# On the use of integrated radial basis function schemes in weighted residual statements for elliptic problems

N. Mai-Duy & T. Tran-Cong
*Computational Engineering and Science Research Centre, University of Southern Queensland, Australia*

## Abstract

In this paper, we discuss the use of integrated radial basis functions (IRBFs) in solving elliptic differential equations. Various formulations, namely point collocation, subregion collocation, Galerkin and inverse statements, are considered. IRBFs are incorporated into these formulations to represent the field variables. Numerical results indicate that this use of IRBFs leads to a considerable improvement in accuracy and convergence rate over the case of using conventional low-order polynomials.
*Keywords: integrated radial basis functions, collocation method, Galerkin method, control volume method, boundary element method.*

## 1 Introduction

Mathematical modelling of physical processes usually leads to partial/ordinary differential equations (PDEs/ODEs). Consider a differential problem governed by

$$\mathcal{L}\bar{u} = b, \quad \mathbf{x} \in \Omega, \tag{1}$$

where $\bar{u}$ is the field variable, $b$ a given function, $\mathcal{L}$ a differential operator, $\mathbf{x}$ the position vector and $\Omega$ the domain. A function $\bar{u}(\mathbf{x})$ can be sought in the form of truncated series

$$\bar{u}(\mathbf{x}) \approx u(\mathbf{x}) = \sum_{i=1}^{n} u^{(i)}\phi^{(i)}(\mathbf{x}) + \gamma, \quad \mathbf{x} \in \Omega, \tag{2}$$

WIT Transactions on Modelling and Simulation, Vol 49, © 2009 WIT Press
www.witpress.com, ISSN 1743-355X (on-line)
doi:10.2495/BE090161

where $\{\phi^{(i)}(\mathbf{x})\}_{i=1}^{n}$ is the set of basis/trial functions which are known, $\{u^{(i)}\}_{i=1}^{n}$ the set of nodal variable values to be found, and $\gamma$ a known value. The weighted residuals approach tries to reduce the residual of (1) to a minimum through

$$\int_{\Omega} \psi(\mathbf{x})\,(Lu - b)\,d\Omega = 0, \quad \mathbf{x} \in \Omega, \tag{3}$$

where $\psi(\mathbf{x})$ is a weighting function. Different choices of $\psi(\mathbf{x})$ result in different discretisation formulations such as point collocation, subregion collocation, Galerkin and inverse statements. More details can be found in [1].

RBF networks (RBFNs) have emerged as a powerful approximation tool [2]. A network relies on a set of points that can be uniformly/nonuniformly distributed throughout the domain for the representation of a function. Some RBFs such as the multiquadric and Gaussian basis functions can offer an exponential rate of convergence. To avoid the problem of reduced convergence rate caused by differentiation, integrated RBFNs (IRBFNs) have been proposed [3]. In this paper we discuss the use of IRBFNs as an interpolating method for different discretisation schemes for the solution of elliptic DEs. This discussion is based on our previous works on IRBFNs reported in [4–7].

The remainder of the paper is organised as follows. Section 2 gives a brief review of IRBFNs. Section 3 is concerned with the discussion of using IRBFNs as trial functions for the solution of DEs, in which several representative examples are given. Section 4 concludes the paper.

## 2 Integrated radial-basis-function networks

RBFNs allow a conversion of a function $f$ from a low-dimensional space (e.g. 1D-3D) to a high-dimensional space in which the function will be expressed as a linear combination of RBFs

$$f(\mathbf{x}) = \sum_{i=1}^{m} w^{(i)} g^{(i)}(\mathbf{x}), \tag{4}$$

where the superscript $(i)$ is the sum index, $\mathbf{x}$ the input vector, $\{w^{(i)}\}_{i=1}^{m}$ the set of network weights to be found, and $\{g^{(i)}(\mathbf{x})\}_{i=1}^{m}$ the set of RBFs. An example of RBFs is the multiquadric (MQ) basis function $g^{(i)}(\mathbf{x}) = \sqrt{(\mathbf{x} - \mathbf{c}^{(i)})^T(\mathbf{x} - \mathbf{c}^{(i)}) + a^{(i)2}}$, where $\mathbf{c}^{(i)}$ and $a^{(i)}$ are the centre and width of the $i$th MQ-RBF, respectively.

