

Computation of Optimal Shape Parameter for Multipoint Gaussian RBF-FD Approximation of Poisson Equation

Oleg Davydov* and Dang Thi Oanh^{†‡}

August 25, 2011

Abstract

We study the behaviour of the optimal shape parameter for the meshless Gaussian RBF-FD method with multipoint stencils on irregular centres for Poisson equations with smooth solution. Numerical experiments show that its value does not depend significantly on the density of the centres. Therefore an algorithm based on comparison of RBF solutions on two nested sets of centres effectively computes a near-optimal shape parameter and helps to obtain more accurate solutions of the PDE than those computed by linear finite elements on the same discretisation centres and with comparable density/bandwidth of the system matrix.

1 Introduction

In this paper we continue investigation of the multipoint stencil RBF-FD method introduced in [3]. Recall that RBF-FD is a generalised finite difference method for numerical solution of partial differential equations based on numerical differentiation stencils on irregular centres arising from interpolation with radial basis functions. This is a promising, truly meshless approach that has only recently become a research subject, see for example [1, 8, 10, 11, 13]. Even though little has been done on its theoretical justification, numerical evidence suggests that this method produces highly accurate solutions, comparable with and often even outperforming the accuracy of the well established mesh based methods for the same number of degrees of freedom and density of the system matrix.

*Department of Mathematics, University of Strathclyde, 26 Richmond Street, Glasgow G1 1XH, Scotland, oleg.davydov@strath.ac.uk

[†]Department of Computer Science, Faculty of Information Technology - Thai Nguyen University, Quyet Thang Commune, Thai Nguyen City, Viet Nam, dtoanh@tnu.edu.vn

[‡]The second author was supported in part by the National Foundation for Science and Technology Development (NAFOSTED) and a Natural Science Research Project of the Ministry of Education and Training.

The ordinary (single point) RBF-FD method for Poisson equation discretises the Laplace operator at a centre x_0 by evaluating the RBF interpolant s satisfying $s(x_i) = u(x_i)$, $i = 0, \dots, n$, where $\{x_1, \dots, x_n\}$ is a set of centres close to x_0 ,

$$\Delta u(x_0) \approx \Delta s(x_0) = \sum_{i=0}^n w_i u(x_i).$$

The stencil $w = [w_0 \dots, w_n]$ depends only on the differences $x_1 - x_0, \dots, x_n - x_0$ and on the choice of RBF. Hermite RBF stencils [13] rely on Hermite RBF interpolants s_H and corresponding numerical differentiation formulas of the type

$$\Delta u(x_0) \approx \Delta s_H(x_0) = \sum_{i=0}^n w_i u(x_i) + \sum_{i=1}^n \tilde{w}_i \Delta u(x_i).$$

Multipoint stencils are obtained by discretising a linear combination of shifted Laplacians $Du := \sum_{k=1}^{\ell} \sigma_k \Delta u(\cdot - x_0 + y_k)$, where y_1, \dots, y_{ℓ} are collocation centres close to x_0 ,

$$\sum_{k=1}^{\ell} \sigma_k \Delta u(y_k) = Du(x_0) \approx Ds(x_0) = \sum_{i=0}^n w_i u(x_i).$$

To some extent, multipoint stencils resemble discretisations arising (after applying a quadrature rule) from the finite element method.

In [3] we suggested meshless algorithms for the selection of stencil supports $\{x_1, \dots, x_n\}$ for RBF-FD methods as well as for the adaptive local refinement of the sets of centres and presented numerical experiments on typical test problems usually employed for assessing adaptive finite element methods. The performance of single point and multipoint stencils was comparable in these experiments. However, in [3] we did not address the question of optimal shape parameter c , which is known to be crucial for RBF methods, see [5, Chapter 17] and references therein. In [3] we adopted a simple strategy of choosing a ‘safe’ value as small as possible such that the RBF matrices used to compute the stencils w are still numerically nonsingular. Recent paper [4] is devoted to the optimal shape parameter for the standard single point stencils in the case of Gaussian RBF $\varphi(r) = e^{-c^2 r^2}$, where the stencils can be computed for arbitrary $c \geq 0$ by using QR method of [6, 7].

In this paper we investigate optimal shape parameter for the multipoint stencils. Similar to [4], numerical evidence is presented that the optimal shape parameter is different for different test problems, and therefore cannot be estimated from the distribution of centres alone. The accuracy achieved with optimal shape parameter is often several times higher than the one with a ‘safe’ value, or in the ‘flat limit’ case of $c = 0$. We observe that for non-adaptive (‘quasi-uniform’) centres the optimal shape parameter does not depend significantly on the density of the centres, and therefore the multilevel algorithm for the estimation of near-optimal shape parameter introduced in [4] performs very well for the multipoint stencils. Moreover, in most test cases multipoint stencil solutions are significantly more accurate than their single point counterparts, which may make them preferable for certain applications even though they are more expensive to compute.

The paper is organised as follows. In Section 2 we discuss general discretisation methods for the Dirichlet problem obtained with the help of numerical differentiation formulae, and derive both single point and multipoint RBF-FD stencils. Section 3 is devoted to the results of the numerical tests on the optimal shape parameter, and Section 4 presents an algorithm for the computation of near-optimal shape parameter and results of its numerical testing.

2 Single point and multipoint RBF-FD methods

2.1 Discretisation of Poisson equation on irregular centres

Let D be a linear differential operator, and $X = \{x_i\}_{i=0}^n$ a fixed irregular set of centres in \mathbb{R}^d . A linear *numerical differentiation formula* for the operator D ,

$$Du(x) \approx \sum_{i=0}^n w_i(x)u(x_i), \quad (1)$$

is determined by the *weights* $w_i = w_i(x)$. The vector $w = [w_0, \dots, w_n]^T$ is called *stencil*.

In the *finite difference method* stencils are used for the discretisation of partial differential equations. Consider the Dirichlet problem for the Poisson equation in a bounded domain $\Omega \subset \mathbb{R}^d$: given a function f defined on Ω , and a function g defined on $\partial\Omega$ find u such that

$$\Delta u = f \quad \text{on } \Omega, \quad (2)$$

$$u|_{\partial\Omega} = g. \quad (3)$$

This problem can be discretised with the help of differentiation formulae (1) as follows.

Let $\Xi \subset \bar{\Omega}$ be a finite set of *discretisation centres*, $\partial\Xi := \Xi \cap \partial\Omega$ and $\Xi_{\text{int}} := \Xi \setminus \partial\Xi$. Assume that for each $\zeta \in \Xi_{\text{int}}$ a set $\Xi_\zeta \subset \Xi$ is chosen such that $\zeta \in \Xi_\zeta$ and

$$\Xi = \bigcup_{\zeta \in \Xi_{\text{int}}} \Xi_\zeta. \quad (4)$$

For each $\zeta \in \Xi_{\text{int}}$, choose a linear numerical differentiation formula for Laplace operator Δ ,

$$\Delta u(\zeta) \approx \sum_{\xi \in \Xi_\zeta} w_{\zeta,\xi} u(\xi), \quad (5)$$

with stencil $[w_{\zeta,\xi}]_{\xi \in \Xi_\zeta}$, and replace (2)–(3) by the system of linear equations

$$\sum_{\xi \in \Xi_\zeta} w_{\zeta,\xi} \hat{u}(\xi) = f(\zeta), \quad \zeta \in \Xi_{\text{int}}, \quad (6)$$

$$\hat{u}(\xi) = g(\xi), \quad \xi \in \partial\Xi. \quad (7)$$

If (6)–(7) is nonsingular, then its solution $\hat{u} : \Xi \rightarrow \mathbb{R}$ can be compared with the vector $u|_{\Xi} = [u(\xi)]_{\xi \in \Xi}$ of the discretised exact solution of (2)–(3).

A standard finite difference method is obtained from the above if we take $\Omega \subset \mathbb{R}^2$ to be a square domain, Ξ a uniformly spaced grid, and (5) the classical 5-point differentiation formula for the Laplacian.

A more general scheme is obtained if we introduce a different set $\Theta \subset \Omega$ of *collocation centres* to replace Ξ_{int} in (6):

$$\sum_{\xi \in \Xi_\theta} w_{\theta,\xi} u(\xi) = f(\theta), \quad \theta \in \Theta, \quad (8)$$

where Ξ_θ and $w_{\theta,\xi}$, $\xi \in \Xi_\theta$, define a suitable numerical differentiation formula for the centre θ ,

$$\Delta u(\theta) \approx \sum_{\xi \in \Xi_\theta} w_{\theta,\xi} u(\xi). \quad (9)$$

Then the system (8),(7) is another discretisation of the problem (2)–(3).

If the cardinality of Θ is greater than that of Ξ , then the linear system (8),(7) is overdetermined, and an approximate solution can be found by the least squares method, for example. Alternatively, the number of equations can be reduced by using local averages of the equations in (8), leading to the following generalised finite difference method.

