An integrated-RBF technique based on Galerkin formulation for elliptic differential equations

N. Mai-Duy* and T. Tran-Cong

Faculty of Engineering and Surveying,

The University of Southern Queensland, Toowoomba, QLD 4350, Australia

Submitted to *EABE*, 22-Feb-2008; revised, 1-May-2008

*Corresponding author: Telephone +61 7 4631 1324, Fax +61 7 4631 2526, E-mail maiduy@usq.edu.au

Abstract This paper presents a new radial-basis-function (RBF) technique for solving elliptic differential equations (DEs). The RBF solutions are sought to satisfy (a) the boundary conditions in a local sense using the point-collocation formulation, and (b) the governing equation in a global sense using the Galerkin formulation. In contrast to Galerkin finite-element techniques, the present Neumann boundary conditions are imposed in an exact manner. Unlike conventional RBF techniques, the present RBF approximations are constructed "locally" on grid lines through integration and they are expressed in terms of nodal variable values. The proposed technique can provide an approximate solution that is a C^p function across the subdomain interfaces (p–the order of the DE). Several numerical examples are presented to demonstrate the attractiveness of the present implementation.

Key words: integrated RBFNs, Galerkin formulation, Neumann boundary conditions, multiple boundary conditions, domain decomposition

1 Introduction

The mathematical modelling of engineering problems usually leads to sets of ordinary/partial differential equations (ODEs/PDEs) and their boundary conditions. To seek solutions to differential problems, for most cases, it is necessary to employ discretisation methods to reduce the sets of DEs to systems of algebraic equations. Principal discretisation methods (e.g. finite-difference, finite-element and boundary-element techniques) can be viewed as variants of the method of weighted residuals that can be stated in three well-known formulations, namely the strong, weak and inverse statements [1]. By means of weighting functions in a statement, the residuals for the DE and boundary conditions are made small in some senses. Two popular ways used are (i) the point-collocation approach, where the residuals are zero at certain points and (ii) the Galerkin-type approach, in which the residuals are zero in an average sense over the space of interest. Each approach has some

advantages in certain areas of application. The former is cost-effective as no integrations are required, while the latter has a smoothing capability owing to its integral nature.

Radial-basis-function (RBF) collocation methods are considered as a powerful tool for the approximation of scattered data as well as for the solution of differential problems [2]. RBF collocation methods are capable of approximating arbitrarily-well continuous functions. A number of RBFs such as the multiquadric and Gaussian basis functions have spectral approximation power. However, the condition number of the RBF interpolation matrix also grows rapidly with respect to (a) the decrease in distance between the RBF centres and (b) the increase in the RBF width. The methods thus, in practice, suffer from a trade-off between accuracy and stability [3]. Moreover, there is a gap in accuracy between the RBF solutions to Neumann- and Dirichlet-type boundary-value problems. To improve the numerical stability of a RBF solution, there are a number of schemes proposed in the literature: for example, (a) preconditioners (e.g. [4]); (b) local RBF approximations (e.g. [5,6]); (c) compactly-supported RBFs (e.g. [7]); and (d) domain decompositions (e.g. [8,9]). Recently, an approximation scheme, which is based on point collocation, Cartesian grids and one-dimensional integrated RBF networks (1D-IRBFNs), has been proposed in [10,11]. A problem domain, which can be regular or irregular, is discretised by a Cartesian grid. Along grid lines, 1D-IRBFNs are constructed to satisfy the governing DE together with boundary conditions in an exact manner. The "local" 1D-IRBFN approximations at a grid node involve only nodal points that lie on the grid lines intersected at that point rather than the whole set of nodes. This scheme allows a larger number of nodes to be employed.

There are very few papers on the use of RBFs in the context of Galerkin approximation [2]. Galerkin RBF techniques have been considered in [2,12-14]. In those works, conventional RBF approximations were employed. A function is decomposed into RBFs; its derivatives are then obtained through differentiation. In this study, we present a new numerical scheme, which is based on the Galerkin formulation and 1D-IRBFNs, for solving

elliptic problems. From a Galerkin-approach point of view, it will be shown that the proposed technique has several advantages: (a) natural boundary conditions are forced to be satisfied exactly, and (b) multiple boundary conditions are incorporated more efficiently. From a RBF-approach point of view, it will be shown that (a) the proposed method is capable of handling much larger data sets, that (b) its accuracy is considerably better than that of the 1D-IRBFN collocation technique, and that (c) it is able to yield almost the same levels of accuracy for the solutions of Neumann- and Dirichlet-type problems. An additional attractiveness of the proposed technique is that it facilitates a higher-order continuity of the approximate solution across the subdomain interfaces.

The paper is organised as follows. Brief reviews of the Galerkin formulation and 1D-IRBFNs are given in Sections 2 and 3, respectively. The Galerkin 1D-IRBFN method is presented in Section 4, followed by several numerical examples in Section 5 to demonstrate the attractiveness of the proposed method. Section 6 concludes the paper.

2 Galerkin approach

The Galerkin-type approach is well documented in the literature. The reader is referred to, see, for example, [1,15,16], for a full comprehensive description. A brief review of this approach is given below.

Consider a boundary-value problem defined by a linear DE and its boundary conditions

$$L(\bar{u}) = 0, \quad \mathbf{x} \in \Omega, \tag{1}$$

$$B(\bar{u}) = 0, \quad \mathbf{x} \in \Gamma, \tag{2}$$

where $\bar{u}$ is the field/dependent variable (the overbar denotes the exact solution), L and B the prescribed known operators, Ω the domain of interest and Γ the boundaries of the

domain Ω.

An approximate solution, denoted by u, to the set of (1) and (2) can be sought in the form

$$\bar{u}(\mathbf{x}) \approx u(\mathbf{x}) = \sum_{i=1}^{N} \alpha_i \phi_i(\mathbf{x}), \tag{3}$$

where $\{\alpha_i\}_{i=1}^{N}$ is the set of unknown coefficients and $\{\phi_i(\mathbf{x})\}_{i=1}^{N}$ the set of linearly-independent functions. The terms ϕ_i are usually referred to as the trial/basis/approximating functions.

Assume that a function u is constructed to satisfy the DE (1) at every point on the domain Ω, it leads to

$$\int_{\Omega} wL(u)d\Omega = 0, \tag{4}$$

for any function w that is bounded on Ω .

Similarly, assume that the function u also satisfies the boundary conditions (2), it follows that

$$\int_{\Gamma} \tilde{w}B(u)d\Gamma = 0, \tag{5}$$

for any bounded function $\tilde{w}$. The functions w and $\tilde{w}$ are often referred to as the weighting/test functions.

Under assumptions (4) and (5), the approximate solution u is also the exact solution $\bar{u}$ itself, and the system defined by (1) and (2) is equivalent to the following integral statement

$$\int_{\Omega} wL(u)d\Omega + \int_{\Gamma} \tilde{w}B(u)d\Gamma = 0, \tag{6}$$

that is satisfied for all bounded functions w and $\tilde{w}$.

However, in practice, one is able to employ finite sets of w and $\tilde{w}$ which result in an approximate solution.

If the weighting functions w and $\tilde{w}$ have sufficient degrees of continuity, integrations by

parts can be applied to derivative terms in (6), leading to other integral statements, namely the weak and inverse forms, that can be expressed as

$$\int_{\Omega} C(w)D(u)d\Omega + \int_{\Gamma} E(\tilde{w})F(u)d\Gamma = 0, \tag{7}$$

where the order of continuity required for the u solution is reduced. One can thus use either (6) or (7) to determine the approximate solution u. These integral forms of weighted residuals will allow the approximation to be conducted subdomain by subdomain. Different types of w and $\tilde{w}$ will constitute different numerical approaches (e.g. point collocation, subdomain collocation and Galerkin-type ones). For the Galerkin-type approach, the weighting functions are chosen from the same set of functions as the trial functions. This approach usually leads to symmetric matrices.