IRBFNs consist in decomposing the highest-order derivatives of $u$ in (1) into RBFs in the form of (4) ($f(\mathbf{x}) = \partial^p u(\mathbf{x})/\partial x_j^p$) and then integrating them to obtain lower-order derivatives and the function itself

$$\frac{\partial^p u(\mathbf{x})}{\partial x_j^p} = \sum_{i=1}^{m} w_{[x_j]}^{(i)} g^{(i)}(\mathbf{x}), \tag{5}$$

$$\frac{\partial^{p-1} u(\mathbf{x})}{\partial x_j^{p-1}} = \sum_{i=1}^{m+q_1} w_{[x_j]}^{(i)} H_{[x_j]}^{(i)[p-1]}(\mathbf{x}), \tag{6}$$

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

$$u_{[x_j]}(\mathbf{x}) = \sum_{i=1}^{m+q_p} w_{[x_j]}^{(i)} H_{[x_j]}^{(i)[0]}(\mathbf{x}), \tag{7}$$

where the subscript $[x_j]$ is used to denote the process of integration with respect to $x_j$; $q_1$ the number of centres in a subnetwork that is employed to approximate a set of nodal integration constants, $q_2 = 2q_1, \cdots, q_p = pq_1$; and $H_{[x_j]}^{(i)[p-1]} = \int g^{(i)} dx_j, H_{[x_j]}^{(i)[p-2]} = \int H_{[x_j]}^{(i)[p-1]} dx_j, \cdots, H_{[x_j]}^{(i)[0]} = \int H_{[x_j]}^{(i)[1]} dx_j$. For convenience of presentation, we introduce another notation, $H_{[x_j]}^{(i)[p]}(\mathbf{x})$, to denote the RBF (i.e. $H_{[x_j]}^{(i)[p]}(\mathbf{x}) \equiv g^{(i)}(\mathbf{x})$) so that $H_{[x_j]}^{(i)[p-1]} = \int H_{[x_j]}^{(i)[p]} dx_j$. It is noted that the new centres and their associated known basis functions in subnetworks are also denoted by the notations $w^{(i)}$ and $H^{(i)}(\mathbf{x})$, respectively, but with $i > m$. An IRBFN is said to be of order $p$ if its starting point is the $p$th-order derivative.

We seek the solution in terms of nodal variable values for the purpose of having a clear physical meaning and computational efficiency. The evaluation of (5)–(7) at a set of collocation points $\{\mathbf{x}^{(i)}\}_{i=1}^m$, which is selected to coincide with the set of centres $\{\mathbf{c}^{(i)}\}_{i=1}^m$, yields

$$\widetilde{\frac{\partial^p u}{\partial x_j^p}} = \widetilde{\mathcal{H}}_{[x_j]}^{[p]} \widetilde{w}_{[x_j]}, \tag{8}$$

$$\widetilde{\frac{\partial^{p-1} u}{\partial x_j^{p-1}}} = \widetilde{\mathcal{H}}_{[x_j]}^{[p-1]} \widetilde{w}_{[x_j]}, \tag{9}$$

$$\cdots \quad \cdots$$

$$\widetilde{u} = \widetilde{\mathcal{H}}_{[x_j]}^{[0]} \widetilde{w}_{[x_j]}, \tag{10}$$

where

$$\widetilde{w}_{[x_j]} = \left[ w_{[x_j]}^{(1)}, w_{[x_j]}^{(2)}, \cdots, w_{[x_j]}^{(m+q_p)} \right]^T,$$

$$\widetilde{u} = \left[ u(\mathbf{x}^{(1)}), u(\mathbf{x}^{(2)}), \cdots, u(\mathbf{x}^{(m)}) \right]^T = \left[ u^{(1)}, u^{(2)}, \cdots, u^{(m)} \right]^T,$$

$$\widetilde{\frac{\partial^k u}{\partial x_j^k}} = \left[ \frac{\partial^k u(\mathbf{x}^{(1)})}{\partial x_j^k}, \cdots, \frac{\partial^k u(\mathbf{x}^{(m)})}{\partial x_j^k} \right]^T = \left[ \frac{\partial^k u^{(1)}}{\partial x_j^k}, \cdots, \frac{\partial^k u^{(m)}}{\partial x_j^k} \right]^T,$$

$k = \{1, \cdots, p\}$, and the matrices $\widetilde{\mathcal{H}}_{[x_j]}^{[.]}$ have entries $\left( \widetilde{\mathcal{H}}_{[x_j]}^{[.]} \right)^{(l,i)} = H_{[x_j]}^{(i)[.]}(\mathbf{x}^{(l)})$, where $1 \le l \le m$ and $1 \le i \le (m + q_p)$. In (8)–(10), the matrices, $\widetilde{\mathcal{H}}_{[x_j]}^{[p]}, \widetilde{\mathcal{H}}_{[x_j]}^{[p-1]}, \ldots, \widetilde{\mathcal{H}}_{[x_j]}^{[1]}$, are augmented using zero-submatrices so that they have the same size as the matrix $\widetilde{\mathcal{H}}_{[x_j]}^{[0]}$.