For each $\zeta \in \Xi_{\text{int}}$ choose a set $\Theta_\zeta \subset \Theta$ and weights $\sigma_{\zeta,\theta} \in \mathbb{R}$, $\theta \in \Theta_\zeta$, to define a linear combination of shifted Laplacians on Θ_ζ , and choose a numerical differentiation formula

$$\sum_{\theta \in \Theta_\zeta} \sigma_{\zeta,\theta} \Delta u(\theta) \approx \sum_{\xi \in \Xi_\zeta} w_{\zeta,\xi} u(\xi), \quad (10)$$

with support Ξ_ζ and weights $w_{\zeta,\xi}$, $\xi \in \Xi_\zeta$. Then a discretisation of (2)–(3) is given by the following linear system

$$\sum_{\xi \in \Xi_\zeta} w_{\zeta,\xi} \hat{u}(\xi) = \sum_{\theta \in \Theta_\zeta} \sigma_{\zeta,\theta} f(\theta), \quad \zeta \in \Xi_{\text{int}}; \quad \hat{u}(\xi) = g(\xi), \quad \xi \in \partial\Xi. \quad (11)$$

It turns out that the finite element discretisations of (2)–(3) can be written in the form (11) if the load vector is evaluated by a quadrature rule. Indeed, consider for example, the linear triangular finite elements. Given a conforming triangulation of a polygonal domain Ω , we denote by Ξ the set of vertices of all triangles, and by φ_ξ , $\xi \in \Xi$, the hat functions. The finite element approximation is sought in the form $\hat{u}(x) \approx \sum_{\xi \in \Xi} \hat{u}(\xi) \varphi_\xi(x)$, $x \in \Omega$, where the values $\hat{u}(\xi)$ satisfy $\hat{u}(\xi) = g(\xi)$ for $\xi \in \partial\Xi$, and

$$-\sum_{\xi \in \Xi_\zeta} \hat{u}(\xi) \int_{\Omega} \nabla \varphi_\xi \nabla \varphi_\zeta dx = \int_{\Omega} f \varphi_\zeta dx, \quad \zeta \in \Xi_{\text{int}},$$

with Ξ_ζ consisting of ζ and all vertices of the triangulation connected to ζ by an edge. The integrals in the left hand side (entries of the stiffness matrix) can be computed explicitly, whereas those in the right hand side (components of the load vector) require a quadrature formula. A standard scheme for the linear finite element method is the midpoint rule on each triangle in the support of φ_ζ . Hence

$$\int_{\Omega} f \hat{\varphi}_\zeta dx = \sum_{\theta \in \Theta_\zeta} \int_{T_\theta} f \hat{\varphi}_\zeta dx \approx \sum_{\theta \in \Theta_\zeta} \frac{\text{area}(T_\theta)}{3} f(\theta),$$

where Θ_ζ is the set of the barycentres of all triangles attached to ζ , and T_θ denotes the triangle with barycentre θ . Thus, we arrive at (11), where

$$w_{\zeta,\xi} := - \int_{\Omega} \nabla \varphi_\xi \nabla \varphi_\zeta dx, \quad \sigma_{\zeta,\theta} := \frac{\text{area}(T_\theta)}{3}. \quad (12)$$

2.2 Numerical differentiation stencils

Numerical differentiation formulae (1) and stencils for the finite difference method are usually obtained by truncating Taylor expansions, which guarantees polynomial exactness (consistency) up to certain order. On the other hand, stencils can be derived with the help of *scattered data fitting methods* as follows.

Let s be an approximation to u from scattered data given by a set of centres $X = \{x_0, \dots, x_n\} \subset \mathbb{R}^d$ and corresponding function values $u|_X = [u(x_0), \dots, u(x_n)]^T$, in the form

$$s = \sum_{i=0}^m a_i s_i, \quad (13)$$

where s_i , $i = 0, \dots, m$, are some basis functions for which Ds_i exist (and can be effectively evaluated), and the coefficient vector $a = [a_0, \dots, a_m]^T$ depends linearly on $u|_X$,

$$a_i = \sum_{j=0}^n b_{ij} u(x_j), \quad i = 0, \dots, m, \quad (14)$$

or, in matrix form $a = B \cdot u|_X$, where $B = [b_{ij}]_{i=0, j=0}^{m, n}$. Then

$$Du(x) \approx Ds(x) = \sum_{i=0}^m a_i Ds_i(x) = \sum_{j=0}^n \left(\sum_{i=0}^m b_{ij} Ds_i(x) \right) u(x_j) = \sum_{j=0}^n w_j u(x_j),$$

and we get a numerical differentiation formula (1) with a stencil $w = [w_0, \dots, w_n]^T$ given by

$$w_j = \sum_{i=0}^m b_{ij} Ds_i(x), \quad j = 0, \dots, n,$$

or in matrix form

$$w = B^T \cdot [Ds_i(x)]_{i=0}^m. \quad (15)$$

In particular, consider the case when $m = n$ and the coefficients a_j are obtained by solving a non-singular *interpolation problem*

$$s(x_i) = \sum_{j=0}^n a_j s_j(x_i) = u(x_i), \quad i = 0, \dots, n.$$

Then clearly $B = S_X^{-1}$, where $S_X := [s_j(x_i)]_{i,j=0}^n$. Therefore, the stencil w is given by

$$w = S_X^{-T} \cdot [Ds_i(x)]_{i=0}^n,$$

and can be computed by solving the linear system

$$S_X^T w = [Ds_i(x)]_{i=0}^n, \quad (16)$$

that is,

$$\sum_{i=0}^n w_i s_j(x_i) = Ds_j(x), \quad j = 0, \dots, n.$$

Clearly, many other approximation methods, for example least squares or quasi-interpolation are of the type (13)–(14) and therefore lead to numerical differentiation stencils in the form (15). Since

$$Du(x) - \sum_{j=0}^n w_j u(x_j) = D\left(u(x) - \sum_{i=0}^m a_i s_i(x)\right),$$

the accuracy of the stencil directly relates to the accuracy of the approximation method.

2.3 RBF-FD methods

A family of well performing methods for scattered data interpolation is based on *radial basis functions* (RBF), and therefore it is natural to expect that good stencils can be generated from RBF interpolation.

Let $\varphi : \mathbb{R}_+ \rightarrow \mathbb{R}$ be a positive definite function. Given any set $X = \{x_0, \dots, x_n\} \subset \mathbb{R}^d$ of centres and a function $u : \mathbb{R}^d \rightarrow \mathbb{R}$, the *RBF interpolant with a constant term* [2, 5, 12] is sought in the form

$$s(x) = \sum_{j=0}^n a_j \varphi_j(x) + c, \quad \varphi_j(x) = \Phi(x - x_j), \quad \Phi(x) := \varphi(\|x\|), \quad (17)$$

where $\|x\|$ is the Euclidean norm of x , and the coefficients a_j are chosen such that

$$s(x_i) = u(x_i), \quad i = 0, \dots, n, \quad \sum_{j=0}^n a_j = 0. \quad (18)$$

The coefficients a_j are uniquely determined as solutions of the linear system

$$\sum_{j=0}^n a_j \Phi(x_i - x_j) + c = u(x_i), \quad i = 0, \dots, n, \quad \sum_{j=0}^n a_j = 0,$$

which may be written in matrix form as

$$\begin{bmatrix} \Phi_X & \mathbf{1} \\ \mathbf{1}^T & 0 \end{bmatrix} \begin{bmatrix} a \\ c \end{bmatrix} = \begin{bmatrix} u|_X \\ 0 \end{bmatrix}, \quad \Phi_X := [\Phi(x_i - x_j)]_{i,j=0}^n, \quad \mathbf{1} := [1 \ \dots \ 1]^T.$$

The matrix Φ_X is symmetric and positive definite for any set X .

Referring to the general setting of Section 2.2, we see that here $m = n + 1$, $s_i = \varphi_i$, $i = 0, \dots, n$, $s_{n+1} = 1$, and B is obtained from the matrix $\begin{bmatrix} \Phi_X & \mathbf{1} \\ \mathbf{1}^T & 0 \end{bmatrix}^{-1}$ by removing its last column. By introducing an auxiliary real variable v , we can write (15) in the form

$$\begin{bmatrix} w \\ v \end{bmatrix} = \begin{bmatrix} \Phi_X & \mathbf{1} \\ \mathbf{1}^T & 0 \end{bmatrix}^{-1} [Ds_i(x)]_{i=0}^m,$$

which leads to the linear system

$$\begin{bmatrix} \Phi_X & \mathbf{1} \\ \mathbf{1}^T & 0 \end{bmatrix} \begin{bmatrix} w \\ v \end{bmatrix} = \begin{bmatrix} [D\varphi_i(x)]_{i=1}^n \\ 0 \end{bmatrix}, \quad (19)$$

assuming that the differential operator D annihilates constants.