3 One-dimensional integrated RBFNs

Consider a univariate function $f(x)$. The basic idea of the integral RBF scheme [17] is to decompose a pth-order derivative of the function f into RBFs

$$\frac{d^p f(x)}{dx^p} = \sum_{i=1}^{N} w_i g_i(x), \tag{8}$$

where $\{w_i\}_{i=1}^N$ is the set of network weights, and $\{g_i(x)\}_{i=1}^N$ the set of RBFs. For a convenient description of the integral scheme, we replace the notation $g_i(x)$ with the notation $I_i^{(p)}(x)$ that contains information about derivative order of f. By integrating

(8), lower-order derivatives and the function itself are then obtained

$$\frac{d^{p-1}f(x)}{dx^{p-1}} = \sum_{i=1}^{N} w_i I_i^{(p-1)}(x) + c_1, \tag{9}$$

$$\frac{d^{p-2}f(x)}{dx^{p-2}} = \sum_{i=1}^{N} w_i I_i^{(p-2)}(x) + c_1 x + c_2, \tag{10}$$

$$\cdots \quad \cdots \qquad \cdots \qquad \cdots \qquad \cdots$$

$$\frac{df(x)}{dx} = \sum_{i=1}^{N} w_i I_i^{(1)}(x) + c_1 \frac{x^{p-2}}{(p-2)!} + c_2 \frac{x^{p-3}}{(p-3)!} + \cdots + c_{p-2}x + c_{p-1}, \tag{11}$$

$$f(x) = \sum_{i=1}^{N} w_i I_i^{(0)}(x) + c_1 \frac{x^{p-1}}{(p-1)!} + c_2 \frac{x^{p-2}}{(p-2)!} + \cdots + c_{p-1}x + c_p, \tag{12}$$

where $I_i^{(p-1)}(x) = \int I_i^{(p)}(x)dx$, $I_i^{(p-2)}(x) = \int I_i^{(p-1)}(x)dx, \cdots, I_i^{(0)}(x) = \int I_i^{(1)}(x)dx$, and $\{c_1, c_2, \cdots, c_p\}$ are the constants of integration.

Unlike conventional differential schemes, the starting point of the integral scheme can vary in use, depending on the particular application under consideration. The scheme is said to be of order p, denoted by IRBFN-p, if the pth-order derivative is taken as the starting point.

Evaluation of (8)-(12) at a set of collocation points $\{x_j\}_{j=1}^{N}$ leads to

$$\widehat{\frac{d^p f}{dx^p}} = \widehat{\mathcal{I}}_{[p]}^{(p)} \widehat{\alpha}, \tag{13}$$

$$\widehat{\frac{d^{p-1} f}{dx^{p-1}}} = \widehat{\mathcal{I}}_{[p]}^{(p-1)} \widehat{\alpha}, \tag{14}$$

$$\cdots\cdots\cdots$$

$$\widehat{\frac{df}{dx}} = \widehat{\mathcal{I}}_{[p]}^{(1)} \widehat{\alpha}, \tag{15}$$

$$\widehat{f} = \widehat{\mathcal{I}}_{[p]}^{(0)} \widehat{\alpha}, \tag{16}$$

where the subscript [.] and superscript (.) are used to denote the order of the IRBFN

scheme and the order of the corresponding derivative function, respectively;

$$\widehat{\mathcal{I}}_{[p]}^{(p)} = \begin{bmatrix} I_1^{(p)}(x_1), & I_2^{(p)}(x_1), & \cdots, & I_N^{(p)}(x_1), & 0, & 0, & \cdots, & 0, & 0 \\ I_1^{(p)}(x_2), & I_2^{(p)}(x_2), & \cdots, & I_N^{(p)}(x_2), & 0, & 0, & \cdots, & 0, & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ I_1^{(p)}(x_N), & I_2^{(p)}(x_N), & \cdots, & I_N^{(p)}(x_N), & 0, & 0, & \cdots, & 0, & 0 \end{bmatrix},$$

$$\widehat{\mathcal{I}}_{[p]}^{(p-1)} = \begin{bmatrix} I_1^{(p-1)}(x_1), & I_2^{(p-1)}(x_1), & \cdots, & I_N^{(p-1)}(x_1), & 1, & 0, & \cdots, & 0, & 0 \\ I_1^{(p-1)}(x_2), & I_2^{(p-1)}(x_2), & \cdots, & I_N^{(p-1)}(x_2), & 1, & 0, & \cdots, & 0, & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ I_1^{(p-1)}(x_N), & I_2^{(p-1)}(x_N), & \cdots, & I_N^{(p-1)}(x_N), & 1, & 0, & \cdots, & 0, & 0 \end{bmatrix},$$

$\cdots\cdots$,

$$\widehat{\mathcal{I}}_{[p]}^{(0)} = \begin{bmatrix} I_1^{(0)}(x_1), & I_2^{(0)}(x_1), & \cdots, & I_N^{(0)}(x_1), & \frac{x_1^{p-1}}{(p-1)!}, & \frac{x_1^{p-2}}{(p-2)!}, & \cdots, & x_1, & 1 \\ I_1^{(0)}(x_2), & I_2^{(0)}(x_2), & \cdots, & I_N^{(0)}(x_2), & \frac{x_2^{p-1}}{(p-1)!}, & \frac{x_2^{p-2}}{(p-2)!}, & \cdots, & x_2, & 1 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ I_1^{(0)}(x_N), & I_2^{(0)}(x_N), & \cdots, & I_N^{(0)}(x_N), & \frac{x_N^{p-1}}{(p-1)!}, & \frac{x_N^{p-2}}{(p-2)!}, & \cdots, & x_N, & 1 \end{bmatrix};$$

$$\widehat{\alpha} = (w_1, w_2, \cdots, w_N, c_1, c_2, \cdots, c_p)^T;$$

and

$$\widehat{\frac{d^k f}{dx^k}} = \left(\frac{d^k f_1}{dx^k}, \frac{d^k f_2}{dx^k}, \cdots, \frac{d^k f_N}{dx^k} \right)^T, \quad k = \{1, 2, \cdots, p\},$$
$$\widehat{f} = (f_1, f_2, \cdots, f_N)^T,$$

in which $d^k f_j/dx^k = d^k f(x_j)/dx^k$ and $f_j = f(x_j)$ with $j = \{1, 2, \cdots, N\}$.

The use of integrated basis functions is expected to avoid the problem of reduction of convergence rate caused by differentiation [18]. Numerical studies, e.g. [19-21], have shown that the integral collocation approach is more accurate than the differential col-

location approach. Recently, theoretical studies [22] have confirmed superior accuracy of integrated RBFNs over differentiated RBFNs.

4 Galerkin IRBFN technique

For Galerkin finite-element techniques, a weak statement (7), where the continuity requirement for the field variable u is reduced, is a preferred option. Piecewise polynomials of low order such as linear and quadratic interpolations are generally used as approximating and weighting functions in numerous small subdomains called elements. In the case that the shape functions φ_i are algebraic polynomials, only the field variable changes continuously throughout the entire domain, and its high derivatives (e.g. second and higher-order derivatives for linear elements) are not defined. Essential boundary conditions are incorporated into the approximate solution prior to the process of discretising the DE, while natural boundary conditions are imposed by means of weighted residual (i.e. the second term in (7)). It should be emphasised that the natural boundary conditions in the weak formulation are approximated rather than identically satisfied. In engineering practice, such a partial satisfaction of the boundary conditions tends to give poor results for surface fluxes or tractions which make the overall results unreliable for many cases [1].

In the present Galerkin 1D-IRBFN technique, we use a Cartesian grid to generate the finite trial and test spaces. One dimensional IRBFNs are employed to represent the field variable and its derivatives on grid lines. The RBF solutions are constructed to satisfy the boundary conditions using the point-collocation approximation and the governing DE using the Galerkin approximation. A distinguishing feature here is that the networks are sought to satisfy a priori the derivative boundary conditions in an exact manner. There is thus no need to use the second term in (6) and (7). As the trial functions are infinitely-differentiable global functions, the present Galerkin 1D-IRBFN technique permits the employment of (6) to solve the differential problem of any order. Moreover,

any derivative of the field variable is defined and continuous throughout the entire domain.