Owing to the presence of the constants of integration (extra coefficients), one can add extra equations to the conversion system that transforms the RBF space into the physical space

$$\begin{pmatrix} \widetilde{u} \\ \widehat{e}_{[x_j]} \end{pmatrix} = \begin{bmatrix} \widetilde{\mathcal{H}}^{[0]}_{[x_j]} \\ \widehat{\mathcal{K}}_{[x_j]} \end{bmatrix} \widetilde{w}_{[x_j]} = \widetilde{\mathcal{C}}_{[x_j]} \widetilde{w}_{[x_j]}, \tag{11}$$

$$\widetilde{w}_{[x_j]} = \widetilde{\mathcal{C}}^{-1}_{[x_j]} \begin{pmatrix} \widetilde{u} \\ \widehat{e}_{[x_j]} \end{pmatrix}, \tag{12}$$

where $\widetilde{\mathcal{C}}_{[x_j]}$ is the conversion matrix, $\widehat{e}_{[x_j]} = \widehat{\mathcal{K}}_{[x_j]} \widetilde{w}_{[x_j]}$ the set of extra equations. When the boundary data involve derivative values, these extra equations can be used to represent derivative boundary values in the $x_j$ direction.

Substitution of (12) into (5)–(7) yields

$$u(\mathbf{x}) = \frac{1}{N} \sum_{j=1}^{N} \left( \left[ H^{(1)[0]}_{[x_j]}(\mathbf{x}), H^{(2)[0]}_{[x_j]}(\mathbf{x}), \cdots \right] \widetilde{\mathcal{C}}^{-1}_{[x_j]} \begin{pmatrix} \widetilde{u} \\ \widehat{e}_{[x_j]} \end{pmatrix} \right), \tag{13}$$

$$\frac{\partial u(\mathbf{x})}{\partial x_j} = \left[ H^{(1)[1]}_{[x_j]}(\mathbf{x}), H^{(2)[1]}_{[x_j]}(\mathbf{x}), \cdots \right] \widetilde{\mathcal{C}}^{-1}_{[x_j]} \begin{pmatrix} \widetilde{u} \\ \widehat{e}_{[x_j]} \end{pmatrix}, \tag{14}$$

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

$$\frac{\partial^p u(\mathbf{x})}{\partial x_j^p} = \left[ H^{(1)[p]}_{[x_j]}(\mathbf{x}), H^{(2)[p]}_{[x_j]}(\mathbf{x}), \cdots \right] \widetilde{\mathcal{C}}^{-1}_{[x_j]} \begin{pmatrix} \widetilde{u} \\ \widehat{e}_{[x_j]} \end{pmatrix}, \tag{15}$$

where $N$ is the dimension of the problem and the approximate function $u(\mathbf{x})$ is taken to be the average value of the $u_{[x_j]}(\mathbf{x})$ due to numerical error.

The calculation of cross derivatives of $u$ is based on the following relation

$$\frac{\partial^p u}{\partial x_i^r \partial x_j^s} = \frac{1}{2} \left( \frac{\partial^r}{\partial x_i^r} \left( \frac{\partial^s u}{\partial x_j^s} \right) + \frac{\partial^s}{\partial x_j^s} \left( \frac{\partial^r u}{\partial x_i^r} \right) \right), \quad p = r + s,\ i \neq j, \tag{16}$$

which reduces the computation of mixed derivatives to that of lower-order pure derivatives for which IRBFNs involve integration with respect to $x_i$ or $x_j$ only.

Since all integrals involved can be obtained analytically, IRBFNs only require a set of distinct points for the approximation of a function.

## 3 IRBFNs for PDEs

IRBFNs are employed to represent the field variable. The governing equation is discretised using various formulations, namely point-collocation, control-volume, Galerkin and boundary integral equation schemes. We implement IRBFNs with the MQ basis function. The MQ width $a^{(i)}$ is simply chosen to be the minimal distance between the centre $\mathbf{c}^{(i)}$ and its neighbours. The accuracy of a numerical

scheme is measured through the discrete relative $L_2$ error of the solution $u$, denoted by $Ne(u)$. Let $h$ and $n_{ip}$ be the "mesh" size and the number of interior points. Another important measure is the convergence rate of the solution $u$, defined by $N_e(u) \approx \theta h^{\nu} = O(h^{\nu})$ or $N_e(u) \approx \theta n_{ip}^{\nu} = O(n_{ip}^{\nu})$ where $\theta$ and $\nu$ are the exponential model's parameters. Only 1D and 2D elliptic problems are considered here.