Note that the stencil w satisfies $\sum_{j=0}^n w_j = 0$, which implies that the system matrix $[w_{\zeta, \xi}]$ of (6) or (11) will be weakly diagonally dominant if it is an L -matrix. This is why we insist on using RBF interpolation with a constant term even though both the constant term c and the side condition $\sum_{j=0}^n a_j = 0$ can be removed in (17)–(18) because Φ_X is nonsingular. In this case of *ordinary RBF interpolation* the coefficient v and the last equation will disappear in (19), leading to a simpler linear system $\Phi_X w = [D\varphi_i(x_0)]_{i=0}^n$ for the computation of the stencil, as in (16). However, in general, stencils obtained this way do not sum to zero.

In this paper we consider the Gaussian RBF $\varphi(r) = e^{-(cr)^2}$, which is positive definite for any value of the *shape parameter* $c > 0$. For this function, the matrix Φ_X takes the form

$$\Phi_X = [e^{-c^2 \|x_i - x_j\|^2}]_{i,j=1}^n. \quad (20)$$

The Laplacian of the Gaussian function $\Phi(x) = e^{-c^2 \|x\|^2}$ needed for the discretisation of the Poisson equation (2) is given by

$$\Delta\Phi(x) = 2c^2 e^{-c^2 \|x\|^2} (2c^2 \|x\|^2 - d). \quad (21)$$

We now make use of the stencils w determined by (19) to define two types of RBF stencils for the discretisation of the Dirichlet problem (2)–(3), see [3] for more details about these methods. Referring to (11), we describe in each case how local collocation centres in Θ_ζ and the weights $\sigma_{\zeta, \theta}$ are chosen and give a linear system that defines the weight vector w assuming that the set of local discretisation centres Ξ_ζ is known. There are many algorithms in the literature for choosing Ξ_ζ , see [3, Section 5] for an overview. In the numerical results of this paper we make use of the method described in [3, Algorithm 1].

Single point RBF stencil

Similar to the finite difference stencils, a numerical differentiation formula for the value of the Laplacian of u at a single point ζ is used. Hence the discretisation of the Dirichlet problem is given by (6)–(7) with $w = [w_{\zeta, \xi}]_{\xi \in \Xi_\zeta}$ computed according to (19) with $D = \Delta$. Namely, let $\Xi_\zeta = X = \{x_0, \dots, x_n\}$, $\zeta = x_0$, be a local set of discretisation centres. Then w is computed by solving the linear system

$$\begin{bmatrix} \Phi_X & \mathbf{1} \\ \mathbf{1}^T & 0 \end{bmatrix} \begin{bmatrix} w \\ v \end{bmatrix} = \begin{bmatrix} [\Delta\varphi_i(x_0)]_{i=1}^n \\ 0 \end{bmatrix}. \quad (22)$$

Multipoint RBF stencil

Motivated by the stencils resulting from the finite element discretisation, we also considered in [3] a method based on numerical differentiation of the operator D defined by a linear combination of Laplacians

$$Du := \sum_{k=1}^{\ell} \sigma_k \Delta u(\cdot - x_0 + y_k),$$

where $\Theta_\zeta = \{y_1, \dots, y_\ell\}$ and positive weights $\sigma = [\sigma_{\zeta, \theta}]_{\theta \in \Theta_\zeta} = [\sigma_1, \dots, \sigma_\ell]$ are properly chosen. Then $Du(x_0) = \sum_{k=1}^{\ell} \sigma_k \Delta u(y_k)$, and (19) leads to the following linear system for the computation of w ,

$$\begin{bmatrix} \Phi_X & \mathbf{1} \\ \mathbf{1}^T & 0 \end{bmatrix} \begin{bmatrix} w \\ v \end{bmatrix} = \begin{bmatrix} [\sum_{k=1}^{\ell} \sigma_k \Delta \varphi_i(y_k)]_{i=1}^n \\ 0 \end{bmatrix}. \quad (23)$$

Following [3], we take into account the fact that $\Xi_\zeta = X = \{x_0, \dots, x_n\}$ produced by [3, Algorithm 1] contains $\zeta = x_0$ in its convex hull $\langle X \rangle$, and choose $\ell = n$ local collocation centres in our numerical results in 2D as follows. Split $\langle X \rangle$ into n triangles T_1, \dots, T_n by connecting x_0 to the other points $x_1, \dots, x_n \in \Xi_\zeta$ by straight line segments, and define $\Theta_\zeta = \{y_1, \dots, y_n\}$ to be the set of barycentres of T_1, \dots, T_n . Clearly, this resembles Θ_ζ arising from the finite element discretisation as described in Section 2.1. In this paper, motivated by (12) we use $\sigma_k = \text{area}(T_k)/3$, $k = 1, \dots, n$. Therefore, $Du(x_0)$ can be interpreted as discretisation of the integral $\int_{\langle X \rangle} \Delta u(x) dx$.

2.4 Stable computation for small c

Since the matrix (20) is extremely ill-conditioned for small values of the shape parameter c , alternative approaches for solving (19) are needed in this case. Several methods are available, see [6] and references therein. We follow the RBF-QR method of [6] adapted to RBF interpolation with a constant term as in [4]. Without describing further details, available in [6, 4], we mention that for the single point method the stencils w are computed by solving a properly normalised linear system [4, Eq. (25)], whereas in the multipoint case the right hand side $\Delta \tilde{\psi}_i(x_0)$ of this system is replaced by $\sum_{k=1}^{\ell} \sigma_k \Delta \tilde{\psi}_i(y_k)$, where $\tilde{\psi}_i$ are appropriate linear combinations of polar-Chebyshev functions, see [4]. Note that this method also allows the computation in the limiting case $c = 0$.

3 Numerical study of optimal shape parameter

In [4, Section 4] we studied the performance of the Gaussian RBF-FD method with single point stencil depending on the choice of the shape parameter c . Now, using the same test problems, we investigate the multipoint stencil, and obtain similar conclusions. We will see, however, that the errors obtained with multipoint stencil are often significantly better.

As in [4], we consider the Dirichlet problem (2)-(3) on four domains listed below, with the right hand sides given by the functions f_1 - f_8 and boundary conditions defined

by the restriction of the corresponding exact solutions u_1-u_8 , see Table 1. For each domain Ω we consider five sets of discretisation centres $\Xi = \Xi^{(1)}, \dots, \Xi^{(5)}$ generated as follows. First, an initial triangulation $\mathcal{T}^{(1)}$ is computed using MATLAB PDE Toolbox [9] with default mesh generation parameters. This triangulation is uniformly refined four times, which produced the triangulations $\mathcal{T}^{(2)}, \dots, \mathcal{T}^{(5)}$. The sets of discretisation centres $\Xi^{(1)}, \dots, \Xi^{(5)}$ consist of all vertices of corresponding triangulations. The number of interior centres for each $\Xi^{(i)}$ is shown in Table 2.

Domains: (a) the square $(-1, 1)^2$, (b) the unit disk $r < 1$, (c) the unit disk with a square hole $(-0.4, 0.4)^2$, see Figure 1 (left), and (d) a polygonal domain shown in Figure 1 (right).

exact solution	right hand side
$u_1(x, y) = \sin(2xy)$	$f_1(x, y) = -4 \sin(2xy)(x^2 + y^2)$
$u_2(x, y) = e^{-x^2-y^2}$	$f_2(x, y) = 4(x^2 + y^2 - 1)e^{-x^2-y^2}$
$u_3(x, y) = \sin(\pi x) \sin(\pi y)$	$f_3(x, y) = -2\pi^2 \sin(\pi x) \sin(\pi y)$
$u_4(r, \phi) = r^2(r - 1) \sin(2\phi)$	$f_4(r, \phi) = 5r \sin(2\phi)$
$u_5(x, y) = e^{-(x-0.1)^2-0.5y^2}$	$f_5(x, y) = e^{-(x-0.1)^2-0.5y^2}(y^2 + (-2x + 0.2)^2 - 3)$
$u_6(x, y) = \sin(2\pi(x - y))$	$f_6(x, y) = -8\pi^2 \sin(2\pi(x - y))$
$u_7(x, y) = \sin(x^3y) + e^x - x/(1 + y^2)$	$f_7(x, y) = -9 \sin(x^3y)x^4y^2 + 6 \cos(x^3y)xy + e^x - \sin(x^3y)x^6 - \frac{8xy^2}{(1+y^2)^3} + \frac{2x}{(1+y^2)^2}$
$u_8(x, y) = e^x \cos y$	$f_8(x, y) = 0$

Table 1: Test functions u_1, \dots, u_9 (exact solutions of the test problems) and their Laplacians (right hand sides for the test problems) $f_i = \Delta u_i$, $i = 1, \dots, 8$. The functions u_4 and f_4 are given in polar coordinates.

	square		disk		disk with hole		polygon	
	$\#\Xi_{\text{int}}$	c_{dmin}	$\#\Xi_{\text{int}}$	c_{dmin}	$\#\Xi_{\text{int}}$	c_{dmin}	$\#\Xi_{\text{int}}$	c_{dmin}
$\Xi^{(1)}$	33	0.043	28	0.042	15	0.052	15	0.033
$\Xi^{(2)}$	149	0.086	125	0.085	90	0.095	83	0.080
$\Xi^{(3)}$	633	0.172	529	0.169	420	0.191	381	0.161
$\Xi^{(4)}$	2609	0.347	2177	0.347	1800	0.386	1625	0.325
$\Xi^{(5)}$	10593	0.706	8833	0.705	7440	0.758	6705	0.660

Table 2: Number of interior centres and ‘safe’ shape parameter c_{dmin} for each discretisation.