From an engineering viewpoint, one would prefer to work in the physical space. The present approximate solution is sought in terms of nodal variable values rather than the usual network weights. The boundary conditions including derivative information are imposed through the conversion process of the network-weight space into the physical space. RBFNs involve two types of data sets, namely centre and collocation points. In the context of point-collocation approximation, RBFNs tend to result in the most accurate approximations when the two sets of points are identical. Here, the collocation points are chosen to be the centres themselves. Unlike conventional differential formulations, the integral RBF formulation has the ability to generate additional coefficients (the constants of integration). This feature thus facilitates the addition of extra equations to the conversion system to represent extra information such as the natural boundary conditions and even the governing equation at the boundary points. The presence of integration constants thus guarantees that all RBFs are used for function approximation. In contrast, for conventional differentiated RBFNs, the enforcement of derivative function values is done at the price of the non-consideration of the function at some RBF centres, which significantly deteriorates the accuracy of the RBF scheme.

Consider a grid line. The conversion system for an 1D-IRBFN scheme of order p can be described as

$$\begin{pmatrix} \widehat{u} \\ \widehat{e} \end{pmatrix} = \begin{bmatrix} \widehat{\mathcal{I}}^{(0)}_{[p]} \\ \widehat{\mathcal{K}} \end{bmatrix} \widehat{\alpha} = \mathcal{C}\widehat{\alpha}, \tag{17}$$

where $\widehat{e}$, whose length can be up to p, is a vector representing extra information (e.g. normal derivative boundary conditions); $\widehat{e} = \widehat{\mathcal{K}}\widehat{\alpha}$; $\widehat{u}$, $\widehat{\mathcal{I}}^{(0)}_{[p]}$ and $\widehat{\alpha}$ defined as before; and $\mathcal{C}$ the conversion matrix. It can be seen from (17) that the approximate solution u is

collocated at the whole set of centres. Solving (17) for $\widehat{\alpha}$ yields

$$\widehat{\alpha} = \mathcal{C}^{-1} \begin{pmatrix} \widehat{u} \\ \widehat{e} \end{pmatrix}, \tag{18}$$

where $\mathcal{C}^{-1}$ is the inverse or pseudo-inverse of $\mathcal{C}$, depending on its dimension. Substitution of (18) into (8)-(12) leads to

$$u(x) = \left(I_1^{(0)}(x), I_2^{(0)}(x), \cdots \right) \mathcal{C}^{-1} \begin{pmatrix} \widehat{u} \\ \widehat{e} \end{pmatrix}, \tag{19}$$

$$\frac{\partial u(x)}{\partial x} = \left(I_1^{(1)}(x), I_2^{(1)}(x), \cdots \right) \mathcal{C}^{-1} \begin{pmatrix} \widehat{u} \\ \widehat{e} \end{pmatrix}, \tag{20}$$

$$\cdots\cdots\cdots\cdots\cdots\cdots$$

$$\frac{\partial^p u(x)}{\partial x^p} = \left(I_1^{(p)}(x), I_2^{(p)}(x), \cdots \right) \mathcal{C}^{-1} \begin{pmatrix} \widehat{u} \\ \widehat{e} \end{pmatrix}. \tag{21}$$

They can be rewritten in the form

$$u(x) = \sum_{i=1}^{N} \varphi_i(x) u_i + \varphi_{N+1}(x) e_1 + \varphi_{N+2}(x) e_2 + \cdots, \tag{22}$$

$$\frac{\partial u(x)}{\partial x} = \sum_{i=1}^{N} \frac{d\varphi_i(x)}{dx} u_i + \frac{d\varphi_{N+1}(x)}{dx} e_1 + \frac{d\varphi_{N+2}(x)}{dx} e_2 + \cdots, \tag{23}$$

$$\cdots\cdots\cdots\cdots\cdots\cdots$$

$$\frac{\partial^p u(x)}{\partial x^p} = \sum_{i=1}^{N} \frac{d^p\varphi_i(x)}{dx^p} u_i + \frac{d^p\varphi_{N+1}(x)}{dx^p} e_1 + \frac{d^p\varphi_{N+2}(x)}{dx^p} e_2 + \cdots. \tag{24}$$

The Galerkin weighting process applied to (1) produces the results

$$\int_\Omega \varphi_i L(u) = 0, \tag{25}$$

where the values of i depend on the problem under consideration as will be discussed

later. The system of equations, (25), can then be used to solve for the nodal value of the variable u.

5 Numerical results

Numerical results are presented for second- and fourth-order DEs in one and two dimensions. A DE of order p is discretised using the 1D-IRBFN-p scheme. For all numerical examples presented in this study, we employ the multiquadric basis function

$$I_i^{(p)}(x) = \sqrt{(x - c_i)^2 + a_i^2}, \tag{26}$$

where c_i and a_i are the centre and the width/shape-parameter of the ith RBF. Moreover, the latter is simply chosen to be the grid size. A 1D-IRBFN-based collocation method is also employed to provide the basis for the assessment of accuracy of the present Galerkin 1D-IRBFN method. An important difference between the two 1D-IRBFN methods lies in the way that the residual for the DE is constructed: the latter reduces the residual in a global sense, while the former in a local sense. Hereafter, the term 1D-IRBFNs will frequently be dropped out for brevity.

The accuracy of an approximate solution is measured by means of the discrete relative L_2 norm defined as

$$N_e = \frac{\sqrt{\sum_{i=1}^{M} (\bar{u}_i - u_i)^2}}{\sqrt{\sum_{i=1}^{M} (\bar{u}_i)^2}}, \tag{27}$$

where $\bar{u}$ and u are the exact and computed solutions, respectively, and M is the number of unknown nodal values of u. Another important measure is the convergence rate of the computed solution with respect to the centre spacing h

$$N_e \approx \gamma h^{\alpha} = O(h^{\alpha}), \tag{28}$$

in which α and γ are exponential model's parameters. Given a set of observations, these parameters can be found by the general linear least squares technique.

5.1 1D problems

5.1.1 Second-order ODE

Find an approximate solution to the ODE

$$\frac{d^2\bar{u}}{dx^2} + \bar{u} + x = 0, \tag{29}$$

on the interval $0 \leq x \leq 1$, with appropriate boundary conditions. The exact solution of (29) is assumed to be

$$\bar{u} = \frac{\sin(x)}{\sin(1)} - x, \tag{30}$$

from which one can easily derive the boundary values at $x = 0$ and $x = 1$. This problem is similar to that in [1]. To generate the finite spaces for the trial and test functions, we employ 45 sets of uniformly-distributed points, varying from 3 to 91 with increment of 2. Two types of boundary conditions are considered.

Dirichlet boundary conditions: Both $\mathcal{K}$ and $\widehat{e}$ are set to null. An approximate solution simply takes the form $u(x) = \sum_{i=1}^{N} u_i \varphi_i(x)$. The weighting functions are chosen to be the trial functions that are associated with the unknown nodal values of u. For this case, they are $\{\varphi_2(x), \varphi_3(x), \cdots, \varphi_{N-1}(x)\}$. Figure 1 plots the variations of the approximating functions of IRBFN using $N = 6$; the interpolating functions $\{\varphi_i\}_{i=2}^{5}$ satisfy homogeneous boundary conditions. Equation (25) leads to a determinate symmetric system of equations A for $(N-2)$ unknowns (i.e. the values of u at the interior points). Results concerning the error (N_e) and the condition number of the system matrix (cond$A$) are given in Figure 2 and Table 1, respectively. The condition numbers are relatively low. The Galerkin and

collocation solutions converge to the exact solution apparently as $O(h^{3.46})$ and $O(h^{2.96})$, respectively. It can be seen that the Galerkin technique yields much more accurate results and converges faster than the collocation technique.

