A Moving IRBFN-based Integration-Free Meshless Method

Phong B.H. Le[1], Timon Rabczuk[2] Nam Mai-Duy[1] and Thanh Tran-Cong[1]

Abstract: A novel approximation method using integrated radial basis function networks (IRBFN) coupled with moving least square (MLS) approximants, namely moving integrated radial basis function networks (MIRBFN), is proposed in this work. In this method, the computational domain Ω is divided into finite sub-domains Ω_I which satisfy point-wise overlap condition. The local function interpolation is constructed by using IRBFN supported by all nodes in subdomain Ω_I. The global function is then constructed by using Partition of Unity Method (PUM), where MLS functions play the role of partition of unity. As a result, the proposed method is locally supported and yields sparse and banded interpolation matrices. The computational efficiency are excellently improved in comparison with that of the original global IRBFN method. In addition, the present method possesses the Kronecker-δ property, which makes it easy to impose the essential boundary conditions. The proposed method is applicable to randomly distributed datasets and arbitrary domains. In this work, the MIRBFN method is implemented in the collocation of a first-order system formulation [Le, Mai-Duy, Tran-Cong, and Baker (2010)] to solve PDEs governing various problems including heat transfer, elasticity of both compressible and incompressible materials, and linear static crack problems. The numerical results show that the present method offers high order of convergence and accuracy.

Keywords: RBF, Local IRBF, Moving IRBF, meshless, collocation method, elasticity, first order system, locking, crack.

1 Introduction

Meshless methods have been increasingly used since they provide solutions more continuous than the piece-wise continuous ones obtained by the finite element methods (FEM). Several meshless methods have been developed, for example, meshless collocation methods [Atluri, Liu, and Han

[1] CESRC, University of Southern Queensland, Toowoomba, QLD 4350, Australia.
[2] Institute of Structural Mechanics, Bauhaus-University Weimar, Marienstr. 15, 99423 Weimar, Germany.

(2006); Libre, Emdadi, Kansa, Rahimian, and Shekarchi (2008)], global weak form meshless methods [Belytschko, Lu, and Gu (1994)], and local weak form meshless method [Atluri and Shen (2002); Han and Atluri (2003); Li, Shen, Han, and Atluri (2003)]. In recent years, RBF-based meshless methods have received increasing interest from the research community since the associated discretisation of the governing PDEs is very simple for random point distribution and arbitrary domain. Furthermore, global RBFN/IRBFN enjoys spectral accuracy and exponential convergence [Madych (1992); Cheng, Golberg, Kansa, and Zamitto (2003)]. However, the main drawback of the globally supported RBFN/IRBFN is that the resultant interpolation matrix is dense and highly ill-conditioned due to the nature of global approximation. For example, the condition number of such a matrix is about 6×10^{19} with only 20×20 collocation points [Fasshauer (1997)]. Therefore, globally supported RBFN/IRBFN methods are less effective in large-scale computation and in problems concerning with small-scale features such as cracks/strain localization. Attempts to deal with this deficiency include domain decomposition method [Ling and Kansa (2004)], block partitioning and multizone methods [Kansa and Hon (2000)], and preconditioned methods [Baxter (2002); Brown, Ling, Kansa, and Levesley (2005)].

Recently, local RBFN methods have been developed as an alternative approach. Compactly supported RBF truncated from polynomials can improve the condition number, yet a large support is required to obtain a reasonable accuracy [Wendland (1995)]. It is thus considered not a robust method against non-uniform datasets [Tobor, Reuter, and Schlick (2004)]. Moreover, some new local methods that exchange spectral accuracy for a sparse and better-conditioned system, have been proposed, including explicit local RBF [Šaler and Vertnik (2006)], finite difference based local RBF [Wright and Fornberg (2006); Liu, Zhang, Li, Lam, and Kee (2006)], differential quadrature based local RBF [Shu, Ding, and Yeo (2003); Shu and Wu (2007)], and radial point interpolation method [Liu, Liu, and Tai (2005); Liu, Zhang, and Gu (2005)].

Another approach to local RBF is one based on the partition of unity (PU) method. The PU concept was first introduced by Sherpard and known as Sherpard's method. However, Sherpard's method is not widely applied since it is only of constant precision. Since the works of Babuška and Melenk (1997), this method has received more attention and may be considered an underlying concept for many other methods such as, PUFEM [Melenk and Babuška (1996)], XFEM [Moës, Dolbow, and Belytschko (1999); Bordas, Duflot, and Le (2008)], GFEM [Strouboulis, Babuška, and Copps (2000); Strouboulis, Copps, and Babuška (2000)] and certain meshfree methods [Rabczuk and Belytschko (2004); Rabczuk, Areias, and Belytschko (2007)]. For RBF methods, locally supported

RBF based on the PU concept was first introduced in data fitting by Wendland (2002) and has been further expanded by several researchers [Tobor, Reuter, and Schlick (2004, 2006); Ohtake, Belyaev, and Seidel (2006)]. In recent times, the idea of local RBF based on the PU concept was extended by Chen, Hu, and Hu (2008) for solving PDEs. In their method, the reproducing kernel function is employed as PU function to achieve a higher precision than that of Sherpard method.

Motivated by the former works, this paper proposes a new locally supported MIRBFN method, in which the standard globally supported IRBFN is coupled with the moving least square (MLS) approximants via the PU concept to formulate a locally supported MIRBFN interpolation method. Moreover, the present interpolation method is implemented in the collocation of a first-order system formulation, resulting in an integration-free meshless method for solving PDEs. The proposed method is verified by various numerical examples, including heat transfer, elasticity of compressible and incompressible materials, and linear static crack problems. The remaining of this paper is organized as follows. The construction of the present MIRBFN is presented in section 2 followed by the first-order system formulation in section 3. Section 4 reports the numerical experiments and section 5 draws some conclusions.

2 Construction of Moving IRBFN

2.1 The global IRBFN approximation

In the IRBFN method [Mai-Duy and Tran-Cong (2001, 2005); Mai-Duy, Khennane, and Tran-Cong (2007); Le, Mai-Duy, Tran-Cong, and Baker (2007, 2008); Mai-Duy and Tran-Cong (2009)], the formulation of the problem starts with the decomposition of the highest order derivatives under consideration into RBFs. The derivative expressions obtained are then integrated to yield expressions for lower order derivatives and finally for the original function itself. The present work is illustrated with the approximation of a function and its derivatives of order up to 2, the formulation can be thus described as follows.

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

$$u_{,j}(\mathbf{x}) = \int \sum_{i=1}^{m} w^{(i)} g^{(i)}(\mathbf{x}) dx_j + C_1(x_{l;l\neq j}) = \sum_{i=1}^{m+p_1} w^{(i)} H^{(i)}_{[x_j]}(\mathbf{x}), \tag{2}$$

$$u(\mathbf{x}) = \int \sum_{i=1}^{m+p_1} w^{(i)} H^{(i)}(\mathbf{x}) dx_j + C_2(x_{l;l\neq j}) = \sum_{i=1}^{m+p_2} w^{(i)} \overline{H}^{(i)}_{[x_j]}(\mathbf{x}), \tag{3}$$

or in compact form

$$u_{,jj}(\mathbf{x}) = \mathbf{G}(\mathbf{x})\mathbf{w}_{[x_j]}, \tag{4}$$

$$u_{,j}(\mathbf{x}) = \mathbf{H}_{[x_j]}(\mathbf{x})\mathbf{w}_{[x_j]}, \tag{5}$$

$$u(\mathbf{x}) = \overline{\mathbf{H}}_{[x_j]}(\mathbf{x})\mathbf{w}_{[x_j]}, \tag{6}$$

where, the comma denotes partial differentiation, m is the number of RBFs, $\{g^{(i)}(\mathbf{x})\}_{i=1}^{m}$ is the set of RBFs, $\{w^{(i)}\}_{i=1}^{m+p_2}$ is the set of corresponding network weights to be found, $\{H^{(i)}(\mathbf{x})\}_{i=1}^{m}$ and $\{\bar{H}^{(i)}(\mathbf{x})\}_{i=1}^{m}$ are new basis functions obtained by integrating the radial basis function $g^{(i)}(\mathbf{x})$, p_1 and p_2 are the number of centers used to represent integration constants in the first and second derivatives, (2) and (3), respectively ($p_2 = 2p_1$). For the multiquadric function

$$g^{(i)}(\mathbf{x}) = \sqrt{\left\|\mathbf{x} - \mathbf{c}^{(i)}\right\|^2 + \left(a^{(i)}\right)^2}, \tag{7}$$

where $\mathbf{c}^{(i)}$ is the RBF center and $a^{(i)}$ is the RBF width, the width of the i^{th} RBF can be determined according to the following simple relation

$$a^{(i)} = \beta d^{(i)}, \tag{8}$$

where β is a factor, $\beta > 0$, and $d^{(i)}$ is the distance from the i^{th} center to its nearest neighbour.

Now, the "constants" of integration $C_1(x_{l;l\neq j})$ and $C_2(x_{l;l\neq j})$ on the right hand side of (2) and (3) can also be interpolated using the IRBFN method as follows.

$$C_1^{''}(x_l; l \neq j) = \sum_{i=1}^{M} \bar{w}^{(i)} g^{(i)}(x_l; l \neq j), \tag{9}$$

$$C_1^{'}(x_l; l \neq j) = \sum_{i=1}^{M} \bar{w}^{(i)} H^{(i)}(x_l; l \neq j) + \widehat{C}_1, \tag{10}$$

$$C_1(x_l; l \neq j) = \sum_{i=1}^{M} \bar{w}^{(i)} \bar{H}^{(i)}(x_l; l \neq j) + \widehat{C}_1 x_{k;k\neq j} + \widehat{C}_2, \tag{11}$$

where $\{\bar{w}^{(i)}\}_{i=1}^{M}$ are the corresponding weights; M is the number of distinct centers. The unknowns to be found are the sets of weights in (1) and (9), which can be determined by the SVD (singular value decomposition) procedure, for example.

Following Mai-Duy and Tran-Cong (2005), we perform a prior conversion of the unknowns from network weights, i.e. $\{w^{(i)}\}_{i=1}^{m+p_2}$, to nodal function value $\mathbf{u}$ in order to form a square system of equations of smaller size as follows.
The set of network weights are expressed in terms of nodal function value as

$$\mathbf{w}_{[x]} = \left[\overline{\mathbf{H}}_{[x]}\right]^{-1}\mathbf{u}, \tag{12}$$

$$\mathbf{w}_{[y]} = \left[\overline{\mathbf{H}}_{[y]}\right]^{-1}\mathbf{u}, \tag{13}$$

and the substitution of (12) and (13) into the system (4)-(6) yields

$$u_{,xx}(\mathbf{x}) = \mathbf{G}(\mathbf{x})\left[\overline{\mathbf{H}}_{[x]}\right]^{-1}\mathbf{u}, \tag{14}$$

$$u_{,x}(\mathbf{x}) = \mathbf{H}_{[x]}(\mathbf{x})\left[\overline{\mathbf{H}}_{[x]}\right]^{-1}\mathbf{u}, \tag{15}$$

$$u(\mathbf{x}) = \overline{\mathbf{H}}_{[x]}(\mathbf{x})\left[\overline{\mathbf{H}}_{[x]}\right]^{-1}\mathbf{u}, \tag{16}$$

$$u_{,yy}(\mathbf{x}) = \mathbf{G}(\mathbf{x})\left[\overline{\mathbf{H}}_{[y]}\right]^{-1}\mathbf{u}, \tag{17}$$

$$u_{,y}(\mathbf{x}) = \mathbf{H}_{[y]}(\mathbf{x})\left[\overline{\mathbf{H}}_{[y]}\right]^{-1}\mathbf{u}, \tag{18}$$

$$u(\mathbf{x}) = \overline{\mathbf{H}}_{[y]}(\mathbf{x})\left[\overline{\mathbf{H}}_{[y]}\right]^{-1}\mathbf{u}, \tag{19}$$

where $\mathbf{I}$ is the identity matrix. It can be seen from (14)-(19) that the function and its derivatives are all expressed in terms of the function values rather than network weights. Consequently, the system of equations obtained is normally square and the unknowns to be solved for are the nodal function values instead of the network weights.

2.2 Moving least-square approximants

The moving least-square (MLS) procedure presented in Belytschko, Lu, and Gu (1994) is briefly reproduced in this section as follows. The interpolant $u^h(\mathbf{x})$ of the function $u(\mathbf{x})$ is defined in the domain Ω by

$$u^h(\mathbf{x}) = \sum_{j=1}^{\mathscr{M}} a_j(\mathbf{x})p_j(\mathbf{x}) \equiv \mathbf{p}^T(\mathbf{x})\mathbf{a}(\mathbf{x}), \tag{20}$$

where $\mathbf{x}^T = [x,y]$, $\mathbf{p}^T = [1,x,y]$ is a linear basis, $\mathscr{M} = 3$ in $\mathbb{R}^2$.

$\mathbf{a}(\mathbf{x})$ is obtained at any point $\mathbf{x}$ by minimizing the following weighted, discrete L_2 norm

$$J = \sum_{I=1}^{n} w(\mathbf{x}-\mathbf{x}_I)[\mathbf{p}^T(\mathbf{x}_I)\mathbf{a}(\mathbf{x}) - u_I]^2, \tag{21}$$

where n is the number of points in the neighbourhood of $\mathbf{x}$ for which the weight function $w(\mathbf{x}-\mathbf{x}_I) \neq 0$, and u_I is the nodal value of u at $\mathbf{x} = \mathbf{x}_I$.

The minimization of J in (21) with respect to $\mathbf{a}(\mathbf{x})$ leads to the following linear relation between $\mathbf{a}(\mathbf{x})$ and the vector of local nodal values $\mathbf{u}$

$$\mathbf{A}(\mathbf{x})\mathbf{a}(\mathbf{x}) = \mathbf{B}(\mathbf{x})\mathbf{u}, \tag{22}$$

or

$$\mathbf{a}(\mathbf{x}) = \mathbf{A}^{-1}(\mathbf{x})\mathbf{B}(\mathbf{x})\mathbf{u}, \tag{23}$$

where $\mathbf{A}(\mathbf{x})$ and $\mathbf{B}(\mathbf{x})$ are defined by

$$\mathbf{A}(\mathbf{x}) = \sum_{I=1}^{n} w(\mathbf{x}-\mathbf{x}_I)\mathbf{p}(\mathbf{x}_I)\mathbf{p}^T(\mathbf{x}_I) \tag{24}$$

$$\mathbf{B}(\mathbf{x}) = \left[w(\mathbf{x}-\mathbf{x}_1)\begin{pmatrix} 1 \\ x_1 \\ y_1 \end{pmatrix}, w(\mathbf{x}-\mathbf{x}_2)\begin{pmatrix} 1 \\ x_2 \\ y_2 \end{pmatrix}, \ldots, w(\mathbf{x}-\mathbf{x}_n)\begin{pmatrix} 1 \\ x_n \\ y_n \end{pmatrix} \right] \tag{25}$$

$$\mathbf{u}^T = [u_1, u_2, \ldots, u_n]. \tag{26}$$

Substitution of (23) into (20) yields

$$u^h(\mathbf{x}) = \sum_{I=1}^{n} \sum_{j=1}^{\mathscr{M}} p_j(\mathbf{x})(\mathbf{A}^{-1}(\mathbf{x})\mathbf{B}(\mathbf{x}))_{jI} u_I \equiv \sum_{I=1}^{n} \varphi_I(\mathbf{x}) u_I, \tag{27}$$

where the shape function $\varphi_I(\mathbf{x})$ is defined by

$$\varphi_I(\mathbf{x}) = \sum_{j=1}^{\mathscr{M}} p_j(\mathbf{x})(\mathbf{A}^{-1}(\mathbf{x})\mathbf{B}(\mathbf{x}))_{jI}, \tag{28}$$

or in compact form

$$\varphi_I(\mathbf{x}) = \mathbf{c}^T(\mathbf{x}) w(\mathbf{x}-\mathbf{x}_I)\mathbf{p}(\mathbf{x}_I), \tag{29}$$

where $\mathbf{A}(\mathbf{x})\mathbf{c}(\mathbf{x}) = \mathbf{p}(\mathbf{x})$ defines vector $\mathbf{c}(\mathbf{x})$.

