An Internal-Node Adaptation Scheme Applied with the Dual Reciprocity Boundary Element Method

Titikan Moonsan, Krittidej Chanthawara, and Sayan Kaennakham

***Abstract*— This paper proposes a node-adaptive algorithm that aims to enhance the effectiveness of the dual reciprocity boundary element method (DRBEM) for numerically solving PDEs. The adaptation algorithm allows the internal nodes to automatically adapt accordingly to a pre-defined criterion during the computing process. The multiquadric radial basis function (MQ-RBF) is used to link the supports from internal nodes to the boundary ones. The proposed adaptation scheme is driven by the local change in velocity (in both x- and y-directions) using a form of normalized error indicator. The node-adaptation manner falls into the *h*-type of refinement where nodes are automatically added into (or removed from) the computational domain. It is found that the numerical solutions obtained from the proposed adaptation scheme are noticeably improved for all cases under the investigation. This promising aspect certainly provides a numerical tool for better applications of DRBEM toward more complex problems.**

***Index Terms*— Node-Adaptation Scheme, Dual Reciprocity Boundary Element Method (DRBEM), coupled-Burgers' equations, Convection-Diffusion.**

I. Introduction

IN addition to those well-known traditional numerical methods; finite difference method (FDM), finite element method (FEM), and finite volume method (FVM), in 1978, Brebbia [1] proposed an alternative method known as 'Boundary element method (BEM)' and it's been drawing the attention of both scientists and engineers ever since. Unlike FDM or FEM, in BEM the original linear operations are replaced with a set of integral equations defined only on the boundary. This procedure results in the reduction of the problem's dimension, remaining the main feature of BEM over other domain numerical schemes. In addition, the method is usually associated with direct formulations in which the problem unknowns are the physical variables, potential, or fluxes, making it even more attractive [2].

Despite several merits, one can obtain, difficulties and drawbacks have occurred when dealing with more complicated and more complex problems particularly those involving non-linear, non-homogeneous, and time-dependent phenomena. Even though a particular solution can be found for some forms of the equations given, making it possible to transform the domain integrals to boundary integrals, the difficulties are found in the implementation resulted from the storage of the closed-form [3], [4]. Over the past decade, many auxiliary methods have been invented and developed to strengthen their effectiveness [5], [6]. One of which is that known as the Dual Reciprocity BEM (or DRBEM) [7] where the whole process is divided into two parts: complementary solutions of its homogeneous form and the particular solutions of the inhomogeneous counterpart, by discretizing the domain into a series of internal nodes done via. the use of the so-called 'Radial Basis Function (RBF)' [8]-[10] (recent successful uses of RBFs in other contexts can be found in [11] and [12]). More applications of DRBEM can be found in a wide range of science and engineering fields; elastodynamics [10], [13], infinite domain scenario [9], free and forced vibrations [14], convection-diffusion with mild Peclet number [15], dynamic non-linear analysis [16], structural dynamics analysis [17] (see also the references therein).

When it comes to applying a numerical scheme for solving PDEs, it is always desirable to achieve as high accuracy as possible while requiring as little computational effort as possible. To achieve this, one seeks ways to optimize the solution quality by managing the computational nodes involved and this is where the techniques of node/grid adaptation come in [18]. Toward this direction, recent works include radial basis function equidistribution-based adaptive algorithm designed for transient and nearly singular PDEs [19], the adaptation scheme driven by strong-form formulation and residual-based error [20], the one based on detection of highly localized features using residual subsampling technique [21], the new iterative scheme for implicit difference equation using stencil FD [22], the distributed adaptive node-specific signal estimation (DANSE) algorithm applying an additional relaxation in the updating process in fully connected sensor networks [23], and the node-adaptation based on local gradient under the context of a global collocation meshless method [24].

The methods of allowing computational nodes to dynamically adapt accordingly to pre-defined criteria normally consist of two main ingredients; the adaptation manner and the area-of-interest identification means. Based on the way an algorithm adds or removes nodes, the methodology of mesh adaptation can generally be classified into four main manners as follows [25];

Manuscript received January 13, 2021; revised September 15, 2021.

T. Moonsan is a Lecturer at the Department of Mathematics, Faculty of Science, Udon Thani Rajabhat University, Udon Thani, 41000, Thailand (e-mail: titikan.mo@udru.ac.th).

K. Chanthawara is a Lecturer at the Programme of Mathematics, Faculty of Science, Ubon Ratchathani Rajabhat University, Mueang Ubon Ratchathani, Ubon Ratchathani 34000, Thailand (e-mail: krittidej.c@ubru.ac.th).

S. Kaennakham is an Associate Professor at the School of Mathematics, Institute of Science, Suranaree University of Technology, Nakhon Ratchasima 30000, Thailand (corresponding author to provide phone: 0066 (0) 44 22 4747; fax: 0066 (0) 44 22 4185; sayan_kk@g.sut.ac.th).

- ***h-refinement***: in this manner nodes can be included in (or removed from) the computational domain during the adaptation procedure [25], [26].
- ***r-refinement***: this manner keeps the number of computation nodes intact and relocates the target nodes toward the areas of interest [27].
- ***p-refinement***: this technique focuses on improving the accuracy by using higher order term of approximation polynomial while keeping the number and position of nodes fixed [28], [29], [30].
- ***m-refinement:*** the computational mesh is completely rebuilt regarding the error indicator introduced from the discretization step [31], [32].

Over the past decade, it is interesting to see that there is only a small number of research on node adaptation schemes applied with the boundary element method (BEM) when compared to those based on conventional numerical methods; FE, FV, and FD. The reason being is obvious that the attractive nature of the method could well be destroyed if more internal nodes are playing a more important role in the calculation process. Some recent attempts to improve BEM by the means of boundary-node adaptation include the technique based on local error analysis [33], the error estimated on a piecewise finite element of degree '*k*' in 1D [34], and the rather comprehensive review nicely done by Michael Feischl *et.al.* [35] (see also references therein). As well-known to have lost its ability when dealing with problems involving strong inhomogeneous, non-linear, and transient nature with strong instabilities and sharp gradient of variable phenomena [36], DRBEM is now being enhanced by supports benefitted from internal nodes via the means of adaptation schemes. In this work, we propose an automatic internal-node adaptation scheme aiming to handle problems with challenging phenomena in mechanics. For this purpose, two well-known benchmarking problems are focused on in this work; the convection-diffusion, and the coupled-Burgers equations.

The layout of the paper is as follows; the mathematical construction of DRBEM is provided in Section 2 before its implementations to the two benchmarking test cases are explained in Section 3. The proposed internal-node adaptation scheme is then proposed in Section 4. Numerical experiments and solutions obtained are illustrated and discussed in Section 5 before some main findings and conclusions are drawn in Section 6.

II. THE DUAL RECIPROCITY BOUNDARY ELEMENT METHOD (DRBEM)

Starting from the Poisson equation as follows;

$$\nabla^2 u = b(x,y) \tag{1}$$

Its equivalent integral form, given by [1] , is expressed as;

$$c_i u_i + \int_\Gamma q^* u d\Gamma - \int_\Gamma u^* q d\Gamma = \sum_{j=1}^{N+L} \alpha_j \left(c_i \hat{u}_{ij} + \int_\Gamma q^* \hat{u}_j d\Gamma - \int_\Gamma u^* \hat{q}_j d\Gamma \right) \tag{2}$$

Where u^* is the fundamental solution and the term, $\hat{q}_j$ is defined as $\hat{q}_j = \dfrac{\partial \hat{u}_j}{\partial \mathbf{n}}$, where $\mathbf{n}$ is the unit vector outward normal to the boundary Γ , and can be written as follows;

$$\hat{q}_j = \frac{\partial \hat{u}_j}{\partial x}\frac{\partial x}{\partial n} + \frac{\partial \hat{u}_j}{\partial y}\frac{\partial y}{\partial n} \tag{3}$$

Next, we apply the boundary element method as explained in [2], [37], and with N and L being the number of boundary and internal nodes respectively, b can be now approximated using a linear combination form by;

$$b_i(x,y) \approx \sum_{j=1}^{N+L} \alpha_j f_{ij}(x,y) \tag{4}$$

Here, the function f is the radial basis function (RBF), commonly defined as a multivariate function whose values are dependent only on the distance. There are many forms proposed and applied over the past decades (see [38]) and in this work, one of the most popular choices proposed by Hardy (1971) [39], known as *'Multiquadric (MQ) RBF'* is employed throughout and it is defined as;

$$f(r,\varepsilon) = \sqrt{\varepsilon_{MQ}^2 + r^2} \tag{5}$$

where ε_{MQ} is the so-called 'shape parameter' and is known to play a crucial role in determining the quality of the final results and has always been an open topic for decades. $r_j = \left\| \mathbf{x} - \mathbf{x}^j \right\|_2$ is the Euclidean distance expressed in $n-$ dimensional space as;

$$r_j = \left\| \mathbf{x} - \mathbf{x}^j \right\|_2 = \sqrt{(x_1 - x_1^j)^2 + (x_2 - x_2^j)^2 + \ldots + (x_n - x_n^j)^2} \tag{6}$$

With this radial basis function, we then have;

$$\nabla^2 \hat{u}_j = f_j \tag{7}$$

For some particular solution, $\hat{u}_j$. Applying Green's theorem, the boundary element approximation to (2) then it becomes, at a node i^{th} ;

$$\begin{aligned} c_i u_i + \sum_{k=1}^{N} \int_{\Gamma_k} q^* u d\Gamma - \sum_{k=1}^{N} \int_{\Gamma_k} u^* q d\Gamma \\ = \sum_{j=1}^{N+L} \alpha_j \left(c_i \hat{u}_{ij} + \sum_{k=1}^{N} \int_{\Gamma_k} q^* \hat{u}_j d\Gamma - \sum_{k=1}^{N} \int_{\Gamma_k} u^* \hat{q}_j d\Gamma \right) \end{aligned} \tag{8}$$

