# An effective integrated-RBFN Cartesian-grid discretisation for the stream function-vorticity-temperature formulation in non-rectangular domains

K. Le-Cao, N. Mai-Duy* and T. Tran-Cong

Computational Engineering and Science Research Centre

Faculty of Engineering and Surveying

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

Submitted to *Numerical Heat Transfer, Part B* , 11-Dec-2008; revised, 22-Jan-2009

Short title: 1D-IRBFN Cartesian-grid technique

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

**Abstract** This paper presents a new numerical collocation procedure, based on Cartesian grids and one-dimensional integrated radial-basis-function networks (1D-IRBFNs), for the simulation of natural convection defined in two-dimensional multiply-connected domains and governed by the stream function - vorticity - temperature formulation. Special emphasis is placed on the handling of vorticity values at boundary points that do not coincide with grid nodes. A suitable formula for computing vorticity boundary conditions, which is based on the approximations with respect to one coordinate direction only, is proposed. Normal derivative boundary conditions for the stream function are forced to be satisfied identically. Several test problems, including natural convection in the annulus between square and circular cylinders, are considered to investigate the accuracy of the proposed technique.

Keywords: Integrated radial-basis-function networks; Non-rectangular domains; Cartesian grids; Stream function - vorticity - temperature formulation; Vorticity boundary conditions.

## NOMENCLATURE

| | | | |
|---|---|---|---|
| $CM$ | convergence measure | $R$ | radius of inner circle |
| $D_i$ | diameter of inner cylinder | $Ra$ | Rayleigh number |
| $D_o$ | diameter of outer cylinder | $s$ | arc length |
| $g$ | gravitational acceleration | $t$ | tangential direction/time |
| $g_i(x)$ | radial basis function | $T$ | temperature |
| $h$ | grid size | $u, v$ | $x$ and $y$ components of velocity |
| $H$ | side length of outer square | $w$ | RBF weight |
| $I_i^{(.)}(x)$ | integrated radial basis function | $x, y$ | coordinates |
| $k$ | thermal conductivity | $\alpha$ | thermal diffusivity |
| $\bar{k}_{eq}$ | average equivalent conductivity | $\beta$ | thermal expansion coefficient |
| $L$ | distance between two cylinders | $\nu$ | kinematic viscosity |
| | $(= (D_o - D_i)/2)$ | $\psi$ | stream function |
| $n$ | normal direction | $\omega$ | vorticity |
| $n_{ip}$ | number of interior points | **Subscripts** | |
| $Ne$ | discrete relative $L_2$ error | $b$ | boundary value |
| $Nu$ | average Nusselt number | $e$ | exact solution |
| $Pr$ | Prandtl number | [.] | order of IRBFN |

# 1 Introduction

Natural convection, where the motion of a fluid is caused by the combination of density variations and gravity, can be governed by the coupling of the momentum (velocity field) and energy (temperature field) equations within the Boussinesq approximation. In the momentum equation, the fluid is assumed to have a constant density except for the generation of buoyancy forces and in the energy equation, one neglects the viscous dissipation and compressibility effects. The governing equations can be written in different dependent variables, including the velocity - pressure - temperature, stream function - vorticity - temperature, and stream function - temperature formulations. Each formulation has some strengths and weaknesses from a computational point of view.

With the introduction of the stream-function variable, the pressure variable does not have to be considered, resulting in an easy implementation. However, several issues arise, to which special attention should be paid. For example, in the stream function - temperature approach, one has to cope with fourth-order derivatives and double boundary conditions. Fourth-order systems are known to have higher matrix condition numbers than first- and second-order systems. Errors for approximating higher-order derivatives are generally larger. In the implementation of double boundary conditions, special treatments are required because of two values given at a boundary point. For the stream function - vorticity - temperature formulation, one has to derive computational boundary conditions for the transport vorticity equation. The boundary vorticity values are defined through the Poisson equation, which needs to be solved discretely on the boundaries. The stream function - vorticity - temperature approach requires the approximations for derivatives of order up to 2 (instead of 4), leading to a significant improvement in the matrix condition number over the stream function - temperature approach. This feature is very attractive in dealing with flows with a fine structure as a large number of nodes is usually required for an accurate simulation.

Numerical solutions to these formulations can be achieved by means of discretisation, followed by solutions of the resultant algebraic equations. Results have been reported using different numerical techniques such as finite-difference methods (FDMs) (e.g. [1, 2]), finite-element methods (FEMs) (e.g. [3, 4]), finite-volume methods (FVMs) (e.g. [5, 6]), boundary-element methods (BEMs) (e.g. [7, 8]), meshless methods (e.g. [9]) and spectral methods (e.g. [10, 11]). These methods are based on a finite-element mesh, a finite-volume mesh, a Cartesian grid or a set of unstructured points. When dealing with non-rectangular domains, conventional FDMs and pseudospectral techniques require coordinate transformations to convert the physical domains into rectangular ones (e.g. [12, 13]). The relationships between the physical and computational coordinates are given by a set of algebraic equations or a set of PDEs, depending on the level of complexity of the geometry. These transformation processes are, in general, complicated. It is very desirable that one is able to retain the PDEs in their original form (i.e. in terms of $x$ and $y$ coordinates) and then solve them on a Cartesian grid. Such a numerical solution can be very economical. The use of Cartesian grids for solving problems defined on irregular domains has received much increased attention in recent decades.

Over the last fifteen years, RBFNs, which have the property of universal approximation, have been developed to solve different types of differential problems encountered in applied mathematics, science and engineering (e.g. [14, 15, 16, 17]). RBFN-based methods are extremely easy to implement and capable of achieving a high level of accuracy using a relatively-small number of nodes. One can construct the RBF approximations through differentiation or integration. Since integration is a smoothing operator, the latter has higher approximation power than the former in the handling of derivative functions (e.g. [16, 18, 19, 20]).

In this study, we report a Cartesian-grid-based collocation technique incorporating 1D-IRBFNs on grid lines for the simulation of natural convection in multiply-connected domains. The technique combines strengths of the three approaches, namely 1D-IRBFNs (high-order accuracy), Cartesian grids (easy preprocessing) and the stream function - vorticity - temperature formulation (low-order system). It should be emphasised that conventional RBFN methods lead to fully-populated matrices that tend to become ill-conditioned quickly with increasing numbers of RBFs. Instead of using conventional schemes, 1D-IRBFN approximation schemes [18] are utilised in the present work. Unlike our previous work [19], the stream function - vorticity - temperature formulation is adopted here. We develop a new formula for deriving computational vorticity boundary conditions on a Cartesian grid. First derivatives of the stream function along the boundaries are incorporated into computational vorticity boundary values by means of integration constants. The present IRBFN approximations are constructed to satisfy all boundary conditions identically. The matrix condition number is significantly improved over that produced by the stream function - temperature formulation. Since there are no coordinate transformations required, the present technique works in a similar fashion for different shapes of annuli. Results obtained are compared well with available numerical data in literature.

The remainder of the paper is organised as follows. Section 2 gives a brief review of the governing equations. In Section 3, we describe the proposed technique. Emphasis is placed on a novel formula for handling vorticity boundary conditions at irregular boundary points. Numerical results are presented in Section 4. Section 5 concludes the paper.

## 2 Governing equations

The stream function - vorticity - temperature formulation is used here. The non-dimensional basic equations for natural convection under the Boussinesq approximation in the Cartesian $x - y$ coordinate system can be written as (e.g. [21])

$$\frac{\partial^2\psi}{\partial x^2}+\frac{\partial^2\psi}{\partial y^2}=\omega, \tag{1}$$

$$\frac{\partial\omega}{\partial t}+u\frac{\partial\omega}{\partial x}+v\frac{\partial\omega}{\partial y}=\sqrt{\frac{Pr}{Ra}}\left(\frac{\partial^2\omega}{\partial x^2}+\frac{\partial^2\omega}{\partial y^2}\right)-\frac{\partial T}{\partial x}, \tag{2}$$

$$\frac{\partial T}{\partial t}+u\frac{\partial T}{\partial x}+v\frac{\partial T}{\partial y}=\frac{1}{\sqrt{RaPr}}\left(\frac{\partial^2 T}{\partial x^2}+\frac{\partial^2 T}{\partial y^2}\right), \tag{3}$$

where $\psi$ is the stream function, $\omega$ the vorticity, $T$ the temperature, $t$ the time, $u$ and $v$ the velocity components, and $Pr$ and $Ra$ the Prandtl and Rayleigh numbers defined as $Pr = \nu/\alpha$

and $Ra = \beta g \Delta T L^3/\alpha\nu$, in which $\nu$ is the kinematic viscosity, $\alpha$ the thermal diffusivity, $\beta$ the thermal expansion coefficient and $g$ the gravity, respectively. In this dimensionless scheme, $L$, $\Delta T$ (temperature difference), $U = \sqrt{gL\beta\Delta T}$ and $(L/U)$, are taken as scale factors for length, temperature, velocity and time, respectively. Here, the velocity scale is chosen in such a way that the buoyancy and inertial forces are balanced (e.g. [21]).