### 3.1 Point-collocation schemes

For these schemes, the residual of (1) is exactly zero at certain points. There are no integrations required in discretising (1). Two versions of IRBFNs are considered.

#### 3.1.1 Two-dimensional IRBFNs

The problem domain is discretised using a set of scattered points (Figure 1a). We employ (13)–(15) over the domain, i.e. 2D-IRBFNs. It can be seen that the 2D-IRBFN collocation technique is truly meshless. The accuracy of the technique is demonstrated through the solution of $\nabla^2 \bar{u} = \sin(\pi x_1)\sin(\pi x_2)$ on a unit square with homogeneous Dirichlet boundary conditions. Its exact solution is $\bar{u}(x_1, x_2) = -(1/2\pi^2)\sin(\pi x_1)\sin(\pi x_2)$. Four scattered data sets of interior points using $n_{ip} = \{32, 52, 89, 145\}$ are employed . We discretise the boundaries using uniformly-distributed points with $n_{x_1} = n_{x_2} = \sqrt{n_{ip}}$. The value of $q_1$ is taken as $3n_{x_j}$. The variable $u$ is represented using IRBFNs of second order, where $\widehat{e}$ and $\widehat{\mathcal{K}}$ in (13)–(15) are simply set to null. For comparison purposes, the conventional (differentiated) RBFN (DRBFN) approach is also considered here. Results are displayed in Figure 2, indicating that the IRBFN approach is superior to the DRBFN approach regarding accuracy and convergence rate. The solution converges apparently as $O(n_{ip}^{-1.46})$ for IRBFN and $O(n_{ip}^{-0.51})$ for DRBFN.

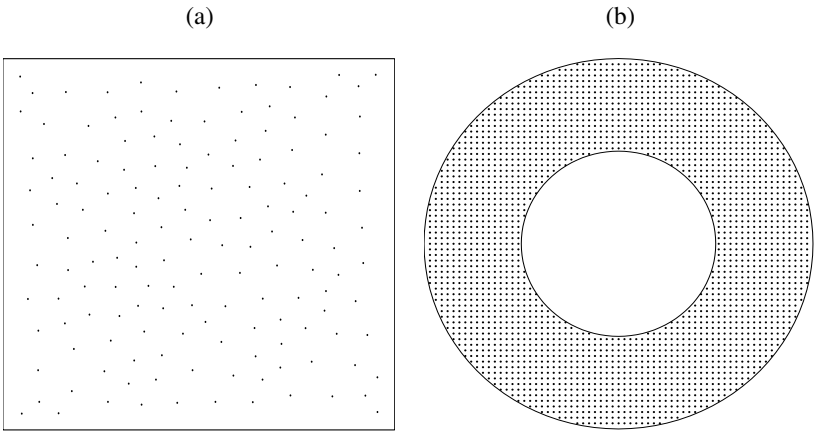


Figure 1: Point-collocation scheme: scattered point (2D-IRBFNs) and Cartesian-grid (1D-IRBFNs) discretisations.

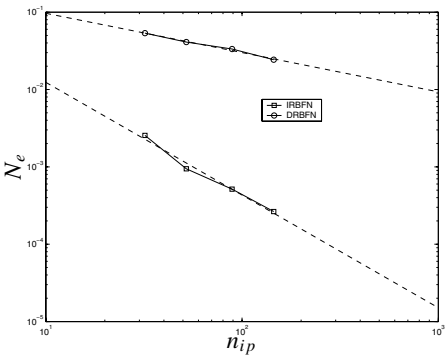


Figure 2: Point collocation scheme, 2D-IRBFNs: accuracy by the IRBFN and DRBFN methods.

### 3.1.2 One-dimensional IRBFNs

The problem domain is discretised using a Cartesian grid (Figure 1b). We employ (13)–(15) on grid lines, i.e. 1D-IRBFNs. The construction of the 1D-IRBFN approximations for a point $\mathbf{x}$ involves only points that lie on grid lines intersecting at $\mathbf{x}$ rather than the whole set of data points. The inversion is now conducted for a series of small matrices rather than for a large matrix. This use of 1D-IRBFNs thus leads to a considerable economy in forming the system matrix over that of 2D-IRBFNs. Consider the biharmonic equation $\nabla^4\bar{u} = 256(\pi^2 - 1)^2[\sin(4\pi x_1)\cosh(4x_2) - \cos(4\pi x_1)\sinh(4x_2)]$ defined on an annulus domain of radii $R_1 = 1/4$ and $R_2 = 1/2$ (Figure 1b) and subject to Dirichlet boundary conditions ($\bar{u}$ and $\partial\bar{u}/\partial n$). The exact solution is $\bar{u} = \sin(4\pi x_1)\cosh(4x_2) - \cos(4\pi x_1)\sinh(4x_2)$. We employ 1D-IRBFNs of fourth order, where $\widehat{e}_{[x_j]}$ is made up of the values of $\partial\bar{u}/\partial x_j$ at the two end points of a grid line. Double boundary conditions are thus incorporated into the system in an accurate manner. Table 1 shows that the proposed method produces a very high convergence rate, $O(h^{5.39})$ with relatively-low matrix condition numbers.