For $i = 1, \ldots, \mathrm{N}$. After introducing the interpolation function and integrating over each boundary element, (8) can be re-written in terms of nodal values as;

$$\begin{aligned} c_i u_i + \sum_{k=1}^{N} H_{ik} u_k - \sum_{k=1}^{N} G_{ik} q_k \\ = \sum_{j=1}^{N+L} \alpha_j \left(c_i \hat{u}_{ij} + \sum_{k=1}^{N} H_{ik} \hat{u}_{kj} - \sum_{k=1}^{N} G_{ik} \hat{q}_{kj} \right) \end{aligned} \tag{9}$$

Where the definition of the terms H_{ik} and G_{ik} can be found in Toutip [2]. The index k is used for the boundary nodes which are the field points. After application to all boundary nodes, using a collocation technique, (9) can be compactly expressed in matrix forms as follows;

$$\mathbf{Hu} - \mathbf{Gq} = \left(\mathbf{H\hat{U}} - \mathbf{G\hat{Q}} \right) \mathbf{F}^{-1} \mathbf{b} \tag{10}$$

By setting $\mathbf{S} = \left(\mathbf{H\hat{U}} - \mathbf{G\hat{Q}} \right) \mathbf{F}^{-1}$, the above equation can be rewritten as;

$$\mathbf{Hu} - \mathbf{Gq} = \mathbf{Sb} \tag{11}$$

By applying boundary condition(s), then the following final form can be reached;

$$\mathbf{Ax} = \mathbf{y} \tag{12}$$

After this equation system is solved using standard techniques such as Gaussian elimination, the values at any internal node can be calculated from the following equation (where each one involving a separate multiplication of known vectors and matrices).

$$\begin{aligned} u_i &= -\sum_{k=1}^{N} H_{ik} u_k + \sum_{k=1}^{N} G_{ik} q_k \\ &= \sum_{j=1}^{N+L} \alpha_j \left(c_i \hat{u}_{ij} + \sum_{k=1}^{N} H_{ik} \hat{u}_{kj} - \sum_{k=1}^{N} G_{ik} \hat{q}_{kj} \right) \end{aligned} \tag{13}$$

This form is to be implemented correspondingly to the governing equations of the problems at hand and shall be detailed in the following section.

III. DRBEM IMPLEMENTATIONS

3.1 A Convection-Diffusion Problem

The first challenging case is the two-dimensional convection-diffusion problem governed by the following equation;

$$\frac{\partial u}{\partial t} + V_x \frac{\partial u}{\partial x} + V_y \frac{\partial u}{\partial y} = \omega_x \frac{\partial^2 u}{\partial x^2} + \omega_y \frac{\partial^2 u}{\partial y^2} - \beta u + g(x,y,t) \tag{14}$$

Where V_x, V_y are convection coefficients, and ω_x, ω_y are diffusion coefficients. The last two terms βu and the source term $g(x,y,t)$ are additional and needed only in specific cases. By setting $\omega_x = \omega_y = \omega$, we obtain;

$$\frac{\partial u}{\partial t} + V_x \frac{\partial u}{\partial x} + V_y \frac{\partial u}{\partial y} + \beta u - g(x) = \omega\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) \tag{15}$$

Leading to;

$$\frac{1}{\omega}\left(\frac{\partial u}{\partial t} + \left(V_x \frac{\partial u}{\partial x} + V_y \frac{\partial u}{\partial y} \right) + \beta u - g(x) \right) = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \tag{16}$$

with the initial condition $u(x,y,0) = \beta(x,y)$, with $(x,y) \in \Gamma$ and the boundary condition $u(x,y,t) = \gamma(x,y,t)$. Where $x,y \in \partial\Gamma$, $t > 0$ and Γ is a domain of the problem, $\partial\Gamma$ is its boundary, β and γ are known functions. The terms $V_x \frac{\partial u}{\partial x}$ and $V_y \frac{\partial u}{\partial y}$ are approximated similarly by;

$$\left.\begin{aligned} V_x \frac{\partial u}{\partial x} &= \mathbf{V_x} \frac{\partial \mathrm{F}}{\partial x} \mathbf{F}^{-1} \mathbf{u} \\ V_y \frac{\partial u}{\partial y} &= \mathbf{V_y} \frac{\partial \mathrm{F}}{\partial y} \mathbf{F}^{-1} \mathbf{u} \end{aligned}\right\} \tag{17}$$

From (11), the form below is defined.

$$b = \frac{1}{\omega}\left(\frac{\partial u}{\partial t} + \left(V_x \frac{\partial u}{\partial x} + V_y \frac{\partial u}{\partial y} \right) + \beta u - g(x) \right) \tag{18}$$

Let $\dot{u} = \partial u / \partial t$ and by substituting (17) and (18) in (11) the following matrix form is reached.

$$\begin{aligned} &\mathbf{Hu} - \mathbf{Gq} \\ &= \mathbf{S}\left(\frac{1}{\omega}\left(\dot{u} + \left(\mathbf{V_x} \frac{\partial \mathrm{F}}{\partial x} \mathbf{F}^{-1}\mathbf{u} + \mathbf{V_y} \frac{\partial \mathrm{F}}{\partial y} \mathbf{F}^{-1}\mathbf{u} \right) + \beta \mathbf{u} - \mathbf{g}(\mathbf{x}) \right) \right) \end{aligned} \tag{19}$$

And then,

$$\begin{aligned} &\mathbf{Hu} - \mathbf{Gq} \\ &= \frac{1}{\omega}\mathbf{S}\left(\dot{u} + \left(\mathbf{V_x} \frac{\partial \mathrm{F}}{\partial x} \mathbf{F}^{-1} + \mathbf{V_y} \frac{\partial \mathrm{F}}{\partial y} \mathbf{F}^{-1} + \beta \mathbf{I} \right) \mathbf{u} - \mathbf{g}(\mathbf{x}) \right) \end{aligned} \tag{20}$$

Let

$$\mathbf{C} = \mathbf{V_x} \frac{\partial \mathrm{F}}{\partial x} \mathbf{F}^{-1} + \mathbf{V_y} \frac{\partial \mathrm{F}}{\partial y} \mathbf{F}^{-1} + \beta \mathbf{I} \tag{21}$$

and by substituting (21) into (20), the following expression is obtained.

$$\mathbf{Hu} - \mathbf{Gq} = \frac{1}{\omega}\mathbf{S}\left(\dot{u} + \mathbf{Cu} - \mathbf{g}(\mathbf{x}) \right) \tag{22}$$

Let $\mathbf{R} = \frac{1}{\omega}\mathbf{S}$, so it becomes.

$$\mathbf{Hu} - \mathbf{Gq} = \mathbf{R}\left(\dot{u} + \mathbf{Cu} - \mathbf{g}(\mathbf{x}) \right) \tag{23}$$

For the time derivative, the forward difference method is used and is expressed as;

$$\frac{\partial u}{\partial t} = \dot{u} = \frac{u^{t+1} - u^t}{\Delta t} \tag{24}$$

Therefore, the final form of the latest equation is expressed as follows.

$$\left(\frac{\mathbf{R}}{\Delta t} + \mathbf{RC} - \mathbf{H} \right)\mathbf{u}^{t+1} + \mathbf{Gq}^{t+1} = \frac{\mathbf{R}}{\mathbf{\Delta t}}\mathbf{u}^t + \mathbf{Rg}(\mathbf{x}) \tag{25}$$

Note that the elements of matrices $\mathbf{H}, \mathbf{G}, \mathbf{R}$ and $\mathbf{g}(\mathbf{x})$ depend only on geometrical data. Thus, they can all be computed once and stored.

3.2 A Nonlinear Transient and Couple- Problem

The second challenging mechanic problem is the famous two-dimensional Burgers' Equations expressed as.

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = \frac{1}{Re}\left(\nabla^2 u \right) \tag{26}$$

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} = \frac{1}{Re}\left(\nabla^2 v \right) \tag{27}$$

These are subject to the initial conditions: $u(x,y,0) = \beta_1(x,y)$, $v(x,y,0) = \beta_2(x,y)$, and the boundary conditions: $u(x,y,t) = \gamma_1(x,y,t)$, and $v(x,y,t) = \gamma_2(x,y,t)$. Where $x,y \in \partial\Gamma$, $t > 0$ and $u(x,y,t)$, and $v(x,y,t)$ are the velocity components to be determined, $\beta_1, \beta_2, \gamma_1$ and γ_2 are known functions and Re is the Reynolds number, described in [40].

The implementation process begins with approximating the terms; $u\frac{\partial u}{\partial x}$, $u\frac{\partial u}{\partial y}$, $v\frac{\partial u}{\partial x}$, and $v\frac{\partial u}{\partial y}$ via the following manners;

$$\left.\begin{aligned} u\frac{\partial u}{\partial x} &= \mathbf{U}\frac{\partial \mathbf{F}}{\partial x}\mathbf{F}^{-1}\mathbf{u} \\ v\frac{\partial u}{\partial y} &= \mathbf{V}\frac{\partial \mathbf{F}}{\partial y}\mathbf{F}^{-1}\mathbf{u} \\ u\frac{\partial v}{\partial x} &= \mathbf{U}\frac{\partial \mathbf{F}}{\partial x}\mathbf{F}^{-1}\mathbf{v} \\ v\frac{\partial u}{\partial y} &= \mathbf{V}\frac{\partial \mathbf{F}}{\partial y}\mathbf{F}^{-1}\mathbf{v} \end{aligned}\right\} \tag{28}$$