The velocity components are defined in terms of the stream function as

$$u = \frac{\partial \psi}{\partial y}, \quad v = -\frac{\partial \psi}{\partial x}.$$

The given velocity boundary conditions, $u$ and $v$, can be transformed into two boundary conditions on the stream function and its normal derivative

$$\psi = \gamma, \quad \frac{\partial \psi}{\partial n} = \xi,$$

where $n$ is the direction normal to the boundary, and $\gamma$ and $\xi$ prescribed functions. In the case of fixed concentric cylinders, non-slip boundary conditions usually lead to $\gamma = 0$ and $\xi = 0$.

# 3 The present technique

The fluid domain is simply embedded in a Cartesian grid. Grid nodes outside the domain are removed from the computations. Boundary points are generated by the intersection of the grid lines and the boundaries of the domain. It can be seen that boundary conditions are over-prescribed for the stream-function equation (1) and under-prescribed for the vorticity equation (2). We use normal derivative boundary conditions for the stream function to derive boundary conditions for the vorticity. For natural-convection problems employed in this study, boundary conditions for the energy equation (3) are the temperature values. Consequently, the three governing equations are all subject to Dirichlet boundary conditions.

On a grid line, 1D-IRBFNs are employed to represent the stream function, vorticity and temperature variables.

## 3.1 1D-IRBFN discretisation

Consider an $x$ grid line. Second-order derivative of the field variable $f$ along a grid line can be decomposed into RBFs

$$\frac{\partial^2 f(x)}{\partial x^2} = \sum_{i=1}^{m} w_i g_i(x) = \sum_{i=1}^{m} w_i I_i^{(2)}(x), \tag{4}$$

where $m$ is the number of RBFs, $\{g_i(x)\}_{i=1}^m \equiv \left\{I_i^{(2)}(x)\right\}_{i=1}^m$ the set of RBFs, $\{w_i\}_{i=1}^m$ the set of weights to be found and $f$ represents $\psi$, $\omega$ and $T$. Approximate expressions for first-order derivative and the field variable are then obtained through integration

$$\frac{\partial f(x)}{\partial x} = \sum_{i=1}^{m} w_i I_i^{(1)}(x) + c_1, \tag{5}$$

$$f(x) = \sum_{i=1}^{m} w_i I_i^{(0)}(x) + c_1 x + c_2, \tag{6}$$

where $I_i^{(1)}(x) = \int I_i^{(2)}(x)dx$ and $I_i^{(0)}(x) = \int I_i^{(1)}(x)dx$.

As shown in Figure 1, a grid line contains two sets of points. The first set consists of $q$ interior points that are also the grid nodes (regular nodes). The function values at the interior points ($\{x_i\}_{i=1}^q$) are unknown. The second set is formed from the two boundary nodes that do not generally coincide with the grid nodes (irregular nodes). At the boundary nodes ($x_{b1}$ and $x_{b2}$), the function values are given (Dirichlet boundary conditions). The boundary conditions are incorporated into the IRBFN approximations through the conversion process of the network-weight space into the physical space. Collocating (6) at the nodal points yields

$$\begin{pmatrix} \widehat{f} \\ \widehat{f}_b \end{pmatrix} = \widehat{\mathcal{I}}^{(0)}_{[2]} \begin{pmatrix} \widehat{w} \\ c_1 \\ c_2 \end{pmatrix}, \tag{7}$$

where

$$\begin{aligned}
\widehat{f} &= \left(f(x_1), f(x_2), \cdots, f(x_q)\right)^T, \\
\widehat{f}_b &= \left(f(x_{b1}), f(x_{b2})\right)^T, \\
\widehat{w} &= \left(w_1, w_2, \cdots, w_m\right)^T, \\
\widehat{\mathcal{I}}^{(0)}_{[2]} &= \begin{bmatrix}
I_1^{(0)}(x_1) & I_2^{(0)}(x_1) & \cdots & I_m^{(0)}(x_1) & x_1 & 1 \\
I_1^{(0)}(x_2) & I_2^{(0)}(x_2) & \cdots & I_m^{(0)}(x_2) & x_2 & 1 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
I_1^{(0)}(x_q) & I_2^{(0)}(x_q) & \cdots & I_m^{(0)}(x_q) & x_q & 1 \\
I_1^{(0)}(x_{b1}) & I_2^{(0)}(x_{b1}) & \cdots & I_m^{(0)}(x_{b1}) & x_{b1} & 1 \\
I_1^{(0)}(x_{b2}) & I_2^{(0)}(x_{b2}) & \cdots & I_m^{(0)}(x_{b2}) & x_{b2} & 1
\end{bmatrix},
\end{aligned}$$

$m = q + 2$ and the subscript [2] indicates that the integral formulation starts with second-order derivatives. Solving (7) for the coefficient vector including the two integration constants results in

$$\begin{pmatrix} \widehat{w} \\ c_1 \\ c_2 \end{pmatrix} = \left(\widehat{\mathcal{I}}^{(0)}_{[2]}\right)^{-1} \begin{pmatrix} \widehat{f} \\ \widehat{f}_b \end{pmatrix}, \tag{8}$$

where $\left(\widehat{\mathcal{I}}^{(0)}_{[2]}\right)^{-1}$ is the generalised inverse.

The values of the first and second derivatives of $f$ with respect to $x$ at the interior points are thus computed in terms of nodal variable values

$$\begin{pmatrix} \frac{\partial f(x_1)}{\partial x} \\ \frac{\partial f(x_2)}{\partial x} \\ \vdots \\ \frac{\partial f(x_q)}{\partial x} \end{pmatrix} = \widehat{\mathcal{I}}^{(1)}_{[2]} \left(\widehat{\mathcal{I}}^{(0)}_{[2]}\right)^{-1} \begin{pmatrix} \widehat{f} \\ \widehat{f}_b \end{pmatrix}, \tag{9}$$

and

$$\begin{pmatrix} \frac{\partial^2 f(x_1)}{\partial x^2} \\ \frac{\partial^2 f(x_2)}{\partial x^2} \\ \vdots \\ \frac{\partial^2 f(x_q)}{\partial x^2} \end{pmatrix} = \widehat{\mathcal{I}}^{(2)}_{[2]} \left(\widehat{\mathcal{I}}^{(0)}_{[2]}\right)^{-1} \begin{pmatrix} \widehat{f} \\ \widehat{f}_b \end{pmatrix}, \tag{10}$$

where

$$\widehat{\mathcal{I}}^{(1)}_{[2]} = \begin{bmatrix} I^{(1)}_1(x_1) & I^{(1)}_2(x_1) & \cdots & I^{(1)}_m(x_1) & 1 & 0 \\ I^{(1)}_1(x_2) & I^{(1)}_2(x_2) & \cdots & I^{(1)}_m(x_2) & 1 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ I^{(1)}_1(x_q) & I^{(1)}_2(x_q) & \cdots & I^{(1)}_m(x_q) & 1 & 0 \end{bmatrix},$$

and

$$\widehat{\mathcal{I}}^{(2)}_{[2]} = \begin{bmatrix} g_1(x_1) & g_2(x_1) & \cdots & g_m(x_1) & 0 & 0 \\ g_1(x_2) & g_2(x_2) & \cdots & g_m(x_2) & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ g_1(x_q) & g_2(x_q) & \cdots & g_m(x_q) & 0 & 0 \end{bmatrix}.$$

Expressions (9) and (10) can be rewritten in compact form

$$\widehat{\frac{\partial f}{\partial x}} = \widehat{\mathcal{D}}_{1x}\widehat{f} + \widehat{k}_{1x}, \tag{11}$$

and

$$\widehat{\frac{\partial^2 f}{\partial x^2}} = \widehat{\mathcal{D}}_{2x}\widehat{f} + \widehat{k}_{2x}, \tag{12}$$

where $\widehat{\mathcal{D}}_{1x}$ and $\widehat{\mathcal{D}}_{2x}$ are the first- and second-order differentiation matrices in the physical space, and $\widehat{k}_{1x}$ and $\widehat{k}_{2x}$ are the known vectors whose components are functions of given boundary conditions.

In the same manner, one can obtain the IRBFN expressions for $\partial f/\partial y$ and $\partial^2 f/\partial y^2$ at the interior points along a vertical grid line.

As with FDMs, FVMs, BEMs and FEMs, the IRBFN approximations will be gathered together to form the global matrices for the discretisation of the PDE.

## 3.2 A new formula for computing vorticity boundary conditions

The values of the vorticity on the boundaries can be computed via

$$\omega_b = \frac{\partial^2 \psi_b}{\partial x^2} + \frac{\partial^2 \psi_b}{\partial y^2}, \tag{13}$$

where the subscript $b$ is used to indicate the boundary quantities. The handling of $\omega_b$ thus involves the evaluation of second-order derivatives of the stream function in both $x$ and $y$ directions.

For regular boundary points (also grid nodes), one can apply (13) directly. The $x$ and $y$ grid lines passing through those points can be used for computing $\partial^2\psi_b/\partial x^2$ and $\partial^2\psi_b/\partial y^2$, respectively. However, in general, the boundary points do not coincide with the grid nodes and hence they lie on either $x$ or $y$ grid lines. Information about $\psi$ is thus given explicitly in one coordinate direction only. A great challenge here is how to compute second derivatives of $\psi$ in (13) with respect to the direction without a grid line. A new formula to overcome this difficulty is proposed as follows.