The quality of a discrete solution \hat{u} of the Dirichlet problem, defined on a set of discretisation centres $\Xi = \Xi^{(i)}$, is measured by its root mean square (rms) error against

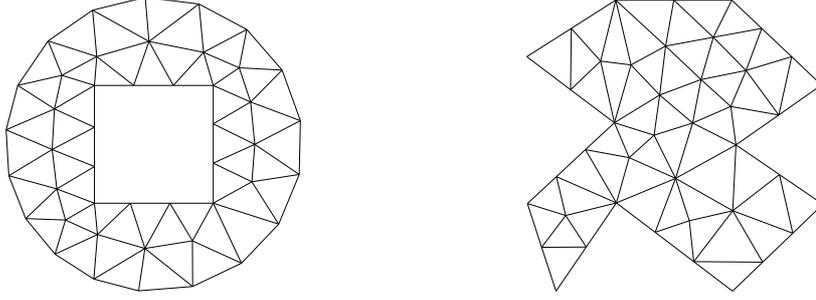


Figure 1: Initial triangulation $\mathcal{T}^{(1)}$ for the disk with a square hole and for the polygonal domain.

the values of the exact solution on Ξ_{int} ,

$$\text{rmse} := \left(\frac{1}{\#\Xi_{\text{int}}} \sum_{\xi \in \Xi_{\text{int}}} (\hat{u}(\xi) - u(\xi))^2 \right)^{1/2}. \quad (24)$$

Apart from the RBF-FD solutions, this formula applies to the standard linear finite element method with midpoint quadrature rule on the corresponding triangulation $\mathcal{T}^{(i)}$. We will use `rmse` of the finite element method as reference. For the RBF-FD single point stencils, we consider in addition the normalised rms error of the numerical differentiation formula (10), given by

$$\text{rmsed} := \left(\frac{1}{\#\Xi_{\text{int}}} \sum_{\zeta \in \Xi_{\text{int}}} r_{\zeta}^2 \right)^{1/2}, \quad r_{\zeta} = \left(\sum_{\theta \in \Theta_{\zeta}} \sigma_{\zeta, \theta} \Delta u(\theta) - \sum_{\xi \in \Xi_{\zeta}} w_{\zeta, \xi} u(\xi) \right) / \sum_{\theta \in \Theta_{\zeta}} \sigma_{\zeta, \theta}, \quad (25)$$

where in the case of the single point stencil the expression for r_{ζ} obviously simplifies to $r_{\zeta} = \Delta u(\zeta) - \sum_{\xi \in \Xi_{\zeta}} w_{\zeta, \xi} u(\xi)$.

As in [4], we select the stencil supports Ξ_{ζ} by the meshless algorithm described in [3, Algorithm 1]. On the quasi-uniform sets of centres used in this paper this algorithm delivers Ξ_{ζ} in most cases not different from the stencil supports used by the linear finite element method. Therefore, the density/bandwidth pattern of the system matrix $[w_{\zeta, \xi}]$ is very close to that of the finite element stiffness matrix.

For each $\Xi^{(i)}$ we computed a ‘safe’ value c_{dmin} of the shape parameter that guarantees that the condition number of the matrix of the system (19) does not exceed 10^{12} for any local set Ξ_{ζ} if $c \geq c_{\text{dmin}}$. We directly solve the system (22) or (23) if $c \geq c_{\text{dmin}}$, and use RBF-QR method if $c < c_{\text{dmin}}$. RBF-QR computations are done by adapting the MATLAB code provided in [6] and available for download from <http://user.it.uu.se/~bette/research.html>

Numerical results are presented in Figures 2–5 and Tables 3 and 4. In particular, Figures 2 and 3 and both tables are devoted to the multipoint stencil results for the test function u_3 on all domains and sets of centres.

The conclusions from these numerical results are similar to those presented in [4, Section 4] for the single point stencil. The key observations:

- For many test problems knowing the optimal shape parameter significantly improves the error of the RBF solution.

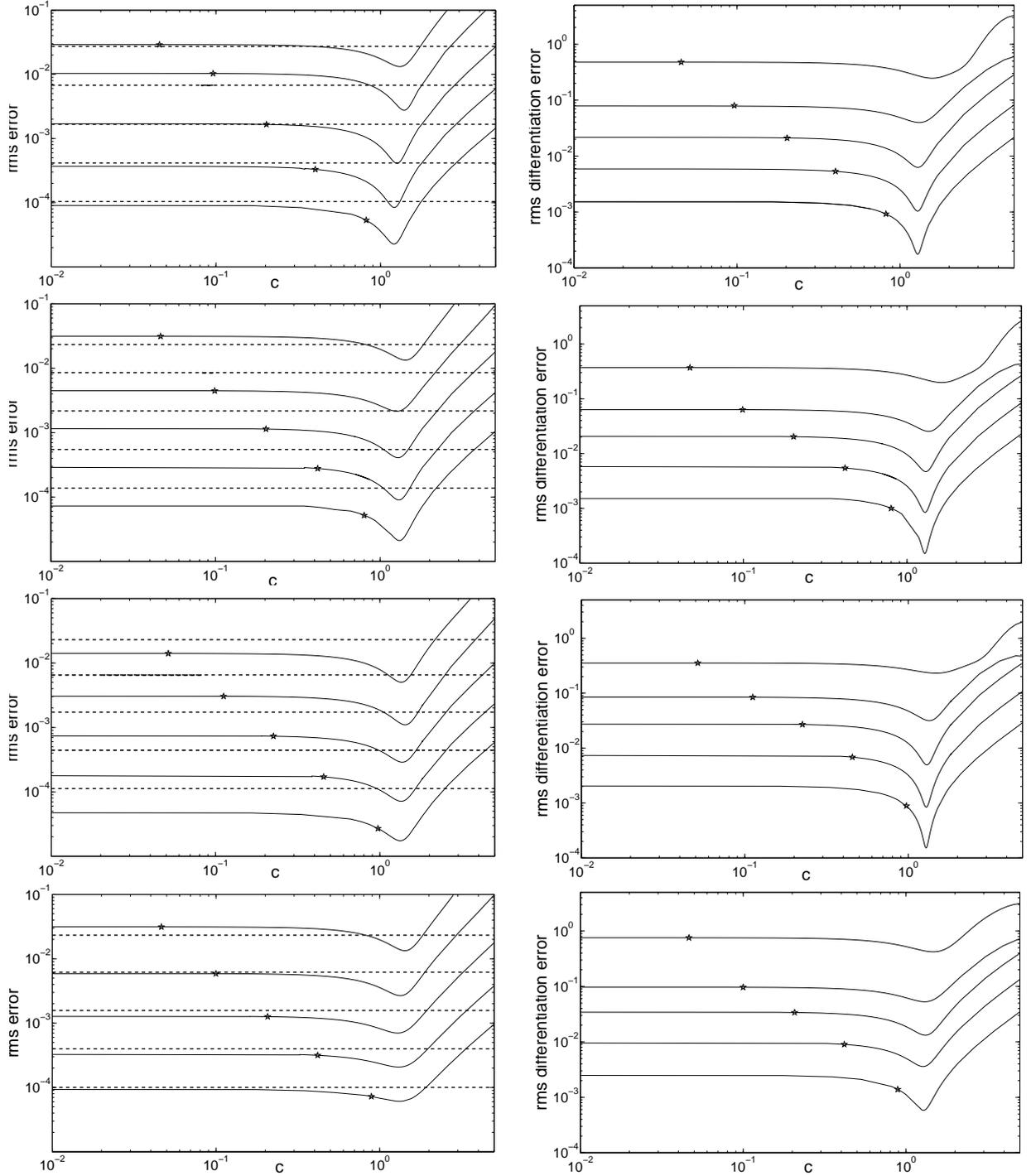


Figure 2: Left: The rms error of the multipoint stencil solutions for the test function u_3 on five sets of centres as a function of the shape parameter c (solid lines) compared to the rms error of the FEM solutions (dashed lines). Right: The numerical differentiation error of the multipoint stencil. From top to bottom: square, disk, disk with hole and polygonal domain. In each subfigure the five solid curves present the error of the multipoint stencil method on the five sets of centres, whereas the dashed constant curves show the error of the finite element method on the five triangulations for comparison. The stars indicate the value of $c = c_{\text{dmin}}$.