By inserting all the above forms in the governing equations and hence, (10) can be rearranged for each of the equations as follows;

$$\mathbf{Hu} - \mathbf{Gq} = \left(\mathbf{H\hat{U}} - \mathbf{G\hat{Q}} \right)\mathbf{F}^{-1}\mathbf{b_1} \tag{29}$$

and

$$\mathbf{Hv} - \mathbf{Gz} = \left(\mathbf{H\hat{V}} - \mathbf{G\hat{Z}} \right)\mathbf{F}^{-1}\mathbf{b_2} \tag{30}$$

where $q = \frac{\partial u}{\partial n}, z = \frac{\partial v}{\partial n}$ and

$$b_1 = Re\left(\frac{\partial u}{\partial t}+u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}\right) \quad (31)$$

$$b_2 = Re\left(\frac{\partial v}{\partial t}+u\frac{\partial v}{\partial x}+v\frac{\partial v}{\partial y}\right) \quad (32)$$

Since $\hat{\mathbf{U}},\hat{\mathbf{V}}$ are generated by using the same redial basis function (MQ-RBF), then;

$$\hat{\mathbf{U}}=\hat{\mathbf{V}} \text{ and } \hat{\mathbf{Q}}=\hat{\mathbf{Z}} \quad (33)$$

For the time derivatives, the forward difference method is employed; $\frac{\partial u}{\partial t}=\frac{\left(u^{t+1}-u^{t}\right)}{\Delta t}$ and $\frac{\partial v}{\partial t}=\frac{\left(v^{t+1}-v^{t}\right)}{\Delta t}$. Setting (34) and (35), expressed as.

$$\mathbf{A}=\left(\mathbf{H}\hat{\mathbf{U}}-\mathbf{G}\hat{\mathbf{Q}}\right)\mathbf{F}^{-1} \quad (34)$$

$$\mathbf{C}= \mathbf{U}\frac{\partial \mathbf{F}}{\partial \mathbf{x}}\mathbf{F}^{-1}+\mathbf{V}\frac{\partial \mathbf{F}}{\partial \mathbf{x}}\mathbf{F}^{-1} \quad (35)$$

Then the final forms of DRBEM for this type of equation can now be finalized as follows;

$$\left(\left(Re\right)\mathbf{AC}+\frac{\mathbf{S}\left(Re\right)}{\Delta t}-\mathbf{H}\right)\mathbf{u}^{t+1}+\mathbf{G}\mathbf{q}^{t+1}=\frac{\left(Re\right)\mathbf{S}}{\Delta t}\mathbf{u}^{t} \quad (36)$$

$$\left(\left(Re\right)\mathbf{AC}+\frac{\mathbf{S}\left(Re\right)}{\Delta t}-\mathbf{H}\right)\mathbf{v}^{t+1}+\mathbf{G}\mathbf{z}^{t+1}=\frac{\left(Re\right)\mathbf{S}}{\Delta t}\mathbf{v}^{t} \quad (37)$$

Note that the elements of matrices $\mathbf{H},\mathbf{G}$ and $\mathbf{A}$ depend only on geometrical data. Thus, they can all be computed once and stored.

IV. THE INTERNAL-NODE ADAPTATION SCHEME

In this section, an *h*-type nodes-adaptive procedure is introduced. The structure was proposed and applied in our previous study [24] and it is now being further modified for tackling problems in conjunction with DRBEM. The internal nodes are automatically added to or removed from the computational domain subjecting to a pre-defined criterion. The proposed scheme mainly consists of three components;

- The means to perform node-adaptive: *Node Adaptive Manner*
- The means to indicate the area of interest: *Error Indicator*
- The steps to execute: *The Adaptation Algorithm*

Each of which is explained below.

4.1 Node Adaptation Manner

Node-Refining: When a node is marked and subjected to a refinement process, it will then generate four child nodes around itself, as depicted in Fig. 1 Assuming node $\mathbf{x}_c=(x_i,y_j)$ is marked for refinement, then its four child nodes, denoted as $\mathbf{x}_c^{(1)}$, $\mathbf{x}_c^{(2)}$, $\mathbf{x}_c^{(3)}$, $\mathbf{x}_c^{(4)}$, are defined as follows;

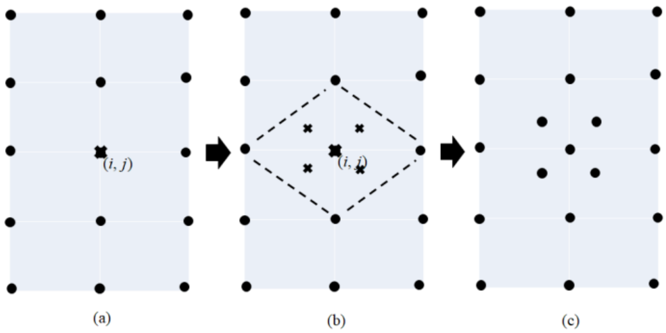


Fig. 1. Internal Node Adaptation Manner

$$\left.\begin{aligned}
\mathbf{x}_c^{(1)} &= \left(\frac{x_{i-1}+x_i+x_i}{3},\frac{y_j+y_j+y_{j+1}}{3}\right),\\
\mathbf{x}_c^{(2)} &= \left(\frac{x_i+x_{i+1}+x_i}{3},\frac{y_j+y_j+y_{j+1}}{3}\right),\\
\mathbf{x}_c^{(3)} &= \left(\frac{x_{i-1}+x_i+x_i}{3},\frac{y_j+y_j+y_{j-1}}{3}\right),\\
\mathbf{x}_c^{(4)} &= \left(\frac{x_i+x_{i+1}+x_i}{3},\frac{y_j+y_j+y_{j-1}}{3}\right)
\end{aligned}\right\} \quad (38)$$

Node-Coarsening: The original or parents nodes are kept untouched but the child ones closest to their parents will be removed.

4.2 The Proposed Error Indicator

To identify the areas where the numerical solutions can possibly be improved based upon, this work proposes a tool that focuses on the local change in a pre-chosen flow variable in both x- and y- directions. This is carried out simply by using the central-finite differences expressed as follows.

$$\left(\tilde{U}_{(i,j)}\right)_{xx}=\frac{\tilde{U}_{i+1,j}-2\tilde{U}_{ij}+\tilde{U}_{i-1,j}}{h_x^{\,2}}, \quad (39)$$

$$\left(\tilde{U}_{(i,j)}\right)_{yy}=\frac{\tilde{U}_{i,j+1}-2\tilde{U}_{ij}+\tilde{U}_{i,j-1}}{h_y^{\,2}} \quad (40)$$

For $i,j=1,2,...,L$ and where $h_x=\left|x_{i+1}-x_i\right|$ and $h_y=\left|y_{i+1}-y_i\right|$. Then the proposed local error indicator, denoted by $\xi^{(k)}$, can be defined for a k^{th}-node as follows;

$$\xi^{(k)}=\sqrt{\frac{1}{2}\left[\left(\tilde{U}^{(k)}\right)_{yy}^{2}+\left(\tilde{U}^{(k)}\right)_{xx}^{2}\right]} \quad (41)$$

Note that (41) is to be applied only to all internal nodes whereas the boundary ones are remained intact throughout the computation process.

Once $\xi^{(k)}=\xi_{(i,j)}$ is calculated for each internal node, this means of normalizing the values obtained from the previous step is crucial as it prevents the proposed adaptation scheme from generating extreme changes of the raw values during the calculation. This is also to fit the re-adjustment of the refining thresholds. For this purpose, the following normalization form is defined and used in the work.

$$\Theta_{(i,j)}=\frac{\xi_{(i,j)}}{\max\,\{\xi_{(m,n)}\}} \quad (42)$$

For $i,i,m,n=1,2,...N$. It is obvious that $0\le\Theta_{(i,j)}\le 1$, and with a pre-defined lower threshold, θ_{low} and upper threshold, θ_{up}, where $0<\theta_{low}<\theta_{up}<1$, the computational domain is divided into 3 sub-areas;

- *Refinement area:* nodes with $\Theta_{(i,j)} > \theta_{up}$
- *Coarsen area:* nodes with $\Theta_{(i,j)} < \theta_{low}$
- *Intact area:* nodes with $\theta_{low} \le \Theta_{(i,j)} \le \theta_{up}$

4.3 The Adaptation Algorithm

The following steps are how the adaptation process proceeds;

Step 1: Uniformly distribute nodes over the computational domain.

Step 2: Construct all the matrices involved and perform DRBEM until reaching time, or iteration for steady case, step $(t-1)^{th}$.

Step 3: Perform the node adaptation scheme by;

3.1) Specifying the lower (θ_{low}) and upper thresholds (θ_{up}), such that $0<\theta_{low}<\theta_{up}<1$. (This is totally up to a user's judgment).

3.2) Computing error indicator, $\xi^{(k)}$ using (39-41).

3.3) Normalizing $\xi^{(k)}$ obtained from Step 3.2 by using (42), getting $\Theta_{(i,j)}$.

3.4) Performing the adaptation process by comparing $\Theta_{(i,j)}$ of each node to the thresholds $(\theta_{low},\theta_{up})$ determined from Step 3.1 and it is done in the manner explained in Fig. 1.

3.5) Interpolating the computed solutions obtained at time, or iteration, $(t-2)^{th}$ on all newly generated nodes.

Step 4: Construct all the matrices using information residing on all nodes involved and perform DRBEM for time, or iteration, $(t-1)^{th}$ and $(t)^{th}$ step respectively.

V. NUMERICAL EXPERIMENTS AND RESULTS

This section provides the results obtained from numerical experiments carried out covering many cases. The multiquadric type of RBF, (5), was used throughout the investigation and an optimal shape parameter (ε_{MQ}) was reached by carrying out a large number of experiments. For the sake of simplicity when being referred therein, the abbreviations used are given as follows;

- *InN*: The number of initial/parent nodes.
- *FnN*: The number of final nodes after applying the node-adaptive algorithm.
- *BdN*: The number of boundary nodes.
- *Adp-DRBEM*: DRBEM with the node-adaptive algorithm.
- *DRBEM*: DRBEM without the node-adaptive algorithm.
- *CPU-Tm*: Computational CPU-time spent.

The proposed node-adaptation algorithm presented in this work was now applied in conjunction with DRBEM and the following three benchmarking test cases were tackled numerically;

- The Poisson equation with a nonrectangular domain.
- A convective-dominated phenomenon in steady-state.
- The highly viscous nonlinear, unsteady, and coupled-Burgers' equations.