Dirichlet $(x = 0)$ **and Neumann** $(x = 1)$ **boundary conditions:** We employ one extra equation to represent the derivative value at $x = 1$. Expression (22) becomes $u(x) = \sum_{i=1}^{N} u_i \varphi_i(x) + (du_N/dx)\, \varphi_{N+1}(x)$ that also contains derivative information. The approximate solution thus satisfy a priori both the Dirichlet and Neumann boundary conditions in an exact manner. The unknown vector consists of the values of u at $\{x_j\}_{j=2}^{N}$. As a result, $\{\varphi_2(x), \varphi_3(x), \cdots, \varphi_N(x)\}$ are taken as the weighting functions. The condition numbers of the present system matrix vary from 2.39×10^1 to 7.94×10^4. The Galerkin and collocation techniques yield a convergence rate of $O(h^{3.54})$ and $O(h^{1.98})$, respectively (Figure 3). The proposed method achieves a very high level of accuracy. At $N = 91$, the value of $N_e(u)$ is 2.03×10^{-8}; the solution is accurate up to at least 8 significant digits. Again, the Galerkin approach outperforms the collocation approach regarding accuracy.

It has been generally observed that the RBF results for boundary-value problems involving Neumann boundary conditions are generally much less accurate than for those involving only Dirichlet boundary conditions. An attractive point here is that the present RBF method yield essentially the same degrees of accuracy for both types of problems (Figures 2 and 3).

It is also noted that the use of algebraic polynomials in N equally-spaced points will lead to the approximations that not only fail to converge in general as $N \to \infty$ but also get worse at a rate that may be as great as 2^N (Runge phenomenon) [23]. It can be seen that the present global approximations, which are based on uniform centre sets, do not suffer from this phenomenon.

5.1.2 Fourth-order ODE

This example is governed by the biharmonic equation

$$\frac{d^4\bar{u}}{dx^4} + \frac{d^2\bar{u}}{dx^2} + b = 0, \tag{31}$$

on the interval $-1/2 \leq x \leq +1/2$, and Dirichlet boundary conditions

$$\bar{u}(-\frac{1}{2}) = -\frac{1}{2}\sin(-\frac{k}{2}), \qquad \frac{d\bar{u}}{dx}(-\frac{1}{2}) = \sin(-\frac{k}{2}) - \frac{k}{2}\cos(-\frac{k}{2}), \tag{32}$$

$$\bar{u}(+\frac{1}{2}) = +\frac{1}{2}\sin(+\frac{k}{2}), \qquad \frac{d\bar{u}}{dx}(+\frac{1}{2}) = \sin(+\frac{k}{2}) + \frac{k}{2}\cos(+\frac{k}{2}), \tag{33}$$

where $b = (4k^3 - 2k)\cos(kx) - x(k^4 - k^2)\sin(kx)$. The exact solution to this problem can be verified to be

$$\bar{u} = x\sin(kx). \tag{34}$$

We employ $k = 7\pi/2$ that makes all boundary data nonzero. The present conversion process involves the enforcement of u at the whole set of centres and du/dx at the two boundary points, from which the approximate solution will take the form $u(x) = \sum_{i=1}^{N} u_i\varphi_i(x) + (du_1/dx)\,\varphi_{N+1}(x) + (du_N/dx)\,\varphi_{N+2}(x)$. The system matrix for solving $\{u_i\}_{i=2}^{N-1}$ is then generated using the weighting functions $\{\varphi_2(x), \varphi_3(x), \cdots, \varphi_{N-1}(x)\}$.

A number of uniform centre sets, namely $5, 7, 9, \cdots, 45$ points, are employed. Figure 4 shows that the accuracy of the Galerkin method is far superior to that of the collocation method. For example, a convergence rate is $O(h^{6.31})$ for the former and $O(h^{4.24})$ for the latter. The present technique produces system matrices with their condition numbers being 2.69×10^1 to 5.85×10^5.

A comparison of the results shown in Figures 2 and 4 indicates that the convergence for fourth-order ODE is much faster than that for second-order ODE. On the other hand, it can be seen that the matrix condition numbers for fourth-order ODE are higher than

those for second-order ODE. Nevertheless, the present Galerkin matrices associated with fourth-order problems still exhibit relatively-low condition numbers.

5.2 2D problems

5.2.1 Dirichlet boundary conditions

This example is concerned with the following Poisson equation

$$\frac{\partial^2 \bar{u}}{\partial x^2} + \frac{\partial^2 \bar{u}}{\partial y^2} + \frac{2\pi^2}{1+2\pi^2}\cos(\pi x)\cos(\pi y) = 0, \tag{35}$$

defined on the domain $-1 \leq x, y \leq 1$ and subject to Dirichlet boundary conditions. This problem has the following exact solution

$$\bar{u}(x,y) = \frac{1}{1+2\pi^2}\cos(\pi x)\cos(\pi y). \tag{36}$$

A uniform grid of $N_x \times N_y$ points ($N = N_x N_y$) is employed to generate the trial and test spaces. The field variable $\bar{u}$ is approximated in the form

$$u(x,y) = \sum_{i=1}^{N_x}\sum_{j=1}^{N_y} \varphi_i^{(x)}(x)\varphi_j^{(y)}(y)u_{i,j}, \tag{37}$$

where $\varphi_i^{(x)}(x)$ and $\varphi_j^{(y)}(y)$ are the known basis functions derived from integrating one-dimensional RBFs associated with the x and y directions, respectively; and $u_{i,j}$ is the value of u at the intersection of the ith horizontal grid line and the jth vertical grid line. In (37), the basis functions are products of 1D-IRBFs in each direction.

A determinate system of algebraic equations for the interior nodal values of u is generated

by making the following residual equal to zero

$$R_{r,s} = \int_\Omega \varphi_r^{(x)}(x)\varphi_s^{(y)}(y) \left(\sum_{i=1}^{N_x}\sum_{j=1}^{N_y} \frac{\partial^2 \varphi_i^{(x)}(x)}{\partial x^2}\varphi_j^{(y)}(y) + \sum_{i=1}^{N_x}\sum_{j=1}^{N_y} \varphi_i^{(x)}(x)\frac{\partial^2 \varphi_j^{(y)}(y)}{\partial y^2} \right) u_{i,j} dxdy$$
$$+ \int_\Omega \varphi_r^{(x)}(x)\varphi_s^{(y)}(y) \left(\frac{2\pi^2}{1+2\pi^2}\cos(\pi x)\cos(\pi y) \right) dxdy, \tag{38}$$

where $r = (2, 3, \cdots, N_x - 1)$ and $s = (2, 3, \cdots, N_y - 1)$.

For the present computation, double integrals in (38) are replaced with repeated integrals, and their evaluation is then carried out using Gaussian points on grid lines.

As shown in Figure 5, error reduces rapidly with decreasing h for both the Galerkin and collocation solutions. The former outperforms the latter regarding accuracy and convergence rate. Furthermore, Table 2 indicates that condition numbers of the present system matrix are relatively low. For example, its value is only 1.30×10^4 for the case of using 5041 RBFs.

5.2.2 Neumann boundary conditions

This problem is exactly the same as the previous one, except that Dirichlet boundary conditions prescribed along the two horizontal boundaries are replaced with Neumann ones.

The approximate solution can be sought in the form

$$u(x,y) = \sum_{i=1}^{N_x} \varphi_i^{(x)}(x) \left(\varphi_1^{(y)}(y)u_{i,1} + \cdots + \varphi_{N_y}^{(y)}(y)u_{i,N_y} + \varphi_{N_y+1}^{(y)}(y)\frac{\partial u_{i,1}}{\partial y} + \varphi_{N_y+2}^{(y)}(y)\frac{\partial u_{i,N_y}}{\partial y} \right), \tag{39}$$

where $\partial u_{i,1}/\partial y$ and $\partial u_{i,N_y}/\partial y$ with $i = (1, 2, \cdots, N_x)$ are also considered as nodal values. In addition, the present indices r and s in the residual equation that is similar to (38) will run from 2 to $(N_x - 1)$ and from 1 to N_y, respectively. In comparison with the previous

problem, it can be seen that more algebraic equations are generated here. Figure 6 indicates that the accuracy of the Galerkin solution is far superior to that of the collocation solution. The condition numbers of the Galerkin approach are relatively low, varying from 3.24×10^0 to 1.16×10^4.