$\mathbf{c}(\mathbf{x})$ can efficiently be computed by the LU factorization of $\mathbf{A}(\mathbf{x})$ with backward substitution [Belytschko, Krongauz, Fleming, Organ, and Liu (1996); Nguyen, Rabczuk, Bordas, and Duflot (2008)] as follows.

$$\mathbf{LUc}(\mathbf{x}) = \mathbf{p}(\mathbf{x}), \qquad \mathbf{Uc}(\mathbf{x}) = \mathbf{L}^{-1}\mathbf{p}(\mathbf{x}), \qquad \mathbf{c}(\mathbf{x}) = \mathbf{U}^{-1}\mathbf{L}^{-1}\mathbf{p}(\mathbf{x}). \tag{30}$$

The partial derivatives of $\varphi_I(\mathbf{x})$ can be obtained by

$$\varphi_{I,i}(\mathbf{x}) = \mathbf{c}_{,i}^T(\mathbf{x}) w(\mathbf{x}-\mathbf{x}_I)\mathbf{p}(\mathbf{x_I}) + \mathbf{c}^T(\mathbf{x}) w_{,i}(\mathbf{x}-\mathbf{x}_I)\mathbf{p}(\mathbf{x_I}), \tag{31}$$

where $(.)_{,i} = \frac{\partial(.)}{\partial x_i}$ and

$$\mathbf{c}_{,i}(\mathbf{x}) = \mathbf{A}_{,i}^{-1}(\mathbf{x})\mathbf{p}(\mathbf{x}) + \mathbf{A}^{-1}(\mathbf{x})\mathbf{p}_{,i}(\mathbf{x}), \tag{32}$$

with

$$\mathbf{A}_{,i}(\mathbf{x}) = \sum_{I=1}^{n} w_{,i}(\mathbf{x}-\mathbf{x}_I)\mathbf{p}(\mathbf{x}_I)\mathbf{p}^T(\mathbf{x}_I). \tag{33}$$

It is noted that the following circular kernel function [Schilling, Caroll, and Al-Ajlouni (2001)] is used to compute the present MLS shape function

$$w(r) = \begin{cases} [1+\cos(\pi\frac{r}{R_s})]/2r^{\alpha}, & \frac{r}{R_s} \le 1, \quad \alpha \quad \text{even}, \\ 0, & \frac{r}{R_s} > 1, \end{cases} \tag{34}$$

where R_s is the radius of the support domain of the weight function $w(r)$, $r = \|\mathbf{x}-\mathbf{x}_I\|$ and $\|.\|$ denotes the Euclidean norm.

2.3 Moving IRBFN interpolation

We propose a locally supported IRBFN, constructed by using the partition of unity concept [Melenk and Babuška (1996); Babuška and Melenk (1997)] as follows.

Let the open and bounded domain of interest $\Omega \subseteq \mathbb{R}^d$ be discretised by a set of N points $\mathscr{X}$

$$\mathscr{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}, \qquad \mathbf{x}_I \in \Omega, \qquad I = 1, 2, \ldots, N, \tag{35}$$

$\mathscr{X}$ is used to define an open cover of Ω, i.e. $\{\Omega_I\}$ such that $\Omega \subseteq \bigcup_{I=1}^{N} \Omega_I$ and $\{\Omega_I\}$ satisfies a point-wise overlap condition

$$\forall \mathbf{x} \in \Omega \qquad \exists k \in \mathbb{N} \qquad : \qquad \text{card}\{I | \mathbf{x} \in \Omega_I\} \leq k. \tag{36}$$

We choose a family of compactly supported, non-negative, continuous functions ψ_I supported on the closure of Ω_I, such that at every point $\mathbf{x}$ we have the following property

$$\sum_{I=1}^{N} \psi_I(\mathbf{x}) = 1, \qquad \forall \mathbf{x} \in \Omega, \tag{37}$$

where $\{\psi_I\}$ is called a partition of unity subordinate to the cover $\{\Omega_I\}$.

For every subdomain Ω_I, a local approximation u_I is constructed by using IRBFN supported by all nodes in Ω_I as presented in section 2.1, i.e.

$$u_I^h(\mathbf{x}) \in V_I, \qquad V_I = \text{span}\{\overline{H}_I^{(1)}(\mathbf{x}), \overline{H}_I^{(2)}(\mathbf{x}), \ldots, \overline{H}_I^{(M)}(\mathbf{x})\}, \tag{38}$$

where $\{V_I\}$ are referred to as the local approximation spaces.

The global approximation of $u(\mathbf{x})$, $u^h(\mathbf{x})$ is obtained via

$$u^h(\mathbf{x}) = \sum_{I=1}^{N} \psi_I(\mathbf{x}) u_I^h(\mathbf{x}), \qquad u^h(\mathbf{x}) \in V, \tag{39}$$

where $\psi_I(\mathbf{x})$ and $u_I^h(\mathbf{x})$ are associated with the subdomain Ω_I, and V is called PU method space and defined by

$$V := \sum_{I=1}^{N} \psi_I V_I. \tag{40}$$

In the present work, the partition of unity function ψ_I is chosen to be identical to the MLS shape function φ_I in (27), the subdomain Ω_I is centered at $\mathbf{x}_I$ as shown in Figure 1.

Replacing ψ_I with MLS shape function φ_I, (39) can be rewritten as follows.

$$u^h(\mathbf{x}) = \sum_{I=1}^{N} \varphi_I(\mathbf{x}) u_I^h(\mathbf{x}), \tag{41}$$

and the associated derivatives of $u^h(\mathbf{x})$ are given by

$$u_{,x}^h(\mathbf{x}) = \sum_{I=1}^{N} \left[\varphi_{I,x}(\mathbf{x}) u_I^h(\mathbf{x}) + \varphi_I(\mathbf{x}) u_{I,x}^h(\mathbf{x}) \right], \tag{42}$$

$$u_{,y}^h(\mathbf{x}) = \sum_{I=1}^{N} \left[\varphi_{I,y}(\mathbf{x}) u_I^h(\mathbf{x}) + \varphi_I(\mathbf{x}) u_{I,y}^h(\mathbf{x}) \right], \tag{43}$$

where $u_{I,x}^h(\mathbf{x})$ and $u_{I,y}^h(\mathbf{x})$, are derived in (15) and (18).

$u^h(\mathbf{x})$ and its derivatives can be rewritten in a compact form as

$$u^h(\mathbf{x}) = \sum_{I=1}^{N} \varphi_I(\mathbf{x}) u_I^h(\mathbf{x}) = \Phi^T(\mathbf{x}) \mathbf{u}, \tag{44}$$

$$u_{,x}^h(\mathbf{x}) = \Phi_x^T(\mathbf{x}) \mathbf{u}, \tag{45}$$

$$u_{,y}^h(\mathbf{x}) = \Phi_y^T(\mathbf{x}) \mathbf{u}, \tag{46}$$

where $\mathbf{u} = \{u_1, u_2, \ldots, u_N\}$, $\Phi(\mathbf{x})$ is the vector of shape functions.

It is noted that $\Phi_I(\mathbf{x}_J) = \delta_{IJ}$ as shown in Figures 3. Consequently, this MIRBFN method possesses the Kronecker-δ property which makes it easy to impose the essential boundary conditions. Owing to the locally supported property, MIRBFN yields symmetric, sparse and banded interpolation matrices as shown in Figure 2. This feature makes the method very efficient in storage and computation.

2.4 Selection of RBF centers and support radius

In the present MIRBFN method, the selection of local RBF centers $\{\mathbf{c}_i\}_I$ is very flexible. Generally, they can be different from the set of local data points $\{\mathbf{x}_i\}_I$ associated with subdomain Ω_I. For example, if a two-dimensional IRBFN is used, the size of the matrices to be inverted $\overline{\mathbf{H}}_{[x]}$ and $\overline{\mathbf{H}}_{[y]}$ in (12) and (13), respectively, will be $n_I \times (m_I + p_{2I})$, where n_I is the number of data points,

m_I the number of RBF centers $\{\mathbf{c}_i\}_I$ and p_{2I} the number of centers used to represent integration constants in the second derivatives. Therefore, the number of columns of the matrices will be p_{2I} larger than the number of rows when $\{\mathbf{c}_i\}_I$ is the same as $\{\mathbf{x}_i\}_I$. To obtain square matrices, we choose the number of centers to be less than the number of data points ($m_I < n_I$) and p_{2I} to be appropriately small.

On the other hand, the selection of support radius for each subdomain Ω_I also affects the numerical results significantly. The larger support radius is, the higher accuracy and convergence rate are. However, the higher cost of storage and computation, and the deterioration of the condition number of the matrices are consequential trade-offs. Hence, to make the method more local and efficient, smaller values of support radius are preferred in this work.

3 First-order system formulation

For the sake of completeness, the first-order system formulation, which was proposed in a previous work of the authors [Le, Mai-Duy, Tran-Cong, and Baker (2010)], is reproduced briefly as follows. It is noticed that in general higher-order differential equations can be transformed into a system of first-order differential equations by introducing some new dual variables, which is the procedure followed here. Both primary and dual variables are then independently interpolated and have the shape functions of the same order. The resultant first-order system of governing equations can be written as follows.

$$\mathscr{L}\mathbf{u} = \mathbf{f}, \qquad \text{in} \quad \Omega \tag{47}$$

$$\mathbf{B}\mathbf{u} = \mathbf{g}, \qquad \text{on} \quad \Gamma \tag{48}$$

where Ω is a bounded domain in $\mathbb{R}^d$, $d = 1,2,3$, Γ the boundary of Ω, $\mathscr{L}$ is a first-order linear differential operator

$$\mathscr{L}\mathbf{u} = \mathscr{L}_0\mathbf{u} + \sum_{i=1}^{d} \mathscr{L}_i \frac{\partial \mathbf{u}}{\partial x_i}, \tag{49}$$

in which $\mathbf{u}^T = [u_1, u_2, ..., u_m]$ is a vector of m unknown functions (including primary and dual variables) of $\mathbf{x}^T = [x_1, x_2, ..., x_d]$, $\mathscr{L}_i$ the coefficient matrices which characterize the differential operator $\mathscr{L}$, $\mathbf{f}$ a given function in the domain, $\mathbf{B}$ a boundary algebraic operator, and $\mathbf{g}$ a given function on the boundary.

Substituting a discrete approximation of $\mathbf{u}$ and its first-order derivatives as given, respectively, in (41) and (42)-(43) into (47) and (48), and using the collocation method at all the nodes of Ω and Γ, one obtains a linear algebraic system as presented below.

Let N_Ω denote the number of interior nodes, N_D the number of nodes on the Dirichlet boundary, N_N the number of nodes on the Neumann boundary, m_p the number of primary unknowns and m_d the number of dual unknowns associated with a node, the number of nodal unknowns is generally $(N_\Omega + N_D + N_N)(m_p + m_d)$. The governing equation (47) is collocated at all the interior and boundary nodes, yielding $(N_\Omega + N_D + N_N)(m_p + m_d)$ equations. The boundary conditions are imposed by collocating (48) at all the boundary nodes, i.e. the obtained system has $(N_\Omega + N_D + N_N)(m_p + m_d) + N_D k_D + N_N k_N$ equations, where k_D and k_N are the number of equations from the boundary conditions per node on the Dirichlet and Neumann boundaries, respectively. The final system is obtained by removing $N_D k_D + N_N k_N$ appropriate equations corresponding to the governing equations collocated at the boundary nodes. Consequently, the number of equations of the resultant system is equal to the number of nodal unknowns and it can be rewritten in a compact form as

$$\mathbf{A}\mathbf{u} = \bar{\mathbf{f}}. \tag{50}$$

3.1 Two-dimensional Poisson equation

Consider the following two-dimensional Poisson equation

$$\frac{\partial^2 \phi(x,y)}{\partial x^2} + \frac{\partial^2 \phi(x,y)}{\partial y^2} = f(x,y) \qquad \text{in} \quad \Omega, \tag{51a}$$

$$\phi(x,y) = g(x,y) \qquad \text{on} \quad \Gamma_D, \tag{51b}$$

$$\frac{\partial \phi(x,y)}{\partial n} = h(x,y) \qquad \text{on} \quad \Gamma_N, \tag{51c}$$

where Ω is a bounded domain in $\mathbb{R}^2$, Γ_D and Γ_N the boundary of Ω on which the Dirichlet and Neumann boundary conditions are imposed, respectively, $\mathbf{n} = (n_x, n_y)^T$ the outward unit normal to Γ_N, and f, g and h given functions on Ω, Γ_D and Γ_N, respectively.

A first-order formulation is obtained by introducing the dual variables in (51) as follows.

$$\frac{\partial \phi(x,y)}{\partial x} - \xi(x,y) = 0 \qquad \text{in} \quad \Omega \quad \text{and on} \quad \Gamma_D \bigcup \Gamma_N, \tag{52a}$$

$$\frac{\partial \phi(x,y)}{\partial y} - \eta(x,y) = 0 \qquad \text{in} \quad \Omega \quad \text{and on} \quad \Gamma_D \bigcup \Gamma_N, \tag{52b}$$

$$\frac{\partial \xi(x,y)}{\partial x} + \frac{\partial \eta(x,y)}{\partial y} = f(x,y) \qquad \text{in} \quad \Omega \quad \text{and on} \quad \Gamma_D \bigcup \Gamma_N, \tag{52c}$$

$$\phi(x,y) = g(x,y) \qquad \text{on} \quad \Gamma_D, \tag{52d}$$

$$n_x \xi + n_y \eta = h(x,y) \qquad \text{on} \quad \Gamma_N. \tag{52e}$$

3.2 Linear elasticity problems

Consider the following two-dimensional problem on a domain Ω bounded by $\Gamma = \Gamma_u \bigcup \Gamma_t$

$$\nabla \cdot \sigma = \mathbf{b} \qquad \text{in} \quad \Omega, \tag{53a}$$

$$\mathbf{u} = \bar{\mathbf{u}} \qquad \text{on} \quad \Gamma_u, \tag{53b}$$

$$\sigma \cdot \mathbf{n} = \bar{\mathbf{t}} \qquad \text{on} \quad \Gamma_t, \tag{53c}$$

in which σ is the stress tensor, which corresponds to the displacement field $\mathbf{u}$ and $\mathbf{b}$ is the body force, $\mathbf{n}$ the outward unit normal to Γ_t. The superposed bar denotes prescribed value on the boundary.

The governing equations (53) are closed when a constitutive relation is specified for σ. Here the linear Hooke's law is used to describe the $\sigma - \mathbf{u}$ relation. By choosing displacement $\mathbf{u}$ as primary variable and stress σ as dual variable, the governing equations remain first-order, which are written

for plane stress case as follows.

$$\frac{\partial u}{\partial x} - \frac{1}{E}\sigma_x + \frac{\mu}{E}\sigma_y = 0, \tag{54a}$$

$$\frac{\partial v}{\partial y} + \frac{\mu}{E}\sigma_x - \frac{1}{E}\sigma_y = 0, \tag{54b}$$

$$\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} - \frac{2(1+\mu)}{E}\tau_{xy} = 0, \tag{54c}$$

$$\frac{\partial \sigma_x}{\partial x} + \frac{\partial \tau_{xy}}{\partial y} = b_x, \tag{54d}$$

$$\frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \sigma_y}{\partial y} = b_y, \tag{54e}$$

$$\mathbf{u} = \bar{\mathbf{u}} \qquad \text{on} \quad \Gamma_u, \tag{54f}$$

$$\sigma \cdot \mathbf{n} = \bar{\mathbf{t}} \qquad \text{on} \quad \Gamma_t, \tag{54g}$$

where μ is the Poisson ratio and E the Young's modulus. By introducing the dimensionless stress tensor $\mathbf{s} = \sigma/E$, the above first-order system can be rewritten as follows.

$$\frac{\partial u}{\partial x} - s_x + \mu s_y = 0, \tag{55a}$$

$$\frac{\partial v}{\partial y} + \mu s_x - s_y = 0, \tag{55b}$$

$$\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} - 2(1+\mu)s_{xy} = 0, \tag{55c}$$

$$\frac{\partial s_x}{\partial x} + \frac{\partial s_{xy}}{\partial y} = b_x, \tag{55d}$$

$$\frac{\partial s_{xy}}{\partial x} + \frac{\partial s_y}{\partial y} = b_y, \tag{55e}$$

$$\mathbf{u} = \bar{\mathbf{u}} \qquad \text{on} \quad \Gamma_u, \tag{55f}$$

$$\mathbf{s} \cdot \mathbf{n} = \bar{\mathbf{t}} \qquad \text{on} \quad \Gamma_t. \tag{55g}$$

4 Numerical examples

For an error estimation and convergence study, the discrete relative L_2 norm of errors of primary and dual variables are defined as

$$L_2^{\phi} = \frac{\sqrt{\sum_{i=1}^{N}\left(\phi_e^{(i)} - \phi^{(i)}\right)^2}}{\sqrt{\sum_{i=1}^{N}\left(\phi_e^{(i)}\right)^2}}, \tag{56}$$

$$L_2^{\xi\eta} = \frac{\sqrt{\sum_{i=1}^{N}\left[\left(\xi_e^{(i)} - \xi^{(i)}\right)^2 + \left(\eta_e^{(i)} - \eta^{(i)}\right)^2\right]}}{\sqrt{\sum_{i=1}^{N}\left[\left(\xi_e^{(i)}\right)^2 + \left(\eta_e^{(i)}\right)^2\right]}}, \tag{57}$$

for Poisson equation and

$$L_2^{u} = \frac{\sqrt{\sum_{i=1}^{N}\left((u_x)_e^{(i)} - (u_x)^{(i)}\right)^2\left((u_y)_e^{(i)} - (u_y)^{(i)}\right)^2}}{\sqrt{\sum_{i=1}^{N}\left[\left((u_x)_e^{(i)}\right)^2 + \left((u_y)_e^{(i)}\right)^2\right]}}, \tag{58}$$

$$L_2^{\sigma} = \frac{\sqrt{\sum_{i=1}^{N}\left[\left((s_x)_e^{(i)} - s_x^{(i)}\right)^2 + \left((s_y)_e^{(i)} - s_y^{(i)}\right)^2 + \left((s_{xy})_e^{(i)} - s_{xy}^{(i)}\right)^2\right]}}{\sqrt{\sum_{i=1}^{N}\left[\left((s_x)_e^{(i)}\right)^2 + \left((s_y)_e^{(i)}\right)^2 + \left((s_{xy})_e^{(i)}\right)^2\right]}}, \tag{59}$$

for elasticity problems, where N is the number of unknown nodal values and the subscript “e" denotes the exact solution. The convergence order of the solution with respect to the refinement of spatial discretization is assumed to be in the form of

$$L_2(h) \approx \zeta h^{\lambda} = O(h^{\lambda}), \tag{60}$$

where h is the maximum nodal spacing, ζ and λ are the parameters of the exponential model, which are found by general linear least square formula in this work.