The result validation process was carried out by comparing the numerical solutions (both with and without the node-adaptation scheme), against both the corresponding exact solutions and those previously presented in literature if available. To achieve this, several forms of error measurement norms were involved and they are listed in Table. I. All numerical investigations presented in the work were carried out on a computer laptop with Intel(R) Core (TM) i7-2620M CPU @ 2.70 GHz and RAM 4.00 GB.

5.1 Poisson Equation with Nonrectangular Domain

In the first test case, the Poisson equation, (43), was numerically solved by DRBEM with and without the use of the proposed node-adaptation algorithm.

$$\nabla^2 u=\left(\frac{\partial^2}{\partial x^2}+\frac{\partial^2}{\partial y^2}\right)u=-x^2 \tag{43}$$

This is defined on the domain with an elliptical boundary expressed as;

$$\frac{x^2}{4}+y^2=1 \tag{44}$$

The boundary condition is taken directly from the exact solution expressed as follows;

$$u(x,y)=-\frac{1}{246}\left(50x^2-8y^2+33.6\right)\left(\frac{x^4}{4}+y^2-1\right) \tag{45}$$

The investigation began with running simulations with keeping the number of the boundary nodes the same in both cases (i.e. *BdN*=40), for the internal nodes to play the main role in determining the final solution quality. For this, two internal node densities were considered; $InN=68$ and 91 and the main findings are shown in Table. II. Under the same adaptation criteria; $(\theta_{low},\theta_{up})=(0.10,0.30)$ and ε_{MQ} = 5.50, both error norms, L_{RMS} and L_∞, revealed the same trends where more internal nodes were found to lead to a noticeable improvement. In particular, the use of the adaptation scheme proposed in this work was clearly seen to yield a promising reduction in errors for both initial node densities. The best result measured by $L_{RMS}=0.210E-04$ was found when using two levels of refinement with

TABLE I
ERROR NORMS ADOPTED IN THIS WORK $(\hat{N}=N+L)$

Error Norm	Noted by	Mathematical Formula
Maximum	L_∞	$\max\limits_{1\le i\le \hat{N}} \left\lvert u^{ext.}(\mathbf{x}_i)-u^{appx.}(\mathbf{x}_i)\right\rvert JN$
Root-Mean-Square	L_{RMS}	$\sqrt{\frac{1}{\hat{N}}\sum_{j=1}^{\hat{N}}\left(u^{ext.}(\mathbf{x}_i)-u^{appx.}(\mathbf{x}_i)\right)^2}$
Absolute	L_{Abs}	$\left\lvert u^{ext.}(\mathbf{x}_i)-u^{appx.}(\mathbf{x}_i)\right\rvert$, $1\le i\le \hat{N}$
Relative	L_{Rlv}	$\left\lvert\frac{u^{ext.}(\mathbf{x}_i)-u^{appx.}(\mathbf{x}_i)}{u^{ext.}(\mathbf{x}_i)}\right\rvert$, $1\le i\le \hat{N}$
Root-Relative-Square	L_{RRS}	$\sqrt{\frac{\sum_{i=1}^{N}\left(u^{ext.}(\mathbf{x}_i)-u^{appx.}(\mathbf{x}_i)\right)^2}{\sum_{i=1}^{N}\left(u^{ext.}(\mathbf{x}_i)\right)^2}}$

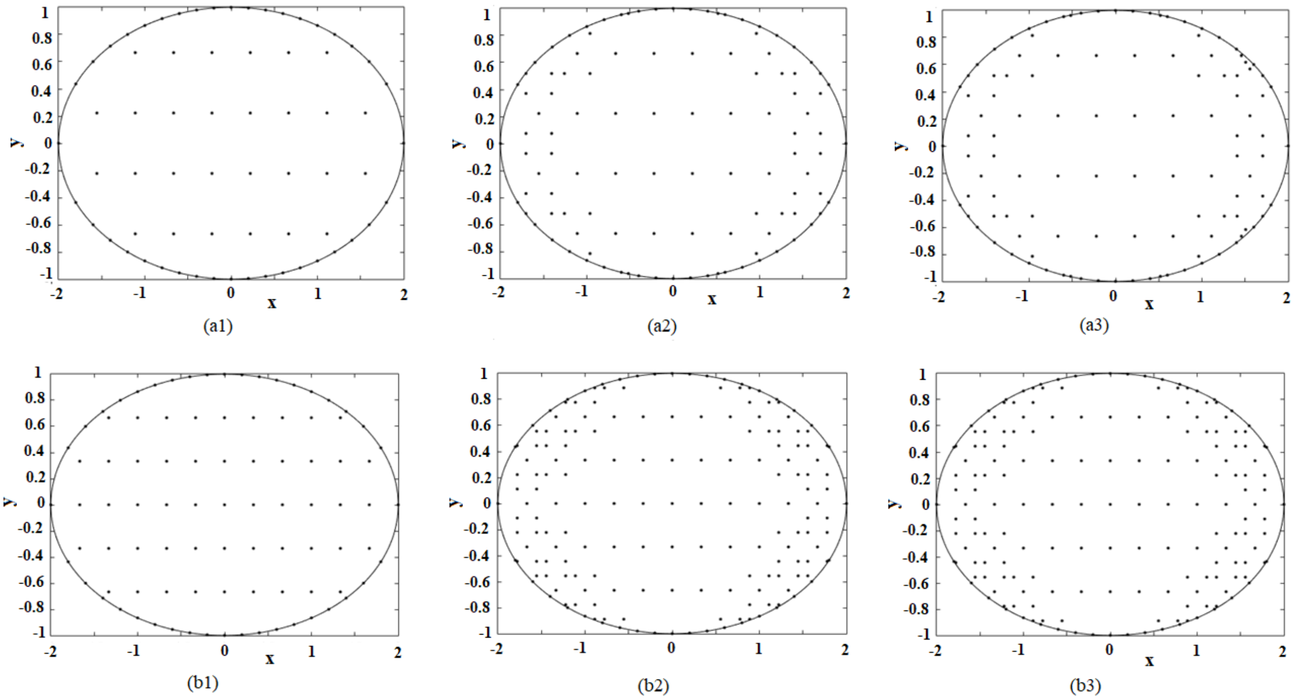


Fig. 2. Node distribution after undergoing the adaptation algorithm with 1- and 2- level of refinement where (a-) starting with $InN = 68$, and (b-) starting with $InN = 91$.

$InN = 91$. Node distribution obtained under this first experiment is displayed in Fig. 2.

5.2 Steady Convection-Dominated Problem

Convection-diffusion is known as one of the most challenging phenomena for numerical mathematics research areas. The most important factor under this kind of problem is the so-called 'Peclet number', a measure of the relative importance of advection versus diffusion. A large Peclet number indicates an advectively dominated distribution. It determines the stability of the numerical solution and has drawn great attention from researchers in the areas. Over decades, many attempts to numerically approximate the solutions to the problems have been made (see [41],[42]). Amongst those, in 2006, Gu and Liu [43] studied four numerical techniques for tackling the instability issues; the enlargement of the upwind support domain, the local support domain, the nodal refinement, the adaptive upwind support domain, the adaptive analysis, and the biased support domain, their work has been recognized as a 'benchmarking case' ever since.

In this work, a 2D convection-diffusion problem in steady-state, as given and studied in Gu and Liu [43] was under investigation. The governing equation is expressed as follows;

TABLE II

ERRORS OBTAINED FROM BOTH CASES; WITH AND WITHOUT ADAPTATION ALGORITHM, COMPUTED WITH *BDN*=40, $(\theta_{low}, \theta_{up}) = (0.10, 0.30)$ AND $\varepsilon_{MQ} = 5.50$.

Error Norm	$InN = 68$			$InN = 91$		
	Adp-DRBEM		*DRBEM*	*Adp-DRBEM*		*DRBEM*
	1-level (*FnN*=108)	2-level (*FnN*=113)		1-level (*FnN*=175)	2-level (*FnN*=175)	
L_{RMS}	0.511E-04	0.380E-04	0.807E-04	0.322E-04	0.210E-04	0.642E-04
L_{∞}	1.125E-04	1.094E-04	1.514E-04	0.815E-04	0.718E-04	1.214E-04

TABLE III

L_{RRS} – ERROR OBTAINED AT DIFFERENT DIFFUSION COEFFICIENTS WITH $InN = 441$ AND $(\theta_{low}, \theta_{up}) = (0.35, 0.50)$.

γ	Gu&Liu [43]	CS [37]	NAA [44]	CNAM [24]	This work		
					ε_{MQ}	*DRBEM*	*Adp-DRBEM*
100	0.245	-	0.435	0.0175	0.448	0.45159	0.06691
10	0.255	0.138	0.371	0.0071	0.901	0.31501	0.06845
1	0.346	0.215	0.589	0.0088	1.525	0.74110	0.08612
0.1	1.276	3.521	38.307	0.8271	8.769	11.0552	2.09499
0.05	-	-	-	1.4389	14.20	64.3320	1.53109
0.01	15.832	73.363	1970.006	6.9517	22.15	107.0244	19.0049
0.001	195.345	-	-	9.5576	42.45	1054.2218	132.9733
5.00E-04	-	-	-	10.0669	54.59	2820.5574	210.3124
1.00E-04	-	-	-	-	67.25	7017.2258	326.4100
5.00E-05	-	-	-	-	82.15	10,227.4802	466.4531

$$L(u)=\mathbf{v}^T\cdot\nabla u-\nabla^T(\mathbf{D}\nabla u)+\beta u-q(\mathbf{x})=0 \quad (46)$$

Defined for $(x,y)\in\Omega=[0,1]\times[0,1]$, and the coefficients $\mathbf{D}=\begin{bmatrix}\gamma & 0\\ 0 & \gamma\end{bmatrix}$, $\mathbf{v}=\{3-x,4-y\}$, and $\beta=1$ in which γ is a given constant of diffusion coefficient (The smaller this number, the more convective-dominated the phenomenon becomes). The boundary conditions were set with $u(x,y)=0$ on all four sides. The exact solution for this problem is given by;

$$u(x,y)=\sin(x)\left(1-\exp(-\frac{2(1-x)}{\gamma})\right)y^2\left(1-\exp(-\frac{3(1-y)}{\gamma})\right) \quad (47)$$

As β is now fixed, the only crucial factor is γ as it determines how difficult the phenomena can numerically be simulated. For this reason, the proposed node-adaptation scheme was tested out at different values of γ and some observations have been recorded. Table. III reports the root-relative-square (L_{RRS}) observed in this work together with those reported in the literature. By using $InN=441$ and $(\theta_{low},\theta_{up})=(0.35,0.50)$, it was found that numerical solutions obtained from DRBEM in both cases; with and without the adaptation algorithm, remained in a reasonably good agreement with other works. This, however, was the case only for $\gamma\le 1.00$ before the difficulty began to take place when γ goes beyond this point.

