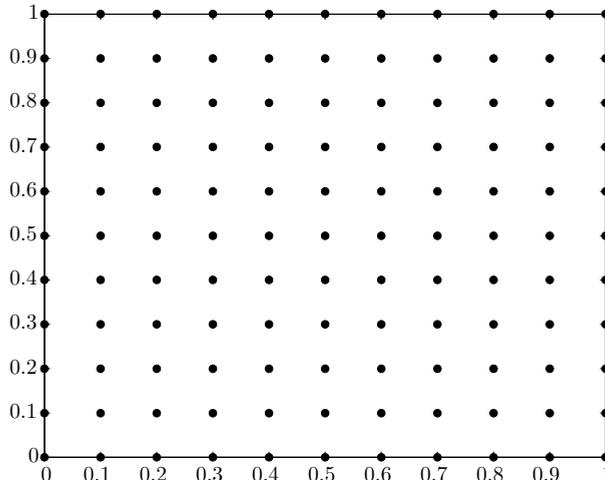


Fig. 4: Gridded data set of 121 collocation points.

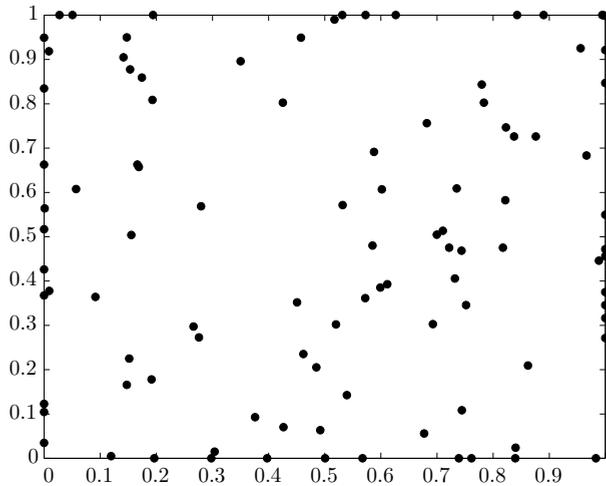


Fig. 5: Scattered data set of 100 collocation points.

In Table 3 we report the matrix dimensions and the approximated condition numbers for some uniform distributions. We remark again that the matrices arising from our approximation scheme are smaller and better conditioned in comparison with those from MQ-RBF scheme.

TABLE 3: *Matrix dimensions (MD) and condition numbers (CN) for some gridded data sets.*

N	5 × 5	7 × 7	9 × 9	11 × 11
MD _K	25	49	81	121
CN _K	1.9153e06	4.0658e09	7.0456e12	7.4703e15
MD _C	9	25	49	81
CN _C	2.5342e00	3.9645e00	5.3981e00	6.8281e00
MD _{Ce}	9	25	49	81
CN _{Ce}	5.8284e00	1.3922e01	2.5233e01	3.9137e01

Tables 4 and 5 show the values of the root mean square and absolute maximum errors computed on the nodes, obtained with gridded data sets and scattered data sets, respectively. The number of boundary data points of the scattered data sets is indicated in brackets. Both tables show that in this example the CRBI method performs better than the MQ-RBF one. In particular, increasing the number of collocation points the errors increase for the latter, while on the contrary our scheme slowly improves.

TABLE 4: *Root mean square errors (RMSE) and absolute maximum errors (MAE) for some gridded data sets.*

N	6 × 6	8 × 8	10 × 10	12 × 12	14 × 14
RMSE _K	1.4017e−2	1.6773e−2	1.9210e−2	2.6067e−2	7.4839e−2
MAE _K	3.3644e−2	4.7334e−2	5.8919e−2	9.4331e−2	3.5157e−1
RMSE _C	6.5439e−2	8.3554e−2	9.8267e−2	1.0984e−1	1.1944e−1
MAE _C	2.1083e−1	2.6841e−1	3.0426e−1	3.3161e−1	3.5143e−1
RMSE _{Ce}	3.2226e−3	2.5500e−3	2.0659e−3	1.6933e−3	1.5749e−3
MAE _{Ce}	1.0400e−2	7.5742e−3	6.6763e−3	6.4897e−3	5.2316e−3