Consider a curved boundary, along which the values for $\psi$ and $\partial\psi/\partial n$ are prescribed (Figure 2). It can be seen that the values of $\partial\psi/\partial x$ and $\partial\psi/\partial y$ on the boundary can then be obtained in a straightforward manner. Let $s$ be the arc length of the boundary. By introducing an interpolating scheme (e.g. 1D-IRBFNs), one is able to derive derivatives of $\partial\psi/\partial x$ and $\partial\psi/\partial y$ with respect to $s$ such as $\partial^2\psi/\partial x\partial s$ and $\partial^2\psi/\partial y\partial s$.

A tangential derivative of a function $f$ at a boundary point can be computed using the following formula

$$\frac{\partial f}{\partial s} = \frac{\partial f}{\partial x} t_x + \frac{\partial f}{\partial y} t_y, \tag{14}$$

where $t_x$ and $t_y$ are the two $x$ and $y$ components of the unit vector $\widehat{t}$ tangential to the curve ($t_x = \partial x/\partial s$ and $t_y = \partial y/\partial s$).

Replacing $f$ with $\partial \psi_b/\partial x$, one has

$$\frac{\partial^2 \psi_b}{\partial x \partial s} = \frac{\partial^2 \psi_b}{\partial x^2} t_x + \frac{\partial^2 \psi_b}{\partial x \partial y} t_y, \tag{15}$$

or

$$\frac{\partial^2 \psi_b}{\partial x \partial y} = \frac{1}{t_y} \left( \frac{\partial^2 \psi_b}{\partial x \partial s} - \frac{\partial^2 \psi_b}{\partial x^2} t_x \right), \tag{16}$$

where $\partial^2 \psi_b/\partial x \partial s$ is considered as a known quantity.

Similarly, taking $f$ as $\partial \psi_b/\partial y$ results in

$$\frac{\partial^2 \psi_b}{\partial x \partial y} = \frac{1}{t_x} \left( \frac{\partial^2 \psi_b}{\partial y \partial s} - \frac{\partial^2 \psi_b}{\partial y^2} t_y \right), \tag{17}$$

where $\partial^2 \psi_b/\partial y \partial s$ is a known value.

From (16) and (17), one can derive the relationship between $\partial^2 \psi/\partial x^2$ and $\partial^2 \psi/\partial y^2$ at a boundary point

$$\frac{1}{t_y} \left( \frac{\partial^2 \psi_b}{\partial x \partial s} - \frac{\partial^2 \psi_b}{\partial x^2} t_x \right) = \frac{1}{t_x} \left( \frac{\partial^2 \psi_b}{\partial y \partial s} - \frac{\partial^2 \psi_b}{\partial y^2} t_y \right). \tag{18}$$

Consider an $x$ grid line. The interpolating scheme employed along this line does not facilitate the computation of second-order derivative of $\psi$ with respect to the $y$ coordinate. However, such a derivative at a boundary point can be found by using (18)

$$\frac{\partial^2 \psi_b}{\partial y^2} = \left( \frac{t_x}{t_y} \right)^2 \frac{\partial^2 \psi_b}{\partial x^2} + q_y, \tag{19}$$

where $q_y$ is a known quantity defined by

$$q_y = -\frac{t_x}{t_y^2}\frac{\partial^2 \psi_b}{\partial x \partial s} + \frac{1}{t_y}\frac{\partial^2 \psi_b}{\partial y \partial s}. \tag{20}$$

By substituting (19) into (13), a boundary condition for the vorticity at a boundary point on a horizontal grid line will be computed by

$$\omega_b = \left[1 + \left(\frac{t_x}{t_y}\right)^2\right]\frac{\partial^2 \psi_b}{\partial x^2} + q_y, \tag{21}$$

where only the approximations in the $x$ direction are needed.

In the same manner, on a vertical grid line, a boundary condition for the vorticity at a boundary point will be computed by

$$\omega_b = \left[1 + \left(\frac{t_y}{t_x}\right)^2\right]\frac{\partial^2 \psi_b}{\partial y^2} + q_x, \tag{22}$$

where $q_x$ is a known quantity defined by

$$q_x = -\frac{t_y}{t_x^2}\frac{\partial^2 \psi_b}{\partial y \partial s} + \frac{1}{t_x}\frac{\partial^2 \psi_b}{\partial x \partial s}. \tag{23}$$

The boundary conditions for the vorticity are thus written in terms of second derivative of the stream function with respect to $x$ or $y$ only.

## 3.3 Numerical implementation of vorticity boundary conditions

As mentioned earlier, normal derivative boundary conditions for the stream function are used for solving the transport vorticity equation. As a result, the values of $\partial\psi/\partial x$ and $\partial\psi/\partial y$ at the boundary points have to be incorporated into (21) and (22), respectively.

It is well known that the computational vorticity boundary conditions strongly affect the performance of a numerical discretisation scheme. To evaluate the values of a second-order derivative using integrated RBFs only, the 1D-IRBFN scheme of at least second order needs be employed. As shown in [20], higher-order IRBFNs can give more accurate results. In this study, we attempt to employ 1D-IRBFNs of order 2 (Scheme 1) and 4 (Scheme 2), in which second and fourth derivatives of $\psi$ are respectively decomposed into RBFs, to evaluate $\partial^2\psi/\partial x^2$ and $\partial^2\psi/\partial y^2$ in (21) and (22). A distinguishing feature here is that derivative boundary values, $\partial\psi/\partial x$ and $\partial\psi/\partial y$, are incorporated into the IRBFN approximations by means of the constants of integration.

Along an $x$ grid line, the conversion process of the network-weight space into the physical space can be described as follows

$$\widehat{\mathcal{B}}\left(\begin{array}{c}\widehat{w}\\ \widehat{c}\end{array}\right)=\left(\begin{array}{c}\widehat{\psi}\\ \widehat{\psi}_b\\ \widehat{\frac{\partial\psi_b}{\partial x}}\end{array}\right), \qquad (24)$$

where $\widehat{c}$ is a vector of integration constants, $\widehat{w}$ and $\widehat{\psi}$ are defined as before, $\widehat{\psi}_b=(\psi(x_{b1}),\psi(x_{b2}))^T$,

$$\widehat{\frac{\partial\psi_b}{\partial x}}=\left(\frac{\partial\psi(x_{b1})}{\partial x},\frac{\partial\psi(x_{b2})}{\partial x}\right)^T,$$

and

$$\widehat{\mathcal{B}}=\left(\begin{array}{c}\widehat{\mathcal{B}_1}\\ \widehat{\mathcal{B}_2}\end{array}\right),$$

in which

$$\widehat{\mathcal{B}_1}=\widehat{\mathcal{I}}^{(0)}_{[2]},$$

$$\widehat{\mathcal{B}_2}=\left[\begin{array}{cccccc}I_1^{(1)}(x_{b1}) & I_2^{(1)}(x_{b1}) & \cdots & I_m^{(1)}(x_{b1}) & 1 & 0\\ I_1^{(1)}(x_{b2}) & I_2^{(1)}(x_{b2}) & \cdots & I_m^{(1)}(x_{b2}) & 1 & 0\end{array}\right],$$

for Scheme 1, and

$$\widehat{\mathcal{B}_1} = \widehat{\mathcal{I}}^{(0)}_{[4]} = \begin{bmatrix} I_1^{(0)}(x_1) & I_2^{(0)}(x_1) & \cdots & I_m^{(0)}(x_1) & x_1^3/6 & x_1^2/2 & x_1 & 1 \\ I_1^{(0)}(x_2) & I_2^{(0)}(x_2) & \cdots & I_m^{(0)}(x_2) & x_2^3/6 & x_2^2/2 & x_2 & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots \\ I_1^{(0)}(x_m) & I_2^{(0)}(x_m) & \cdots & I_m^{(0)}(x_m) & x_m^3/6 & x_m^2/2 & x_m & 1 \end{bmatrix},$$

$$\widehat{\mathcal{B}_2} = \begin{bmatrix} I_1^{(1)}(x_{b1}) & I_2^{(1)}(x_{b1}) & \cdots & I_m^{(1)}(x_{b1}) & x_{b1}^2/2 & x_{b1} & 1 & 0 \\ I_1^{(1)}(x_{b2}) & I_2^{(1)}(x_{b2}) & \cdots & I_m^{(1)}(x_{b2}) & x_{b2}^2/2 & x_{b2} & 1 & 0 \end{bmatrix},$$

for Scheme 2, where $I_i^{(4)}(x) = g_i(x)$, $I_i^{(3)}(x) = \int I_i^{(4)}(x)$, $\cdots$, $I_i^{(0)}(x) = \int I_i^{(1)}(x)$.