As reported in the Table, the error increases rapidly (from 64.332 (obtained at $\gamma=0.05$) to over 10,000 (at $\gamma=0.00005$) when using DRBEM without supports from the adaptation scheme. This is more than 20 times higher than those errors produced when receiving supports from the proposed algorithm. A clear piece of evidence supporting this is the small value of L_{RRS}=210 obtained even at $\gamma=0.00005$ (see *Adp-DRBEM*). It should be mentioned that at this very small value of γ, to our knowledge, no other numerical works have yet been reported. The closest one found in the literature is the case with $\gamma=0.0005$, is that recently carried out by [24] using an automatic-node adaptation scheme with the global collocation meshfree method. It can also be observed that the optimal choice for multiquadric type (ε_{MQ}), is increasing when γ gets smaller;from $\varepsilon_{MQ}\approx 0.45$ (obtained at $\gamma=100$) to about $\varepsilon_{MQ}\approx 82.15$ (at $\gamma=0.00005$).

At a highly convection-dominated situation, Table. IV provides information on both types of errors; L_{RMS} and L_∞ obtained from *DRBEM* and *Adp-DRBEM* using at $(\theta_{low},\theta_{up})=(0.25,0.45)$. It can be clearly seen that better solutions can be achieved if DRBEM is used in conjunction with the proposed node-adaptation algorithm. The table also reports that at different node densities used at the beginning of simulations, (*InN*), *Adp-DRBEM* generates different numbers of final nodes (*FnN*) at different values of γ. More nodes were seen to be needed once the phenomenon

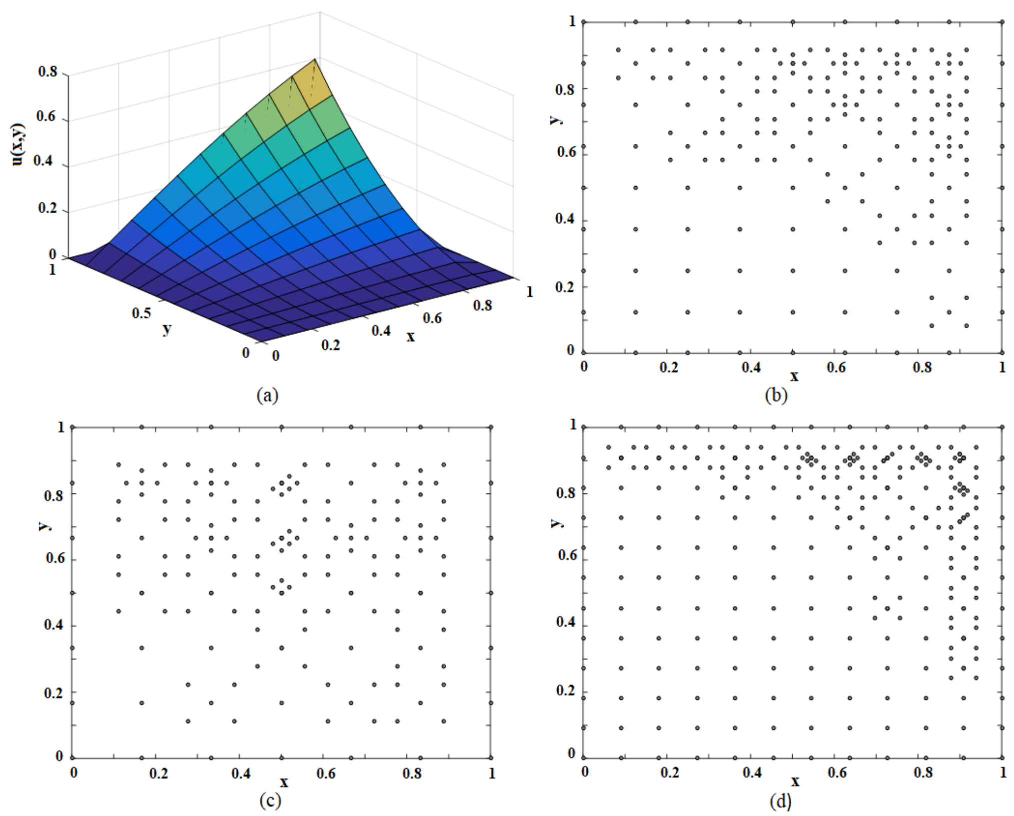
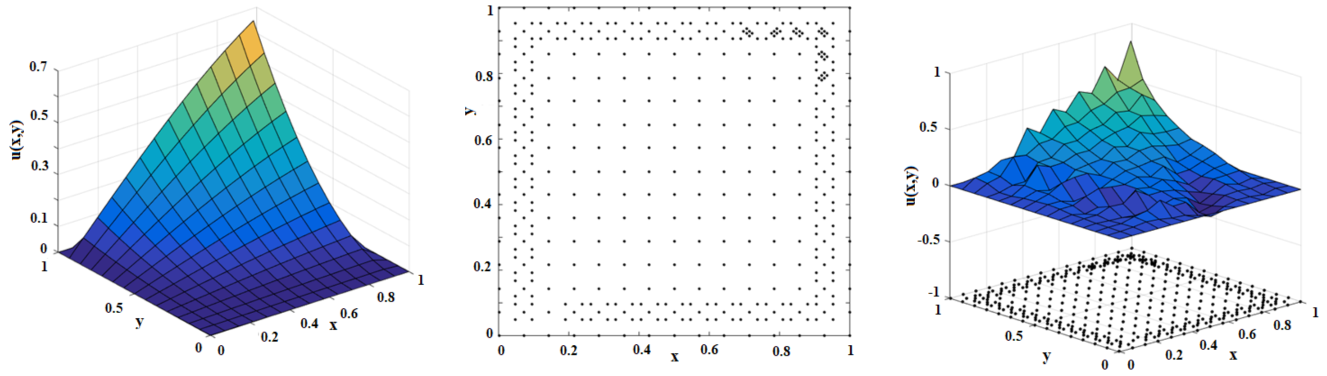


Fig. 3. Node distributions after undergoing the adaptation algorithm using $\varepsilon_{MQ}=50$ and $(\theta_{low},\theta_{up})=(0.25,0.45)$ measured at $\gamma=5.00E-03$; (a) the exact solution profile, (b) *InN* = 81, (c) *InN*=49, and (d) *InN* = 144.



(a) (b) (c)

Fig. 4. Solution profiles at $\gamma = 1.00E-04$; (a) The exact profile, (b) Node distribution after undergoing the node-adaptation process with $(\theta_{low}, \theta_{up}) = (0.25, 0.45)$, $InN = 225$, $\varepsilon_{MQ} = 50$, and (c) Numerical solution profile with its corresponding node-distribution with FnN = 443.

became more convective-dominated, i.e. smaller γ. Appropriate values of ε_{MQ} for these cases were observed to reside in the interval of $(48.50, 59.25)$. Node distributions produced by the algorithm for some interesting cases are illustrated in Fig. 3 and Fig. 4, with the corresponding solution profiles.

5.3 Transient Couple-Burgers' equations

Another challenging structure of PDEs is found in the classical non-linear equation system acknowledged as Burgers Equations, named after the great Physicist Johannes Martinus Burgers (1895-1981). The increase of the so-called 'Reynolds number, (Re)', introduces one of the major difficulties due to inviscid boundary layers produced by the steepening effect of the nonlinear advection term. This is encountered also in the inviscid Navier-Stokes equation for convection-dominated flows. The equations retain the nonlinear of the governing equation in several applications; flow through a shock wave traveling in a viscous fluid, the phenomena of turbulence, sedimentation of two kinds of particles in fluid suspensions under the effect of gravity (see [45], [46]).

Amongst recent attempts to numerically solve the equations system, a well-known analytic solution obtained by using the Hopf-Cole transformation is nicely provided in 1993 by Fletcher [47] and it is of the following forms.

$$u(x,y,t) = \frac{3}{4} - \frac{1}{4}\left[1 + \exp\left((-4x + 4y - t)\frac{\mathrm{Re}}{32}\right)\right]^{-1} \tag{48}$$

$$v(x,y,t) = \frac{3}{4} + \frac{1}{4}\left[1 + \exp\left((-4x + 4y - t)\frac{\mathrm{Re}}{32}\right)\right]^{-1} \tag{49}$$

While the exact solution can produce results regardless of the impact of the Reynolds number, it remains a great challenge for numerical works to effectively approximate the solutions for high Reynolds numbers when the instability starts to kick in [22], [48]-[51]. For this reason, this work focuses on cases with Reynolds numbers greater than 500 only.

TABLE V

L_{RMS} AND L_∞ RECORDED AT MODERATE-TO-HIGH REYNOLDS NUMBERS; WITH $\varepsilon_{MQ} \in (5,10)$, $InN = 144$ AND $(\theta_{low}, \theta_{up}) = (0.15, 0.30)$.