TABLE 5: *Root mean square errors (RMSE) and absolute maximum errors (MAE) for some scattered data sets.*

N(N₂)	60(28)	100(36)	140(44)	240(60)	400(80)
RMSE _K	6.2720e−1	2.3253e−2	1.6883e+0	1.0566e−1	3.6538e−1
MAE _K	2.0397e+0	1.3569e−1	1.8115e+1	7.7725e−1	9.8808e−1
RMSE _C	6.3918e−2	8.8239e−2	1.1258e−1	1.2086e−1	1.2542e−1
MAE _C	2.6953e−1	4.7216e−1	3.6631e−1	3.8096e−1	5.2064e−1
RMSE _{Ce}	4.7033e−2	5.3307e−2	5.3559e−2	5.8081e−2	5.6451e−2
MAE _{Ce}	1.7072e−1	3.8215e−1	1.5142e−1	4.6622e−1	3.7112e−1

Fig.6 shows the plot of the exact solution. In Fig.7 and Fig.8 are plotted the numerical approximations obtained with the MQ-RBF method and the CRBI

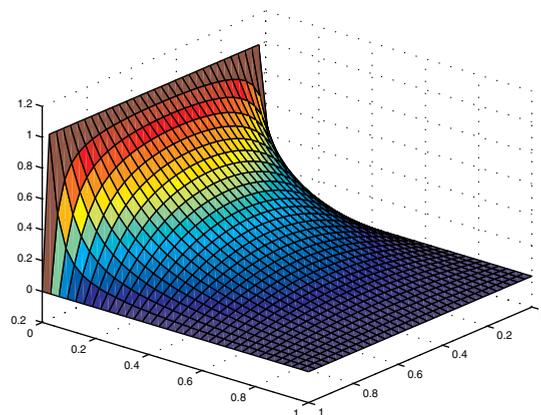


Fig. 6: Exact solution computed on the 26×26 grid.

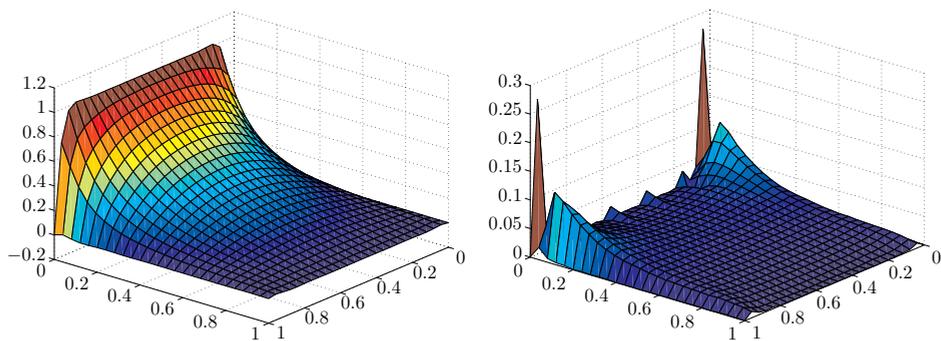


Fig. 7: Approximation with the MQ-RBF method using 121 collocation points and absolute error.

method with localizing weight using 121 collocation points and the related absolute errors computed on a 26×26 grid. We remark that, in comparison with the solution of the Poisson equation in Example 1, this solution has a behaviour difficult to be captured near the boundaries, where the values of the solution are prescribed equal to zero and one by the Dirichlet conditions. A comparison between the two plots points out that the CRBI method is more accurate near the boundaries where the solution is constrained to zero, while the MQ-RBF method is better approximating near the boundary $y = 0$, when the number of collocation points used is equal for the two scheme.

To improve the performance of the CRBI method we increase the number of collocation points. Fig. 9 shows the approximation obtained with 441 points and absolute errors. Extending data set of collocation points is justified by the properties of the discretization scheme and by the numerical stability.