Taking (24) into account, second derivatives of $\psi$ at the two boundary points can be expressed in terms of the values of $\psi$ at every point on the grid line and the values of $\partial\psi/\partial x$ at the two boundary points

$$\widehat{\frac{\partial^2 \psi_b}{\partial x^2}} = \widehat{\mathcal{D}}\widehat{\mathcal{B}}^{-1} \begin{pmatrix} \widehat{\psi} \\ \widehat{\psi_b} \\ \widehat{\frac{\partial \psi_b}{\partial x}} \end{pmatrix}, \tag{25}$$

where

$$\widehat{\mathcal{D}} = \begin{bmatrix} g_1(x_{b1}) & g_2(x_{b1}) & \cdots & g_m(x_{b1}) & 0 & 0 \\ g_1(x_{b2}) & g_2(x_{b2}) & \cdots & g_m(x_{b2}) & 0 & 0 \end{bmatrix}$$

for Scheme 1, and

$$\widehat{\mathcal{D}} = \begin{bmatrix} I_1^{(2)}(x_{b1}) & I_2^{(2)}(x_{b1}) & \cdots & I_m^{(2)}(x_{b1}) & x_{b1} & 1 & 0 & 0 \\ I_1^{(2)}(x_{b2}) & I_2^{(2)}(x_{b2}) & \cdots & I_m^{(2)}(x_{b2}) & x_{b2} & 1 & 0 & 0 \end{bmatrix}$$

for Scheme 2.

It should be emphasised that the IRBFN approximation for $\partial^2\psi/\partial x^2$ satisfies normal derivative of $\psi$ at the two boundary points identically. Substituting (25) into (21), one is able to obtain the boundary conditions for the vorticity equation.

The process of computing the values of the vorticity at the two boundary points on a vertical line is similar to that on a horizontal line.

## 3.4 Solution procedure

The three governing equations must be solved simultaneously to find the values of the temperature, vorticity and stream function at the discrete points within the domain. In this paper, a time marching approach is adopted and at each time step, to minimise the memory requirement, the three equations are solved in a sequential manner. The solution procedure involves the following main steps.

1. Guess the distributions of $T, \omega$ and $\psi$.

2. Discretise the governing equations in time using a first-order accurate finite-difference scheme, where the diffusive and convective terms are treated implicitly and explicitly, respectively.

3. Discretise the governing equations in space using 1D-IRBFNs. Since the differentiation matrices are the same for all variables, the construction process only needs to be carried out for one variable. The system matrices, which involve the IRBFN approximations for the Laplacian operator in the governing equations, stay the same during the iteration process. All equations are subject to Dirichlet boundary conditions.

4. Solve the energy equation (3) for $T$.

5. Derive computational boundary conditions for $\omega$.

6. Solve the vorticity equation (1) for $\omega$.

7. Solve the stream-function equation (2) for $\psi$.

8. Check to see whether the solution has reached a steady state using the following con-

vergence measure ($CM$)

$$CM = \frac{\sqrt{\sum_{i=1}^{n_{ip}} \left(\psi_i^{(k)} - \psi_i^{(k-1)}\right)^2}}{\sqrt{\sum_{i=1}^{n_{ip}} \left(\psi_i^{(k)}\right)^2}} < \epsilon, \tag{26}$$

where $n_{ip}$ is the number of interior points, $k$ the time level and $\epsilon$ the tolerance (in this study, $\epsilon$ is taken to be $10^{-12}$).

9. If it is not satisfied, advance time step and repeat from step 4. Otherwise, stop the computation and output the results.

# 4 Numerical examples

The first two examples, for which analytic solutions are available, are used to verify the vorticity boundary formula and its numerical implementations on both simply- and multiply-connected domains. In the last two examples, the proposed technique is applied for the simulation of natural convection in the region between two concentric cylinders. The thermal boundary conditions are prescribed as $T = 0$ and $T = 1$ along the stationary outer and inner walls, respectively.

The present technique implements the multiquadric (MQ) basis function whose form is

$$g_i(x) = \sqrt{(x - c_i)^2 + a_i^2}, \tag{27}$$

where $c_i$ and $a_i$ are the centre and the width of the $i$th MQ function.

For all numerical examples presented in this study, the problem domain is discretised with a uniform Cartesian grid. The width of the $i$th MQ-RBF, $a_i$, is simply chosen to be the grid spacing $h$ (grid size), and the interior points that fall very close to the boundary (within the distance of $h/8$) are removed from the set of nodal points. In the first two examples, the accuracy of an approximation scheme is measured by means of the discrete relative $L_2$ error defined as

$$Ne = \frac{\sqrt{\sum_{i=1}^{M}(f_i^e - f_i)^2}}{\sqrt{\sum_{i=1}^{M}(f_i^e)^2}}, \tag{28}$$

where $M$ is the number of unknown nodal values of $f$, and $f^e$ and $f$ are the exact and approximate solutions, respectively. Another important measure is the convergence rate of the solution with respect to the refinement of spatial discretisation

$$Ne(h) \approx \gamma h^{\alpha} = O(h^{\alpha}), \tag{29}$$

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

## 4.1 Example 1: Circular shape domain

The present technique is first verified through the solution of the following test problem governed by

$$\frac{\partial^2\psi}{\partial x^2} + \frac{\partial^2\psi}{\partial y^2} = \omega, \tag{30}$$

$$\frac{\partial^2\omega}{\partial x^2} + \frac{\partial^2\omega}{\partial y^2} = b(x,y), \tag{31}$$

on a unit circular domain with the boundary conditions in terms of $\psi$ and $\partial\psi/\partial n$. The exact solution of this problem is taken as

$$\psi_e = \cos(\sqrt{x^2+y^2}), \tag{32}$$

from which the driving function $b(x,y)$ and the boundary conditions can be derived analytically. The problem domain and its typical discretisation are shown in Figure 3.

It can be seen that the two components of the unit vector $\widehat{t}$ tangential to the boundary of the present domain are simply computed by

$$t_x = \frac{-y}{\sqrt{x^2+y^2}}, \tag{33}$$

$$t_y = \frac{x}{\sqrt{x^2+y^2}}. \tag{34}$$

Expressions for the vorticity at the boundary nodes on the $x$ and $y$ grid lines thus reduce to

$$\omega_b = \left[1 + \left(\frac{y}{x}\right)^2\right] \frac{\partial^2 \psi_b}{\partial x^2} + q_y, \tag{35}$$

$$\omega_b = \left[1 + \left(\frac{x}{y}\right)^2\right] \frac{\partial^2 \psi_b}{\partial y^2} + q_x, \tag{36}$$

each of which only requires the approximation of second-order derivative of $\psi$ with respect to one coordinate direction. Both Scheme 1 and Scheme 2 are employed to compute the above expressions.

We conduct two investigations here: (i) the accuracy of Scheme 1 and Scheme 2 for approximating $\partial^2\psi/\partial x^2$ in (35), and (ii) the accuracy of the RBF solution to the PDEs.

For the former, calculations are carried out for various grids from 5 × 5 to 90 × 90. Results of $Ne$ obtained by the two different order interpolating schemes are displayed in Table 1, which show that Scheme 2 gives much more accurate results than Scheme 1. Scheme 2 is thus recommended for use in practice.

For the latter, a number of grids, namely $(12 \times 12, 22 \times 22, \cdots, 62 \times 62)$, are employed to study the convergence behaviour of the solution. Results concerning the condition number of the system matrix, denoted by cond$(A)$, and the error $Ne$ are given in Table 2. The present technique produces system matrices with relatively-low condition numbers. For example, the matrix condition number is only $6.0 \times 10^3$ for a grid of 62 × 62. It can be seen that the choice of Scheme 1 (1D-IRBFN-2s) and Scheme 2 (1D-IRBFN-4s) for computing $\omega_b$ in (35) and (36) has a profound influence on the overall accuracy of the IRBFN solution. The fourth-order boundary scheme outperforms the second-order one regarding both accuracy and convergence rate. The recommended scheme yields a fast rate of convergence, $O(h^{3.1})$.

## 4.2 Example 2: Multiply-connected domain

This test problem is also governed by the two coupled equations (30) and (31) with Dirichlet boundary conditions, $\psi$ and $\partial\psi/\partial n$. The driving function is taken as

$$b(x,y) = 256(\pi^2-1)^2\left[\sin(4\pi x)\cosh(4y)+\cos(4\pi x)\sinh(4y)\right], \tag{37}$$

and the domain of interest is the region lying between a circle of radius 1/2 and a square of dimensions $1/2 \times 1/2$ which are both centered at the origin (Figure 4). The exact solution for this problem is

$$\psi_e = \sin(4\pi x)\cosh(4y) - \cos(4\pi x)\sinh(4y). \tag{38}$$

A number of uniform grids, $(10 \times 10, 20 \times 20, ..., 50 \times 50)$, are considered. Since inner boundaries are parallel to the $x$ and $y$ axes, one can use the original formula for deriving a computational vorticity boundary condition. The stream-function equation can be rewritten as

$$\omega_b = \frac{\partial^2\psi_b}{\partial n^2} + \frac{\partial^2\psi_b}{\partial t^2}, \tag{39}$$

where $\partial^2\psi_b/\partial t^2$ is a known quantity derived from the boundary conditions for $\psi$, and $\partial^2\psi_b/\partial n^2$ can be evaluated using a grid line passing through that point. On the outer boundary, the handling of $\omega_b$ is similar to that in the previous example. For brevity, only the recommended scheme (i.e. Scheme 2) for computing $\omega_b$ is employed here. Results concerning $Ne(\psi)$ are given in Table 3, from which one can also make remarks that are similar to those in the case of simply-connected domains. It can be seen that the present matrix condition numbers are relatively low and the approximate solution converges fast to the exact solution with a rate of $O(h^{3.78})$.