Through Figures 5 (Dirichlet-type problem) and 6 (Neumann-type problem), it can be seen that the order of accuracy reduces from $O(h^{3.28})$ to $O(h^{2.60})$ for the collocation solution, but slightly increases from $O(h^{3.84})$ to $O(h^{3.89})$ for the Galerkin solution. The proposed technique is able to work well for Neumann boundary conditions without the need for refining the grid near the boundaries, as is often the case with conventional techniques. This is a clear advantage of the present implementation.

5.2.3 Helmholtz problem

This test problem, which is taken from [2,13], is governed by

$$\frac{\partial^2 \bar{u}}{\partial x^2} + \frac{\partial^2 \bar{u}}{\partial y^2} - \bar{u}(x, y) + \cos(\pi x) \cos(\pi y) = 0, \tag{40}$$

on $-1 \leq x, y \leq 1$, with Neumann boundary conditions. Its exact solution is also given by (36).

Fasshauer [2] provided a MATLAB program for the Galerkin-RBF solution of this problem (Program 45.1). In this program, conventional 2D-RBF approximations (i.e. the field variable is represented by 2D-RBFNs, followed by successive differentiations to obtain its derivatives) are employed. To maintain consistency, we made two minor modifications to the program: (a) resolution of evaluation grid is chosen to be the same as that of the RBF centre grid, and (b) errors are computed in the relative L_2 norm. Errors and condition numbers of the system matrix are shown in Figure 7. It can be seen that the matrix condition number grows much faster with the conventional RBF approach than with the

proposed RBF approach. Moreover, the solution of the former becomes unstable for small values of h. In contrast, the error N_e of the present solution reduces consistently with decreasing h at a rate of $O(h^{3.96})$.

5.3 Domain decomposition

The most time-consuming part of the proposed technique lies in the computation of volume integrals to form the algebraic system. One strategy to overcome this problem is to use domain decomposition. The problem is divided into a number of non-overlapping subdomains. Relevant RBF matrices are constructed for a generic subdomain and they can then be directly applied to subdomains involved. A substructuring technique consists of two main stages: (i) find the solution on the interfaces and (ii) find the solutions to subproblems defined on subdomains. The interface system can be constructed by requiring continuity of the field variable and its derivatives of order up to $(p-1)$ ($p-$the order of the DE) across the subdomain interfaces.

The combination of the substructuring technique and the proposed method that is presented above will lead to an approximate solution u that is a C^1 function for second-order problems (e.g. example in Section 5.1.1) and C^3 function for fourth-order problems (e.g. example in Section 5.1.2).

However, as shown in [24], a C^p solution can be achieved if the DE is enforced to be satisfied at the interface points in the subdomain solutions. The solution procedure described in [24] can be straightforwardly applied here. Satisfaction of the DE on the interfaces can be made through the transformation of the network-weight space into the physical space. For simplicity, consider a second-order problem. The presence of integration constants allows the addition of extra equation representing the DE at the interface points. The network used for a typical subdomain is thus constructed to satisfy not only the values

of u at the grid points, but also the DE at the interface points.

Two versions of the present multidomain Galerkin technique are applied to the following ODE

$$\frac{d^2\bar{u}}{dx^2} + \frac{d\bar{u}}{dx} + \bar{u} = -\exp(-5x)\left[9979\sin(100x) + 900\cos(100x)\right], \quad 0 \leq x \leq 1, \tag{41}$$

with Dirichlet boundary conditions $\bar{u}(0) = 0$ and $\bar{u}(1) = \sin(100)\exp(-5)$. The exact solution can be verified to be

$$\bar{u}(x) = \sin(100x)\exp(-5x). \tag{42}$$

It is noted that the variation of (42) is highly oscillatory. The domain is partitioned into 6 and 11 subdomains that are then identically represented using grids of $\{3, 5, \cdots, 91\}$ uniform points. Figure 8 clearly shows that the present C^2 solution is, as expected, more stable and accurate than the C^1 solution. It also appears that the convergence rate is mainly decided by the subdomain solver used, while the accuracy level depends on continuity order of the solution across the interfaces.

6 Concluding remarks

In this paper, a numerical technique, based on 1D-IRBFNs and Galerkin approximation, is developed for solving elliptic differential equations. Prior to the process of discretising the differential equation, all boundary conditions are incorporated into the RBF approximations in an exact manner. The proposed technique has a clear advantage over Galerkin finite-element methods in the implementation of Neumann and multiple boundary conditions. In the context of RBF techniques, the proposed technique produces a system of equations that is often symmetric and has a relatively-low matrix condition number.

These facilitates the employment of much larger numbers of nodes. To avoid the problem of high cost associated with the evaluation of volume integrals, the use of domain decomposition is discussed, where continuity order can be improved. Numerical results have shown that (a) the proposed technique achieves a high rate of convergence, (b) the accuracy of the proposed method is much higher than that of the RBF collocation technique, and (c) the obtained solutions have similar levels of accuracy for both types of boundary conditions, Dirichlet and Neumann ones.

Acknowledgement

This work is supported by the Australian Research Council. We would like to thanks the referees for their helpful comments.

References

1. Brebbia CA. Dominguez J. *Boundary Elements–An Introductory Course*. Southampton: Computational Mechanics Publications; 1992.

2. Fasshauer GE. *Meshfree Approximation Methods With Matlab* (Interdisciplinary Mathematical Sciences - Vol. 6). Singapore: World Scientific Publishers; 2007.

3. Schaback R. Error estimates and condition numbers for radial basis function interpolation. *Advances in Computational Mathematics* 1995; 3:251–264.

4. Ling L. Kansa EJ. A least-squares preconditioner for radial basis functions collocation methods. *Advances in Computational Mathematics* 2005; 23:31–54.

5. Shu C. Ding H. Yeo KS. Computation of incompressible Navier-Stokes equations by local RBF-based differential quadrature method. *Computer Modeling in Engineering & Sciences* 2005; 7(2):195–206.

6. Sarler B. Vertnik R. Meshfree explicit local radial basis function collocation method for diffusion problems. *Computers & Mathematics with Applications* 2006; 51(8):1269–1282.

7. Wendland H. *Scattered Data Approximation.* Cambridge: Cambridge University Press; 2005.

8. Ingber MS. Chen CS. Tanski JA. A mesh free approach using radial basis functions and parallel domain decomposition for solving three-dimensional diffusion equations. *International Journal for Numerical Methods in Engineering* 2004; 60(13):2183–2201.

9. Divo E. Kassab A. Iterative domain decomposition meshless method modeling of incompressible viscous flows and conjugate heat transfer. *Engineering Analysis with Boundary Elements* 2006; 30(6):465–478.

10. Mai-Duy N. Tanner RI. A collocation method based on one-dimensional RBF interpolation scheme for solving PDEs. *International Journal of Numerical Methods for Heat & Fluid Flow* 2007; 17(2):165–186.

11. Mai-Duy N. Tran-Cong T. A Cartesian-grid collocation method based on radial-basis-function networks for solving PDEs in irregular domains. *Numerical Methods for Partial Differential Equations* 2007; 23(5):1192–1210.

12. Wendland H. Meshless Galerkin methods using radial basis functions. *Mathematics of Computation* 1999; 68:1521–1531.

13. Wendland H. Numerical solution of variational problems by radial basis functions. In: Chui CK, Schumaker LL, editors. *Approximation Theory IX, Vol. II: Computational Aspects.* Tennessee: Vanderbilt University Press, 361–368, 1999.