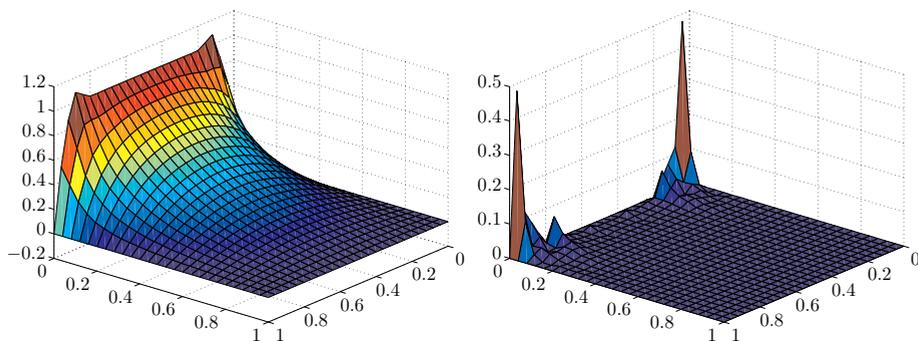


Fig. 8: Approximation with the CRBI method with localizing weight using 121 collocation points and absolute error.

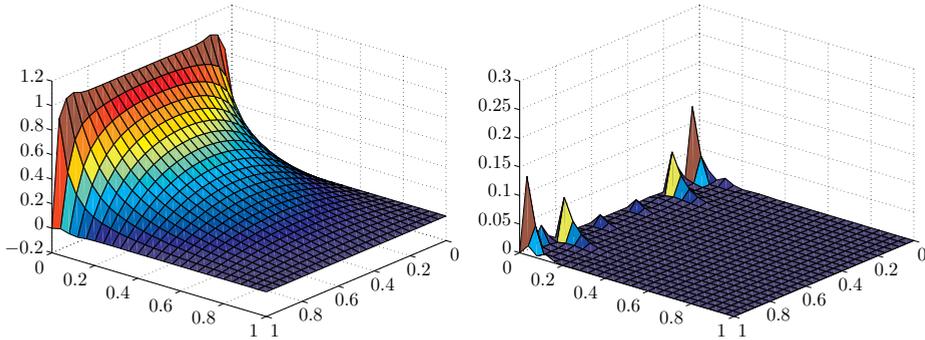


Fig. 9: Approximation with the CRBI method with the localizing weight using 441 collocation points and absolute error.

In order to test accuracy near the boundaries, we computed the local relative error $(u_t - u_n)/u_t$ where u_t and u_n are the analytical and numerical values, respectively, at the internal collocation points.

Fig. 10 and Fig. 11 show the relative error between the analytical and the numerical solutions when we used 81 uniformly distributed collocation points.

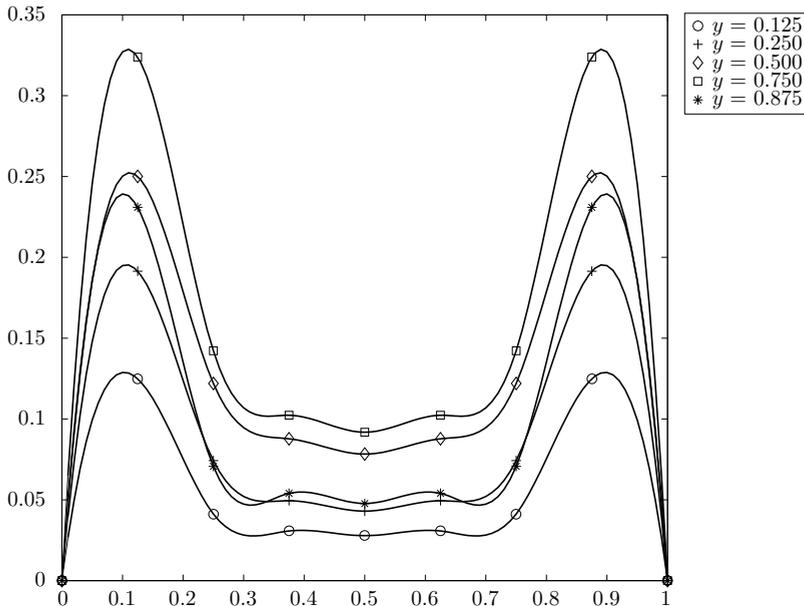


Fig. 10: Relative error obtained with the MQ-RBF method on the uniform distribution of 81 collocation points.