## 3.2 Boundary integral equation (BIE) schemes

These schemes are based on free-space fundamental solutions. One attractive feature of BIE schemes is that the differential equation is satisfied exactly. Consider a Dirichlet biharmonic problem. The BIE analog of $\nabla^4\bar{u} = b$ can be written as

$$C(\mathbf{y})\bar{u}(\mathbf{y}) + \int_\Gamma \frac{\partial G^H(\mathbf{y},\mathbf{x})}{\partial n}\bar{u}(\mathbf{x})d\Gamma = \int_\Gamma G^H(\mathbf{y},\mathbf{x})\frac{\partial\bar{u}(\mathbf{x})}{\partial n}d\Gamma - \int_\Gamma \left(\frac{\partial G^B(\mathbf{y},\mathbf{x})}{\partial n}\bar{v}(\mathbf{x}) - G^B(\mathbf{y},\mathbf{x})\frac{\partial\bar{v}(\mathbf{x})}{\partial n}\right)d\Gamma - \int_\Omega G^B(\mathbf{y},\mathbf{x})b(\mathbf{x})d\Omega, \quad (17)$$

Table 1: Point collocation scheme, 1D-IRBFNs: Condition number of the system matrix $\widetilde{\mathcal{A}}$ and accuracy of the solution $u$. Notice that $a(-b)$ means $a \times 10^{-b}$.

| Grid | cond$\widetilde{\mathcal{A}}$ | $N_e(u)$ |
|---|---|---|
| $11 \times 11$ | 2.93(1) | 1.15(−2) |
| $17 \times 17$ | 5.40(2) | 1.05(−3) |
| $21 \times 21$ | 2.12(3) | 5.56(−4) |
| $27 \times 27$ | 3.53(3) | 4.81(−5) |
| $31 \times 31$ | 1.41(4) | 2.47(−5) |
| $37 \times 37$ | 1.24(4) | 1.46(−5) |
| $41 \times 41$ | 3.41(4) | 8.37(−6) |
| $47 \times 47$ | 5.80(4) | 1.97(−6) |
| $51 \times 51$ | 8.37(4) | 1.77(−6) |
| $57 \times 57$ | 1.50(5) | 1.37(−6) |
| $61 \times 61$ | 2.29(5) | 8.85(−7) |
| $67 \times 67$ | 2.70(5) | 5.87(−7) |
| | | $O(h^{5.39})$ |

where $\mathbf{y}$ is the source point, $\mathbf{x}$ the field point, $\Gamma$ the piecewise smooth boundary of a domain $\Omega$ in $\mathbf{R}^2$, $C(\mathbf{y})$ the free term coefficient, $\bar{v}$ the new variable defined as $\bar{v} = \nabla^2 \bar{u}$, $n$ the outward normal direction at a point on the boundary, and $G^H$ and $G^B$ the harmonic and biharmonic fundamental solutions.

For traditional BIEMs, two BIEs are required and often solved in a coupled manner. Lagrange polynomials such as constant, linear and quadratic functions are usually employed to approximate the variations of $\bar{v}$ and $\partial\bar{v}/\partial n$ along the boundary.

A domain-type interpolation scheme is adopted here to represent the variable $u$, from which approximations to the unknown variables $\bar{v}$ and $\partial\bar{v}/\partial n$ are derived. From the prescribed boundary conditions $\bar{u}$ and $\partial\bar{u}/\partial n$, the values of $\partial\bar{u}/\partial x_1$ and $\partial\bar{u}/\partial x_2$ at a boundary point can be easily obtained. We implement 2D-IRBFNs of fourth order, where $\widehat{e}_{[x_j]}$ is made up of the values of $\partial\bar{u}/\partial x_j$ at the boundary points. The 2D-IRBFN BIE technique requires only one BIE, namely (17). The present unknowns are the values of $u$ at the interior points. The algebraic system is generated by applying the BIE (17) at the interior points. It should be emphasised that the present equation system consists of the interior equations only, thus completely avoiding all difficulties in numerical computation caused by the singularity of boundary integrals.