14. Hu H-Y. Li Z-C. Cheng AH-D. Radial basis collocation methods for elliptic boundary value problems. *Computers & Mathematics with Applications* 2005; 50(1-2):289–

320.

15. Fletcher CAJ. *Computational Galerkin Methods*. New York: Springer-Verlag; 1984.

16. Zienkiewicz OC. Taylor RL. *The Finite Element Method — Volume 1: The Basis* (Fifth edition). Oxford: Butterworth Heinemann; 2000.

17. Mai-Duy N. Tran-Cong T. Approximation of function and its derivatives using radial basis function networks. *Applied Mathematical Modelling* 2003; 27:197–220.

18. Kansa EJ. Power H. Fasshauer GE. Ling L. A volumetric integral radial basis function method for time-dependent partial differential equations: I. Formulation. *Engineering Analysis with Boundary Elements* 2004; 28:1191–1206.

19. Mai-Duy N. Tran-Cong T. Numerical solution of differential equations using multiquadric radial basis function networks. *Neural Networks* 2001; 14(2):185–199.

20. Ling L. Trummer MR. Adaptive multiquadric collocation for boundary layer problems. *Journal of Computational and Applied Mathematics* 2006; 188(2):265–282.

21. Shu C. Wu YL. Integrated radial basis functions-based differential quadrature method and its performance. *International Journal for Numerical Methods in Fluids* 2007; 53(6):969–984.

22. Sarra SA. Integrated multiquadric radial basis function approximation methods. *Computers & Mathematics with Applications* 2006; 51(8):1283–1296.

23. Trefethen LN. *Spectral Methods in MATLAB*. Philadelphia: SIAM; 2000.

24. Mai-Duy N. Tran-Cong T. A multidomain integrated radial basis function collocation method for elliptic problems. *Numerical Methods for Partial Differential Equations* (in press).

Table 1: 1D problem, Dirichlet boundary conditions: Condition number, cond$A$, versus a number of RBFs, N, for the Galerkin solution.

N	cond$A$	N	cond$A$
3	1.00e+0	49	3.73e+3
5	1.75e+1	51	4.04e+3
7	6.75e+1	53	4.37e+3
9	1.33e+2	55	4.71e+3
11	1.94e+2	57	5.07e+3
13	2.57e+2	59	5.43e+3
15	3.34e+2	61	5.81e+3
17	4.27e+2	63	6.21e+3
19	5.36e+2	65	6.61e+3
21	6.59e+2	67	7.03e+3
23	7.96e+2	69	7.46e+3
25	9.45e+2	71	7.90e+3
27	1.10e+3	73	8.36e+3
29	1.28e+3	75	8.83e+3
31	1.46e+3	77	9.31e+3
33	1.67e+3	79	9.81e+3
35	1.88e+3	81	1.03e+4
37	2.10e+3	83	1.08e+4
39	2.34e+3	85	1.13e+4
41	2.59e+3	87	1.19e+4
43	2.86e+3	89	1.24e+4
45	3.14e+3	91	1.30e+4
47	3.43e+3		

Table 2: 2D problem, Dirichlet boundary conditions: Condition number, cond$A$, versus a number of RBFs, N, for the Galerkin solution.

N	cond$A$	N	cond$A$
9	1.00e+0	1521	3.90e+3
25	2.54e+1	1681	4.32e+3
49	1.30e+2	1849	4.75e+3
81	2.69e+2	2025	5.21e+3
121	3.63e+2	2209	5.69e+3
169	4.53e+2	2401	6.19e+3
225	5.72e+2	2601	6.71e+3
289	7.25e+2	2809	7.25e+3
361	9.05e+2	3025	7.82e+3
441	1.11e+3	3249	8.40e+3
529	1.33e+3	3481	9.00e+3
625	1.58e+3	3721	9.63e+3
729	1.85e+3	3969	1.02e+4
841	2.14e+3	4225	1.09e+4
961	2.45e+3	4489	1.16e+4
1089	2.78e+3	4761	1.23e+4
1225	3.13e+3	5041	1.30e+4
1369	3.51e+3		

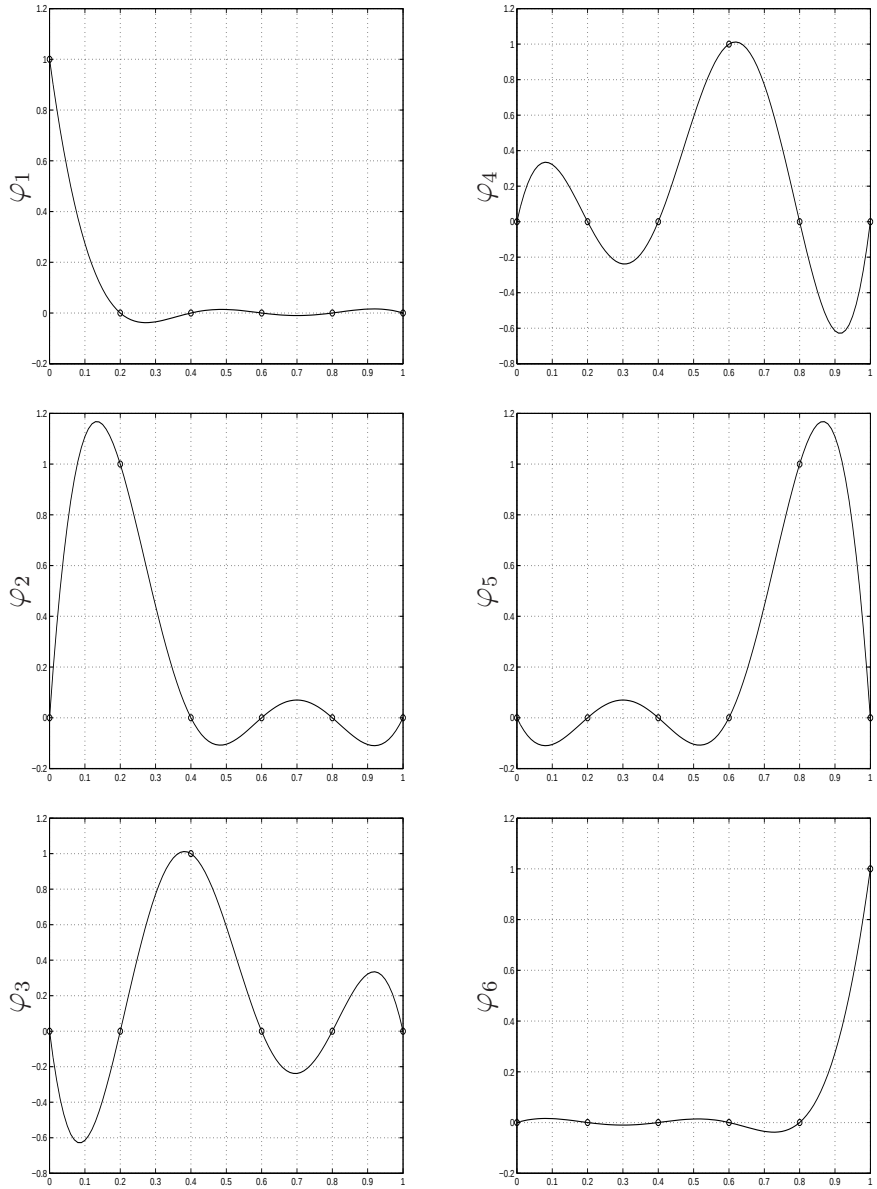

Figure 1: IRBFN-2's trial/weighting functions with 6 RBFs.