It is noted that the CPU time in the following sections is associated with a computer which has 8.0

GB of RAM and two Intel(R) Xeon(R) CPUs of 3.0 GHz each. The code is written in MATLAB® language.

4.1 Poisson equation

4.1.1 Poisson equation in a regular domain

Consider the following Poisson equation

$$\frac{\partial^2 \phi(x,y)}{\partial x^2} + \frac{\partial^2 \phi(x,y)}{\partial y^2} = -2\pi^2 \cos(\pi x)\cos(\pi y), \tag{61}$$

defined in $\Omega = [0,1] \times [0,1]$, subjected to the Dirichlet boundary condition

$$\phi(0,y) = \cos(\pi y), \qquad \text{on} \quad x = 0, \tag{62}$$

and the following Neumann boundary conditions

$$\frac{\partial \phi(1,y)}{\partial x} = 0, \qquad \text{on} \quad x = 1, \tag{63a}$$

$$\frac{\partial \phi(x,0)}{\partial y} = 0, \qquad \text{on} \quad y = 0, \tag{63b}$$

$$\frac{\partial \phi(x,1)}{\partial y} = 0, \qquad \text{on} \quad y = 1. \tag{63c}$$

$$\tag{63d}$$

The corresponding exact solution is given by

$$\phi(x,y) = \cos(\pi x)\cos(\pi y). \tag{64}$$

Two discretisations are considered for this problem: uniform and nonuniform distributions of nodes/collocation points (CPs) as shown in Figures 4 and 8, respectively. For both cases, the radius of support domains is set at $\frac{R_s}{h} = 2.1$, where h is the maximum spacing between two nearest nodes in x or y direction. The maximum number of uniformly distributed RBF centers m_I in each subdomain is 5 as shown in Figure 4. The numbers of centers to represent the integration constants p_{1I} and p_{2I} are 3 and 6, respectively. The values of β in (8) for both cases are listed in Tables 1 and 3.

Table 1: Poisson equation in a regular domain: uniform discretisations with MIRBFN

No. points	L_2^{ϕ}	$L_2^{\xi\eta}$	cond(A)	β	$\frac{R_s}{h}$	$\frac{CPU\ time}{second}$
3×3	0.4415	1.1413	51.7250	12	2.1	0.15
7×7	0.0252	0.0219	512.8116	12	2.1	0.30
11×11	0.0036	0.0041	813.8110	12	2.1	0.59
21×21	4.4864e-4	5.5402e-4	2.2514e3	12	2.1	2.07
25×25	2.5203e-4	3.1671e-4	2.9034e3	12	2.1	3.07
31×31	1.2419e-4	1.5922e-4	4.1964e3	12	2.1	5.13
41×41	5.0132e-5	6.5620e-5	6.1935e3	12	2.1	10.59
61×61	1.4217e-5	1.9006e-5	1.5362e4	12	2.1	35.84
81×81	5.9951e-6	8.0377e-6	3.5862e4	12	2.1	90.0
91×91	4.2892e-6	5.6966e-6	5.2312e4	12	2.1	136.40
101×101	3.2363e-6	4.2199e-6	7.4923e4	12	2.1	197.11
121×121	2.1324e-6	2.6352e-6	9.037e4	12	2.1	374.7
	$O(h^{3.32})$	$O(h^{3.38})$				

Table 2: Poisson equation in a regular domain: uniform discretisations with global IRBFN

No. points	L_2^{ϕ}	$L_2^{\xi\eta}$	cond(A)	β	$\frac{CPU\ time}{second}$
7×7	0.0245	0.0273	1.6043e4	1	0.161
11 × 11	0.0038	0.0048	2.5617e4	1	0.179
21 × 21	7.4562e-5	1.5070e-4	5.8907e4	1	2.462
31 × 31	1.2924e-5	2.3775e-5	1.2225e5	1	30.064
41× 41	4.3906e-6	7.9095e-6	2.0292e5	1	149.319
51 × 51	2.1210e-6	3.7691e-6	3.0404e5	1	535.049
61 × 61	1.3851e-6	2.1592e-6	7.0649e4	1	1674.980
	$O(h^{4.71})$	$O(h^{4.52})$			

Table 3: Poisson equation in a regular domain: unstructured nodes with MIRBFN

No. points	L_2^{ϕ}	$L_2^{\xi\eta}$	cond(A)	β	$\frac{R_s}{h}$	h	$\frac{CPU\ time}{second}$
88	0.2833	0.1438	1.6887e5	10	2.1	0.1250	0.73
108	0.0402	0.0613	4.5345e5	10	2.1	0.1200	0.80
327	0.0077	0.0057	6.2091e7	10	2.1	0.0685	2.23
691	0.0018	0.0019	4.5704e8	10	2.1	0.0507	5.65
1723	7.2107e-4	5.7631e-4	1.3461e8	10	2.1	0.0308	22.12
2248	3.3681e-4	2.5718e-4	1.2765e8	12	2.1	0.0272	35.58
	$O(h^{3.78})$	$O(h^{3.82})$					

The influences of local support radius $\frac{R_s}{h}$ and β on the accuracy of the solution are numerically studied in this example. Figure 5 shows the relative error norms (L_2^{ϕ} and $L_2^{\xi\eta}$) obtained by the present MIRBFN method with different values of $\frac{R_s}{h}$ while β is fixed. On the other hand, the results with different values of β and fixed local support radius are displayed in Figure 6. It can be seen that the values around 2 for $\frac{R_s}{h}$ are not only able to capture well the solution but also keep the matrix small, as long as β is large enough.

To study the convergence of the method, a number of discretization refinements and the relative L_2 error norms for function values L_2^{ϕ} and its derivatives $L_2^{\xi\eta}$ are reported in Tables 1 and 3 for uniform and unstructured cases, respectively. As shown in these tables and Figures 7 and 9, very good accuracy and stability are obtained. The convergence rates for $\phi(x,y)$ and $(\xi(x,y),\eta(x,y))$ are $O(h^{3.32})$ and $O(h^{3.38})$, respectively, for uniform distribution, and, $O(h^{3.78})$ and $O(h^{3.82})$, respectively, for unstructured nodes. It can be seen that the condition numbers in the case of uniform distribution are relatively smaller than those in the case of random distribution (Table 3) since there is a relatively larger number of nodes in each subdomain in the case of random distribution.

The results in Tables 1-2 and Figure 7 indicate that the global IRBFN gives higher orders of convergence. Nonetheless, the condition numbers by the MIRBFN method are slightly better in comparison with those by the global IRBFN method, as listed in Tables 1 and 2, although β is set quite large for the MIRBFN method. Furthermore, the MIRBFN method is much more efficient than the global IRBFN method as can be seen in Figure 10.

4.1.2 Poisson equation in an irregular domain

The Poisson equation in example 4.1.1 is examined in a complicated irregular domain as shown in Figure 11. The Dirichlet boundary conditions on the upper edge and the left edge are given as below

$$\phi(0,y) = \cos(\pi y), \qquad \text{on} \quad x = 0, \tag{65a}$$

$$\phi(x,0) = \cos(\pi x), \qquad \text{on} \quad y = 0. \tag{65b}$$

Table 4: Poisson equation in an irregular domain: structured dicerizations with MIRBFN

No. points	L_2^{ϕ}	$L_2^{\xi\eta}$	cond(A)	β	$\frac{R_s}{h}$	h	$\frac{CPU\ time}{second}$
51	3.4762e-1	5.4441e-1	3.1001e5	9	4.1	0.25	0.83
87	2.8487e-2	5.2716e-2	1.9944e6	9	4.1	0.181	1.31
266	1.4620e-3	3.8399e-3	2.0634e7	9	4.1	0.0095	3.86
595	5.0421e-4	8.7207e-4	6.4519e8	9	4.1	0.0065	11.55
1029	1.7279e-4	4.0659e-4	6.2724e8	9	4.1	0.0048	23.05
1574	8.5957e-5	2.4792e-4	4.1291e8	9	4.1	0.039	43.17
2266	3.6035e-5	8.0371e-5	6.5102e9	9	4.1	0.039	84.48
3413	3.0210e-5	5.0281e-5	1.9016e8	9	4.1	0.033	172.53
	$O(h^{4.06})$	$O(h^{3.96})$					

Table 5: Poisson equation in an irregular domain: unstructured discretisation with MIRBFN

No. points	L_2^{ϕ}	$L_2^{\xi\eta}$	cond(A)	β	$\frac{R_s}{h}$	h	$\frac{CPU\ time}{second}$
51	1.9465e-1	1.9142e-1	7.5387e4	14	3.1	2.7337e-1	4.775
338	2.4059e-3	6.7564e-3	4.4212e6	14	3.1	1.1182e-1	22.017
1046	7.1240e-4	1.7302e-3	9.0038e6	12	3.1	5.9731e-2	89.633
1486	4.2708e-4	8.4299e-4	7.5913e7	12	3.1	5.3098e-2	203.883
1711	1.4251e-4	2.1264e-4	1.4224e8	8	3.1	4.8722e-2	
	$O(h^{3.80})$	$O(h^{3.50})$					

The Neumann boundary conditions on the inner arc and the outer arc are, respectively

$$n_x\frac{\partial\phi(x,y)}{\partial x}+n_y\frac{\partial\phi(x,y)}{\partial y}=q(x,y), \qquad \text{on} \quad x^2+y^2=1, \tag{66a}$$

$$n_x\frac{\partial\phi(x,y)}{\partial x}+n_y\frac{\partial\phi(x,y)}{\partial y}=q(x,y), \qquad \text{on} \quad x^2+y^2=4, \tag{66b}$$

where $q(x,y)=-n_x\pi\sin(\pi x)\cos(\pi y)-n_y\pi\cos(\pi x)\sin(\pi y)$.

The complexity is increased with the Neumann boundary conditions on two curved boundaries. The structured domain discretisation is described as follows. A uniformed grid covering the domain is generated, then the points outside the domain and on the curves are removed. Finally, the points on the inner and outer arcs are generated uniformly.

In the case of structured discretisation (Figure 11), the local support radius $\frac{R_s}{h}$ is set at 4.1, β is 9, the maximum number of centers in each subdomain is 13. The relative L_2 error norms L_2^{ϕ} and $L_2^{\xi\eta}$ associated with the structured node discretizations are presented in Table 4 and in Figure 13. It

can be observed that high orders of convergence are obtained with a large support radius, namely $O(h^{4.06})$ and $O(h^{3.96})$ for the function and its derivatives, respectively. However, the condition numbers are much larger than those in the previous example. For unstructured node distributions (Figure 12), the corresponding parameters and obtained results are presented in Table 5 and Figure 14. The results indicate that the solution by the proposed method apparently converges at the rates of $(O^{3.80})$ and $(O^{3.50})$, using L_2^{ϕ} and $L_2^{\xi\eta}$, respectively.

4.2 Linear elasticity problems

4.2.1 Cantilever Beam

A cantilever beam subjected to a parabolic shear load at the end $x = 0$ as shown in Figure 15 is considered in this example.

Table 6: Cantilever beam: uniform discretizations with MIRBFN ($\mu = 0.3$).

No. points	$L_2^{\mathbf{u}}$	L_2^{σ}	cond(A)	β	$\frac{R_s}{h}$	h	$\frac{CPU\ time}{second}$
20×5	1.9598e-1	3.3652e-1	4.1516e6	8	2.1	0.240	0.60
36×9	1.4986e-2	2.5489e-2	1.4193e8	10	2.1	0.133	1.84
68×17	1.2182e-3	2.1326e-3	3.0383e6	14	2.1	0.070	6.98
124×31	5.8434e-4	5.7764e-4	4.0336e6	14	2.1	0.039	43.78
164×41	2.2892e-4	2.3983e-4	8.3453e6	14	2.1	0.029	109.42
204×51	1.1069e-4	1.2366e-4		14	2.1	0.024	230.01
244×61	5.9462e-5	7.2455e-5		14	2.1	0.020	438.98
	$O(h^{3.04})$	$O(h^{3.26})$					

The following parameters are used for the problem: $L = 4.8$ and $D = 1.2$. The beam has a unit thickness. Young's modulus is $E = 3\times 10^6$, Poisson's ratio $\mu = 0.3$ (also $\mu = 0.5$) and the integrated parabolic shear force $P = 100$. Plane stress condition is assumed and there is no body force.

The exact solution to this problem was given by Timoshenko and Goodier (1970) as

$$\sigma_{xx}(x,y) = \frac{-Pxy}{I}, \tag{67a}$$

$$\sigma_{yy}(x,y) = 0, \tag{67b}$$

$$\tau_{xy}(x,y) = \frac{-P}{2I}\left(\frac{D^2}{4} - y^2\right). \tag{67c}$$

Table 7: Cantilever beam: uniform discretizations with MIRBFN ($\mu = 0.5$).

No. points	$L_2^{\mathbf{u}}$	L_2^{σ}	cond(A)	β	$\frac{R_s}{h}$	h	$\frac{CPU\ time}{second}$
20×5	1.0069e-1	1.9291e-1	2.3672e7	8	2.1	0.240	0.60
36×9	2.0936e-2	5.3772e-2	4.2607e7	10	2.1	0.133	1.81
68×17	7.8576e-4	1.6020e-3	2.1090e6	14	2.1	0.070	6.77
124×31	4.3029e-4	4.1872e-4	2.8678e6	14	2.1	0.039	41.12
164×41	1.6292e-4	1.6988e-4	5.7418e6	14	2.1	0.029	106.48
204×51	7.7595e-5	8.7489e-5		14	2.1	0.024	235.5
244×61	4.1951e-5	5.2041e-5		14	2.1	0.020	475.7
	$O(h^{3.07})$	$O(h^{3.39})$					

Table 8: Cantilever beam: uniform discretizations with global IRBFN ($\mu = 0.3$).

No. points	$L_2^{\mathbf{u}}$	L_2^{σ}	cond(A)	β	h	$\frac{CPU\ time}{second}$
20×5	4.5356e-2	2.5571e-1	1.7953e6	1	0.2400	0.408
36×9	5.2822e-3	4.0279e-2	5.2505e6	1	0.1333	2.068
68×17	1.5706e-3	2.6022e-3	6.5476e7	1	0.0706	68.088
124×31	3.8901e-4	4.3698e-4	3.1351e8	1	0.0387	2351.78
164×41	2.1295e-4	2.2075e-4		1	0.0293	51201.338
	$O(h^{3.06})$	$O(h^{3.39})$				

Table 9: Cantilever beam: unstructured nodes with MIRBFN ($\mu = 0.3$).

No. points	$L_2^{\mathbf{u}}$	L_2^{σ}	cond(A)	β	$\frac{R_s}{h}$	h	$\frac{CPU\ time}{second}$
43	6.5385e-1	6.9895e-1	2.6549e6	10	2.1	4.6860e-1	0.715
170	2.7461e-2	5.5154e-2	3.9549e7	10	2.1	2.4000e-1	2.079
616	7.2999e-3	3.1141e-2	7.4558e7	10	2.1	1.2507e-1	7.888
1112	4.9025e-4	3.0318e-3	1.0345e9	10	2.1	1.0454e-1	20.190
	$O(h^{4.21})$	$O(h^{3.07})$					

Table 10: Cantilever beam: structured FEM mesh with four-node quadrilateral element (Q4) ($\mu = 0.3$).