## 4.3 Example 3: Concentric annulus between two circular cylinders

The present method is now applied to the simulation of buoyancy-driven flow in an annulus between two concentric cylinders which are separated by a distance $L$, the inner cylinder heated ($T = 1$) and the outer cylinder cooled ($T = 0$) (Figure 5a). A comprehensive review of this problem can be found in [2]. Most cases have been reported with $Pr = 0.7$ and $L/D_i = 0.8$, in which $D_i$ is the diameter of the inner cylinder. To these conditions, results by Kuehn and Goldstein [2] using FDM for $Ra = 10^2$ to $Ra = 7 \times 10^4$ are often cited in the literature for comparison purposes. Later on, Shu [11], who employed a differential quadrature method (DQM), has provided very accurate solutions for the values of $Ra$ in the range of $10^2$ to $5 \times 10^4$. It is noted that those works required a computational domain be rectangular.

The three governing equations (1), (2), (3) are presently solved with respective to Cartesian coordinates (Figure 5a). Numerical simulations are also conducted for $Pr = 0.7$ and $L/D_i = 0.8$. Like in the case of [2], we consider the values of the Rayleigh number from $Ra = 10^2$ to $Ra = 7 \times 10^4$, which is broader than those reported in [11]. Different grids, namely $(12 \times 12, 22 \times 22, \cdots, 52 \times 52)$, are employed.

The stream function and its normal derivative are set to zero along the inner and outer cylinders. Expressions (35) and (36) derived in Example 1 are applicable here to compute the values of the vorticity on the cylinder walls. Since the boundary values for $\partial\psi/\partial x$ and $\partial\psi/\partial y$ are simply zeros, the terms $q_x$ and $q_y$ vanish. The values for the vorticity at the boundary nodes on the $x$ and $y$ grid lines can thus be computed by

$$\omega_b = [1 + (\frac{y}{x})^2]\frac{\partial^2\psi_b}{\partial x^2}, \tag{40}$$

$$\omega_b = [1 + (\frac{x}{y})^2]\frac{\partial^2\psi_b}{\partial y^2}, \tag{41}$$

respectively. As mentioned earlier, Neumann boundary conditions for the stream function, $\partial\psi/\partial x = 0$ and $\partial\psi/\partial x = 0$, are presently incorporated into the computational vorticity boundary conditions via integration constants and they are satisfied identically.

Table 4 indicates a significant improvement in the matrix condition number of the present formulation over the stream function-temperature formulation reported in [19]. The former yields the matrix condition number several orders of magnitude lower than the latter. This feature is very attractive in the context of RBF techniques. One is thus able to use a larger number of nodes with the present approach in the RBF simulation of fluid flow problems.

Both accuracy and grid convergence of the present technique are investigated.

The solution accuracy is assessed through the average equivalent conductivity defined as (e.g. [2, 11])

$$\bar{k}_{eq} = \frac{-\ln(D_o/D_i)}{2\pi} \oint \frac{\partial T}{\partial n} ds, \tag{42}$$

in which $D_o$ and $D_i$ are the diameters of the outer cylinder and the inner cylinder, respectively. Results concerning $\bar{k}_{eq}$ together with those of Kuehn and Goldstein [2] and of Shu [11] for $Ra = \{10^2, 10^3, 3 \times 10^3, 6 \times 10^3, 10^4, 5 \times 10^4, 7 \times 10^4\}$ are presented in Table 5. It can be seen that there is a good agreement between these numerical solutions.

Figures 6 and 7 show the convergence of the stream function and temperature fields with respect to grid refinement for $Ra = 10^4$ and $Ra = 7 \times 10^4$, respectively. It can be seen that the present technique is able to capture complex structures of the stream function and temperature fields even at coarse grids, and those patterns are improved in quality (smoothness) with increasing grid-densities. At a grid of $42 \times 42$ for $Ra = 10^4$ and a grid of $52 \times 52$ for $Ra = 7 \times 10^4$, the plots look reasonable when compared with those reported in [2, 11].

## 4.4 Example 4: Concentric annulus between a square outer cylinder and a circular inner cylinder

For this example, natural convection between a heated inner circular cylinder and a cooled square enclosure is considered (Figure 5b). It is noted that the transformation of this domain into a rectangular one in other methods where it is required is much more complicated than that in the previous problem.

An aspect ratio of $H/(2R) = 0.25$ ($H$: the side length of the outer square and $R$: the radius of the inner circle), $Pr = 0.71$ and $Ra = \{10^4, 5 \times 10^4, 10^5, 5 \times 10^5, 10^6\}$ are employed here to investigate the accuracy of the technique.

An attractive feature here is that the present technique does not require any coordinate transformations. The problem domain is simply replaced with a Cartesian grid (Figure 5b). Numerical results are obtained for three uniform grids of 32 × 32, 42 × 42, and 52 × 52.

For $Ra = 10^4$, we start to simulate the flow from rest. For higher values of $Ra$, the initial solution is taken as the solution obtained at lower and nearest $Ra$. Figure 8 presents the behaviour of the convergence measure $CM$ against the number of time steps. It can be seen that the decrease in $CM$ is rather monotonic. As expected, the simulation of high-$Ra$ flows requires a larger number of iterations.

The obtained results are presented in Table 6 and Figure 9. Table 6 is concerned with the accuracy of the solution. Following the work of Moukalled and Acharya [12], the local heat transfer coefficient is defined as

$$h = -k\frac{\partial T}{\partial n}, \tag{43}$$

where $k$ is the thermal conductivity. The average Nusselt number (the ratio of the temperature gradient at the wall to a reference temperature gradient) is computed by

$$Nu = \frac{\overline{h}}{k}, \tag{44}$$

where $\overline{h} = -\oint \frac{\partial T}{\partial n} ds$. Since the computational domain in [12] is taken as one-half of the physical domain, the values of $Nu$ in the present work (Table 6) are divided by 2 for comparison purposes. The present results agree well with those in [12] and [13].

Figure 9 displays streamlines and isotherms versus grid densities for $Ra = 10^6$, which shows a very fast convergence of all fields. The qualitative behaviour of these fields and those in [13] is similar.

# 5 Concluding remarks

In this article, a new numerical discretisation scheme for the stream function - vorticity - temperature formulation using Cartesian grids and 1D-IRBFNs is reported. Attractive features of the proposed technique include (i) the preprocessing is simple and (ii) the boundary conditions for the vorticity are implemented in a new and effective manner. Numerical results show that (i) the matrix condition number is significantly improved over the stream function - temperature formulation and (ii) accurate results are obtained using a relatively-coarse grid.

## Acknowledgements

This research is supported by the Australian Research Council. K. Le-Cao wishes to thank the CESRC, FoES and USQ for a postgraduate scholarship.

## References

[1] G. de Vahl Davis, Natural convection of air in a square cavity: a bench mark numerical solution, Int. J. for Numerical Methods in Fluids, vol. 3, pp. 249-264, 1983.

[2] T.H. Kuehn, and R.J. Goldstein, An experimental and theoretical study of natural convection in the annulus between horizontal concentric cylinders, J. of Fluid Mechanics, vol. 74(4), pp. 695-719, 1976.

[3] M.T. Manzari, An explicit finite element algorithm for convection heat transfer problems, Int. J. of Numerical Methods for Heat and Fluid Flow, vol. 9(8), pp. 860-877, 1999.

[4] H. Sammouda, A. Belghith, C. Surry, Finite element simulation of transient natural convection of low-Prandtl-number fluids in heated cavity, Int. J. of Numerical Methods for Heat and Fluid Flow, vol. 9(5), pp. 612-624, 1999.

[5] E.K. Glakpe, C.B. Watkins, and J.N. Cannon, Constant heat flux solutions for natural convection between concentric and eccentric horizontal cylinders, Numerical Heat Transfer - Part B, vol. 10, pp. 279-295, 1986.

[6] D.A. Kaminski, and C. Prakash, Conjugate natural convection in a square enclosure: effect of conduction in one of the vertical walls, Int. J. for Heat and Mass Transfer, vol. 29(12), pp. 1979-1988, 1986.

[7] M. Hribersek, and L. Skerget, Fast boundary-domain integral algorithm for the computation of incompressible fluid flow problems, Int. J. for Numerical Methods in Fluids, vol. 31, pp. 891-907, 1999.

[8] K. Kitagawa, L.C. Wrobel, C.A. Brebbia, and M. Tanaka, A boundary element formulation for natural convection problems, Int. J. for Numerical Methods in Fluids, vol. 8, pp. 139-149, 1988.

[9] G. Kosec, and B. Sarler, Solution of thermo-fluid problems by collocation with local pressure correction, Int. J. of Numerical Methods for Heat & Fluid Flow, vol. 18 (7/8), pp. 868-882, 2008.

[10] P. Le Quere, Accurate solutions to the square thermally driven cavity at high Rayleigh number, Computers & Fluids, vol. 20(1), pp. 29-41, 1991.