Table 2: BIE scheme: accuracy and convergence rate.

| | Linear | | IRBFN-4 | |
|---|---|---|---|---|
| | $Ne(v)$ | | $Ne(v)$ | |
| $n_{x_1} = n_{x_2}$ | Boundary | Interior | Boundary | Interior |
| 5 | 4.6349(0) | 1.7829(−1) | 1.0325(−1) | 2.6475(−1) |
| 7 | 4.3553(0) | 1.2632(−1) | 2.1247(−2) | 3.1643(−2) |
| 9 | 4.0599(0) | 9.0515(−2) | 9.8104(−3) | 1.0121(−2) |
| 11 | 3.7946(0) | 7.0285(−2) | 5.6267(−3) | 4.6423(−3) |
| 13 | 3.5625(0) | 5.7887(−2) | 3.5808(−3) | 2.5094(−3) |
| 15 | 3.3602(0) | 4.9755(−2) | 2.4541(−3) | 1.5060(−3) |
| 17 | 3.1830(0) | 4.4192(−2) | 1.7799(−3) | 9.7711(−4) |
| 19 | 3.0262(0) | 4.0332(−2) | 1.3494(−3) | 6.7476(−4) |
| 21 | 2.8860(0) | 3.7703(−2) | 1.0595(−3) | 4.9015(−4) |
| 23 | 2.7592(0) | 3.6049(−2) | 8.5548(−4) | 3.7082(−4) |
| 25 | 2.6431(0) | 3.5238(−2) | 7.0667(−4) | 2.8978(−4) |
| 27 | 2.5356(0) | 3.5226(−2) | 5.9459(−4) | 2.3265(−4) |
| 29 | 2.4348(0) | 3.6038(−2) | 5.0821(−4) | 1.9050(−4) |
| 31 | 2.3393(0) | 3.7778(−2) | 4.4080(−4) | 1.5844(−4) |
| | $O(h^{0.35})$ | $O(h^{0.84})$ | $O(h^{2.58})$ | $O(h^{3.52})$ |

A test problem chosen here is $\nabla^4 u = 0$ on $-2 \leq x_1, x_2 \leq 2$ with Dirichlet boundary conditions. The exact solution of this problem is $\bar{u} = (1/2)x_1(\sin x_1 \cosh x_2 - \cos x_1 \sinh x_2)$ Results concerning $Ne$ are shown in Table 2, together with those obtained by a linear-BIEM. The present method yields a much faster convergence rate.

### 3.3 Galerkin schemes

For these schemes, the residual of (1) is zero in an average sense. Galerkin schemes have a smoothing capability owing to their integral nature. Consider a rectangular domain. We use a Cartesian grid to generate the finite trial and test spaces. 1D-IRBFNs are employed on grid lines. The present 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. Moreover, any derivative of the field variable is defined and continuous throughout the entire domain.

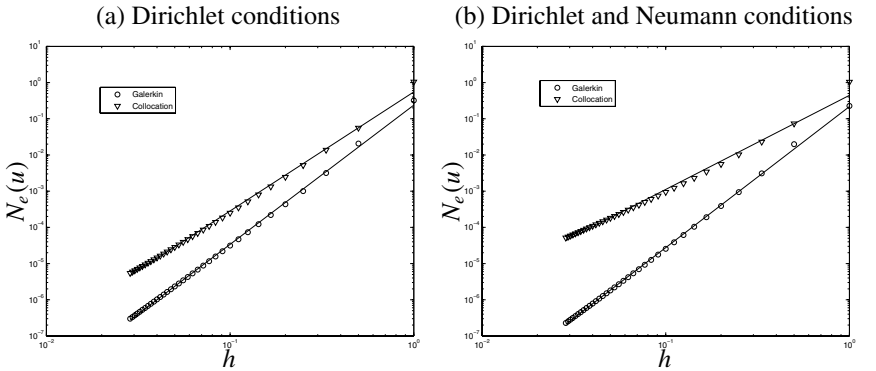


Figure 3: Galerkin scheme, 1D-IRBFNs: Error $N_e(u)$ versus the centre spacing $h$ for the Galerkin and collocation solutions. They converge as $O(h^{3.84})$ and $O(h^{3.28})$ for (a) and $O(h^{3.89})$ and $O(h^{2.60})$ for (b).