No. elements	$L_2^{\mathbf{u}}$	h	$\frac{CPU\ time}{second}$
16×4	1.3991e-1	0.40	0.1806
32×8	3.8516e-2	0.1714	0.4395
40×10	2.5191e-2	0.1333	1.7111
80×20	6.9048e-3	0.0631	8.4087
160×40	1.6994e-3	0.0307	21.5620
240×60	9.1261e-4	0.0203	47.9957
320×80	6.1308e-4	0.0152	307.579
	$O(h^{1.84})$		

The displacements are given by

$$u_x = -\frac{Px^2y}{2EI} - \frac{\mu Py^3}{6EI} + \frac{Py^3}{6IG} + y\left(\frac{PL^2}{2EI} - \frac{PD^2}{8IG}\right), \tag{68}$$

$$u_y = \frac{\mu Pxy^2}{2EI} + \frac{Px^3}{6EI} - \frac{PL^2x}{2EI} + \frac{PL^3}{3EI}, \tag{69}$$

where $I = D^3/12$ is the moment of inertia of the cross section of the beam, $G = E/(2(1+\mu))$ the modulus of elasticity in shear. The exact displacement (68) and (69) are imposed on $x = L$ while the shear load is applied on $x = 0$ and the upper and lower edges are traction free.

Both regular and irregular distributions of nodes used for this problem are displayed in Figures 16 and 18, respectively. The local support radius is $\frac{R_x}{h} = 2.1$. The values of β are listed in Tables 6, 7 and 9. The scheme for selection of RBF centers for both regular and irregular node distributions is similar to that in example 4.1.1. In addition, the effect of incompressibility, i.e. $\mu = 0.5$, is also studied here.

Figure 19 shows the shear stress s_{xy} for $\mu = 0.3$ at $x = 2.4686$ obtained by the present method with 36×9 nodes. A very good agreement between the obtained result and the exact solution can be observed in this figure.

To study the convergence of the method, a number of different uniform node distributions is used for computation as presented in the Tables 6 and 7. For $\mu = 0.3$, the relative L_2 error norms for displacement and stress are shown in Table 6 and Figure 20, the convergence rates of displacement and stress are $O(h^{3.04})$ and $O(h^{3.26})$, respectively. In the case of incompressible materials ($\mu = 0.5$), the relative L_2 error norms for displacement and stress are presented in Table 7 and Figure 20. Very good orders of convergence are achieved, namely $O(h^{3.07})$ and $O(h^{3.39})$ for displacement and stress, respectively. Furthermore, the results shown in Figure 20 indicate that the present method does not suffer from any volumetric locking.

The behaviour of the MIRBFN method in the case of irregular discretisation is also examined with four nodal configurations as shown in Figure 18. The obtained results with the MIRBFN method and $\mu = 0.3$ are shown in Table 9 and Figure 21. The orders of convergence of the present method are $O(h^{4.21})$ and $O(h^{3.07})$ for displacement and stress, respectively.

In comparison with the global IRBFN method, the MIRBFN method achieves similar accuracy and convergence rates as can be observed in Tables 6 and 8, and in Figure 20 as well. The present method is apparently much more efficient than the global IRBFN method (Figure 22).

The obtained results are also compared with those by FEM using four-node quadrilateral element (Table 10). Figure 20 shows that both accuracy and order of convergence of the MIRBFN method are superior to those of FEM, e.g. using L_2^u, the convergence rates are $O(h^{3.04})$ and $O(h^{1.84})$ for the MIRBFN method and the FEM, respectively. The computing cost of the MIRBFN method is higher than that of the FEM for the same number of nodes. However, the MIRBFN method is more efficient than the FEM for the same accuracy, for example, it takes the MIRBFN method 6.98 seconds for $L_2^u = 1.2182 \times 10^{-3}$ while the FEM needs 21.56 seconds to achieve $L_2^u = 1.6994 \times 10^{-3}$ as exhibited in Figure 22, Table 6 and Table 10.

4.2.2 Infinite plate with a circular hole

In this example, an infinite plate with a circular hole subjected to unidirectional tensile load of 1.0 in the x direction is analyzed as shown in Figure 23. The radius of hole is taken as 1 unit. Owing to symmetry, only the upper right quadrant $[0,3] \times [0,3]$ of the plate is modeled as shown in Figure 24.

In this problem, plane stress conditions are assumed with elastic isotropic properties $E = 10^3$, $\mu = 0.3$ (also $\mu = 0.5$). The exact solution to this problem was given by Timoshenko and Goodier (1970) as follows

$$\sigma_x(x,y) = \sigma\left[1 - \frac{a^2}{r^2}\left[\frac{3}{2}\cos(2\theta) + \cos(4\theta)\right] + \frac{3a^4}{2r^4}\cos(4\theta)\right], \tag{70a}$$

$$\sigma_y(x,y) = -\sigma\left[\frac{a^2}{r^2}\left[\frac{1}{2}\cos(2\theta) - \cos(4\theta)\right] + \frac{3a^4}{2r^4}\cos(4\theta)\right], \tag{70b}$$

$$\tau_{xy}(x,y) = -\sigma\left[\frac{a^2}{r^2}\left[\frac{1}{2}\sin(2\theta) + \sin(4\theta)\right] - \frac{3a^4}{2r^4}\sin(4\theta)\right], \tag{70c}$$

where (r,θ) are the polar coordinates, a the radius of the hole.

The corresponding displacements are given by

$$u_x(x,y) = \sigma\frac{(1+\mu)}{E}\left[\frac{1}{1+\mu}r\cos(\theta) + \frac{2}{1+\mu}\frac{a^2}{r}\cos(\theta) + \frac{1}{2}\frac{a^2}{r}\cos(3\theta) - \frac{1}{2}\frac{a^4}{r^3}\cos(3\theta)\right] \tag{71a}$$

$$u_y(x,y) = \sigma\frac{(1+\mu)}{E}\left[\frac{-\mu}{1+\mu}r\sin(\theta) + \frac{1-\mu}{1+\mu}\frac{a^2}{r}\sin(\theta) + \frac{1}{2}\frac{a^2}{r}\sin(3\theta) - \frac{1}{2}\frac{a^4}{r^3}\sin(3\theta)\right] \tag{71b}$$

The boundary conditions of the problem are as follows. The traction boundary conditions cor-

Table 11: Infinite plate with a circular hole: structured discretisation with MIRBFN ($\mu = 0.3$).

No. points	$L_2^{\mathbf{u}}$	L_2^{σ}	cond(A)	β	$\frac{R_s}{h}$	h	$\frac{CPU\ time}{second}$
50	3.0520e-1	2.6147e-1	6.7532e4	4	2.1	0.50	0.54
119	9.2110e-2	8.1240e-2	8.3533e6	4	2.1	0.30	1.03
409	1.0837e-2	1.2229e-2	6.1059e4	4	2.1	0.15	3.25
1129	8.7872e-4	2.6677e-3	2.0085e5	4	2.1	0.088	10.56
3085	1.8647e-4	4.2703e-4	4.4334e5	4	2.1	0.052	44.36
	$O(h^{3.61})$	$O(h^{3.02})$					

Table 12: Infinite plate with a circular hole: structured discretisation with MIRBFN ($\mu = 0.5$).

No. points	$L_2^{\mathbf{u}}$	L_2^{σ}	cond(A)	β	$\frac{R_s}{h}$	h	$\frac{CPU\ time}{second}$
50	7.8208e-1	5.7769e-1	1.1433e5	4	2.1	0.50	0.54
119	1.0186e-1	8.2598e-2	5.0328e6	4	2.1	0.30	1.01
409	1.3343e-2	1.4314e-2	5.5146e4	4	2.1	0.15	3.20
1129	9.5928e-4	2.7873e-3	2.0372e5	4	2.1	0.088	10.48
3085	4.0203e-4	4.6366e-4	6.0161e5	4	2.1	0.052	43.06
	$O(h^{3.68})$	$O(h^{3.27})$					

Table 13: Infinite plate with a circular hole: structured discretisation with global IRBFN ($\mu = 0.3$).

No. points	$L_2^{\mathbf{u}}$	L_2^{σ}	cond(A)	β	h	$\frac{CPU\ time}{second}$
119	1.3243e-1	1.1085e-1	7.0056e5	1	0.2727	0.413
409	2.3900e-2	1.5925e-2	1.4568e6	1	0.1429	2.222
886	4.8966e-3	3.3027e-3	3.3118e6	1	0.0968	21.323
3085	2.5075e-4	7.2314e-4	4.3415e6	1	0.0517	977.988
	$O(h^{3.78})$	$O(h^{3.08})$				

responding to the exact solution for the infinite plate are applied on the top and right edges, the symmetric conditions are applied on the left and bottom edges, and the curved edge is traction free.

To solve the problem, the computational domain is discretized in the same manner as in example 4.1.2. The support radius is $\frac{R_s}{h} = 2.1$, the value of β varies between 3 and 4 as in Tables 11, 12 and 14, and the RBF centers are identical to the nodes in each subdomain.

A comparison between the stress s_x along $x = 0$ obtained by the MIRBFN with a structured discretisation of 409 nodes and the exact solution are presented in Figure 26. The result indicates that the solution obtained by the proposed method agrees well with the exact one.

Table 14: Infinite plate with a circular hole: unstructured node distribution with MIRBFN ($\mu = 0.3$).

No. points	$L_2^{\mathbf{u}}$	L_2^{σ}	cond(A)	β	$\frac{R_s}{h}$	h	$\frac{CPU\ time}{second}$
68	7.5923e-1	8.0880e-1	4.9714e4	4	2.1	5.0000e-1	1.383
156	2.3616e-1	3.4100e-1	2.9975e5	4	2.1	3.0888e-1	2.911
479	1.0531e-2	4.0582e-2	2.7228e6	4	2.1	1.6343e-1	8.729
1024	5.0684e-3	2.2821e-2	6.2775e6	3	2.1	1.1346e-1	23.620
2439	9.9303e-4	8.2974e-3	1.2450e8	3	2.1	7.5139e-2	81.186
	$O(h^{3.60})$	$O(h^{2.50})$					

The convergence of the present method in the case of structured node distribution (Figure 24) is reported in Table 11 and Figure 27 for $\mu = 0.3$, and in Table 12 and Figure 27 for the case of incompressible materials. The present method appears to converge at the rates of $O(h^{3.61})$ for displacement and $O(h^{3.02})$ for stress in the case of $\mu = 0.3$. In the case of incompressible materials, the orders of convergence are $O(h^{3.68})$ and $O(h^{3.27})$ for displacement and stress, respectively.

The performance of the MIRBFN method is also tested with irregular node distributions as shown in Figure 25. The obtained results are presented in Table 14 and Figure 28, which show that the convergence rates are $O(h^{3.60})$ and $O(h^{2.50})$ for displacement and stress, respectively.

Again, the MIRBFN method achieves similar accuracy and convergence rates in comparison with those of the global IRBFN method as shown in Table 11 and 13, and in Figure 27. Clearly, the efficiency of the present method is superior to that of the global IRBFN (Figure 29).

4.2.3 Mode I crack problem

Consider an infinite plate containing a straight crack of length $2a$ and loaded by a remote uniform stress field σ as shown in Figure 30. Along ABCD the closed form solution in terms of polar coordinates in a reference frame (r,θ) centered at the crack tip is given by

$$\sigma_x = \frac{K_I}{\sqrt{r}}\cos\frac{\theta}{2}\left(1-\sin\frac{\theta}{2}\sin 3\frac{\theta}{2}\right), \tag{72a}$$

$$\sigma_y = \frac{K_I}{\sqrt{r}}\cos\frac{\theta}{2}\left(1+\sin\frac{\theta}{2}\sin 3\frac{\theta}{2}\right), \tag{72b}$$

$$\tau_y = \frac{K_I}{\sqrt{r}}\sin\frac{\theta}{2}\cos\frac{\theta}{2}\cos 3\frac{\theta}{2}, \tag{72c}$$

Table 15: Center crack problem: uniform discretisations with MIRBFN ($\mu = 0.3$).

No. points	$L_2^{\mathbf{u}}$	L_2^{σ}	cond(A)	β	$\frac{R_s}{h}$	h	$\frac{CPU\ time}{second}$
10×10	0.2017	1.3794	2.7726e5	0.01	1.1	1.0	0.53
14×14	0.13882	0.6583	6.7641e5	0.01	1.1	0.714	0.94
16×16	0.0909	0.5043	4.7002e5	0.01	1.1	0.625	1.22
20×20	0.0374	0.2327	5.1297e5	0.01	1.1	0.50	1.85
24×24	0.0269	0.1887	1.6910e6	0.01	1.1	0.416	2.68
	$O(h^{2.47})$	$O(h^{2.38})$					

for stress and

$$u_x = \frac{2(1+\mu)}{\sqrt{2\pi}} \frac{K_I}{E} \sqrt{r} \cos\frac{\theta}{2} \left(2 - 2\mu - \cos^2\frac{\theta}{2}\right), \tag{73a}$$

$$u_y = \frac{2(1+\mu)}{\sqrt{2\pi}} \frac{K_I}{E} \sqrt{r} \sin\frac{\theta}{2} \left(2 - 2\mu - \cos^2\frac{\theta}{2}\right), \tag{73b}$$

for displacement, where $K_I = \sigma\sqrt{\pi a}$ is the stress intensity factor, μ Poisson's ratio and E Young modulus, ABCD a square of $10 \times 10\ mm^2$, $a = 100\ mm$, $E = 10^7 N/mm^2$, $\mu = 0.3$ (also $\mu = 0.5$), $\sigma = 10^4 N/mm^2$. Plane strain condition is assumed and the body force is zero.

The computational domain ABCD is shown in Figure 30. Owing to symmetry, only upper half of ABCD, namely CDEFG as shown in Figure 31, is analyzed. The segment of crack denoted by EF has a length of $b = 5\ mm$. The boundary condition of the problem is as follows. The traction free boundary condition is applied on the crack while the displacement field given in (73) is imposed on the remaining boundaries.

It is known that stress tends to infinity when r tends to 0. Thus, to alleviate the oscillation due to the effect of singularity, the support radius R_s and β are selected as small as possible. For this example, $\frac{R_s}{h}$ and β are set at 1.1 and 0.01, respectively, and the RBF centers are chosen to be identical to the nodes in each subdomain.

The performance of the present method in this singular problem is examined by employing a number of uniform data point distributions as displayed in the Tables 15 and 16. The results with 24×24 nodes ($\mu = 0.3$) are plotted in Figures 32-34 as follows. Figures 32 and 33 exhibit displacement u_x and u_y, respectively, in comparison with those of an analytical solution. Figures 34 (a) and (b) depict stress s_x and s_y by MIRBFN method, respectively, and the corresponding exact solutions. Some oscillations can be observed in these figures due to singularity of stress with C_0

Table 16: Center crack problem: uniform discretizations with MIRBFN ($\mu = 0.5$).

No. points	$L_2^{\mathbf{u}}$	L_2^{σ}	cond(A)	β	$\frac{R_s}{h}$	h	$\frac{CPU\ time}{second}$
10×10	0.1087	0.9622	1.4859e5	0.01	1.1	1.0	0.53
14×14	0.1064	0.6401	2.9922e5	0.01	1.1	0.714	0.94
16×16	0.0477	0.3233	2.8071e5	0.01	1.1	0.625	1.22
20×20	0.0366	0.2627	5.3818e5	0.01	1.1	0.50	1.85
24×24	0.0379	0.2613	4.9657e5	0.01	1.1	0.416	2.73
	$O(h^{1.44})$	$O(h^{1.64})$					

continuity of displacement and C_∞ property of IRBFNs. This oscillation is known as the Gibbs phenomenon in RBF-based methods [Jung (2007)] where numerical oscillations occur around a jump discontinuity because of high order approximation by RBF. Nevertheless, the obtained results are in good agreement with the analytical ones and the present MIRBFN method is able to capture highly steep gradients.

The convergence of the method can be seen in Tables 15 and 16, and in Figure 35. In the case of $\mu = 0.3$, high convergence rates of $O(h^{2.47})$ and $O(h^{2.38})$ for displacement and stress, respectively, are obtained. It is apparent that accuracy of the stress field is considerably reduced in comparison with that of displacement due to the presence of singularity (Figure 35,). For incompressible materials, the convergence rates reduce to $O(h^{1.44})$ and $O(h^{1.64})$ for displacement and stress, respectively.