[11] C. Shu, Application of differential quadrature method to simulate natural convection in a concentric annulus, Int. J. for Numerical Methods in Fluids, vol. 30, pp. 977-993, 1999.

[12] F. Moukalled, and S. Acharya, Natural convection in the annulus between concentric horizontal circular and square cylinders, J. of Thermophysics and Heat Transfer, vol. 10(3), pp. 524-531, 1996.

[13] C. Shu, and Y.D. Zhu, Efficient computation of natural convection in a concentric annulus between an outer square cylinder and an inner circular cylinder, Int. J. for Numerical Methods in Fluids, vol. 38, pp. 429-445, 2002.

[14] G.E. Fasshauer, Solving partial differential equations by collocation with radial basis functions, Surface Fitting and Multiresolution Methods (editors: A. Le Mehaute, C.

Rabut, and L.L. Schumaker), pp. 131-138, Vanderbilt University Press, Nashville, TN, 1997.

[15] E.J. Kansa, Multiquadrics- A scattered data approximation scheme with applications to computational fluid-dynamics-II. Solutions to parabolic, hyperbolic and elliptic partial differential equations, Computers and Mathematics with Applications, vol. 19(8/9), pp. 147-161, 1990.

[16] N. Mai-Duy, and T. Tran-Cong, Approximation of function and its derivatives using radial basis function networks, Applied Mathematical Modelling, vol. 27, pp. 197-220, 2003.

[17] B. Sarler, A radial basis function collocation approach in computational fluid dynamics, Computer Modeling in Engineering & Sciences, vol. 7(2), pp. 185-194, 2005.

[18] N. Mai-Duy, and T. Tran-Cong, A Cartesian-grid collocation method based on radial-basis-function networks for solving PDEs in irregular domains, Numerical Methods for Partial Differential Equations, vol. 23(5), pp. 1192-1210, 2007.

[19] N. Mai-Duy, K. Le-Cao, and T. Tran-Cong, A Cartesian grid technique based on one-dimensional integrated radial basis function networks for natural convection in conccentric annuli, Int. J for Numerical Methods in fluids, vol. 57, pp. 1709-1730, 2008.

[20] N. Mai-Duy, Solving high order ordinary differential equations with radial basis function networks, Int. J. for Numerical Methods in Engineering, vol. 62, pp. 824-852, 2005.

[21] S. Ostrach, Natural convection in enclosures, J. of Heat Transfer, vol. 110, pp. 1175-1190, 1988.

| Grid | $Ne$ | |
|---|---|---|
| | Scheme 1 | Scheme 2 |
| $5 \times 5$ | 3.5(-2) | 4.0(-3) |
| $10 \times 10$ | 3.7(-2) | 5.3(-4) |
| $30 \times 30$ | 6.8(-2) | 1.1(-4) |
| $50 \times 50$ | 9.6(-2) | 9.5(-5) |
| $70 \times 70$ | 6.9(-2) | 1.9(-5) |
| $90 \times 90$ | 7.3(-2) | 1.3(-5) |

Table 1: Example 1: Errors by 1D-IRBFN-2s (Scheme 1) and 1D-IRBFN-4s (Scheme 2) in the computation of second derivatives of $\psi$ at the boundary points. It is noted that $a(b)$ represents $a \times 10^b$.

| Grid | $Ne(\psi)$ | | cond$(A)$ |
|---|---|---|---|
| | Scheme 1 | Scheme 2 | |
| $12 \times 12$ | 1.5(-3) | 6.5(-5) | 8.9(1) |
| $22 \times 22$ | 9.4(-4) | 1.1(-5) | 3.6(2) |
| $32 \times 32$ | 7.4(-4) | 3.9(-6) | 8.0(2) |
| $42 \times 42$ | 6.4(-4) | 1.4(-6) | 2.4(3) |
| $52 \times 52$ | 6.4(-4) | 6.4(-7) | 3.3(3) |
| $62 \times 62$ | 4.5(-4) | 3.1(-7) | 6.0(3) |

Table 2: Example 1: Overall accuracy of the solution $\psi$ by the present technique employed with two different computational vorticity boundary schemes, namely Scheme 1 (1D-IRBFN-2s) and Scheme 2 (1D-IRBFN-4s). The condition numbers of the IRBFN system matrix are also included.

| Grid | cond($A$) | $Ne$ |
|---|---|---|
| $10 \times 10$ | 1.2(1) | 2.6(−2) |
| $20 \times 20$ | 4.5(1) | 9.4(−4) |
| $30 \times 30$ | 1.0(2) | 2.0(−4) |
| $40 \times 40$ | 1.8(2) | 8.5(−5) |
| $44 \times 44$ | 2.5(2) | 6.0(−5) |
| $50 \times 50$ | 2.9(2) | 4.2(−5) |

Table 3: Example 2: Condition numbers of the system matrix and relative $L_2$ errors of the solution.

| Grid | cond($A$) | |
|---|---|---|
| | $\psi-\omega-T$ | $\psi-T$ |
| $12\times12$ | 1.7(1) | 8.4(1) |
| $22\times22$ | 9.1(1) | 1.6(3) |
| $32\times32$ | 2.4(2) | 6.9(3) |
| $42\times42$ | 6.7(2) | 1.0(5) |
| $52\times52$ | 9.3(2) | 2.1(5) |
| $62\times62$ | 1.7(3) | 1.2(6) |

Table 4: Example 3 (circular cylinders): Condition numbers of the 1D-IRBFN system matrix by the two formulations.

| $Ra$ | $10^2$ | $10^3$ | $3 \times 10^3$ | $6 \times 10^3$ | $10^4$ | $5 \times 10^4$ | $7 \times 10^4$ |
|---|---|---|---|---|---|---|---|
| | | | | $k_{eqi}$ | | | |
| Present Method | 1.000 | 1.083 | 1.396 | 1.709 | 1.975 | 2.962 | 3.207 |
| FDM [2] | 1.000 | 1.081 | 1.404 | 1.736 | 2.010 | 3.024 | 3.308 |
| DQM [11] | 1.001 | 1.082 | 1.397 | 1.715 | 1.979 | 2.958 | |
| | | | | $k_{eqo}$ | | | |
| Present Method | 0.999 | 1.080 | 1.393 | 1.712 | 1.970 | 2.942 | 3.246 |
| FDM [2] | 1.002 | 1.084 | 1.402 | 1.735 | 2.005 | 2.973 | 3.226 |
| DQM [11] | 1.001 | 1.082 | 1.397 | 1.715 | 1.979 | 2.958 | |

Table 5: Example 3 (circular cylinders): Comparison of the average equivalent conductivity on the outer and inner cylinders, $k_{eqi}$ and $k_{eqo}$, between the present IRBFN technique using a grid of $52 \times 52$ and some other techniques for $Ra$ in the range of $10^2$ to $7 \times 10^4$.

| $Ra$ | $10^4$ | $5 \times 10^4$ | $10^5$ | $5 \times 10^5$ | $10^6$ |
|---|---|---|---|---|---|
| | | | $Nu_o$ | | |
| Present Method | 3.22 | 4.04 | 4.89 | 7.43 | 8.70 |
| FDM [12] | 3.24 | | 4.86 | | 8.90 |
| DQM [13] | 3.33 | | 5.08 | | 9.37 |
| | | | $Nu_i$ | | |
| Present Method | 3.21 | 4.04 | 4.89 | 7.51 | 8.85 |
| FDM [12] | 3.24 | | 4.86 | | 8.90 |
| DQM [13] | 3.33 | | 5.08 | | 9.37 |

Table 6: Example 4 (square-circular cylinders): Comparison of the average Nusselt number on the outer and inner cylinders, $Nu_o$ and $Nu_i$, for $Ra$ from $10^4$ to $10^6$ between the present technique (grid 52 × 52) and some other techniques.

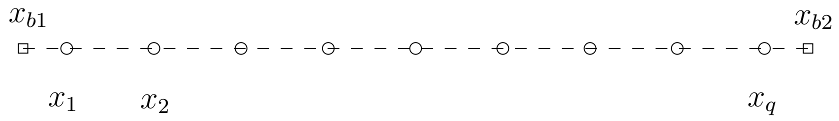


Figure 1: Points on a grid line consist of interior points $x_i$ ($\circ$) and boundary points $x_{bi}$ ($\Box$).

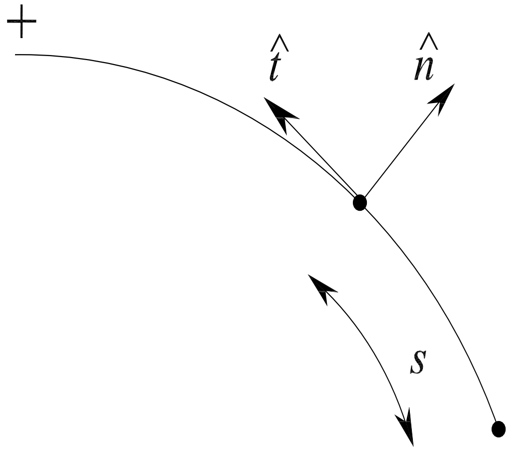


Figure 2: Curved boundary.