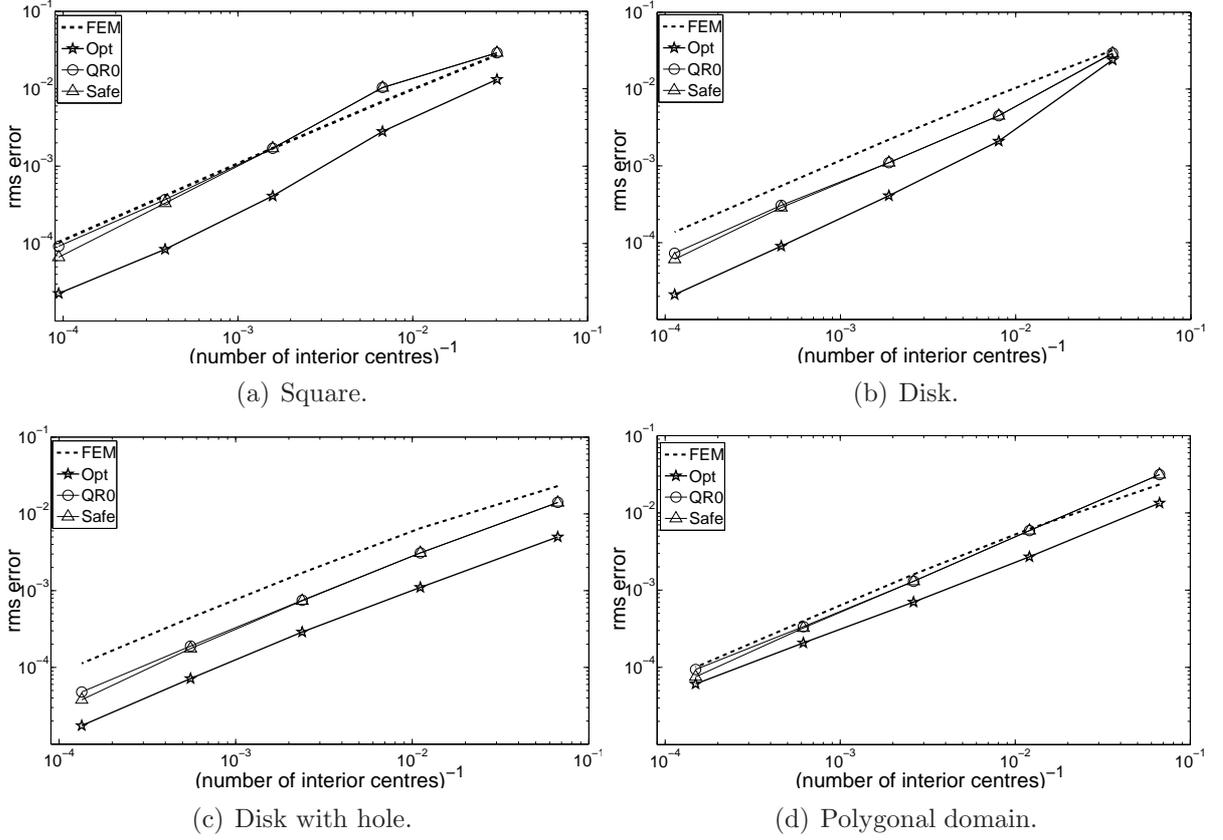


Figure 3: The rms error of the multipoint stencil solutions for the test function u_3 on five sets of centres as function of the number of degrees of freedom, for three values of the shape parameter: **Safe** refers to $c = c_{\text{dmin}}$, as shown in Table 2, **QR0** refers to ‘flat limit’ case $c = 0$, and **Opt** to the optimal values of c shown in Table 3.

	square		disk		disk with hole		polygon	
$\Xi^{(1)}$	1.31	[0.36,1.73]	1.24	[0.4,2.11]	1.35	[0.79,1.73]	1.43	[0.61,1.91]
$\Xi^{(2)}$	1.39	[1.00,1.69]	1.26	[0.085,1.74]	1.43	[0.83,1.82]	1.34	[0.08,1.81]
$\Xi^{(3)}$	1.26	[0.95,1.51]	1.29	[0.76,1.65]	1.37	[0.72,1.79]	1.29	[0,1.87]
$\Xi^{(4)}$	1.21	[0.93,1.44]	1.30	[0.88,1.61]	1.35	[0.73,1.76]	1.31	[0,2.00]
$\Xi^{(5)}$	1.21	[0.90,1.45]	1.31	[0.93,1.60]	1.33	[0.75,1.73]	1.32	[0,2.08]

Table 3: Optimal shape parameters for the rms error of the multipoint stencil solution \hat{u} for the test function u_3 . For each domain, the number in the first column is the optimal shape parameter, whereas the second column indicates the range of values of the shape parameter, for which the rms error is at most twice the optimal error.

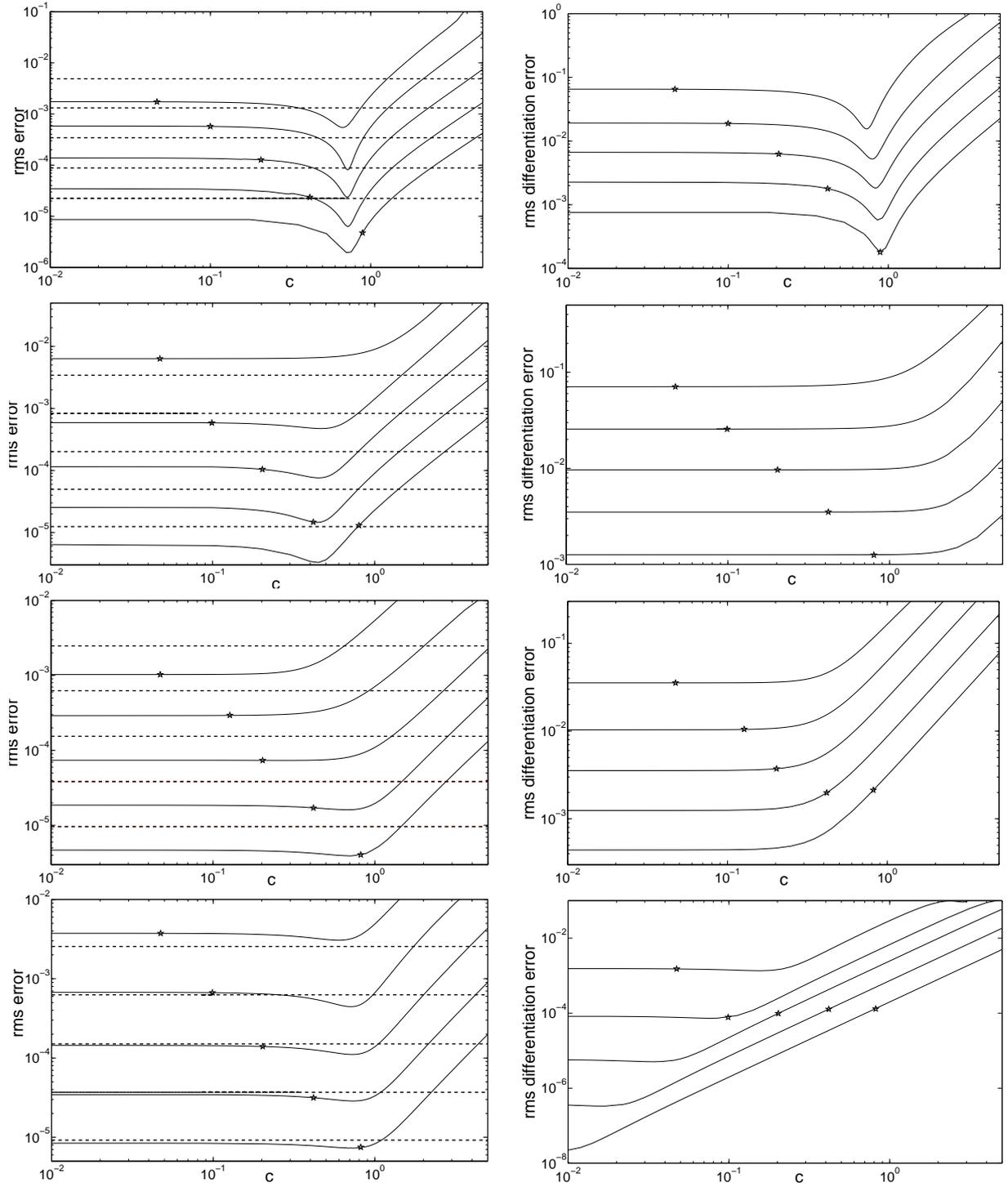


Figure 4: The rms error of the multipoint stencil solutions (left) and the rms differentiation error (right) as in Figure 2. From top to bottom: u_5 on the polygonal domain, u_7 , u_8 and u_1 on the disk.

	square	disk	disk with hole	polygon
$\Xi^{(1)}$	1.57	1.63	1.48	1.47
$\Xi^{(2)}$	1.31	1.36	1.34	1.30
$\Xi^{(3)}$	1.29	1.30	1.30	1.31
$\Xi^{(4)}$	1.28	1.28	1.29	1.27
$\Xi^{(5)}$	1.28	1.29	1.29	1.27

Table 4: Optimal shape parameters for the rms differentiation error of the multipoint stencil for the test function u_3 .