### 3.3.1 Dirichlet boundary conditions

The accuracy of the 1D-IRBFN Galerkin method is demonstrated through the solution of $\nabla^2\bar{u} = -(2\pi^2/(1+2\pi^2))\cos(\pi x_1)\cos(\pi x_2)$ on $-1 \le x_1, x_2 \le 1$ with Dirichlet boundary conditions. Its exact solution is $\bar{u}(x_1, x_2) = (1/(1+2\pi^2))\cos(\pi x_1)\cos(\pi x_2)$. Uniform grids, $3\times 3, 5\times 5, \cdots, 71\times 71$, are employed. As shown in Figure 3a, error reduces rapidly with decreasing $h$ for both the Galerkin and collocation solutions. The former outperforms the latter regarding accuracy and convergence rate. Condition numbers of the present system matrix are in the range of 1.0 to $1.3 \times 10^4$.

### 3.3.2 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. Figure 3b 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 $3.24 \times 10^0$ to $1.16 \times 10^4$.

Through Figures 3a (Dirichlet-type problem) and 3b (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 1D-IRBFN Galerkin 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.

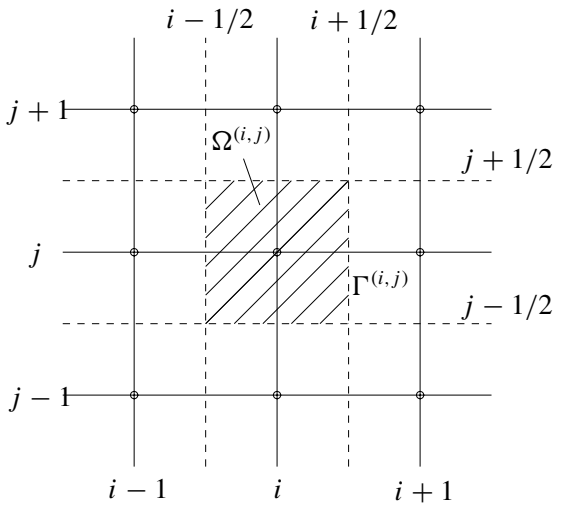


Figure 4: Control-volume scheme: Node $(i, j)$ and its associated control volume $\Omega^{(i,j)}$. Note that the dash lines represent the faces of the control volume.

### 3.4 Subregion-collocation/control-volume schemes

For these schemes, the weighting function is chosen to be unity over a control volume. Control-volume (CV) formulations are based on the actual satisfaction of the physical laws (i.e. the conservations of mass, momentum and energy) rather than on the satisfaction of approximate discrete expressions controlled by means of mesh size. The accuracy of a CV technique depends on both the approximation of gradients (e.g. diffusive fluxes) and the evaluation of integrals involving these gradients. For the latter, assume that the flux evaluations are sufficiently accurate, the midpoint rule is capable of yielding second-order accuracy only. Consider the diffusion equation $\nabla \cdot \nabla u = 0$ on $0 \leq x_1, x_2 \leq \pi$. Its exact solution is $\bar{u} = (1/\sinh(\pi)) \sin(x_1) \sinh(x_2)$. The problem domain is discretised using a uniform Cartesian grid. For each grid point $(x_1^{(i)}, x_2^{(j)})$, one can construct a CV $\Omega^{(i,j)}$ with its interfaces $\Gamma^{(i,j)}$ as shown in Figure 4. There is a full CV for an interior node and only a half CV for a boundary node. The CV equation of the governing equation takes the form

$$\int_{\Gamma^{(i,j)}} \nabla u \cdot \mathbf{n} \, d\Gamma = 0, \tag{18}$$

which involves first derivatives of $u$ only. Flux integrals over line segments of $\Gamma^{(i,j)}$ are evaluated using Gaussian quadrature which facilitates a high-order accurate solution. Two different cases of boundary conditions, namely (i) Dirichlet conditions only and (ii) Dirichlet ($x_1 = 0$ and $x_1 = \pi$) and Neumann ($x_2 = 0$ and $x_2 = \pi$) conditions, are considered. The 1D-IRBFN approximations are constructed in the same manner as in the case of Galerkin schemes. Figure 5 shows