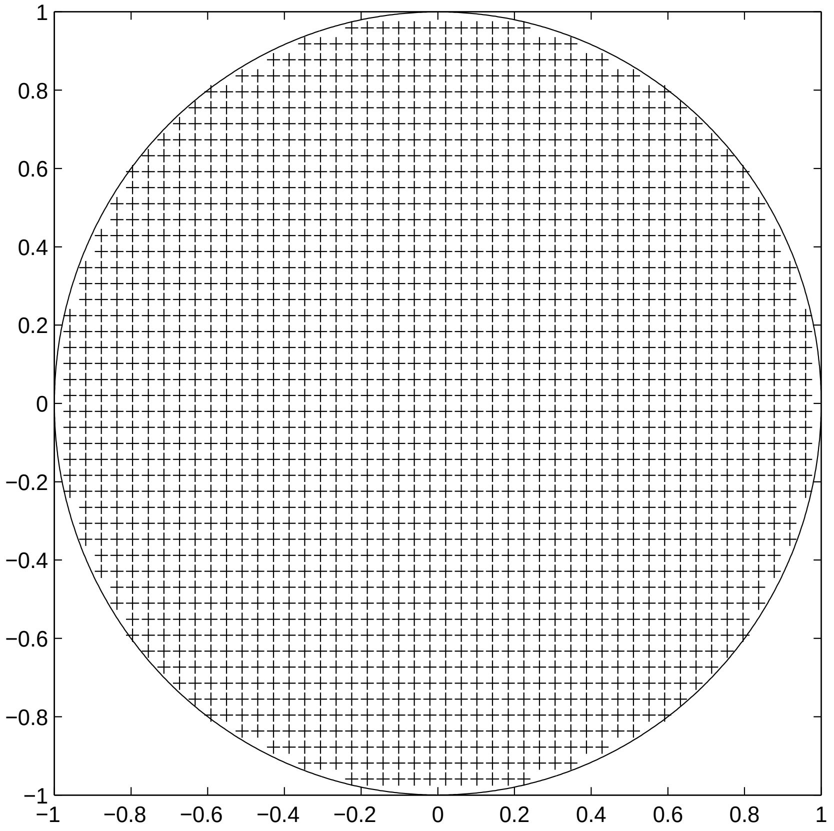


Figure 3: Example 1: Domain of interest and its typical discretisation. It is noted that the nodes outside the domain are removed.

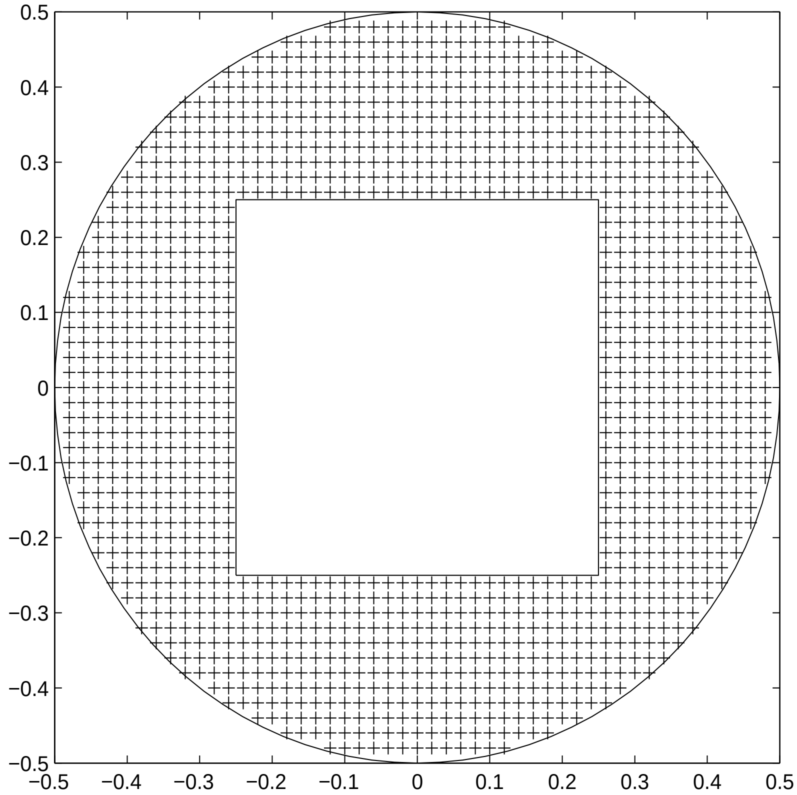


Figure 4: Example 2: Multiple-connected domain and its typical discretisation. It is noted that the nodes outside the domain are removed.

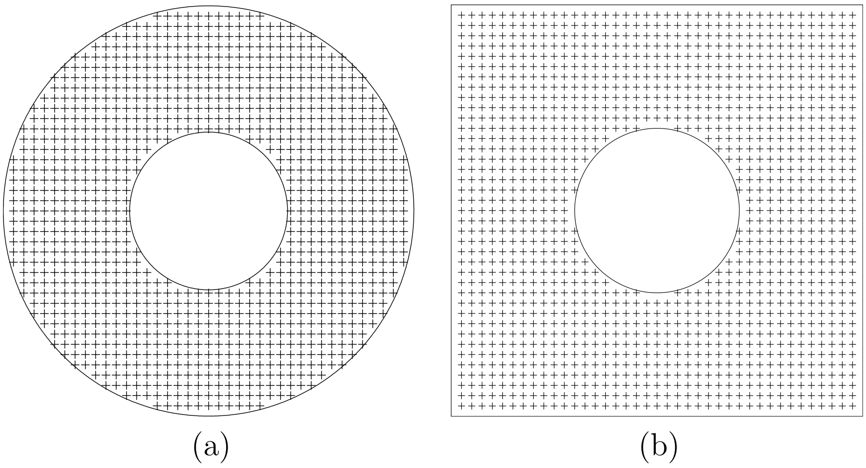


Figure 5: Computational domains and discretisations: Annulus between two circular cylinders (a) and annulus between inner circular cylinder and outer square cylinder (b).

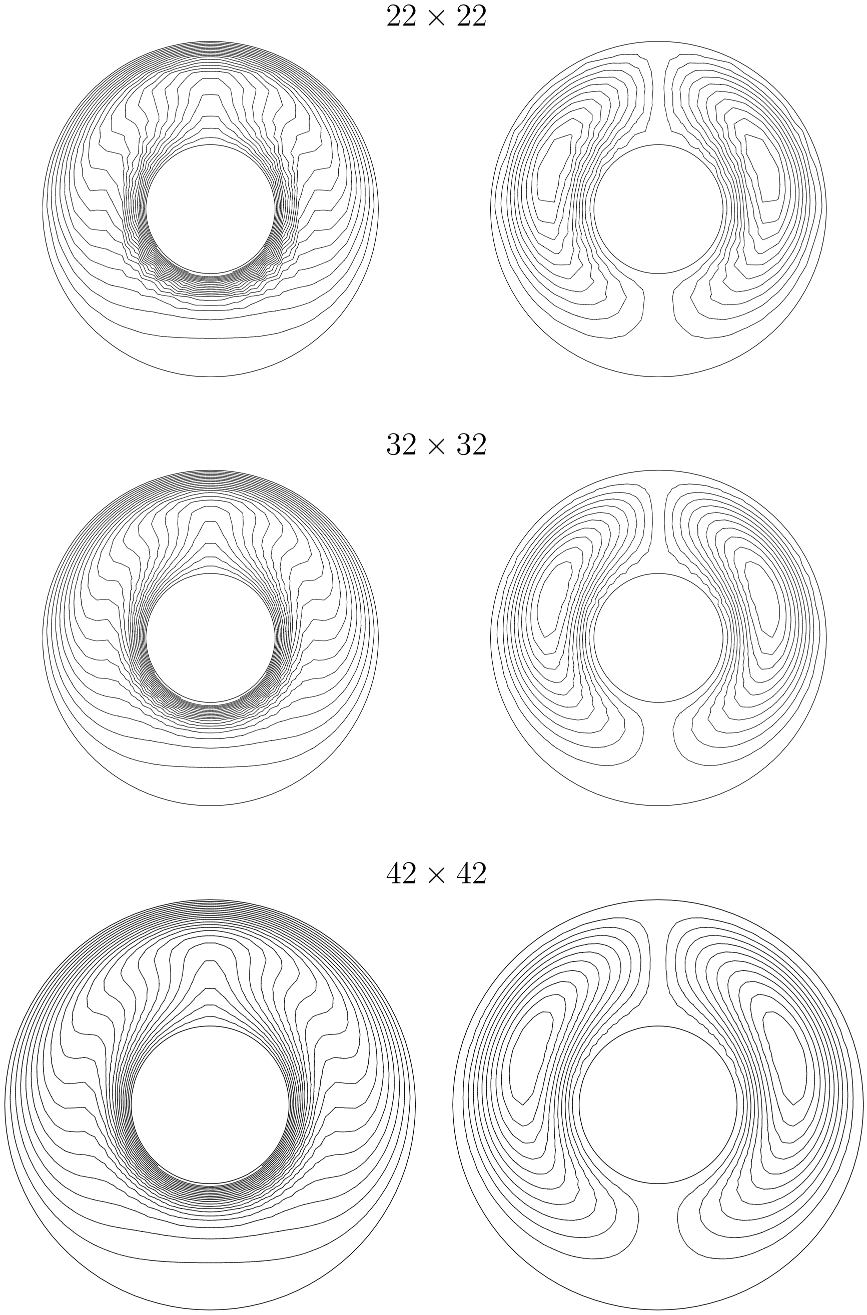


Figure 6: Circular-circular cylinders: Convergence of the temperature (left) and stream function (right) fields with respect to grid refinement for the flow at $Ra = 10^4$. Each plot contains 21 contour lines whose levels vary linearly from the minimum to maximum values.