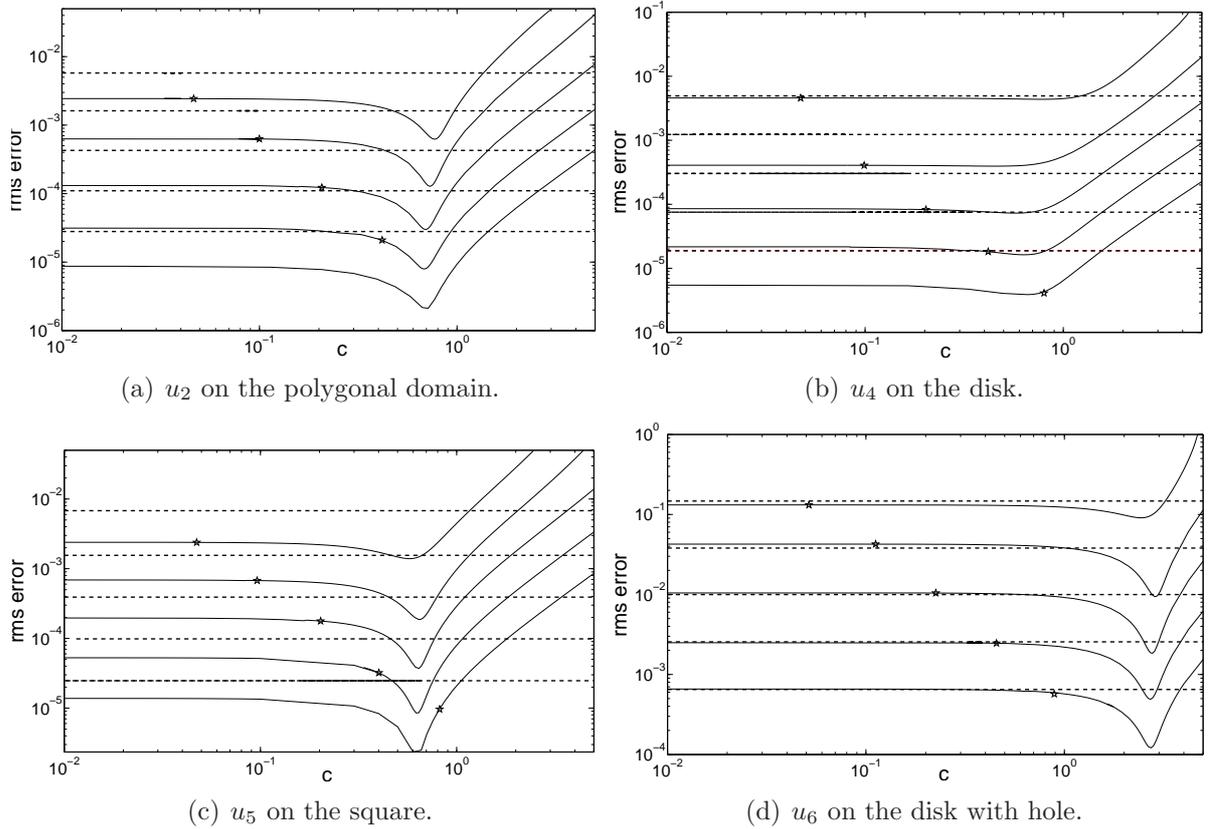


Figure 5: The rms error of the multipoint stencil solutions for the test functions u_2, u_4, u_5, u_6 . The layout of the figures is the same as in Figure 2 (left).

- Comparing to the results in [4, Section 4], we see that multipoint stencils perform better than single point stencils, and in almost every test the error of the multipoint stencil solution is better than the error of the finite element solution.
- The optimal value of the shape parameter c depends on the test function. However, it does not vary much when the number of centres or even the domain is changed. Comparing Tables 3 and 4 with Tables 4 and 5 of [4], we also conclude that the optimal shape parameters for single point and multipoint stencils do not differ much.
- The flat limit case $c = 0$ gives a good solution comparable to the finite element solution, albeit often significantly worse than the solution obtained with the optimal shape parameter. Comparing to [4, Section 4], we can say that the results with $c = 0$ are better for multipoint stencil than for the single point stencil approach.
- As in the single point case, the ‘safe’ shape parameter seems a dangerous choice for finer sets of centres.
- The value of c optimal for the error of the PDE correlates well with the optimal value of c for the numerical differentiation, which is particularly emphasised by comparing Tables 3 and 4.

4 Estimation of optimal shape parameter

In this section we investigate the performance of Algorithm 1 of [4] for the multipoint stencil. Recall that this algorithm is based on the observation that the optimal shape parameter does not change much when the set of centres is refined, and therefore a good estimation of the optimal shape parameter can be obtained by minimising the error with respect to a solution on a refined set of centres.

More precisely, given two values c, c_{ref} of the shape parameter corresponding to two sets of centres Ξ, Ξ^{ref} , such that $\Xi \subset \Xi^{\text{ref}}$, we assume that a cost function $\text{cost}(c, c_{\text{ref}})$ is defined that measures how well the RBF method with parameter c on centres Ξ performs comparing to the RBF solution with parameter c_{ref} on centres Ξ^{ref} .

Algorithm 1 ([4]). *Input:* two sets of centres Ξ, Ξ^{ref} such that $\Xi \subset \Xi^{\text{ref}}$ and initial estimate of the optimal shape parameter c_{ref} . *Output:* estimated optimal shape parameter c_{opt} . *Parameters:* tolerances $\delta > \varepsilon > 0$, maximum number of iterations m , upper bound C for the shape parameter. In the numerical tests below the following parameter values have been used: $\varepsilon = 0.01$, $\delta = 0.1$, $m = 4$, and $C = 5$.

- I. Compute Gaussian RBF solution \hat{u}_{ref} on Ξ^{ref} with shape parameter c_{ref} and find $c \in [c_{\text{min}}, c_{\text{max}}]$ such that $\text{cost}(c, c_{\text{ref}})$ is minimised, where $[c_{\text{min}}, c_{\text{max}}] = [0, C]$ if $c_{\text{ref}} = 0$ and $[c_{\text{min}}, c_{\text{max}}] = [c_{\text{ref}} - \delta, c_{\text{ref}} + \delta]$ otherwise.
 - II. For $i = 1, \dots, m$:
 - If $|c - c_{\text{ref}}| < \varepsilon$: STOP and return $c_{\text{opt}} = c$.
 - ElseIf $c = c_{\text{min}}$ or $c = c_{\text{max}}$: STOP and return $c_{\text{opt}} = \text{NaN}$.
 - Else: Set $c_{\text{ref}} = c$ and repeat Step I.
- Return:* $c_{\text{opt}} = \text{NaN}$

Note that Algorithm 1 fails if it returns NaN. If this happens, it should be rerun with different input parameters. We discuss below how we do this in our numerical experiments.

Depending on the choice of the cost function, we distinguish two versions of Algorithm 1, the first referred to as Algorithm 1a, and the second as Algorithm 1b.

For **Algorithm 1a**, let \hat{u} and \hat{u}_{ref} be the the RBF solutions of the Dirichlet problem (2)–(3) with the shape parameter c on Ξ and c_{ref} on Ξ^{ref} , respectively. Since $\Xi \subset \Xi^{\text{ref}}$, \hat{u}_{ref} is well defined for $\xi \in \Xi$. The cost function in this case is given by the formula

$$\text{cost}(c, c_{\text{ref}}) := \left(\frac{1}{\#\Xi_{\text{int}}} \sum_{\xi \in \Xi_{\text{int}}} (\hat{u}(\xi) - \hat{u}_{\text{ref}}(\xi))^2 \right)^{1/2}. \quad (26)$$

The cost function for **Algorithm 1b** measures the the accuracy of the RBF numerical differentiation formulae (10) on the set Ξ , obtained with the shape parameter value c ,

$$\sum_{\theta \in \Theta_{\zeta}} \sigma_{\zeta, \theta} \Delta u(\theta) \approx \sum_{\xi \in \Xi_{\zeta}} w_{\zeta, \xi} u(\xi), \quad \zeta \in \Xi_{\text{int}},$$

against the ones on the refined set of centres Ξ^{ref} , obtained with the shape parameter value c_{ref} ,

$$\sum_{\theta \in \Theta_{\zeta}^{\text{ref}}} \sigma_{\zeta, \theta}^{\text{ref}} \Delta u(\theta) \approx \sum_{\xi \in \Xi_{\zeta}^{\text{ref}}} w_{\zeta, \xi}^{\text{ref}} u(\xi), \quad \zeta \in \Xi_{\text{int}}^{\text{ref}}.$$

The error between two approximate average Laplacians of \hat{u}_{ref} in the neighborhood of $\zeta \in \Xi_{\text{int}}$ is therefore given by

$$e_{\zeta} = \left(\sum_{\theta \in \Theta_{\zeta}} \sigma_{\zeta, \theta} \right)^{-1} \sum_{\xi \in \Xi_{\zeta}} w_{\zeta, \xi} \hat{u}_{\text{ref}}(\xi) - \left(\sum_{\theta \in \Theta_{\zeta}^{\text{ref}}} \sigma_{\zeta, \theta}^{\text{ref}} \right)^{-1} \sum_{\xi \in \Xi_{\zeta}^{\text{ref}}} w_{\zeta, \xi}^{\text{ref}} \hat{u}_{\text{ref}}(\xi),$$

leading to the cost function in the form

$$\text{cost}(c, c_{\text{ref}}) := \left(\frac{1}{\#\Xi_{\text{int}}} \sum_{\zeta \in \Xi_{\text{int}}} e_{\zeta}^2 \right)^{1/2}. \quad (27)$$

Clearly, definitions (26) and (27) apply to both single point and multipoint stencil versions of RBF-FD method. The cost function (27) is cheaper to evaluate because it does not require solving the linear system (11) to obtain \hat{u} .