Re=600		U-velocity				V-velocity			
		L_{RMS}		L_∞		L_{RMS}		L_∞	
t	Δt	*DRBEM*	*Adp-DRBEM*	*DRBEM*	*Adp-DRBEM*	*DRBEM*	*Adp-DRBEM*	*DRBEM*	*Adp-DRBEM*
0.01	0.0001	3.0441E-03	2.0782E-05	8.0690E-03	1.1221E-04	3.0408E-03	2.0787E-05	8.0250E-03	1.1241E-04
0.5	0.001	5.1149E-03	6.0739E-04	1.0225E-02	4.3296E-03	5.1132E-03	6.0731E-04	1.0280E-02	4.3296E-03
1.0	0.005	7.5910E-02	2.1241E-03	9.7101E-02	9.5390E-03	7.5917E-02	2.1234E-03	9.7127E-02	9.5136E-03
1.0	0.01	9.8051E-02	6.1120E-03	1.2415E-01	2.3101E-02	9.2101E-02	6.2055E-03	1.2950E-01	2.3998E-02
2.0	0.01	5.9005E-01	7.8810E-03	8.0552E-01	4.6612E-02	5.9225E-01	7.5081E-03	8.1125E-01	4.8805E-02
Re=800		U-velocity				V-velocity			
		L_{RMS}		L_∞		L_{RMS}		L_∞	
t	Δt	*DRBEM*	*Adp-DRBEM*	*DRBEM*	*Adp-DRBEM*	*DRBEM*	*Adp-DRBEM*	*DRBEM*	*Adp-DRBEM*
0.01	0.0001	4.5088E-03	3.3721E-05	8.2055E-02	1.7289E-04	4.5014E-03	3.3729E-05	8.2860E-02	1.7309E-04
0.5	0.001	6.2899E-03	8.5851E-04	9.2501E-02	6.1594E-03	6.2825E-03	8.5847E-04	9.5011E-02	6.1594E-03
1.0	0.005	9.2251E-02	2.8172E-03	3.2201E-01	1.1219E-02	9.2208E-02	2.8175E-03	3.2099E-01	1.1216E-02
1.0	0.01	5.1145E-01	8.2501E-03	9.5504E-01	3.0025E-02	5.4133E-01	8.0255E-03	9.6044E-01	3.1125E-02
2.0	0.01	8.2778E-00	4.0556E-02	5.0889E+01	7.2211E-02	8.4415E-00	4.1002E-02	7.225E+01	6.2511E-02

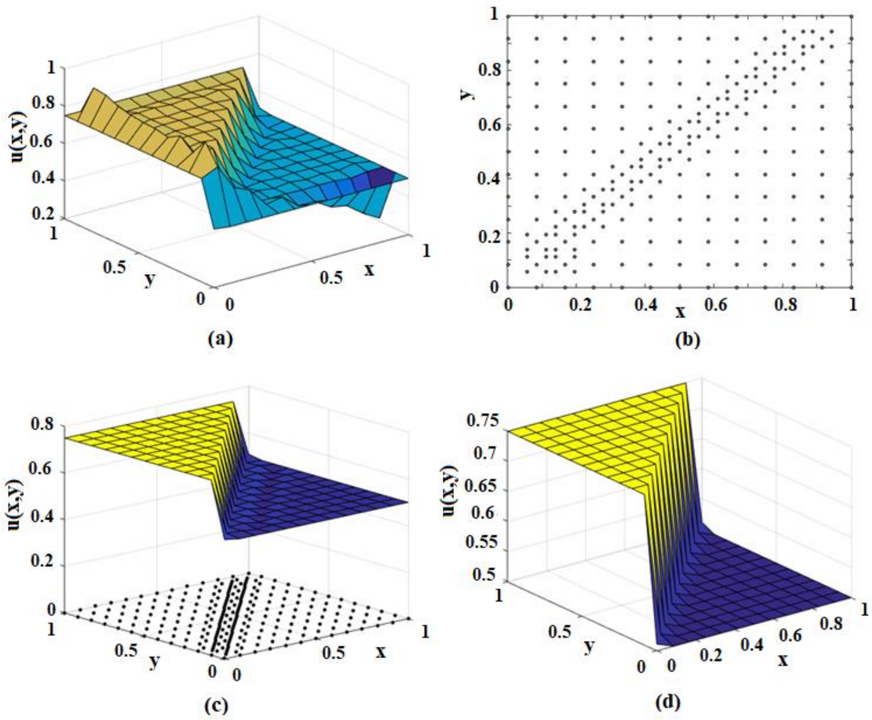


Fig. 5. Numerical simulation of U-velocity component at Re = 1,000 with *InN* = 13×13 where (a) *DRBEM*, (b) Node distribution for *Adp-DRBEM*, (c) Improved solution profile, and (d) Exact surface.

The investigation started at Re=600 and 800 and the main attention was paid to accuracy, measured with L_{RMS}, of the approximate solutions. Table. V. reports a comparison of errors produced by DRBEM with and without using the adaptation scheme. When starting with $InN = 144$ and $(\theta_{low}, \theta_{up}) = (0.15, 0.30)$, the solutions have revealed that the overall performance can be noticeably improved when adopting the proposed node-adaptation algorithm in every case involved. It can also be seen from the table that the solutions in both directions of velocity lose their accuracy when the final target time increases. The optimal multiquadric shape parameter used in Table. V is found to vary within $\varepsilon_{MQ} \in (5,10)$. It has to be mentioned that the increase in Reynolds numbers causes the overall performance to drop as the phenomenon becomes more unstable, as commonly encountered in the literature.

At an even higher Reynolds number Re = 1000, Table. VI shows the effect of node distribution together with CPU-consumption aspect, when adopting the algorithm with $\Delta t = 0.0005,\ t = 0.1$ and $(\theta_{low}, \theta_{up}) = (0.20, 0.35)$. At this very high Reynolds number, as encountered in several other numerical works available in the literature, DRBEM is clearly seen to dramatically have lost its capability to produce good result quality, and the increase of computational nodes is seen to only slightly improve the approximate solutions. With local supports obtained from internal nodes generated by the adaptation algorithm, on the other hand, the approximate solutions obtained by *Adp-DRBEM* are clearly seen to remain promising. Even at the smallest number of initial nodes of $InN = 16$, the algorithm was found with only L_{RMS} = 4.0640E-01 and generated only 28 final nodes, whereas the fixed-conventional DRBEM was found to produce L_{RMS} as high as 2.0015E+02. This aspect persists for every number of initial nodes.

Another interesting finding revealed in Table. VI is that despite the increase of nodes involved in the calculating system, the optimal multiquadric shape parameter is seen not to be significantly affected, varying within $\varepsilon_{MQ} \in (4.00, 5.10)$. Approximate solution surfaces of both x- and y-velocity components at Re=1000 are illustrated in Fig. 5 and Fig. 6 respectively. The figures clearly show the improvement of the accuracy even when using only *InN* =

TABLE VI

L_{RMS} OBTAINED AT DIFFERENT NODES DENSITY AFTER APPLYING THE NODE-ADAPTATION ALGORITHM AT A VERY HIGH REYNOLDS NUMBER $\mathrm{Re} = 1{,}000$ AT , $\Delta t = 0.0005,\ t = 0.1$ $(\theta_{low}, \theta_{up}) = (0.20, 0.35)$.

InN	*FnN*	Optimal ε_{MQ}	CPU-time(s)	U-component		V-component	
				DRBEM	*Adp-DRBEM*	*DRBEM*	*Adp-DRBEM*
4×4	28	5.01	1.3271	2.0015E+02	4.0640E-01	3.2155E+02	3.0016E-01
6×6	75	5.10	1.6239	1.6911E+02	3.1719E-01	2.5110E+02	2.1761E-01
8×8	112	4.98	1.6825	1.0255E+02	2.7016E-01	2.1419E+02	2.0582E-01
10×10	164	5.05	1.8786	8.2871E+01	8.2353E-02	7.3911E+01	9.0607E-02
12×12	236	4.90	2.0504	6.0577E+01	7.9553E-02	6.6018E+01	8.8895E-02
14×14	309	5.02	2.4482	2.5587E+01	7.7774E-02	1.6251E+01	7.0758E-02
16×16	389	5.00	3.1274	5.2889E+00	6.1784E-02	7.5811E+00	5.3695E-02
18×18	465	4.88	3.8970	5.0029E+00	3.9946E-02	4.1660E+00	4.3029E-02
20×20	555	5.04	4.9399	3.3325E+00	1.2429E-02	4.0884E+00	1.8173E-02

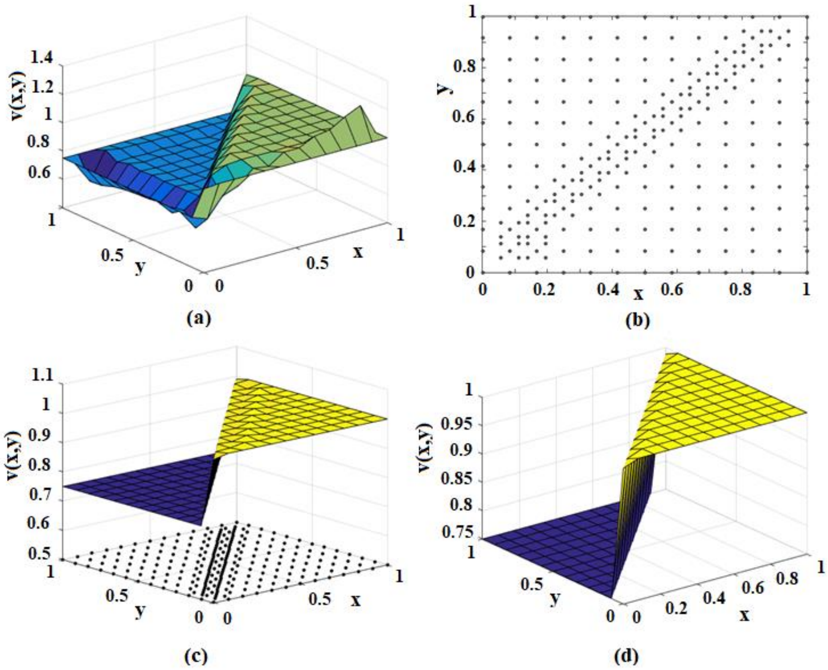


Fig. 6. Numerical simulation of V-velocity component at Re = 1,000 with *InN* = 13×13 where (a) *DRBEM*, (b) Node distribution for *Adp-DRBEM*, (c) Improved solution profile, and (d) Exact surface.