5 Concluding remarks

In this work, we propose a locally supported RBF interpolation method, namely MIRBFN, with the main features as follows.

- The proposed method is a locally supported approximation method. As a result, the resultant interpolation matrices are sparse and banded, resulting in improved efficiency in comparison with those of standard RBF methods.
- The shape functions of the MIRBFN method possesses the Kronecker-δ property that facilitates the imposition of the essential boundary conditions.
- The present method offers high orders of convergence and is applicable to scattered node distribution, arbitrary domain and highly steep gradient problems.

Moreover, the proposed interpolation method is implemented in the collocation of a first-order system formulation resulting in an integration-free meshless method which enjoys high convergence rate and very good accuracy.

Acknowledgement: This work is supported by the Australian Research Council. This support is gratefully acknowledged.

References

Atluri, S.; Shen, S. (2002): The meshless local Petrov-Galerkin (MLPG) method:A simple & less-costly alternative to the finite element and the boundary element methods. *CMES: Computer Modeling in Engineering & Sciences*, vol. 3, pp. 11–51.

Atluri, S. N.; Liu, H. T.; Han, Z. D. (2006): Meshless local Petrov-Galerkin (MLPG) mixed collocation method for elasticity problems. *CMES: Computer Modeling in Engineering & Sciences*, vol. 14, no. 3, pp. 141–152.

Babuška, I.; Melenk, I. (1997): The partition of unity method. *International Journal for Numerical Methods in Engineering*, vol. 40, no. 4, pp. 727–758.

Baxter, B. (2002): Preconditioned conjugate gradient, radial basis functions, and toeplitz matrices. *Compu. Math. Appl.*, vol. 43, pp. 305–318.

Belytschko, T.; Krongauz, Y.; Fleming, M.; Organ, D.; Liu, W. (1996): Smoothing and accelerated computations in the element-free Galerkin method. *J. Comput. Appl. Math*, vol. 74, pp. 111–126.

Belytschko, T.; Lu, Y. Y.; Gu, L. (1994): Element-free Galerkin methods. *International Journal for Numerical Methods in Engineering*, vol. 37, pp. 229–256.

Bordas, S.; Duflot, M.; Le, P. (2008): A simple error estimator for extended finite elements. *Communications In Numerical Methods In Engineering*, vol. 24, pp. 961–971.

Brown, D.; Ling, L.; Kansa, E.; Levesley, J. (2005): On approximate cardinal preconditioning methods for solving pdes with radial basis functions. *Engineering Analysis with Boundary Elements*, vol. 29, pp. 343–353.

Chen, J.; Hu, W.; Hu, H. (2008): Reproducing kernel enhanced local radial basis collocation method. *International Journal for Numerical Methods in Engineering*, vol. 75, pp. 600–627.

Cheng, A.; Golberg, M.; Kansa, E. J.; Zamitto, G. (2003): Exponential convergnce and h-c multiquadric collocation method for partial differential equations. *Numerical Methods for Partial Differential Equations*, vol. 19, pp. 571–594.

Fasshauer, G. (1997): Solving partial differential equations by collocation with radial basis functions. In Méhauté, A. L.; Rabut, C.; Schumaker, L.(Eds): *Surface Fitting and Multiresolution Methods*, pp. 131–138. Vanderbilt University Press.

Han, Z. D.; Atluri, S. N. (2003): Truly meshless local petrov-galerkin (MLPG) solutions of traction & displacement BIEs. *CMES: Computer Modeling in Engineering & Sciences*, vol. 4, no. 6, pp. 665–678.

Jung, J. H. (2007): A note on the Gibbs phenomenon with multiquadratic radial basis functions. *Applied Numerical Mathematics*, vol. 57, pp. 213–229.

Kansa, E.; Hon, Y. (2000): circumventing the ill-conditioning problem with multiquadric radial basis functions: application to elliptic partial differential equations. *Compu. Math. Appl.*, vol. 39, pp. 123–137.

Le, P.; Mai-Duy, N.; Tran-Cong, T.; Baker, G. (2007): A numerical study of strain localization in elasto-therno-viscoplastic materials using radial basis function networks. *CMC: Computers, Materials & Continua*, vol. 5, pp. 129–150.

Le, P.; Mai-Duy, N.; Tran-Cong, T.; Baker, G. (2008): A meshless modeling of dynamic strain localization in quasi-britle materials using radial basis function networks. *CMES: Computer Modeling in Engineering & Sciences*, vol. 25, no. 1, pp. 43–66.

Le, P.; Mai-Duy, N.; Tran-Cong, T.; Baker, G. (2010): A Cartesian-grid Collocation Technique with Integrated Radial Basis Functions for mixed boundary value problems. *International Journal for Numerical Methods in Engineering*, vol. 82, pp. 435–463.

Li, Q.; Shen, S.; Han, Z. D.; Atluri, S. N. (2003): Application of meshless local Petrov-Galerkin (MLPG) to problems with singularities, and material discontinuities, in 3-D elasticity. *CMES: Computer Modeling in Engineering & Sciences*, vol. 4, no. 5, pp. 571–585.

Libre, N. A.; Emdadi, A.; Kansa, E. J.; Rahimian, M.; Shekarchi, M. (2008): A stabilized RBF collocation scheme for Neumann type boundary value problems. *CMES: Computer Modeling in Engineering & Sciences*, vol. 24, pp. 63–82.

Ling, L.; Kansa, E. (2004): Preconditioning for radial basis functions with domain decomposition methods. *Mathematical and Computer Modelling*, vol. 40, pp. 1413–1427.

Liu, G.; Zhang, G.; Gu, Y. (2005): A meshfree radial point interpolation method (RPIM) for three-dimensional solids. *Computational Mechanics*, vol. 36, no. 6, pp. 421–430.

Liu, G.; Zhang, J.; Li, H.; Lam, K.; Kee, B. (2006): Radial point interpolation based finite difference method for mechanics problems. *International Journal for Numerical Methods in Engineering*, vol. 68, pp. 728–754.

Liu, X.; Liu, G.; Tai, K. (2005): Radial point interpolation collocation method (RPICM) for partial differential equations. *Computers & Mathematics With Applications*, vol. 50, no. 8–9, pp. 1425–1442.

Madych, W. (1992): Miscellaneous error bounds for multiquadric and related interpolators. *Compu. Math. Appl.*, vol. 24, pp. 121–138.

Mai-Duy, N.; Khennane, A.; Tran-Cong, T. (2007): Computation of laminated composite plates using indirect radial basis function networks. *CMC: Computers, Materials & Continua*, vol. 5, pp. 63–77.

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

Mai-Duy, N.; Tran-Cong, T. (2005): An efficient indirect RBFN-based method for numerical solution of PDEs. *Numerical Methods for Partial Differetial Equations*, vol. 21, pp. 770–790.

Mai-Duy, N.; Tran-Cong, T. (2009): A Cartesian-Grid discretisation scheme based on local integrated IRBFNs for two-dimensional elliptic problems. *CMES: Computer Modeling in Engineering & Sciences*, vol. 51, pp. 213–238.

Melenk, J. M.; Babuška, I. (1996): The partition of unity finite element method: Basic theory and applications. *Computer Methods in Applied Mechanics and Engineering*, vol. 139, pp. 289–314.

Moës, N.; Dolbow, J.; Belytschko, T. (1999): A finite element method for crack growth without remeshing. *International Journal for Numerical Methods in Engineering*, vol. 46, pp. 131–150.

Nguyen, V.; Rabczuk, T.; Bordas, S.; Duflot, M. (2008): Meshfree method: A review and computer implementation aspects. *Mathematics and Computers in Simulation*, vol. 79, pp. 763–813.

Ohtake, Y.; Belyaev, A.; Seidel, H. (2006): Sparse surface reconstruction with adaptive partition of unity and radial basis functions. *Graphical Models*, vol. 68, pp. 15–24.

Rabczuk, T.; Areias, P.; Belytschko, T. (2007): A meshfree thin shell method for non-linear dynamic fracture. *International Journal for Numerical Methods in Engineering*, vol. 72, no. 5, pp. 524–548.

Rabczuk, T.; Belytschko, T. (2004): Cracking particles: A simplified meshfree method for arbitrary evolving cracks. *International Journal for Numerical Methods in Engineering*, vol. 61, no. 13, pp. 2316–2343.

Schilling, R. J.; Caroll, J. J.; Al-Ajlouni, A. F. (2001): Approximation of nonlinear system with radial basis function neural networks. *IEEE Transaction on Neural Networks*, vol. 12, pp. 1–15.

Shu, C.; Ding, H.; Yeo, K. (2003): Local radial basis functions based differential quadrature method and its application to solve two-dimensional incompressible Navier-Stoke equations. *Computer Methods in Applied Mechanics and Engineering*, vol. 192, pp. 941–954.

Shu, C.; Wu, Y. (2007): Integrated radial basis functions based differential quadrature method and its perfomance. *International Journal for Numerical Methods in Fluids*, vol. 53, pp. 969–984.

Strouboulis, T.; Babuška, I.; Copps, K. (2000): The design and analysis of the generalized finite element method. *Computer Methods in Applied Mechanics and Engineering*, vol. 181, pp. 43–71.

Strouboulis, T.; Copps, K.; Babuška, I. (2000): The generalized finite element method: an example of its implementation and illustration of its performance. *International Journal for Numerical Methods in Engineering*, vol. 47, no. 8, pp. 1401–1417.

Timoshenko, S.; Goodier, J. (1970): *Theory of elasticity*. McGraw-Hill, New York, USA, 3rd edition.

Tobor, I.; Reuter, P.; Schlick, C. (2004): Efficient reconstruction of large scattered geometric datasets using the partition of unity and radial basis functions. *Journal of WSCG*, vol. 12, no. 1–3. ISSN-1213-6972.

Tobor, I.; Reuter, P.; Schlick, C. (2006): Reconstructing multi-scale variational partition of unity implicit surfaces with attributes. *Graphical Models*, vol. 68, pp. 25–41.

Šaler, B.; Vertnik, R. (2006): Meshfree local radial basis function collocation for diffusion problems. *Compu. Math. Appl.*, vol. 51, pp. 1269–1282.

Wendland, H. (1995): Piecewise polynomial, positive definite and compactly supported radial basis functions of minimal degree. *Advances in Computational Mathematics*, vol. 4, pp. 389–396.

Wendland, H. (2002): Fast evaluation of radial basis functions: Method based on partition of unity. In Chui, C.; Schumaker, L.; Stöckler, J.(Eds): *Approximation theory X: Abstract and classical Analysis*, pp. 473–483. Vanderbilt University Press.

Wright, G.; Fornberg, B. (2006): Scattered node compact finite difference-type formulas generated from radial basis functions. *Journal of Computational Physics*, vol. 212, pp. 99–123.

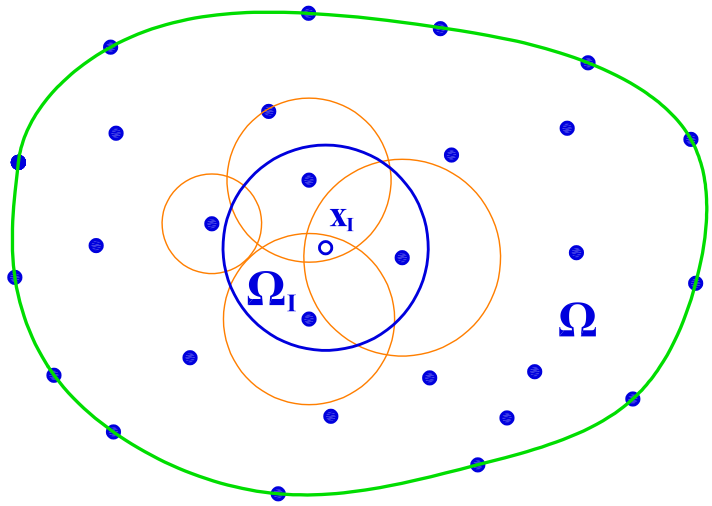


Figure 1: Schematic representation of a moving IRBFN: Ω is the domain of interest which is subdivided into N overlapping subdomains Ω_I centered at $\mathbf{x}_I$.

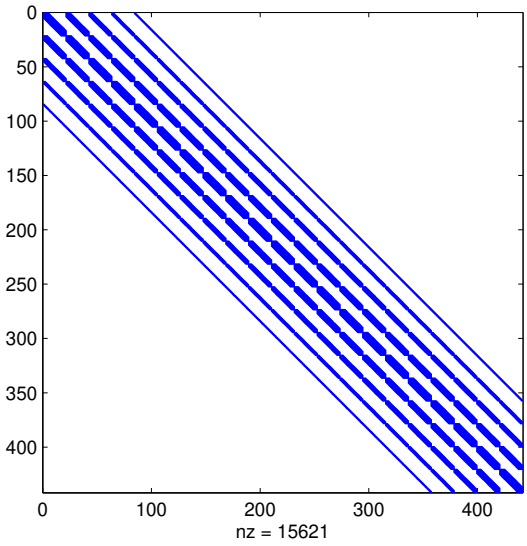
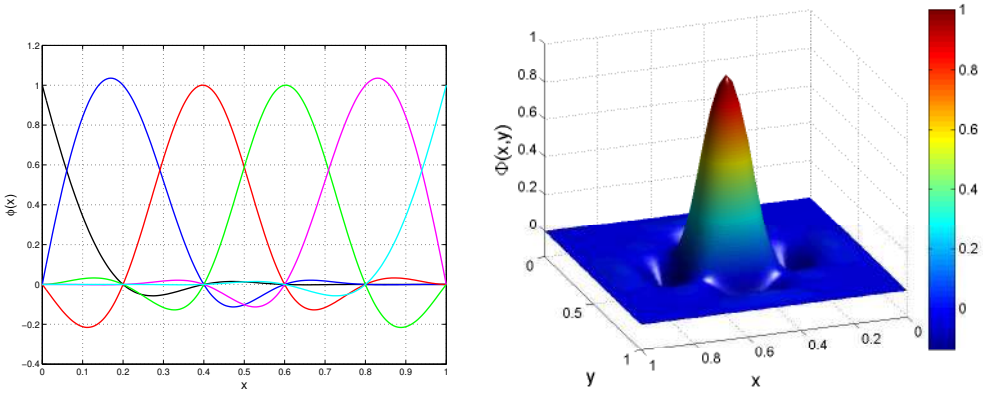
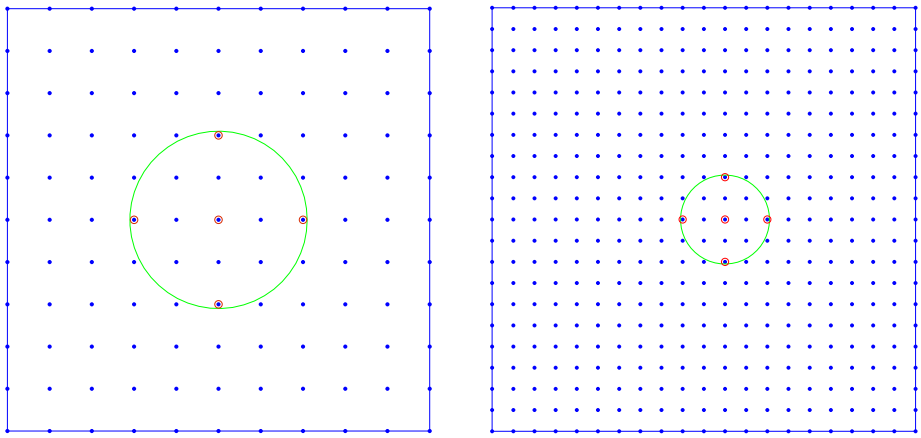


Figure 2: Moving IRBFN yields symmetric, sparse and banded interpolation matrices.



(a) (b)

Figure 3: Example of MIRBFN shape functions: (a) $\Phi_I(x)$ in one dimension and (b) $\Phi_I(x,y)$ in two dimensions.



(a) (b)

Figure 4: Poisson equation in a regular domain: discretisation with uniform distribution of (a) 11×11 nodes, (b) 21×21 nodes. The small circles are RBF centers and the big one is subdomain Ω_I.