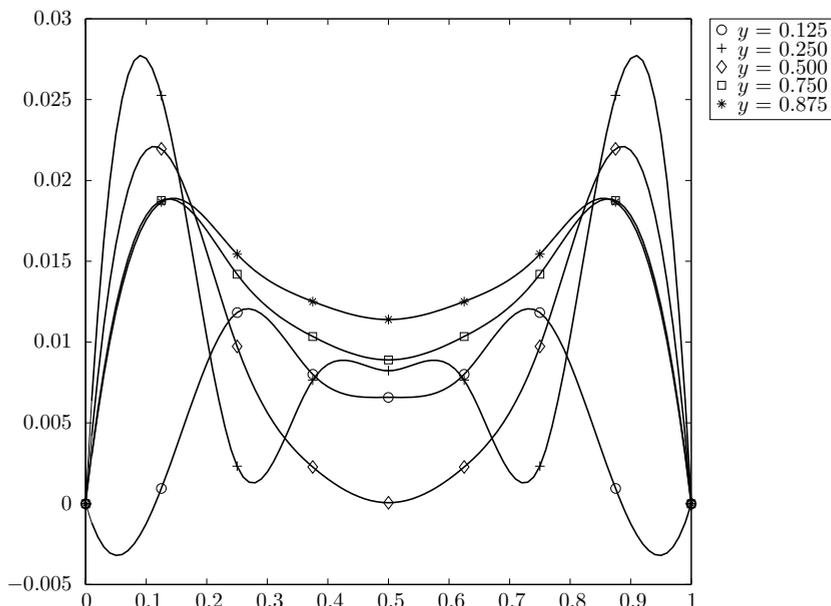


Fig. 11: Relative error obtained with the CRBI method with the localizing weight on the uniform distribution of 81 collocation points.

The numerical solutions were computed with MQ-RBF and the CRBI with localizing weight, respectively. In Fig. 10 we observe that the maximum relative errors are on the lines $y = 0.500$, $y = 0.750$, $y = 0.875$ and near the boundaries where the solution values were prescribed equal to zero; vice versa they decrease in the middle regions. Fig. 11 shows as the approximation with the CRBI method has a different behaviour. In fact the maximum relative error is more uniformly distributed in the domain. As already pointed-out in the literature on functional approximation of scattered data, the MQ function seems more appropriate to approximate rapidly varying functions, while CRBI method performs better when approximating slowly varying functions.

10 – Conclusions

Based on a theoretical establishment, a new method has been constructed to give a numerical solution to the Poisson equation. The method, which makes use of Cardinal Radial Basis Intepolation operators (CRBI), enjoys the following special features:

1. It is well-posed and numerically stable.
2. The discretization matrix is symmetric and strictly diagonally dominant (hence, positive definite).
3. Both error and sensitivity are reasonably small, that is, the method is not affected from the so-called uncertainty principle.
4. Being mesh-free and insensitive to dimension, it is particularly suitable for irregular domains and 3D problems.
5. Boundary effects can be controlled by increasing the number of collocation points on the boundary at the cost of a little computational effort.

Obviously, there are some factors affecting the accuracy of the proposed method:

1. First derivatives of the approximation operator at internal collocation points vanish so that flat spots appear on the rendered surface.
2. In general, when applied on a reduced number of collocation points (of the order of tens or hundreds), it does not work as well as MQ-RBF method.
3. The choice of the weight and the determination of optimal values of the possible parameters play important roles in the accuracy.
4. Numerically the method converges slowly and the rate of convergence has not yet been investigated theoretically.

Additional theoretical and numerical characteristics of the method as well as the application in solving other type of PDEs are currently under investigation by our research group.