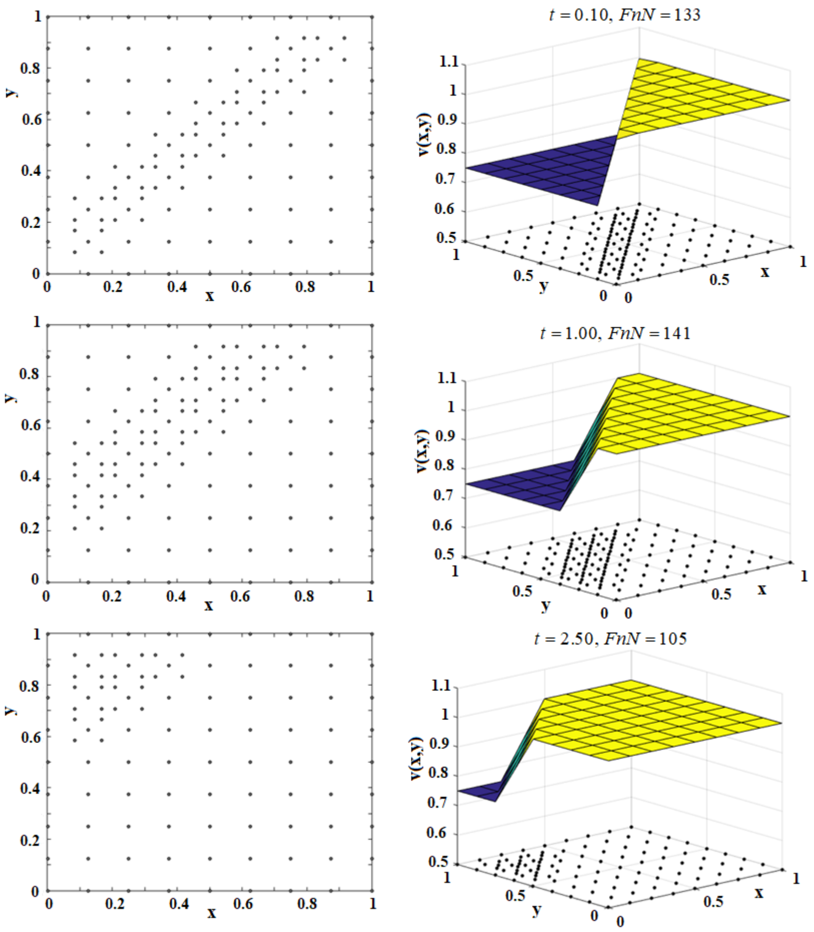


Fig. 7. Node distributions (left) with their corresponding solution profile (right) for V-velocity component numerically produced at Re = 1200 with *InN* = 9×9, $(\theta_{low}, \theta_{up}) = (0.25, 0.40)$ and $\Delta t = 0.005$ measured at $t = 0.1, 1.0$ and 2.5 respectively.

13×13, producing *FnN*=257 (under conditions, $\Delta t = 0.005,\ t = 0.1$ $(\theta_{low}, \theta_{up}) = (0.25, 0.35)$).

Nevertheless, the instability became more serious when the Reynolds number exceeded 1000. It is seen that DRBEM loses its ability to regenerate solutions inside the domain. The use of node-adaptation in this work, on the other hand, has managed to reach a very satisfactory result quality level up to Re=1200 (see Fig. 7).

The nice and smooth surfaces obtained at times t=0.1, 1.0, and 2.5 as illustrated in Fig. 7, required a great amount of effort to carefully test each value of the factors involved; the number of initial nodes (InN), the shape parameter (ε_{MQ}), the refinement thread hold ($\theta_{low}, \theta_{up}$), and time step size (Δt). It was found, for example, that at t=2.5 a smooth surface of approximate solution could be obtained when using $\varepsilon_{MQ} = 0.42$ while significant instabilities (not shown here) were found to take place when using $\varepsilon_{MQ} = 0.50$. The number of initial/parent nodes of InN = 9x9 was found to provide the best results whereas things were observed to be undesirable at other numbers of node density sizes. Attempts to go beyond Re=1200 (up to 1500, in fact) were also made but unfortunately, simulations were not successful. However, at this extremely high Reynolds number, the problem is well known to be highly complicated and is still almost impossible to successfully simulate by numerical schemes [50], [52]. This, as a consequence, remains one of the future investigations of the authors.

From all the test cases under the investigation in this work, some additional and crucial observations are summarized here. Firstly, on CPU time and storage, it has been revealed that using *Adp-DRBEM* requires approximately 25%-50% more CPU time and storage when compared with the traditional DRBEM. This is attributed to the need to construct at least 4 matrices (i.e. $\mathbf{H}$, $\mathbf{G}$, $\hat{\mathbf{U}}$ and $\hat{\mathbf{Q}}$ in (10)) after undergoing the algorithm of adaptation. All simulations in this work underwent only up to 2-level of refinement meaning that the construction of those matrices happens three times throughout each simulation. However, when taking into consideration the much better result quality, it might then be worth investing. Secondly, the choice of good refinement thresholds ($\theta_{low}, \theta_{up}$) in this work is obtained in an ‘ad-hoc’ manner. It is not straightforward to completely be certain about what interval thresholds the simulation would remain in an optimal mode. Thirdly, the same pain faced with setting appropriate refinement thresholds occurs again when it comes to choosing a good (if not ‘optimal’) shape parameter for the multiquadric radial basis function used. Despite several attempts on choosing this kind of parameter proposed and presented in the literature [53]-[55], this work carried out a large number of numerical experiments before a good decision was made. This is also because shape parameters are known to highly be sensitive to the number of interpolation nodes involved. To remedy this shortcoming, parameters that are self-adaptable corresponding to the number of computational nodes may be good alternatives. All these open problems mentioned so far truly deserve further investigation.

VI. Conclusion

In this paper, a node-adaptation scheme is proposed and applied in conjunction with the dual reciprocity boundary element method (DRBEM). The algorithm focuses on internal nodes where the boundary ones are intact. Nodes are automatically inserted into (or removed from) the computational domain during the computation procedure based on a proposed pre-defined refinement criterion. Classical mechanics problems containing challenges are used to perform the algorithm. The solution quality validation is carried out by comparing to the exact ones and/or those documented in the literature. The radial basis function used is the multiquadric type and the following is the list of the main findings obtained from this investigation.

1. Based on all error measurement norms used, the performance of the conventional DRBEM is seen to significantly improve when used with the proposed node-adaptation algorithm (Adp-DRBEM). This figure is found in all test cases.
2. The best performance of Adp-DRBEM is found in the case of a convection-diffusion problem (The results quality was observed to increase approximately 20 times when compared with the conventional DRBEM.)
3. The method of finding an optimal value of the MQ-shape parameter remains unclear due to many factors involved so that all simulations in this work obtained the optimal choice via. an ‘*empirical*’ means.
4. The necessity to generate matrices all over again after undergoing the adaptation algorithm leads to an increase in CPU time and storage.

With all these positive and negative aspects of the proposed adaptative algorithm, our further investigation is set out toward its improvement. This is with the hope that it can enhance DRBEM to be more capable of dealing with more complex problems in the future.

References

[1] C. A. Brebbia, and R. Butterfield, "Formal Equivalence of Direct and Indirect Boundary Element Methods," Applied Mathematical Modelling, vol. 2, 1978

[2] W. Toutip, "The dual reciprocity boundary element method for linear and nonlinear," Ph.D., Department of Mathematics, University of Hertfordshire, United Kingdom, 2001

[3] F. J. Rizzo, and D. J. Shippy, "An advanced boundary integral equation method for three-dimensional thermoelasticity," Int J Numer Meth Eng, vol. 11, pp1753-1768, 1977

[4] R. Zheng, and N. Phan-Thien, "Transforming the domain integrals to the boundary using approximate particular solutions: a boundary element approach for nonlinear problems," Appl Numer Math, vol. 10, pp435-445, 1992

[5] Viktor Popov, and Henry Power, "The DRM-MD integral equation method: an efficient approach for the numerical solution of domain dominant problems," Int J Numer Meth Eng, vol. 44, pp327-353, 1999

[6] M. M. Grigoriev, and G. F. Dargush, "A poly-region boundary element method for incompressible viscous fluid flows," Int J Numer Meth Eng, vol. 46, pp1127-1158, 1999

[7] P. W. Partridge, C. A. Brebbia, and L. C. Wrobel, *The dual reciprocity boundary element method*. Southampton: Computational Mechanics Publications/Elsevier, 1992.