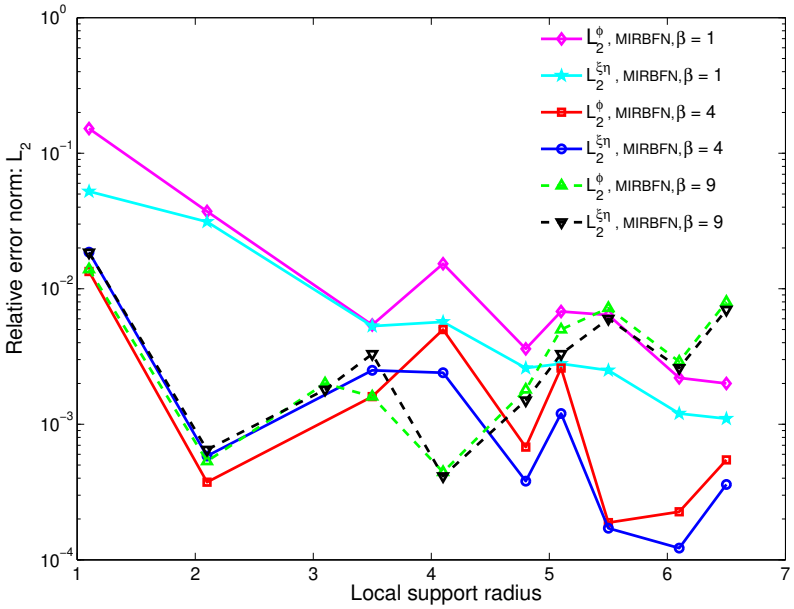


Figure 5: Poisson equation in a regular domain with uniform distribution of 21 $\times$ 21 nodes: influence of the local support radius on the accuracy of the solution.

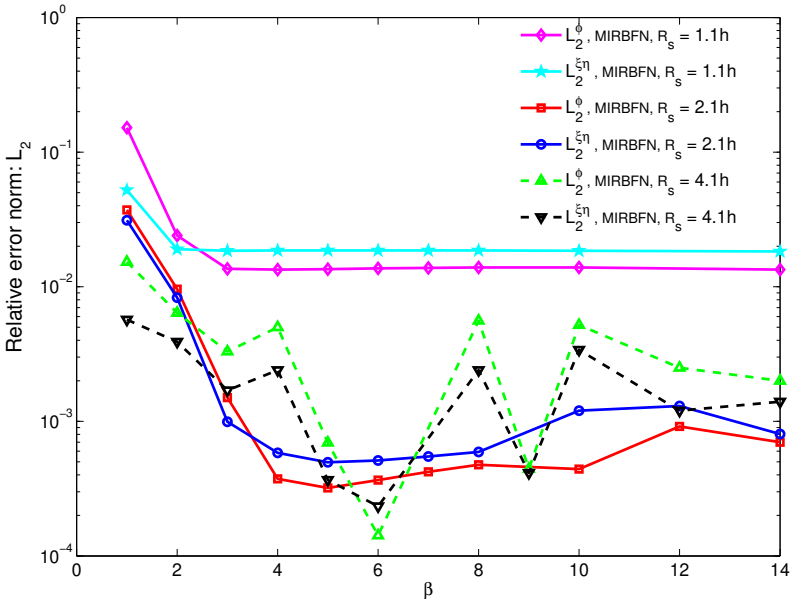


Figure 6: Poisson equation in a regular domain with uniform distribution of 21 $\times$ 21 nodes: influence of the RBF width β on the accuracy of the solution.

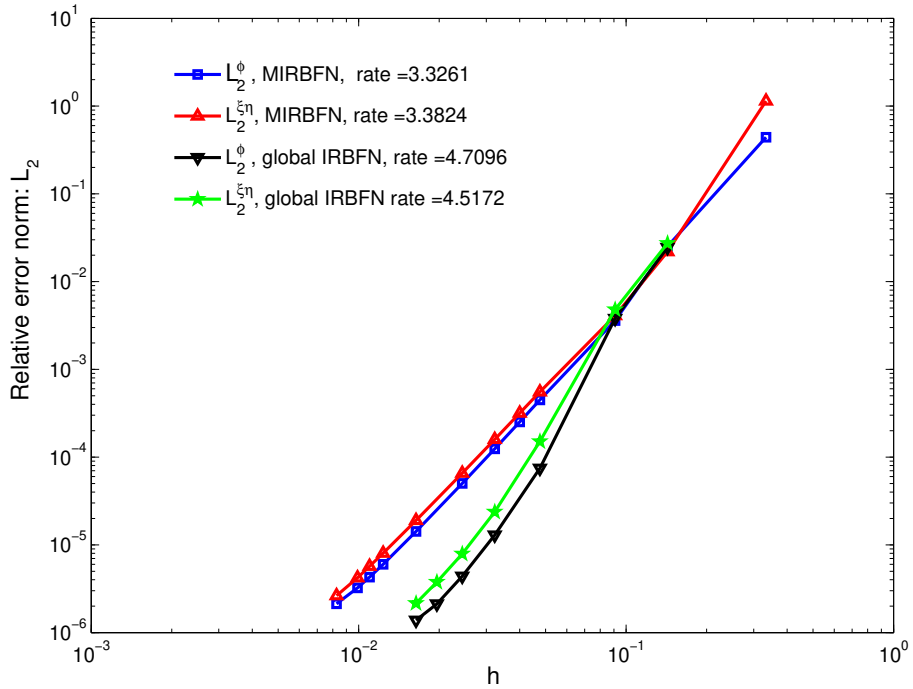


Figure 7: Poisson equation in a regular domain with uniform distribution of nodes: relative error norms L_2^{ϕ} and $L_2^{\xi\eta}$, and associated convergence rates.

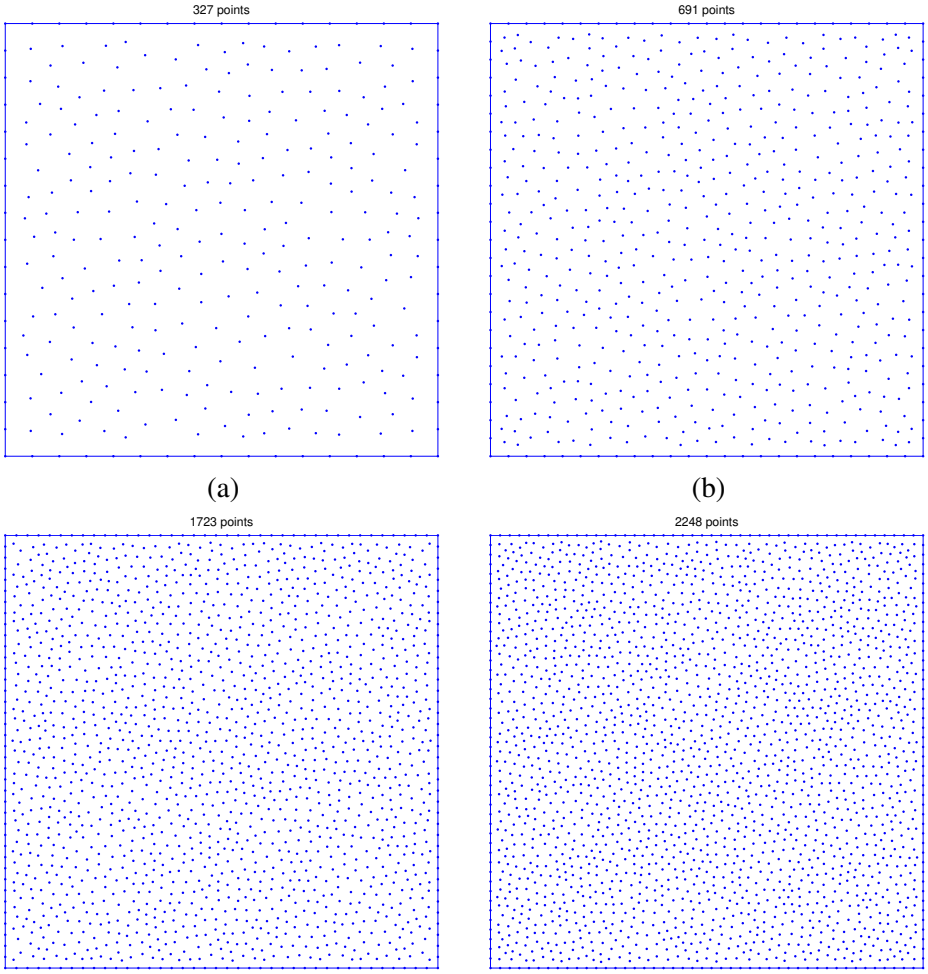


Figure 8: Poisson equation in a regular domain: discretisation with unstructured distribution of (a) 327 nodes, (b) 691 nodes, (c) 1723 nodes and (d) 2248 nodes.

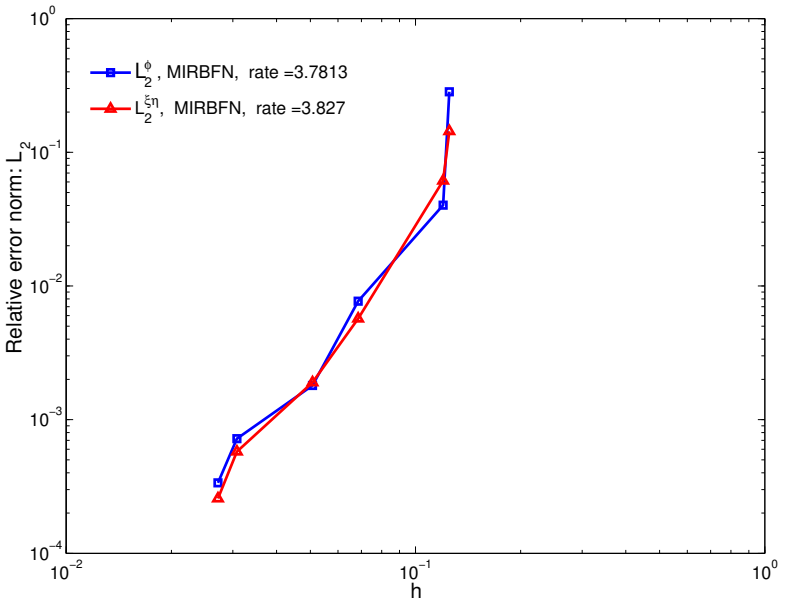


Figure 9: Poisson equation in a regular domain: relative error norms L_2^{ϕ} and $L_2^{\xi\eta}$, and associated convergence rates obtained by MIRBFN method with unstructured nodes.

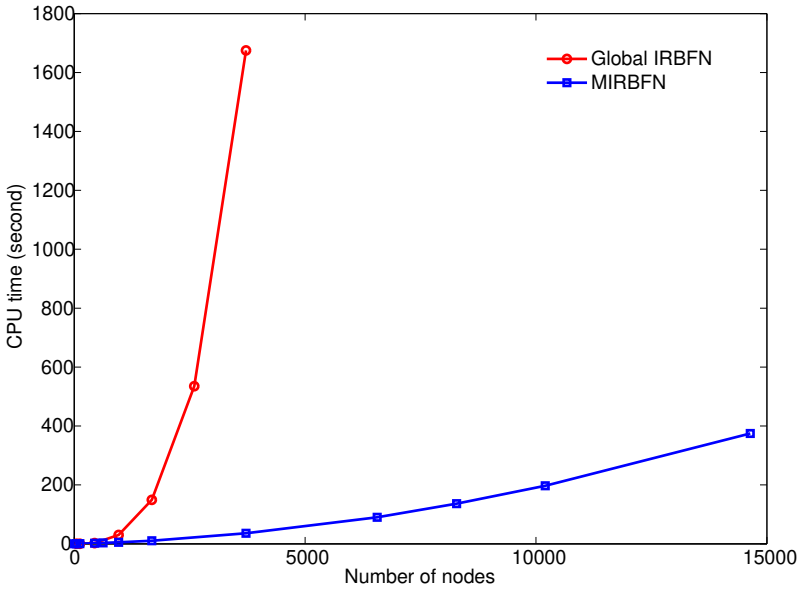


Figure 10: Poisson equation in a regular domain with uniform distribution of nodes: CPU times of MIRBFN method versus that of global IRBFN method.

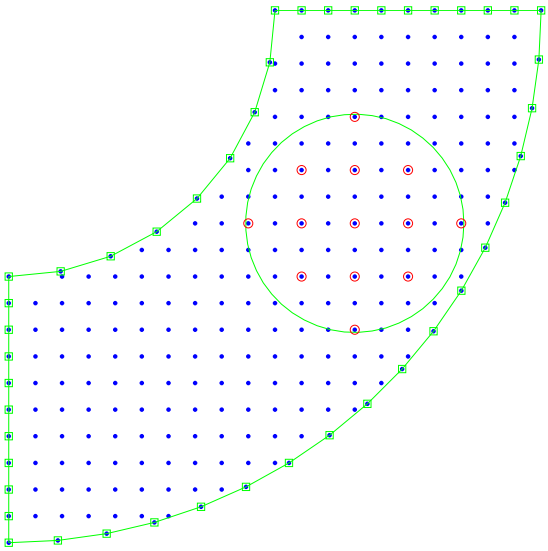


Figure 11: Poisson equation in an irregular domain: structured discretisation with 266 nodes.

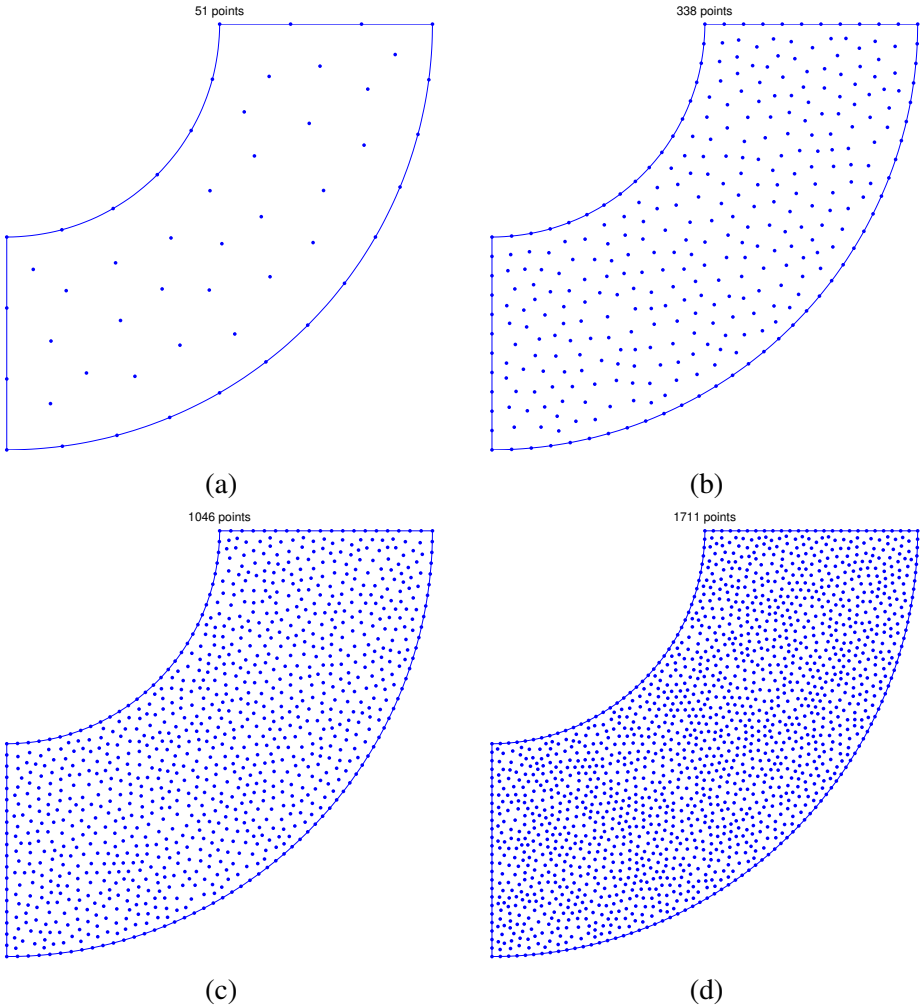


Figure 12: Poisson equation in an irregular domain: discretisation with unstructured distribution of (a) 51 nodes, (b) 338 nodes, (c) 1046 nodes and (d) 1711 nodes.

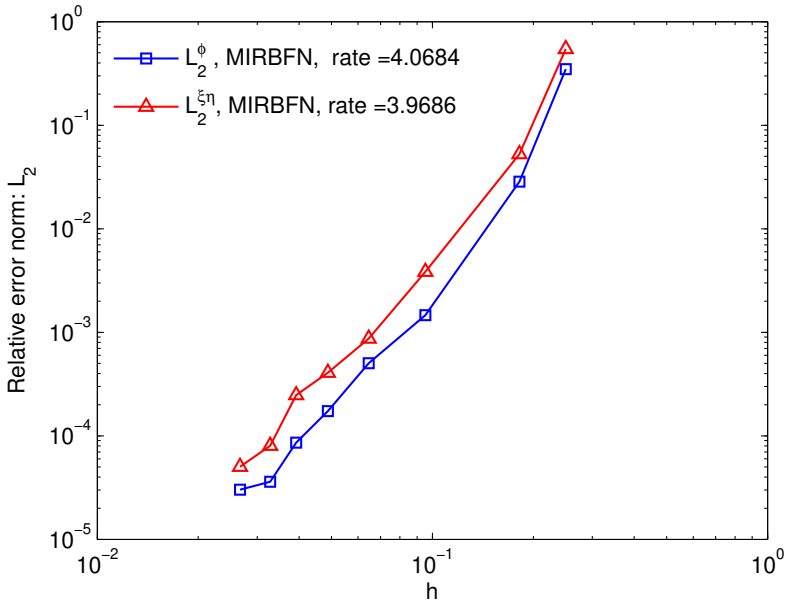


Figure 13: Poisson equation in an irregular domain with regular distribution of nodes: relative error norms L_2^{ϕ} and $L_2^{\xi\eta}$, and associated convergence rates.