REFERENCES

- [1] G. ALLASIA: *A class of interpolating positive linear operators: theoretical and computational aspects*, in: *Approximation Theory, Wavelets and Applications*, S. P. Singh (ed.), Kluwer, Dordrecht, 1995, 1-36.
- [2] G. ALLASIA: *Cardinal basis interpolation on multivariate scattered data*, *Nonlinear Analysis Forum*, (1) **6** (2001), 1-13.
- [3] G. ALLASIA: *Simultaneous interpolation and approximation by a class of multivariate positive operators*, *Numer. Algorithms*, **34** (2003), 147-158.
- [4] G. ALLASIA: *A scattered data approximation scheme for the multidimensional Poisson equation by cardinal radial basis interpolants*, *Curve and surface fitting (Saint-Malo, 2002)*, Nashboro Press, Brentwood, TN, 2003, 11-20.
- [5] H. S. CARSLAW – J. C. JAEGER: *Conduction of heat in solids*, 2nd ed., Clarendon Press, Oxford, 1959.
- [6] C. S. CHEN – M. GANESH – M. A. GOLBERG – A. H.-D. CHENG: *Multilevel compact radial functions based computational schemes for some elliptic problems*, *Computers Math. Applic.*, **43** (2002), 359-378.
- [7] A. I. FEDOSEYEV – M. J. FRIEDMAN – E. J. KANSA: *Improved multiquadric method for elliptic partial differential equations via PDE collocation on the boundary*, in: *Radial Basis Functions and Partial Differential Equations*, E. J. Kansa – Y. C. Hon (eds.), *Computers Math. Applic.*, (3-5) **43** (2002), 439-455.

- [8] B. FORNBERG – T. A. DRISCOLL – G. WRIGHT – R. CHARLES: *Observations on the behavior of radial basis function approximations near boundaries*, in: *Radial Basis Functions and Partial Differential Equations*, E. J. Kansa – Y. C. Hon (eds.), Computers Math. Applic., (3-5) **43** (2002), 473-490.
- [9] G. H. FORSYTHE – C. B. MOLER: *Computer solution of linear algebraic systems*, Prentice-Hall, Englewood Cliffs, 1967.
- [10] T. L. FREEMAN – C. PHILLIPS: *Parallel numerical algorithms*, Prentice Hall, U. K., 1992.
- [11] A. GEORGE – J. W. LIU: *Computer solution of large sparse definite positive systems*, Prentice-Hall, Englewood Cliffs, 1981.
- [12] N. HIGHAM: *Accuracy and stability of numerical algorithms*, SIAM, Philadelphia, 1996.
- [13] E. J. KANSA: *Multiquadrics-A scattered data approximation scheme with applications to computational fluid dynamics-I: Surface approximation and partial derivatives estimates*, Computers Math. Applic., (8/9) **19** (1990), 127-145.
- [14] E. J. KANSA: *Multiquadrics-A scattered data approximation scheme with applications to computational fluid dynamics-II: Solution to parabolic, hyperbolic and elliptic partial differential equations*, Computers Math. Applic., (8/9) **19** (1990), 147-161.
- [15] E. J. KANSA – Y. C. HON: *Circumventing the ill-conditioning problem with multiquadric radial basis functions: Applications to elliptic partial differential equations*, Computers Math. Applic., **39** (2000), 123-137.
- [16] E. J. KANSA – Y. C. HON (eds.): *Radial Basis Functions and Partial Differential Equations*, Computers Math. Applic., (3-5) **43** (2002), 275-619.
- [17] H. POWER – V. BARRACO: *A comparison analysis between unsymmetric and symmetric radial basis function collocation methods for the numerical solution of partial differential equations*, in: *Radial basis functions and partial differential equations*, E. J. Kansa – Y. C. Hon (eds.), Computers Math. Applic., (3) **43** (2002), 551-583.
- [18] C. SCHNEIDER – W. WERNER: *Hermite interpolation: the barycentric approach*, Computing, **46** (1991), 35-51.

*Lavoro pervenuto alla redazione il 15 febbraio 2003
ed accettato per la pubblicazione il 20 dicembre 2003.
Bozze licenziate il 6 dicembre 2004*

INDIRIZZO DEGLI AUTORI:

Giampietro Allasia – Alessandra De Rossi – Department of Mathematics – University of Turin
– Via Carlo Alberto 10 – 10123 Torino – Italy
E-mail: giampietro.allasia@unito.it alessandra.derossi@unito.it