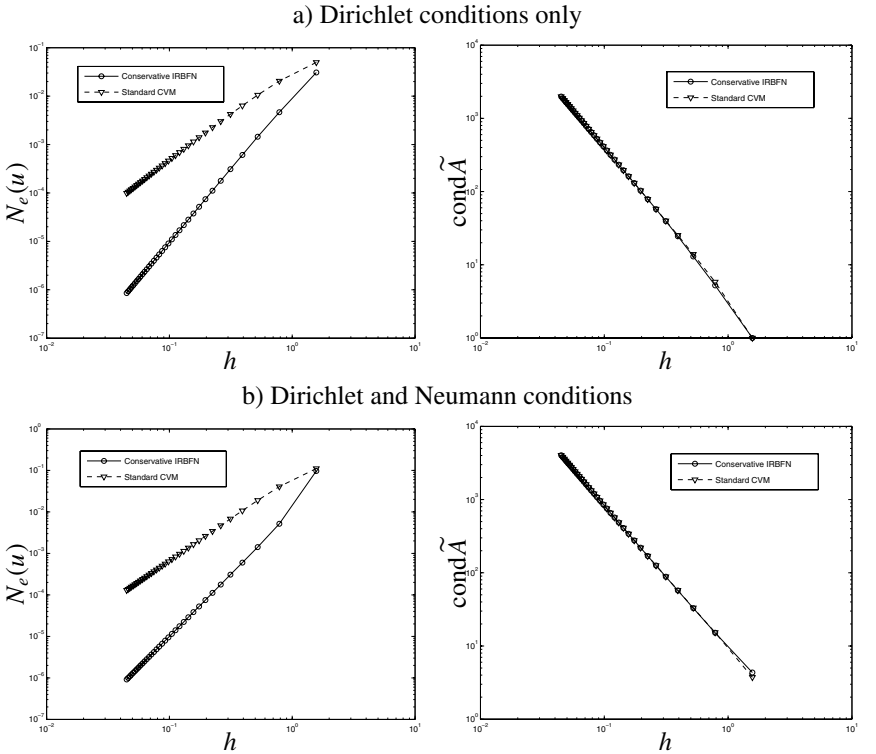


Figure 5: Control-volume scheme, $n = [9, 25, \cdots, 5041]$: Comparisons of the accuracy and condition number between the conservative IRBFN and standard CV methods.

comparisons of the condition number and accuracy of the conservative IRBFN and standard CV methods. For the case of Dirichlet conditions, their rates respectively are $O(h^{3.00})$ and $O(h^{1.84})$ for the accuracy $N_e(u)$, and $O(h^{-2.08})$ and $O(h^{-2.06})$ for the matrix condition number $\widetilde{\text{cond}}A$. For the case of Dirichlet and Neumann conditions, they are $O(h^{3.09})$ and $O(h^{1.98})$ for $N_e(u)$, and $O(h^{-1.94})$ and $O(h^{-1.96})$ for $\widetilde{\text{cond}}A$. Both techniques have similar condition numbers, but the former yields much faster convergence than the latter. Like in the case of Galerkin schemes, conservative IRBFN solutions to Dirichlet and Dirichlet-Neumann problems have similar degrees of accuracy.

## 4 Concluding remarks

In this paper, trial functions are implemented using IRBFNs rather than the usual low-order polynomials for the solution of elliptic DEs. Attractive features of IRBFNs include (i) to result in mesh-free methods for the collocation statement,

(ii) to provide effective treatments of irregular boundary geometries for Cartesian-grid-based methods, (iii) to offer a proper way of implementing derivative boundary conditions, (iv) to avoid the application of the BIE on the boundaries when the domain-type approach is adopted for the inverse statement, and (v) to allow the use of high-order integration schemes to evaluate flux integrals arising from a control volume discretisation. Various examples are presented to demonstrate high-order accurate solutions and accurate implementation of derivative boundary conditions of IRBFNs.

## References

[1] Brebbia, C.A. & Dominguez, J., *Boundary Elements–An Introductory Course*, Computational Mechanics Publications: Southampton, 1992.
[2] Fasshauer G.E., *Meshfree Approximation Methods With Matlab*, World Scientific Publishers: Singapore, 2007.
[3] Mai-Duy, N. & Tran-Cong, T., Approximation of function and its derivatives using radial basis function networks. *Applied Mathematical Modelling*, **27**, pp. 197–220, 2003.
[4] Mai-Duy, N., Indirect RBFN method with scattered points for numerical solution of PDEs. *Computer Modeling in Engineering & Sciences*, **6(2)**, pp. 209–226, 2004.
[5] Mai-Duy, N., Tran-Cong, T. & Tanner, R.I., A domain-type boundary-integral-equation method for two-dimensional biharmonic Dirichlet problem. *Engineering Analysis with Boundary Elements*, **30(10)**, pp. 809–817, 2006.
[6] 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*, **23(5)**, pp. 1192–1210, 2007.
[7] Mai-Duy, N. & Tran-Cong, T., A control volume technique based on integrated RBFNs for the convection-diffusion equation. *Numerical Methods for Partial Differential Equations*(to appear)