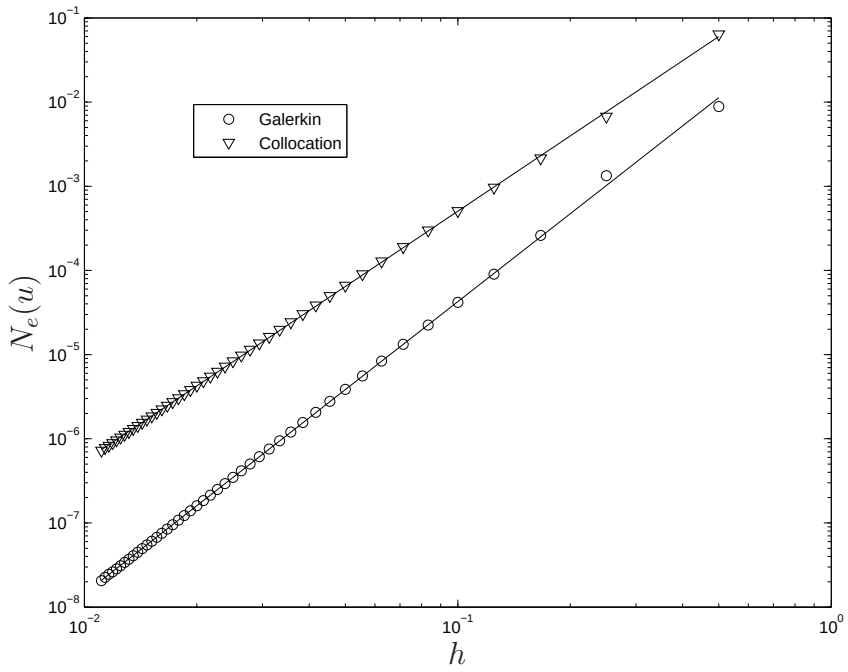

Figure 2: 1D problem, Dirichlet boundary conditions, 1D-IRBFNs: Error $N_e(u)$ versus the centre spacing h for the Galerkin and collocation solutions. They converge as $O(h^{3.46})$ and $O(h^{2.96})$, respectively.

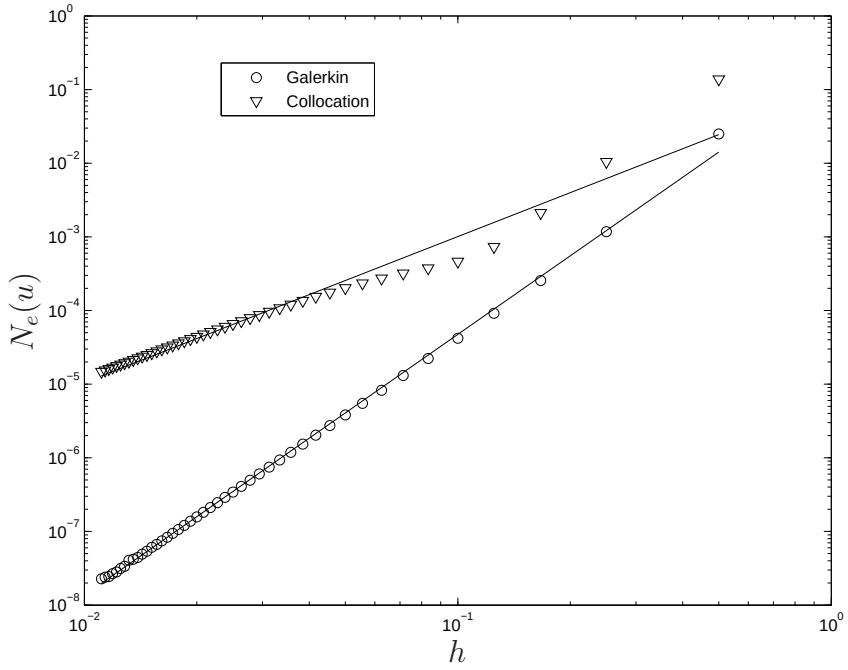

Figure 3: 1D problem, Dirichlet and Neumann boundary conditions, 1D-IRBFNs: Error $N_e(u)$ versus the centre spacing h for the Galerkin and collocation solutions. They converge as $O(h^{3.54})$ and $O(h^{1.98})$, respectively.

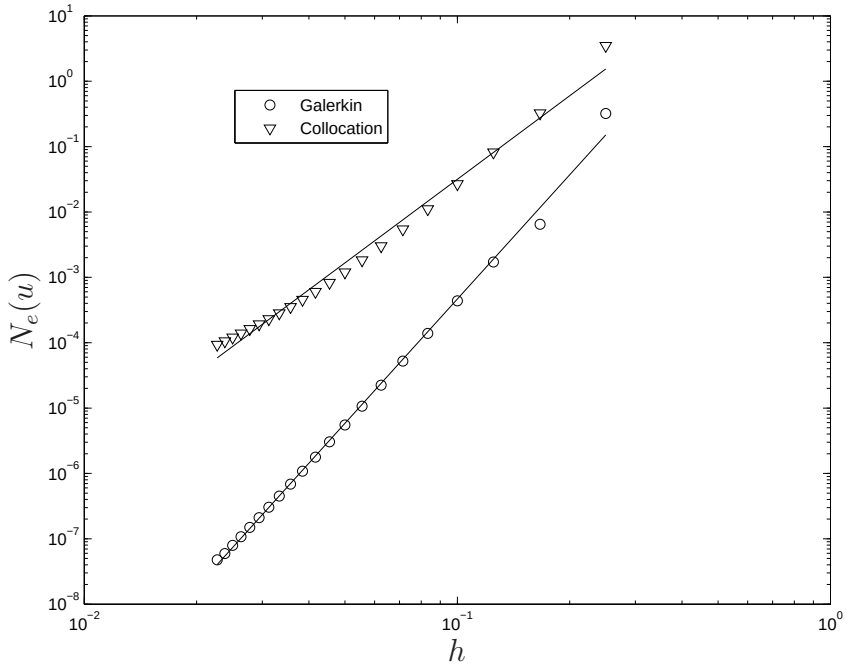

Figure 4: 1D problem, biharmonic equation, Dirichlet boundary conditions, 1D-IRBFNs: Error $N_e(u)$ versus the centre spacing h for the Galerkin and collocation solutions. They converge as $O(h^{6.31})$ and $O(h^{4.24})$, respectively.

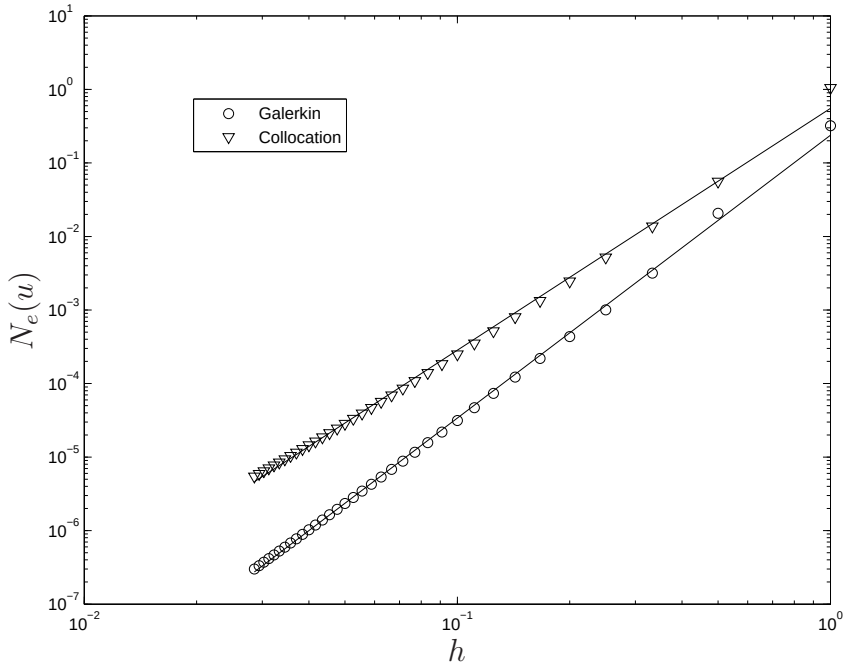

Figure 5: 2D problem, Dirichlet boundary conditions, 1D-IRBFNs: Error $N_e(u)$ versus the centre spacing h for the Galerkin and collocation solutions. They converge as $O(h^{3.84})$ and $O(h^{3.28})$, respectively.