In our numerical tests with the sets $\Xi^{(1)}, \dots, \Xi^{(5)}$, we apply Algorithm 1 as follows, see [4] for more details. We first run it with $\Xi = \Xi^{(1)}$, $\Xi^{\text{ref}} = \Xi^{(3)}$, $c_{\text{ref}} = 0$, to obtain a nearly optimal shape parameter c_1 for $\Xi^{(1)}$. We then run it with $c_{\text{ref}} = c_1$, $\Xi = \Xi^{(2)}$, $\Xi^{\text{ref}} = \Xi^{(3)}$, to obtain a nearly optimal shape parameter c_2 for $\Xi^{(2)}$. On the rare occasion that Algorithm 1 fails and returns NaN, we set $c_{\text{ref}} = 0$ and rerun Algorithm 1 with Ξ^{ref} replaced by its refinement if c_{ref} was zero already. The value c_2 is also used on the sets $\Xi^{(3)}, \Xi^{(4)}, \Xi^{(5)}$. Therefore, multiple values of c are only tested on $\Xi^{(1)}, \Xi^{(2)}, \Xi^{(3)}$, and in a few exceptional cases on $\Xi^{(4)}$, which is executed at a fraction of the cost of the computation with a single c on $\Xi^{(5)}$.

Optimisation with respect to c in Step I is done using MATLAB function `fminbnd`. We set `MaxFunEvals` = 9 and `TolX` = 10^{-2} to reduce computation cost. Parameter m is

an upper bound on the number of computations of the RBF solution on Ξ^{ref} . Setting m to a small value may help to reduce the cost.

Figures 6–8 and Tables 5–7 below illustrate the performance of Algorithm 1 for the test problems in Figures 2, 4 and 5. In these experiment Algorithm 1a effectively finds near-optimal shape parameter, whereas the less expensive Algorithm 1b sometimes returns sub-optimal, albeit acceptable results. The tables also confirm that the number of iterations needed in Step II is small, typically just 2 or 3. Moreover, Figures 6–8 provide comparison with the results of numerical tests with Algorithm 1 for single point stencil method, see also Figures 8–10 in [4]. We see that while for u_3 the results for single point and multipoint stencils are comparable, in most other tests multipoint stencil significantly outperforms single point stencil, with the exception of u_8 , where single point stencils seem to be better. (This might be related to the fact that u_8 is a harmonic function.)

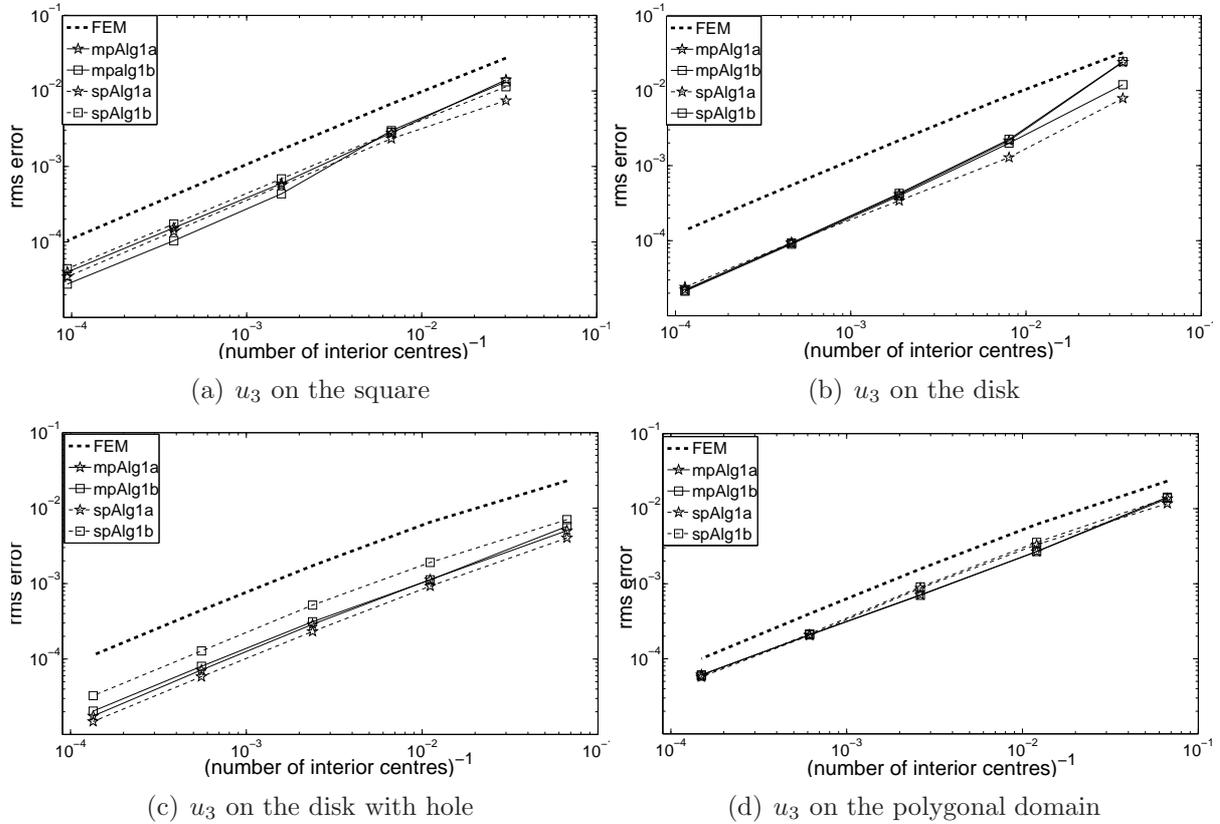
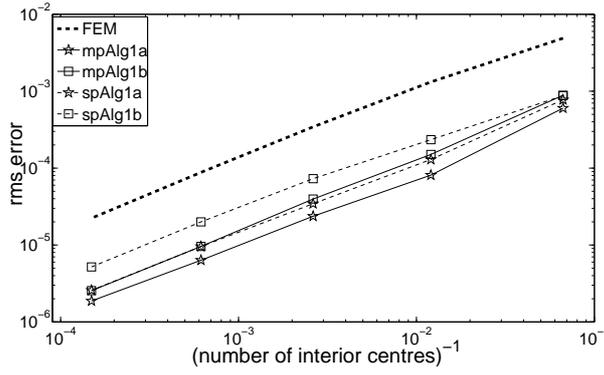
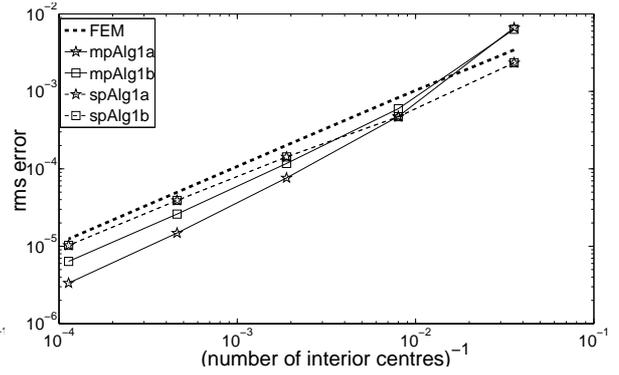


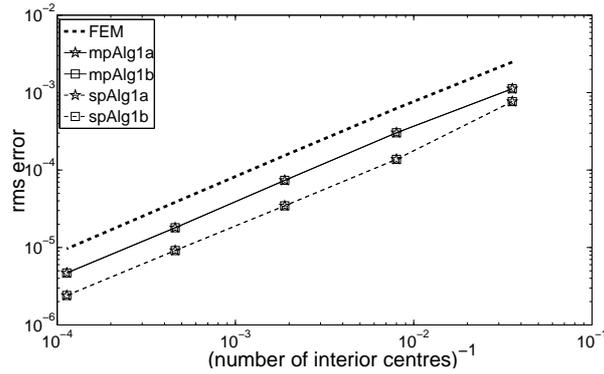
Figure 6: The rms error of the single point (sp, dashed curves) and multipoint (mp, solid curves) stencil solutions for the test function u_3 on five sets of centres as function of the number of degrees of freedom, with the shape parameter values produced by Algorithm 1: Alg1a and Alg1b refer to Algorithm 1a and 1b, respectively. The error of the finite element method is also shown (FEM) for comparison.