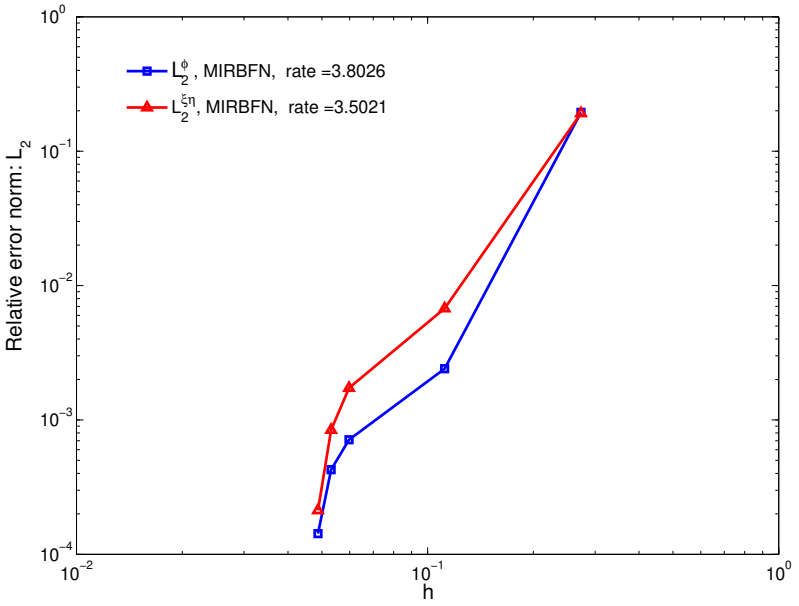


Figure 14: Poisson equation in an irregular domain with unstructured distribution of nodes: relative error norms L_2^{ϕ} and $L_2^{\xi\eta}$, and associated convergence rates.

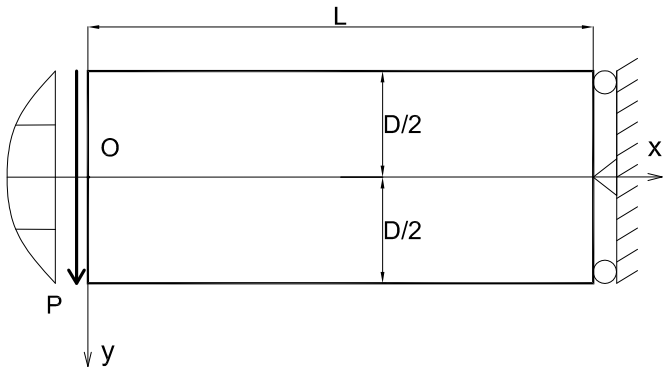


Figure 15: Cantilever beam: a mathematical model.

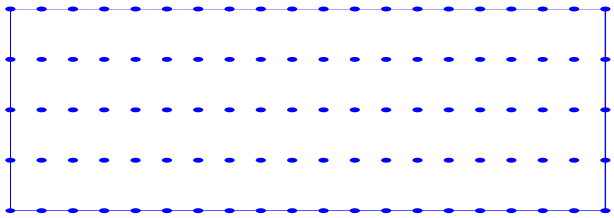


Figure 16: Cantilever beam: discretisation with 20×5 nodes.

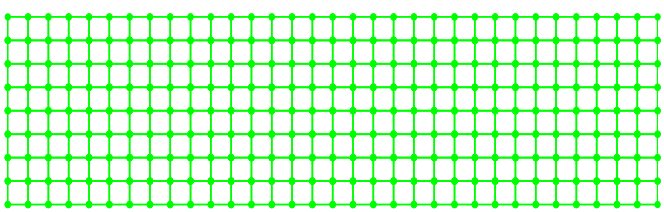


Figure 17: Cantilever beam: a FEM mesh with 8×32 Q4 elements.

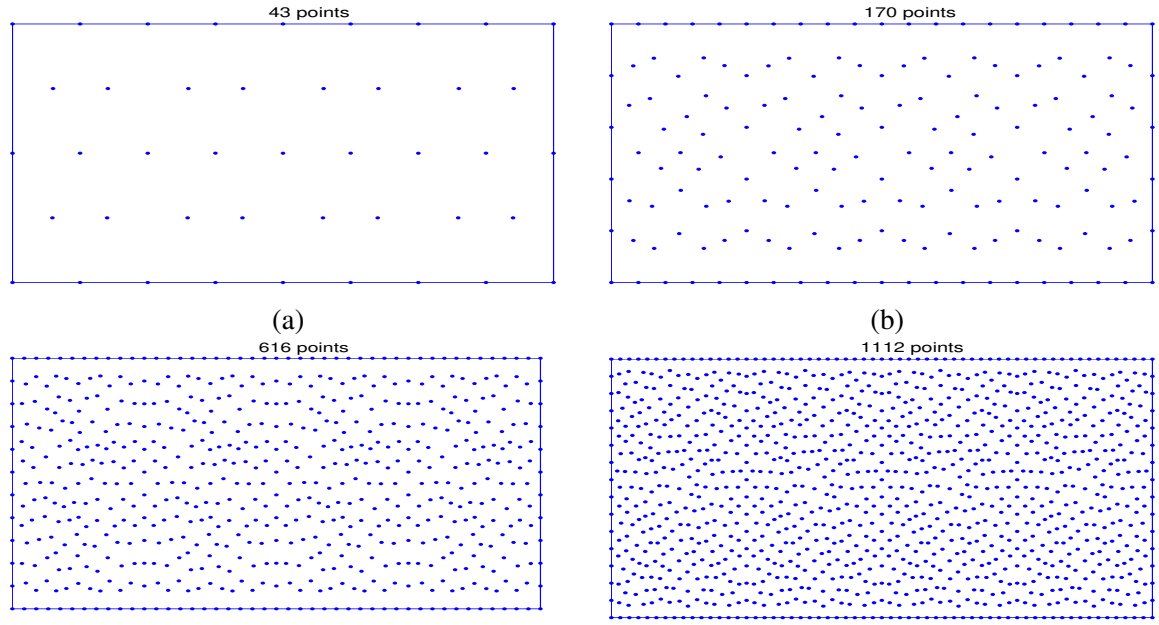


Figure 18: Cantilever beam: discretisation with unstructured distribution of (a) 43 nodes, (b) 170 nodes, (c) 616 nodes, and (d) 1112 nodes.

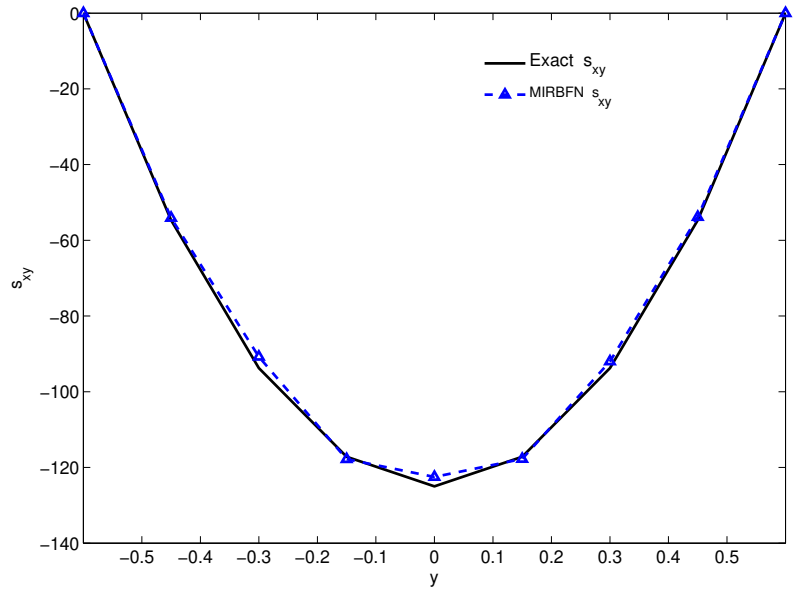


Figure 19: Cantilever beam: s_{xy} at $x = 2.4686$ with 36×9 nodes ($\mu = 0.3$).

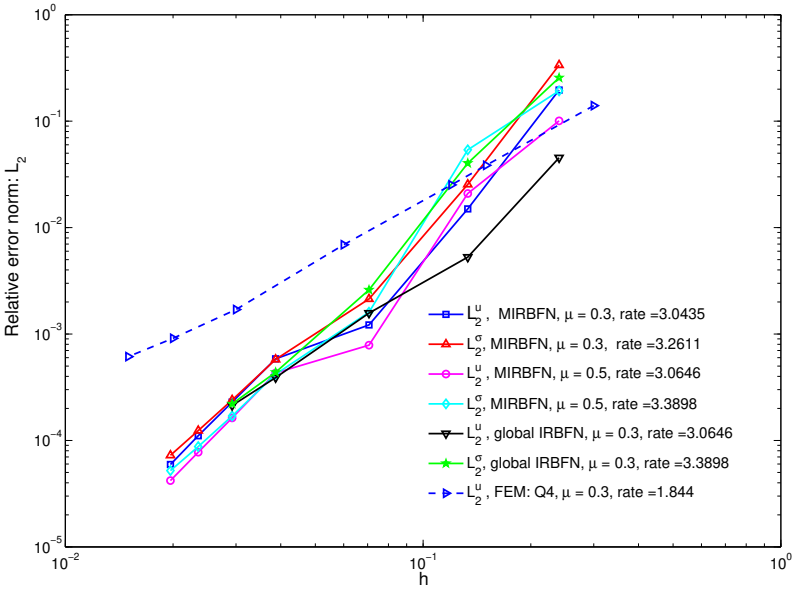


Figure 20: Cantilever beam: L_2 relative error norms for displacement and stress for $\mu = 0.3$ and $\mu = 0.5$, with associated convergence rates.

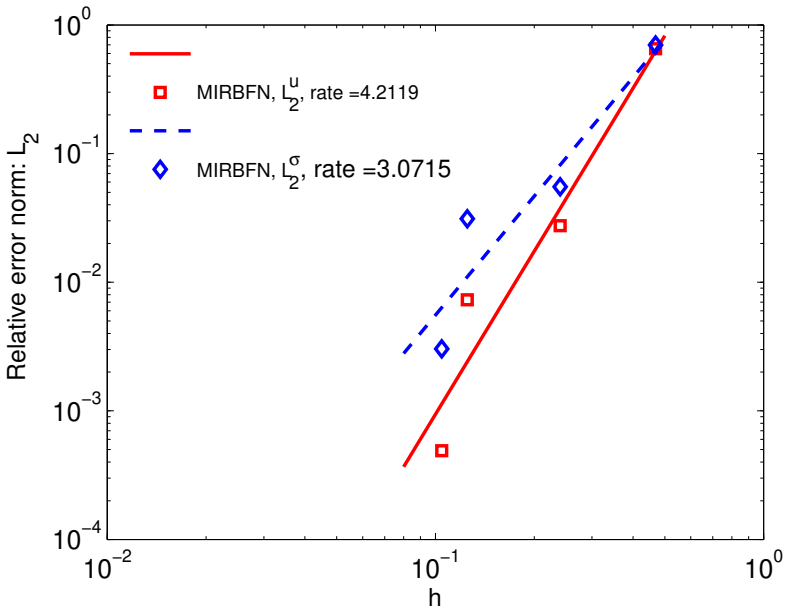


Figure 21: Cantilever beam: L_2 relative error norms for displacement and stress, and associated convergence rates for $\mu = 0.3$ with different unstructured nodal configurations.

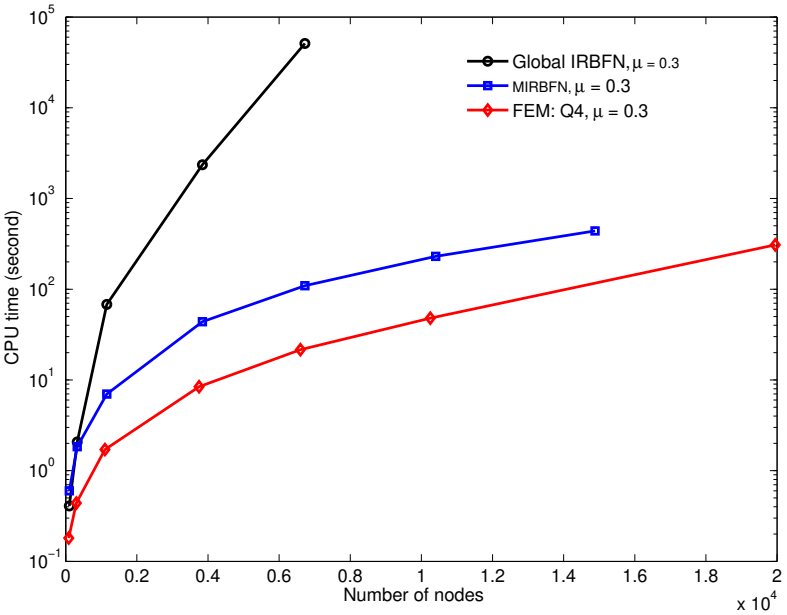


Figure 22: Cantilever beam: CPU times of MIRBFN method versus that of FEM and global IRBFN method.

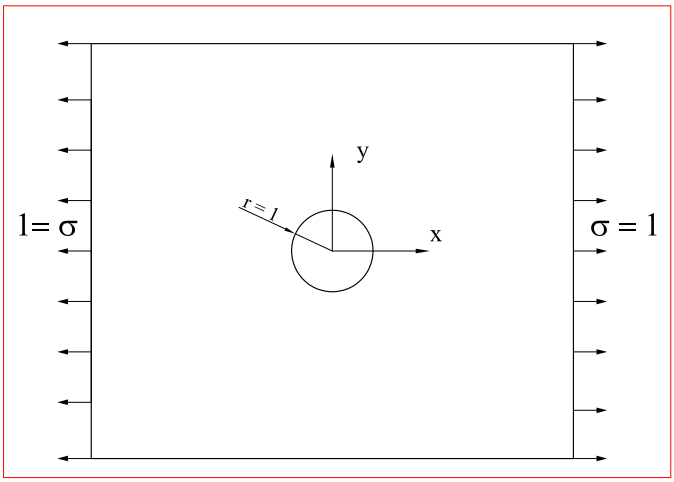


Figure 23: Infinite plate with a circular hole.

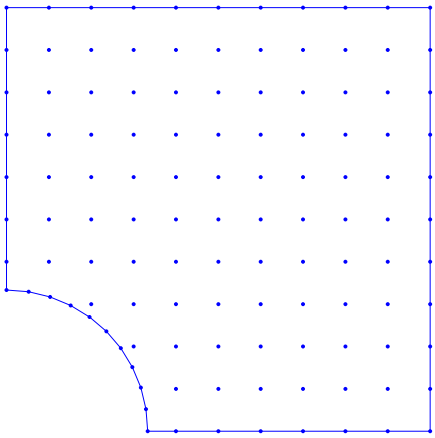


Figure 24: Infinite plate with a circular hole: computational domain with 119 nodes.