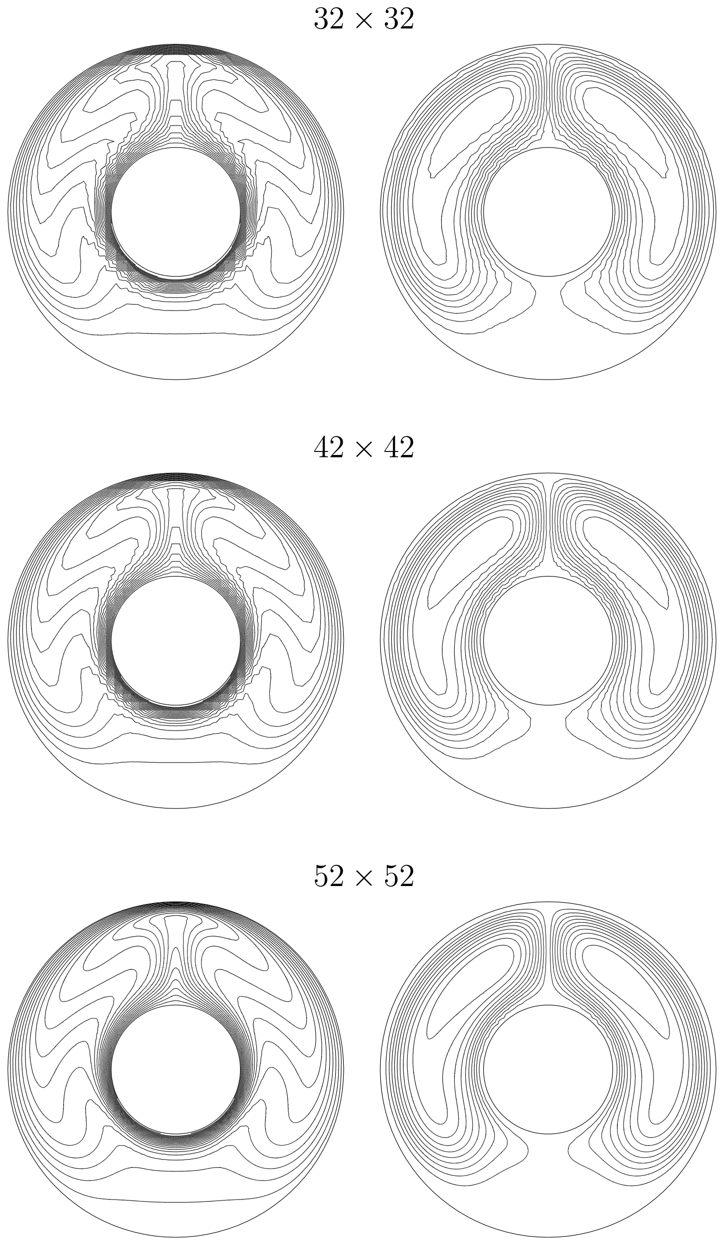


Figure 7: Circular-circular cylinders: Convergence of the temperature (left) and stream function (right) fields with respect to grid refinement for the flow at $Ra = 7 \times 10^4$. Each plot contains 21 contour lines whose levels vary linearly from the minimum to maximum values.

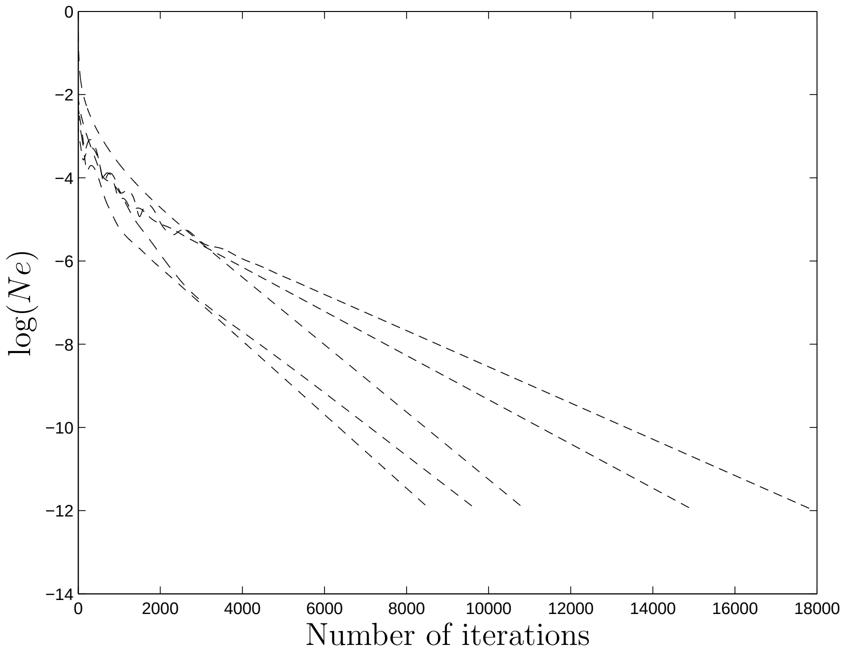
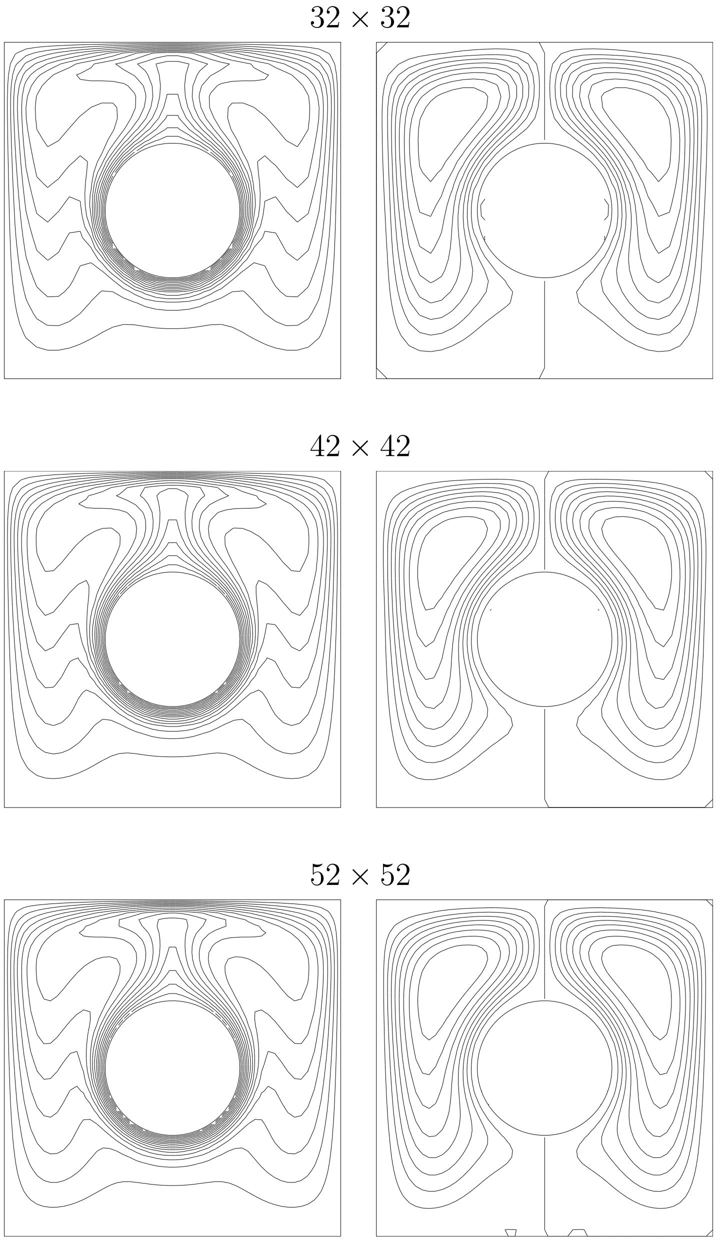


Figure 8: Iterative convergence. Time steps used are 0.002 for $Ra = 10^4$, 0.005 for $Ra = 5 \times 10^4$, and 0.008 for $Ra = \{10^5, 5 \times 10^5, 10^6\}$. The values of CM become less than $10^{-12}$ when the numbers of iterations reach 10925, 9740, 8609, 15017, and 17938 for $Ra = \{10^4, 5 \times 10^4, 10^5, 5 \times 10^5, 10^6\}$, respectively.



32 × 32

42 × 42

52 × 52

Figure 9: Square-circular cylinders: Convergence of the temperature (left) and stream function (right) fields with respect to grid refinement for the flow at $Ra = 10^6$.