[8] T. R. Bridges, and L. C. Wrobel, "A dual reciprocity formulation for elasticity problems with body forces using augmented thin-plate splines," Commun Numer Meth En, vol. 12, pp209–220, 1996

[9] Y. F. Rashed, "BEM for dynamic analysis using compactly supported radial basis functions," Comput Struct, vol. 80, pp1351-1367, 2002

[10] M. F. Samaan, and Y. F. Rashed, "BEM for transient 2D elastodynamics using multiquadric functions," Int J Solids Struct, vol. 44, pp8517-8531, 2007

[11] Khang Jie Liew, Kah Heng Tee, Ahmad Ramli, and Wen Eng Ong, “Integrating Clustering Method in Compactly Supported Radial Basis Function for Surface Approximation,” IAENG International Journal of Computer Science, vol. 46, no. 1, pp38-45, 2019

[12] Y. Alhuri, D. Ouazar, and A. Taik, “Comparison between Local and Global Mesh-free Methods for Ground-Water Modeling,” IJCSI International Journal of Computer Science Issues, vol. 8, no. 2, pp337-342, 2011

[13] C. F. Loeffler, and W. J. Mansur, "Analysis of time integration schemes for boundary element applications to transient wave propagation problems," Proceedings of The BETECH’87, Computational Mechanics Publication, 1987, Southampton, UK

[14] T. R. Bridges, and L. C. Wrobel, "On the calculation of natural frequencies of microstructures using DRBEM," Proceedings of the Boundary Element Method (BEM) XVI Conference, Computational Mechanics Publications, 1994, Southampton, London

[15] K. Chanthawara, S. Kaennakham, and W. Toutip, "The Numerical Study and Comparison of Radial Basis Functions in Applications of the Dual Reciprocity Boundary Element Method to Convection-Diffusion Problems," Progress in Applied Mathematics in Sciences and Engineering, AIP Conference Proceedings, 2016.

[16] D. P. N. Kontoni and D. E. Beskos, "Transient dynamic elastoplastic analysis by the dual reciprocity BEM," Eng Anal Boundary Elem, vol. 12, pp1-16, 1993

[17] J. Dominguez, *Boundary element in dynamics*. Computational Mechanics Publications: Southampton, Elsevier Applied Science, London, 1993.

[18] S. Kaennakham, A. E. Holdø, and C. Lambert, "A new simple h-mesh adaptation for standard Smagorinsky LES: a first step of Taylor scale as a refinement variable," International Journal of Multiphysics, vol. 4, pp33-50, 2010

[19] S. A. Sarra, "Adaptive radial basis function method for time-dependent partial differential equations," Appl. Num. Math, vol. 54, pp79-94, 2005

[20] G. R. Liu, Bernard B. T. Kee, Z. H. Zhong, G. Y. Li, and X. Han, *Adaptive meshfree methods using local nodes and radial basis functions*. New York: Springer, Berlin, 2006.

[21] Tobin A. Driscoll, and Alfa R. H. Heryudono, "Adaptive residual subsampling methods for radial basis function interpolation and collocation problems," Computers and Mathematics with Applications, vol. 53, pp927-939, 2007

[22] Shou-hui Zhang, and Wen-qia Wang, "A stencil of the finite–difference method for the 2D convection diffusion equation and its new iterative scheme," Int J Comput Math, vol. 87, pp2588-2600, 2010

[23] W. Chen, and Y. C. Hon, "Numerical investigation on convergence of boundary knot method in the analysis of homogeneous Helmholtz, modified Helmholtz, and convection-diffusion problems," Computer Methods in Applied Mechanics andd Engineering, vol. 192, pp1859-1875, 2003

[24] S. Kaennakham, and N. Chuathong, "An Automatic Node-Adaptive Scheme applied with a RBF-Collocation Meshless Method," Applied Mathematics and Computation, vol. 348, pp102-125, 2019

[25] S. Kaennakham, "A solution adaptive grid (SAG) for incompressible flows simulations: an attempt towards enhancing SAG algorithm for large eddy simulation," Ph.D., Department of Computational Fluid Dynamics, Coventry University, United Kingdom, 2010.

[26] A. J. Khattak, and Siraj-ul-Islam, "A comparative study of numerical solutions of a class of KdV equation," Appl. Math. Comput, vol. 199, pp425-434, 2008

[27] A. K. Khalifa, K. R. Raslan, and H. M. Alzubaidi, "A finite difference scheme for the MRLW and solitary wave interactions," Appl. Math. Comput, vol. 189, pp346-354, 2007

[28] W. Gui, and I. Babuška, "The h, p and h-p versions of the finite element method in 1 dimension part 1: the error analysis of the p-version," Numer. Math., vol. 49, pp577-612, 1986

[29] O. C. Zienkiewicz, and J. Z. Zhu, "A simple error estimator and adaptive procedure for practical engineering analysis," Int. J. Numer. Methods Eng., vol. 24, pp337-357, 2005

[30] Catherine Mavriplis, "A posteriori error estimators for adaptive spectral element techniques," Lecture Notes in Numerical Fluid Mechanics (NNFM): Proceedings of the Eighth GAMM-Conference on Numerical Methods in Fluid Mechanics, 1990, Wiesbaden, German, pp334-342

[31] Rainald Löhner, "An adaptive finite element solver for transient problems with moving bodies," Comput. Struct., vol. 30, pp303-317, 1988

[32] Jean-Francois Hetu and Dominique H. Pelletier, "Fast, adaptive finite element scheme for viscous incompressible flows," AIAA J., vol. 30, pp2677-2682, 1992

[33] J. J. Rodriguez, and H. Power, "A mesh refinement technique for the boundary element method based on local error analysis," Transactions on Modelling and Simulation, vol. 24, pp291-302, 1999

[34] André Fortin, José M. Urquiza, and Richard Bois, "A mesh adaptation method for 1D-boundary layer problems," International Journal of Numerical Analysis and Modeling, Series B, vol. 3, no. 4, pp408-428, 2012

[35] Michael Feischl, Thomas Führer, Norbert Heuer, Michael Karkulik, and Dirk Praetorius, "Adaptive Boundary Element Methods: A posteriori error estimators, adaptivity, convergence, and implementation," Mathematics Subject Classification, 2000

[36] Krittidej Chanthawara, Sayan Kaennakham, and Wattana Toutip, "Inverse Multiquadric RBF in the Dual Reciprocity Boundary Element Method(DRBEM) for Coupled 2D Burgers' Equations at high Reynolds numbers," Proceeding of the 19th International Annual Symposium on Computational Science and Engineering (ANSCSE19), 17-19 June, 2015, Ubon Ratchathani, Thailand, pp2-22

[37] Krittidej Chanthawara, Sayan Kaennakham, and Wattana Toutip, "The Numerical Study and Comparison of Radial Basis Functions in Applications of the Dual Reciprocity Boundary Element Method to Convection-Diffusion Problems," Progress in Applied Mathematics in Sciences and Engineering, AIP Conference Proceedings, 2016.

[38] Xingping Sun, "Scattered Hermite interpolation using radial basis functions," Linear Algebra Appl., vol. 207, pp135–146, 1994

[39] Rolland L. Hardy, "Multiquadric equations of topography and other irregular surfaces," Journal of Geophysical Research, vol. 76, pp1905-1915, 1971

[40] Ali Kurt, Yücel Çenesiz, and Orkun Tasbozan, "On the solution of Burgers' equation with the new fractional derivative," Open Phys, vol. 13, pp355-360, 2015.

[41] H. Lin, and S. N. Atluri, "Meshless local Petrov-Galerkin (MLPG) method for convection-diffusion problems," Comput. Model. Eng. Sci. , vol. 1, pp45-60, 2000

[42] Guo Zhong, Mingxu Yi, and Jun Huang, "Numerical Method for Solving Fractional Convection Diffusion Equations with Timespace Variable Coefficients," IAENG International Journal of Applied Mathematics, vol. 48, no. 1, pp62-66, 2018

[43] Y. T. Gu, and G. R. Liu, "Meshless techniques for convection dominated problems," Compt. Mech., vol. 38, pp171-182, 2006

[44] Alsoy-Akgun Nagehan, "The dual reciprocity boundary element method solution of Helmholtz type equations in fluid dynamic," Ph.D., Doctoral Program, Middle East Technical University, Turkey, 2013

[45] Janpou Nee, and J. Duan, "Limit Set of Trajectories of the Coupled Viscous Burgers Equations," Applied Mathematics Letters, vol. 11, pp57-61, 1998.

[46] Dingwen Deng, and Jianqiang Xie, "High-order Exponential Time Differencing Methods for Solving One-dimensional Burgers' Equation," IAENG International Journal of Computer Science, vol. 43, no. 2, pp167-175, 2016

[47] Clive A. J. Fletcher, "Generating Exact Solutions of the Two-Dimensional Burgers equations," International Journal for Numerical Methods in Fluids, vol. 3, pp213-216, 1983

[48] A. Refik Bahadır, "A fully implicit finite-difference scheme for two-dimensional Burgers' equations," Applied Mathematics and Computation, vol. 137, pp131-137, 2003.

[49] S. Kaennakham, and N. Chuathong, " A Radial Point Interpolation Method (RPIM) with Matern RBF Applied to a Two-Dimensional Nonlinear Coupled-PDE," IAENG International Journal of Applied Mathematics, vol. 48 , no. 4, pp485-498, 2018

[50] Siraj-ul-Islamac, Božidar Šarler, Robert Vertnik, and Gregor Kosec, "Radial basis function collocation method for the numerical solution of the two-dimensional transient nonlinear coupled Burgers' equations," Applied Mathematical Modelling, vol. 36, pp1148-1160, 2012

[51] Arshed Ali, Siraj-ul-Islam, and Sirajul Haq, "A Computational Meshfree Technique for the Numerical Solution of the Two-Dimensional Coupled Burgers' Equations," International Journal for Computational Methods in Engineering Science and Mechanics, vol. 10, pp406-422, 2009

[52] Fang Liu, and Shi Weiping, "Numerical solutions of two-dimensional Burgers' equations by lattice Boltzmann method," Communications in Nonlinear Science and Numerical Simulation, vol. 16, 2011

[53] E. J. Kansa, and R. E. Carlson, "Improved accuracy of multiquadric interpolation using variable shape parameters," Computers and Mathematics with Applications, vol. 24, pp99-120, 1992

[54] Sayan Kaennakham, and Nissaya Chuathong, "A Numerical Experiment on Optimal Inverse Multiquadric RBF Shape Parameter in the Dual Reciprocity Boundary Element Method for Convective-Dominated Problems," Thai Journal of Mathematics, Special Issue on ICMSA2015, 2016.

[55] Ralph E. Carlson, and Thomas A. Foley, "The parameter R2 in multiquadric interpolation," Computers & Mathematics with Applications, vol. 21, pp29-42, 1991