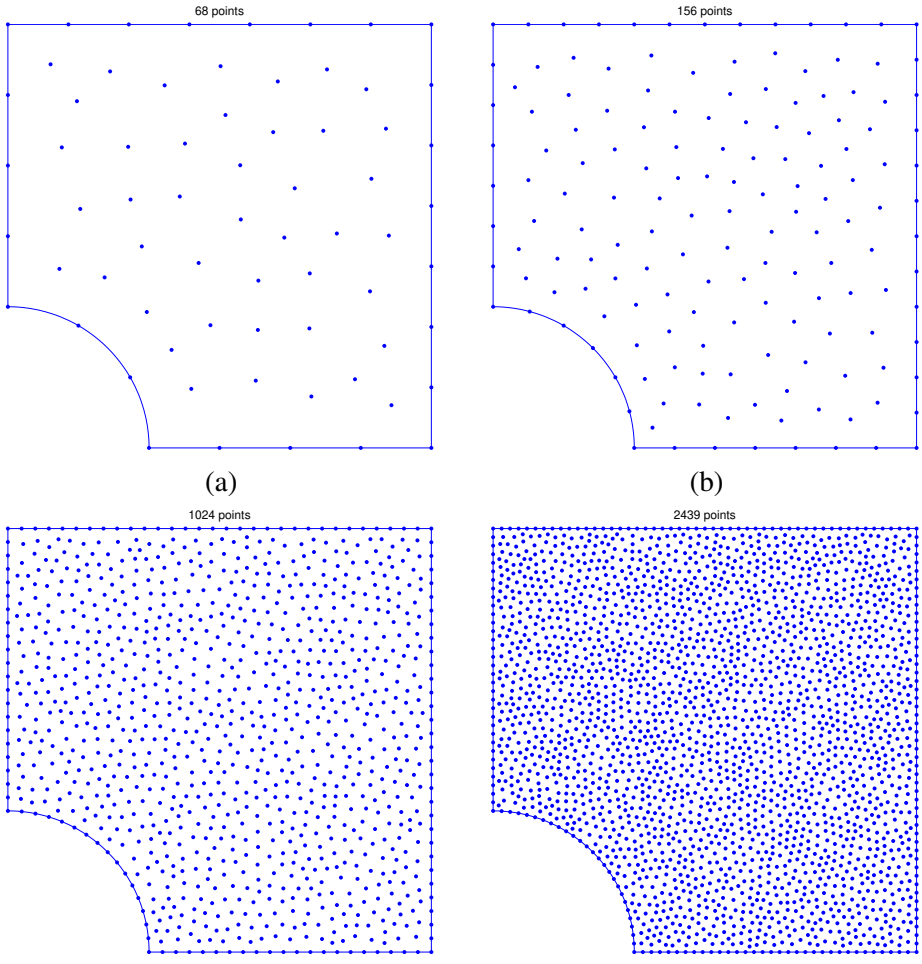


Figure 25: Infinite plate with a circular hole: discretisation with unstructured distribution of (a) 68 nodes, (b) 156 nodes, (c) 1024 nodes, and (d) 2439 nodes.

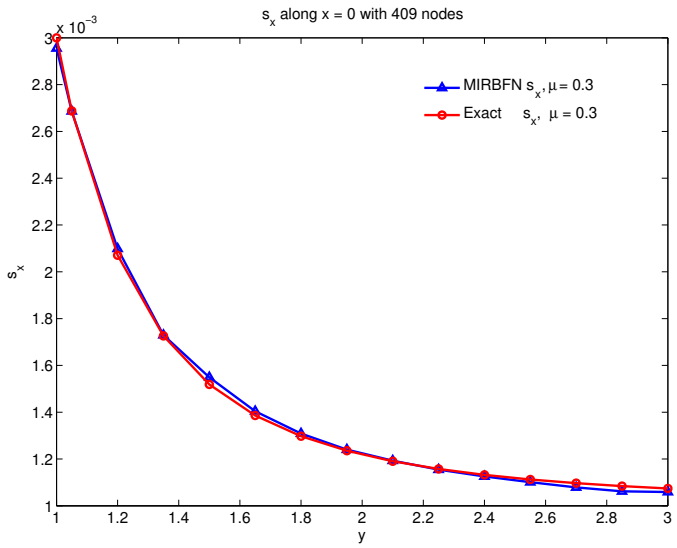


Figure 26: Infinite plate with a circular hole: s_x along $x = 0$ with 409 nodes ($\mu = 0.3$).

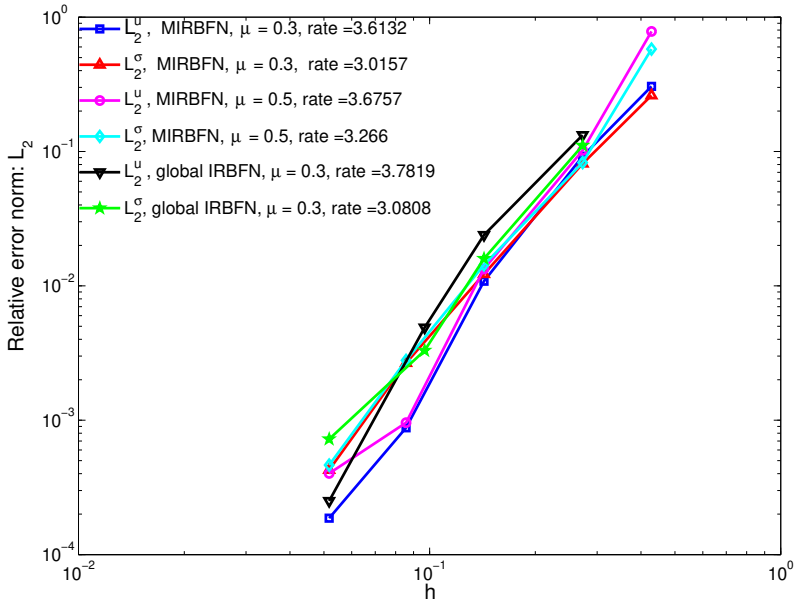


Figure 27: Infinite plate with a circular hole: L_2 relative error norms for displacement and stress for $\mu = 0.3$ and $\mu = 0.5$. Convergence rates are also shown.

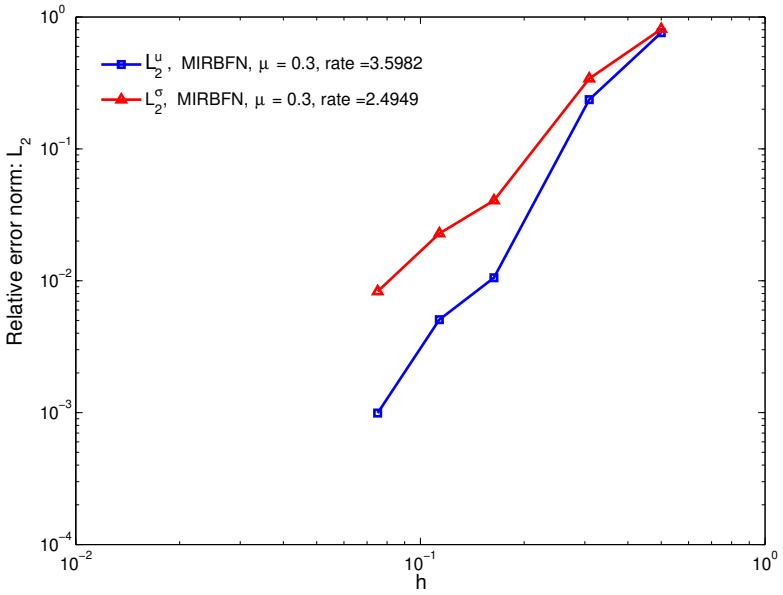


Figure 28: Infinite plate with a circular hole: L_2 relative error norms for displacement and stress for $\mu = 0.3$ with unstructured nodes. Convergence rates are also shown.

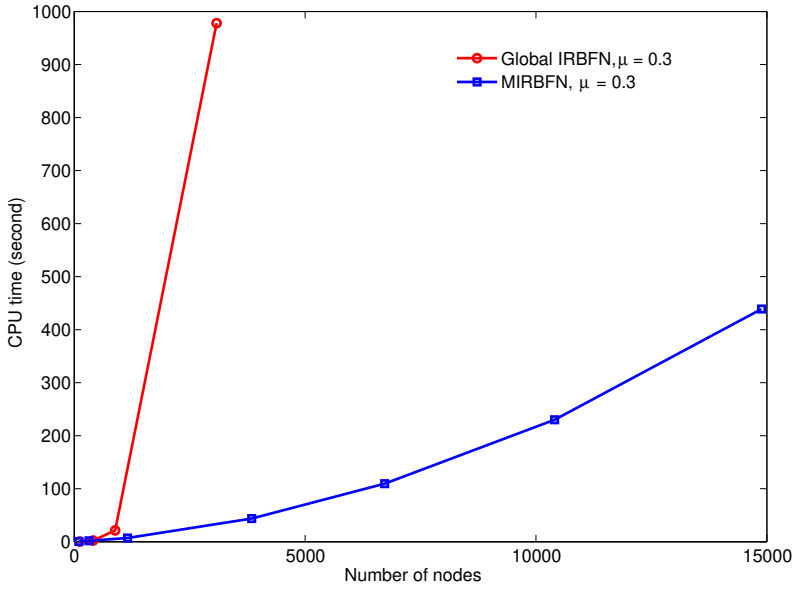


Figure 29: Infinite plate with a circular hole: CPU times of MIRBFN method versus that of global IRBFN method.

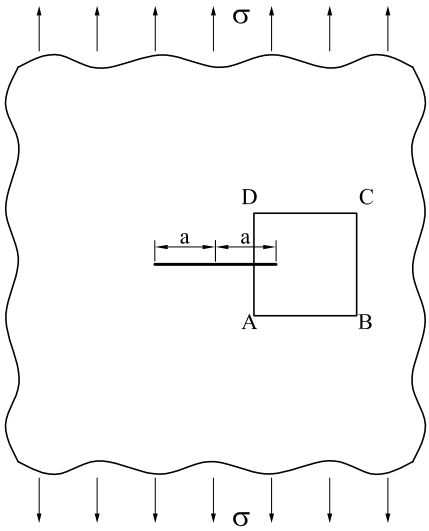


Figure 30: Infinite cracked plate under remote tension.

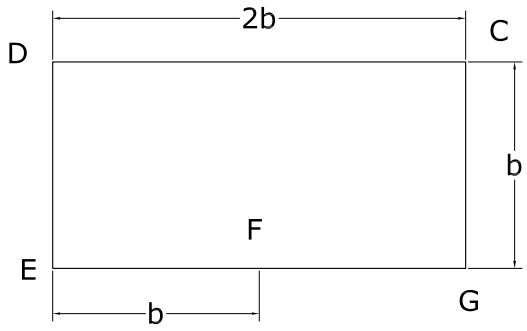
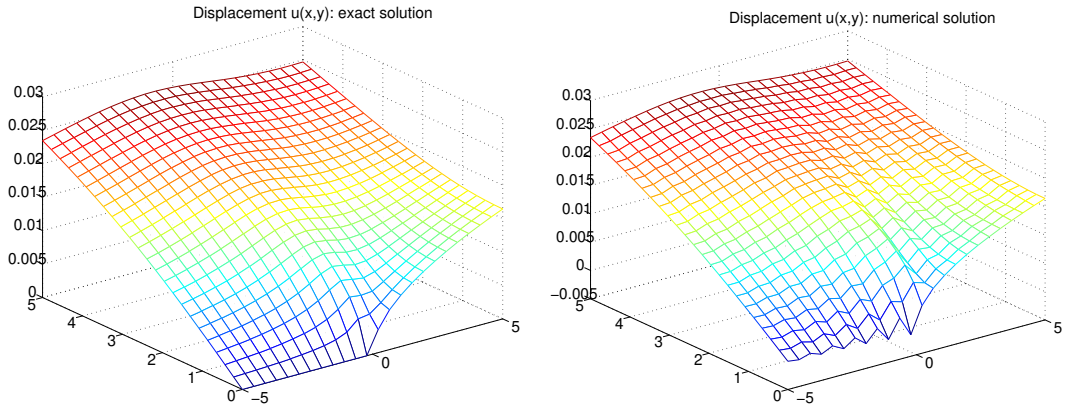
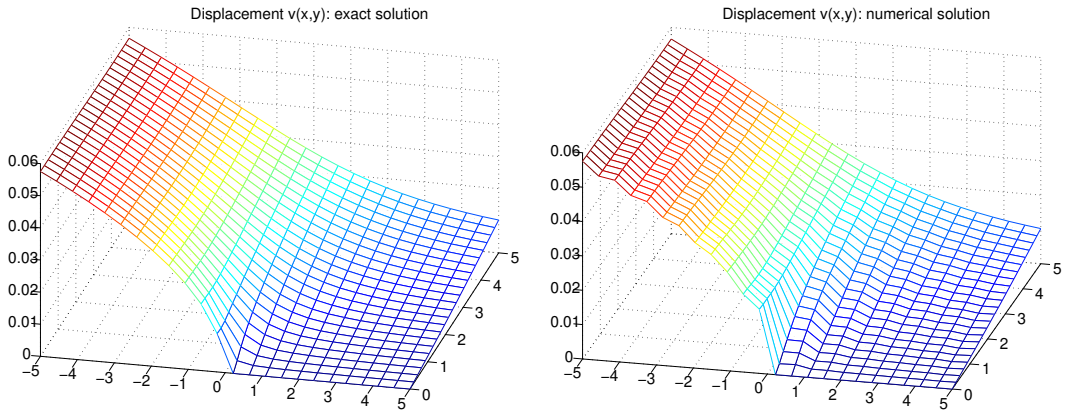


Figure 31: Infinite cracked plate: analyzed portion.



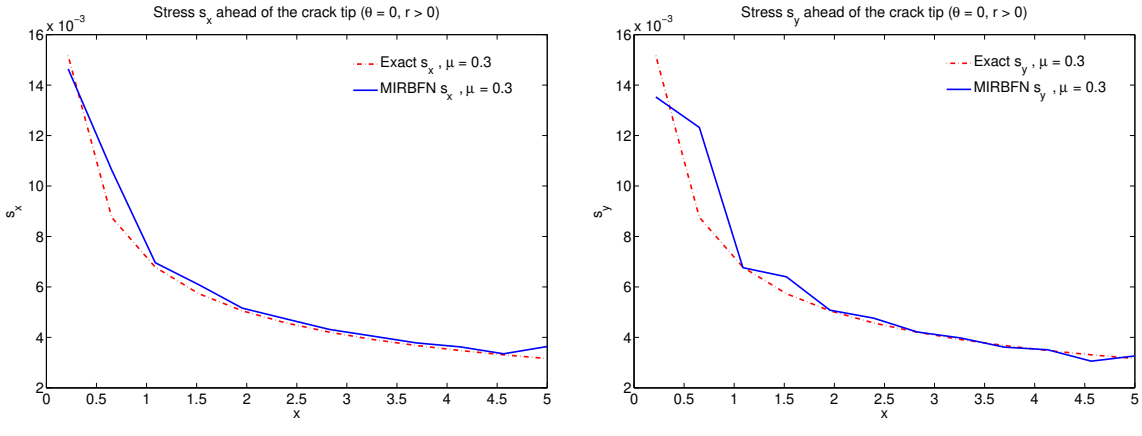
(a) (b)

Figure 32: Infinite cracked plate: u_x obtained by (a) exact solution and (b) MIRBFN method with 24×24 nodes ($\mu = 0.3$).



(a) (b)

Figure 33: Infinite cracked plate: u_y obtained by (a) exact solution and (b) MIRBFN method with 24×24 nodes ($\mu = 0.3$).



(a) (b)

Figure 34: Infinite cracked plate - stress ahead of the crack-tip ($\theta = 0, r > 0$): (a) s_x and (b) s_y obtained by MIRBFN method and exact solutions with 24×24 nodes ($\mu = 0.3$).

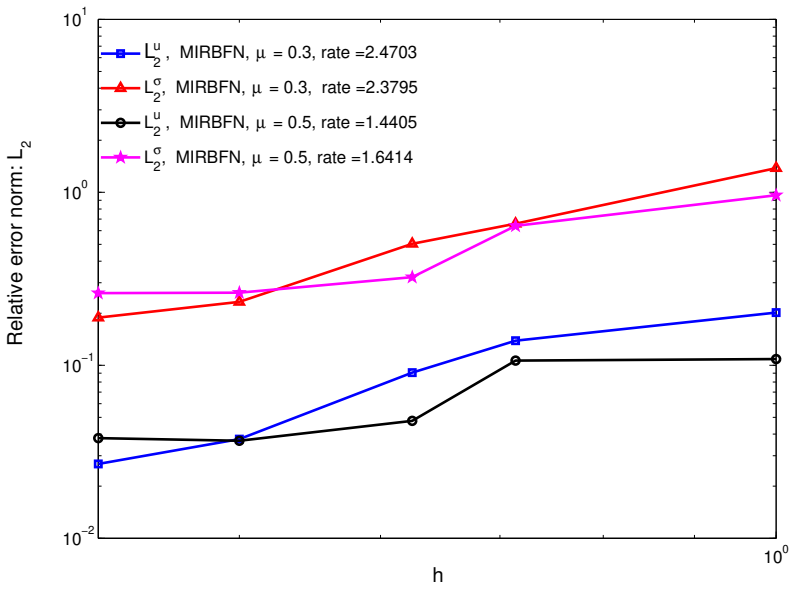


Figure 35: Infinite cracked plate: L_2 relative error norms for displacement and stress, and associated convergence rates for $\mu = 0.3$ and $\mu = 0.5$.