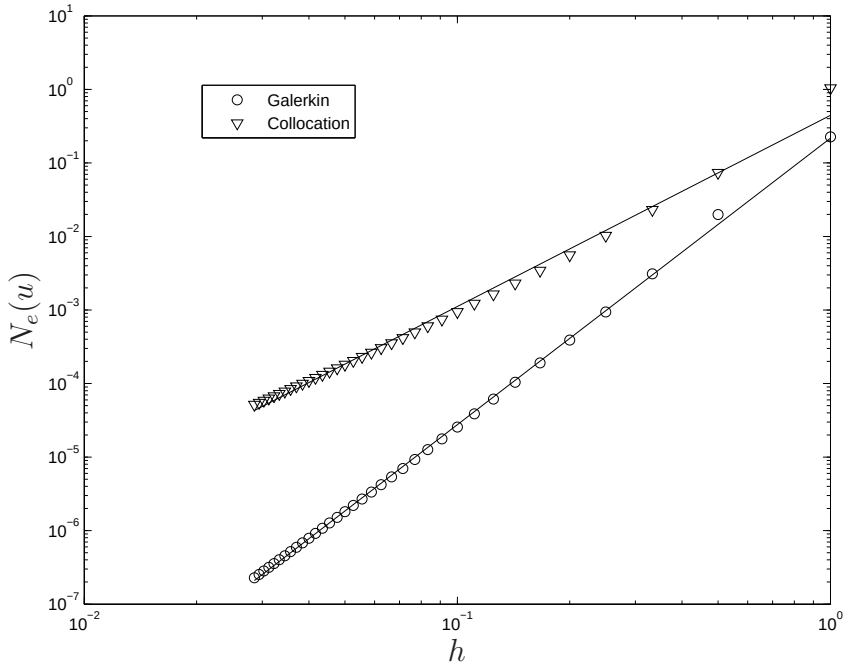

Figure 6: 2D problem, Dirichlet and Neumann boundary conditions, 1D-IRBFNs: Error $N_e(u)$ versus the centre spacing h for the Galerkin and collocation solutions. They converge as $O(h^{3.89})$ and $O(h^{2.60})$, respectively.

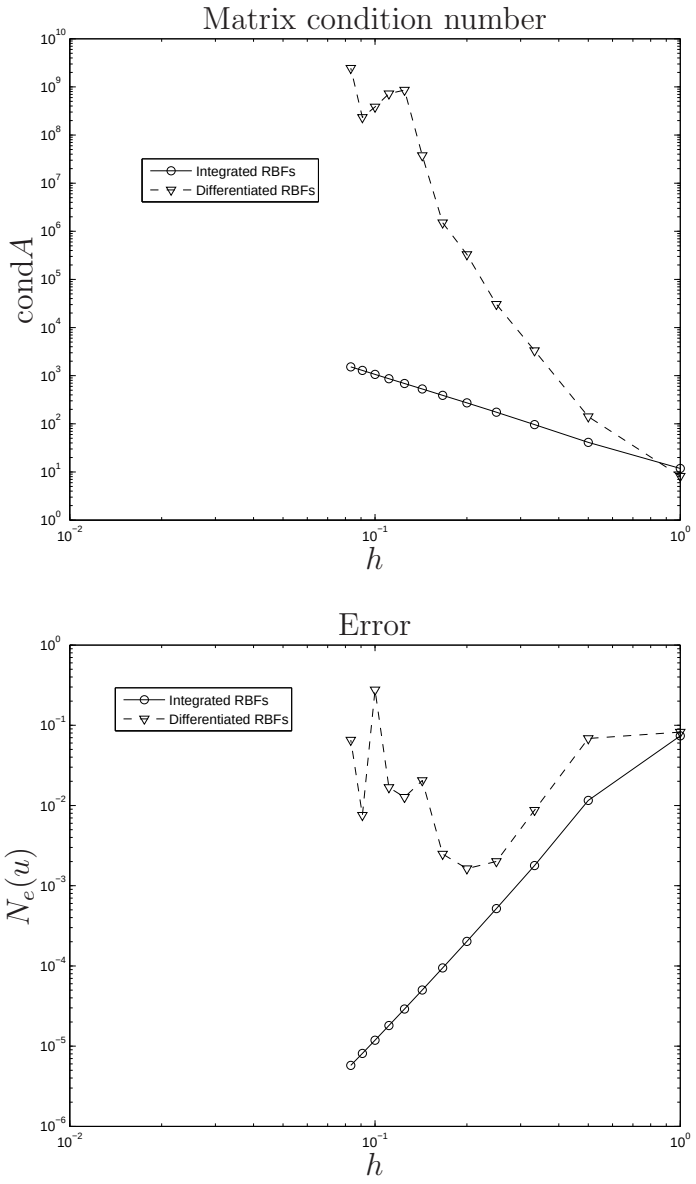

Figure 7: 2D Helmholtz problem, Galerkin formulation: Comparison of the matrix condition number and error between the Galerkin solutions using integrated and differentiated RBFs. The latter is obtained using a MATLAB program provided by Fasshauer [2].

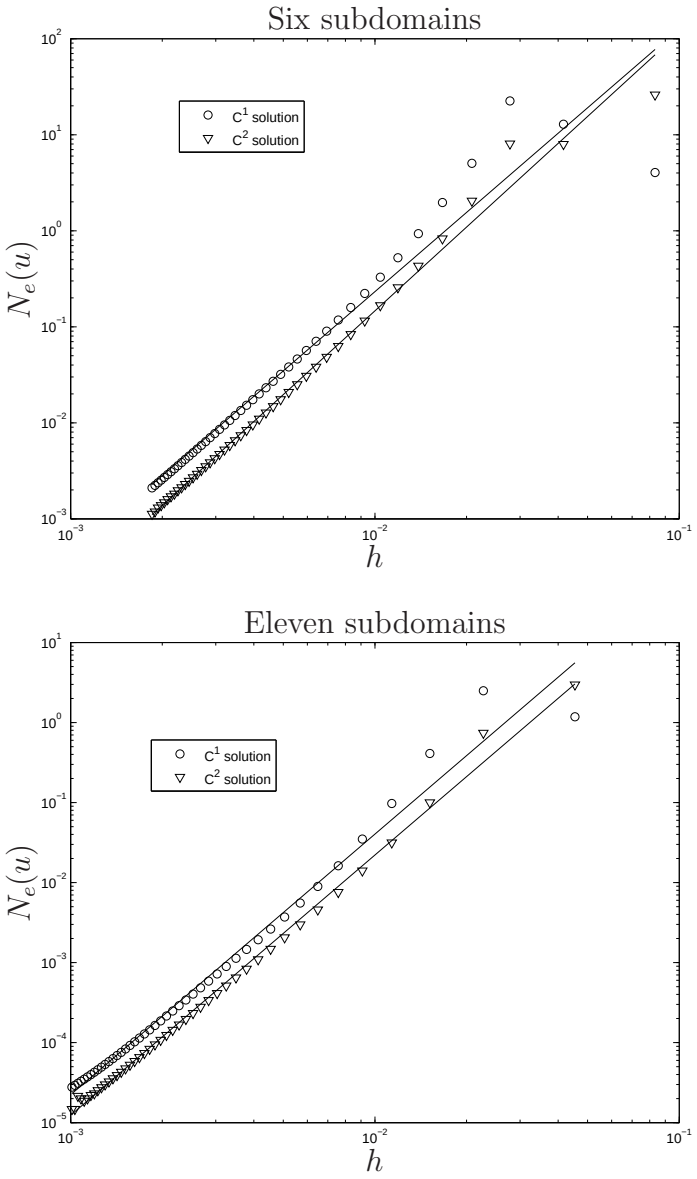

Figure 8: Domain decomposition, second-order problem: Errors of the present C^1 and C^2 Galerkin solutions for 6 and 11 subdomains.