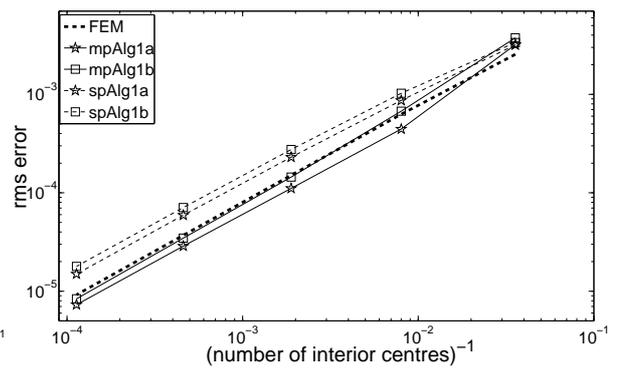
(a) u_5 on the polygonal domain



(b) u_7 on the disk

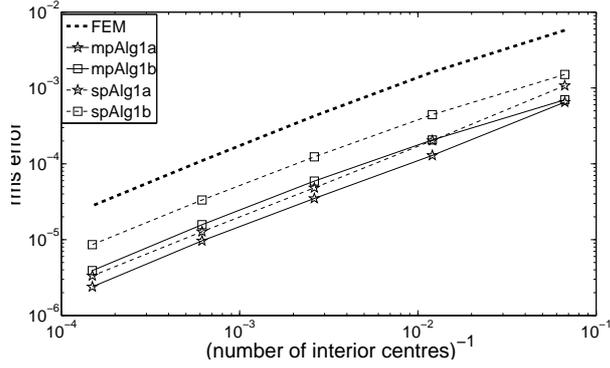


(c) u_8 on the disk

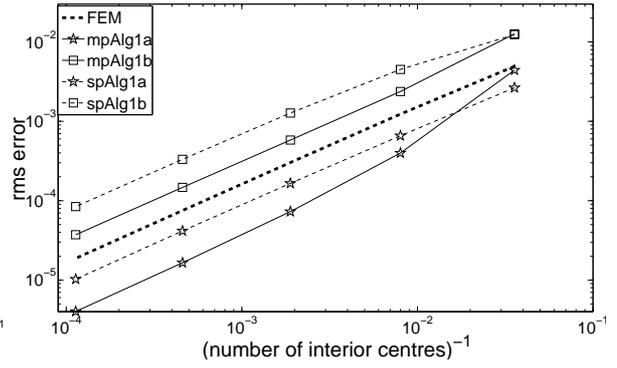


(d) u_1 on the disk

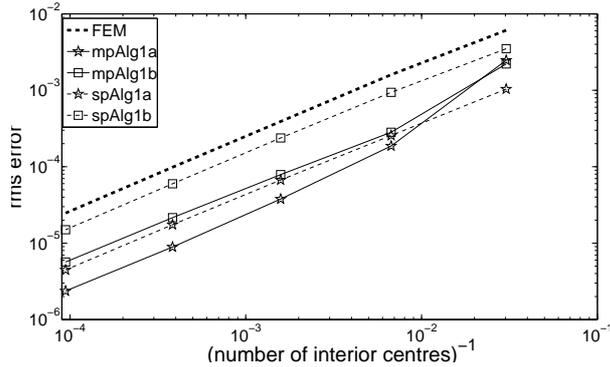
Figure 7: The rms error of the single point and multipoint stencil solutions for the test functions and domains as in Figure 4 on five sets of centres, with the shape parameter values produced by Algorithm 1. The symbols in the legend are the same as in Figure 6.



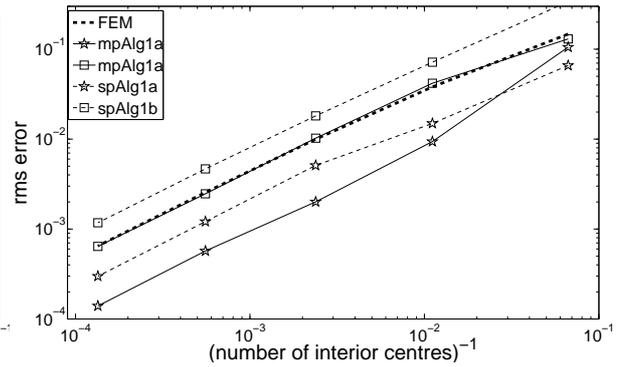
(a) u_2 on the polygonal domain.



(b) u_4 on the disk.



(c) u_5 on the square.



(d) u_6 on the disk with hole.

Figure 8: The rms error of the single point and multipoint stencil solutions for the test functions and domains as in Figure 5 on five sets of centres, with the shape parameter values produced by Algorithm 1. The symbols in the legend are the same as in Figure 6.

	square		disk		disk with hole		polygon	
	c_{opt}	nIter	c_{opt}	nIter	c_{opt}	nIter	c_{opt}	nIter
Algorithm 1a	1.41	3	1.25	2	1.43	2	1.36	2
Algorithm 1b	1.31	3	1.35	3	1.35	2	1.30	2

Table 5: The near optimal shape parameter c_{opt} for the multipoint stencil method and the number of iterations **nIter** in Step II of Algorithm 1 when $\Xi = \Xi^{(2)}$ and $\Xi^{\text{ref}} = \Xi^{(3)}$ for the test function u_3 .

	u_5 polygonal		u_7 disk		u_8 disk		u_1 disk	
	c_{opt}	nIter	c_{opt}	nIter	c_{opt}	nIter	c_{opt}	nIter
Algorithm 1a	0.72	3	0.47	4	0.28	4	0.71	4
Algorithm 1b	0.79	2	0	1	0	1	0	1

Table 6: The near optimal shape parameter c_{opt} and the number of iterations **nIter** for the multipoint stencil method as in Table 5 for the test functions and domains as in Figures 4 and 7.

References

- [1] J. P. Boyd and L. Wang. Truncated Gaussian RBF differences are always inferior to finite differences of the same stencil width. *Commun. Comput. Phys.*, 5:42–60, 2009.
- [2] M. D. Buhmann. *Radial Basis Functions*. Cambridge University Press, New York, NY, USA, 2003.
- [3] O. Davydov and D. T. Oanh. Adaptive meshless centres and RBF stencils for Poisson equation. *J. Comput. Phys.*, 230:287–304, 2011.
- [4] O. Davydov and D. T. Oanh. On optimal shape parameter for Gaussian RBF-FD approximation of Poisson equation. Technical Report 2011-01, University of Strathclyde, Department of Mathematics and Statistics, Jan. 2011. A revised version has appeared in *Comput. Math. Appl.*, 62:2143–2161, 2011.
- [5] G. F. Fasshauer. *Meshfree Approximation Methods with MATLAB*. World Scientific Publishing Co., Inc., River Edge, NJ, USA, 2007.
- [6] B. Fornberg, E. Larsson, and N. Flyer. Stable computations with Gaussian radial basis functions. *SIAM J. Sci. Comput.*, 33(2):869–892, 2011.
- [7] B. Fornberg and C. Piret. A stable algorithm for flat radial basis functions on a sphere. *SIAM J. Sci. Comput.*, 30:60–80, 2007.
- [8] C. K. Lee, X. Liu, and S. C. Fan. Local multiquadric approximation for solving boundary value problems. *Comput. Mech.*, 30(5-6):396–409, 2003.

	u_2 polygonal		u_4 disk		u_5 square		u_6 disk with hole	
	c_{opt}	nIter	c_{opt}	nIter	c_{opt}	nIter	c_{opt}	nIter
Algorithm 1a	0.74	2	0.57	4	0.65	2	2.86	2
Algorithm 1b	0.82	2	2.1	2	0.78	2	0.44	4

Table 7: The near optimal shape parameter c_{opt} and the number of iterations **nIter** for the multipoint stencil method as in Table 5 for the test functions and domains as in Figures 5 and 8.

- [9] *Partial Differential Equation ToolboxTM User's Guide*. The MathWorks, Inc, 2009.
- [10] C. Shu, H. Ding, and K. S. Yeo. Local radial basis function-based differential quadrature method and its application to solve two-dimensional incompressible Navier-Stokes equations. *Comput. Methods Appl. Mech. Eng.*, 192(7-8):941–954, 2003.
- [11] A. I. Tolstykh and D. A. Shirobokov. On using radial basis functions in a ‘finite difference mode’ with applications to elasticity problems. *Computational Mechanics*, 33(1):68–79, 2003.
- [12] H. Wendland. *Scattered Data Approximation*. Cambridge University Press, 2005.
- [13] G. B. Wright and B. Fornberg. Scattered node compact finite difference-type formulas generated from radial basis functions. *J. Comput. Phys.*, 212(1):99–123, 2006.