Available online at www.sciencedirect.com
ScienceDirect
Comput. Methods Appl. Mech. Engrg. 351 (2019) 404–421
Computer methods in applied mechanics and engineering
www.elsevier.com/locate/cma

Strong form-based meshfree collocation method for wind-driven ocean circulation

Andrew Beel[a], Tae-Yeon Kim[b], Wen Jiang[c], Jeong-Hoon Song[a,*]

[a] *Department of Civil, Environmental and Architectural Engineering, University of Colorado, Boulder, CO, 80309, USA*
[b] *Civil Infrastructure and Environmental Engineering, Khalifa University of Science and Technology, Abu Dhabi, 127788, United Arab Emirates*
[c] *Fuels Modeling and Simulation, Idaho National Laboratory, Idaho Falls, ID, 83415, USA*

Received 24 November 2018; received in revised form 20 March 2019; accepted 21 March 2019
Available online 1 April 2019

Abstract

A meshfree point collocation method for modeling large scale wind-driven ocean circulation is proposed. A distinct feature of the method is its ability to represent derivative operators via moving least-square approximation of the Taylor expansion through point-wise computations at collocation points. The method directly discretizes strong forms using the precomputed derivative operators at each collocation point. Numerical studies with three benchmark problems are performed to demonstrate the accuracy and robustness of the method. Along with these studies, an examination is presented regarding the effect of numerical parameters on the error behavior of the proposed method. Finally, wind-driven ocean circulation in the Mediterranean Sea is examined to test the method's ability to model realistic oceanic flow with arbitrary shaped coastal lines.

Keywords: Strong form; Meshfree; Point collocation; Adaptivity; Ocean circulation; Stommel–Munk model

1. Introduction

Understanding large scale wind-driven ocean circulation at mid-latitude is important to predict weather, including extreme events such as cyclones [1]. Common features of wind-driven ocean flows are strong western boundary currents, weak interior flows, and weak eastern boundary currents, as in the north Atlantic and Pacific oceans. Popular mathematical models to capture these phenomena are the quasi-geostrophic equations (QGE), the Stommel model, and the Stommel–Munk model [2,3]. Whereas the QGE are time-dependent nonlinear partial differential equations (PDEs), the Stommel model and the Stommel–Munk model are stationary linear PDEs.

Existing numerical methods that can be employed to solve these mathematical models are the finite-difference method [4] or the finite-volume method [5,6], and the finite-element method [7–9]. For geophysical flows, the finite-volume method is particularly appealing due to its capability of unstructured grids along with preserving conservation properties of the underlying equations. On the other hand, advantages of the finite-element method over the finite-difference approaches are an easy treatment of complex boundaries and grid refinement to achieve a high accuracy in regions of interest. The finite-element methods to solve the QGE have been developed based

* Corresponding author.
E-mail address: jh.song@colorado.edu (J.-H. Song).

https://doi.org/10.1016/j.cma.2019.03.045
0045-7825/

on a mixed formulation [9] and the streamfunction formulation [10]. The mixed formulation results in a second-order PDE, thus requiring C^0-elements. The streamfunction formulation is a fourth-order PDE, which necessitates C^1-elements. While standard finite-element methods using the mixed formulation result in suboptimal convergence rates [9], optimal rates of convergence can be obtained using the streamfunction formulation [10]. As a result, recent developments of numerical techniques have focused on the streamfunction formulation with the goal of obtaining optimal convergence rates. Examples include a conforming Galerkin formulation using C^1-elements [10], a discontinuous Galerkin formulation using C^0-elements [11], and B-spline based finite-element formulations [12–14]. Further development of the B-spline based finite-element method was achieved by introducing an adaptivity technique for the Stommel and Stommel–Munk models [15] and modeling arbitrary shaped coastal boundaries on embedded boundaries [16].

In the context of wind-driven oceanic flow simulation, it is challenging to achieve accurate and efficient models in light of the complex boundaries of arbitrarily shaped coastlines. Furthermore, capturing a strong western boundary layer requires an efficient adaptive refinement technique. In doing so, the computational cost of a conforming Galerkin formulation using C^1-elements is relatively expensive. Using B-splines, modeling arbitrary shape geometries often involves either a mapping (as in isogeometric analysis [17]) or a fictitious-domain approach [18]. Moreover, B-splines have difficulty in applying boundary conditions, particularly in the case of a strong boundary layer [19]. In the present study, we propose a meshfree point collocation method for accurate and efficient simulation of the large scale wind-driven ocean circulation with the Stommel model and the Stommel–Munk model.

The key idea of the proposed method is based on constructing an approximation of higher-order derivative operators with a Taylor expansion through the moving least squares approach; such ideas have been initially adapted for meshfree methods by other researchers [20–24]. It allows for straightforward computation of numerical derivative operators that can directly discretize strong forms of PDEs of any order, as well as their boundary conditions. Other interesting weak form-based meshfree methods can be found in [25–27]. Main advantages of the method include an easy treatment of high-order derivatives and boundary conditions, an easy adaptive refinement, and low computation cost due to the lack of numerical integration. Upon making use of these advantages, the method has been successfully applied to various problems such as incompressible fluid flows [23], elastic crack problems [28], and the asymptotic crack tip singularity in a linear elastic fracture [29,30]. Later, the method has been also applied to weak and strong discontinuities [31,32], moving boundary [33], diffusive interface modeling [34,35], and inelastic material problems [36] by the authors.

Despite the advantages originating from the flexibility in discretization, the strong form-based meshfree collocation method also exhibits drawbacks. For instance, since the proposed method is based on the strong form, it lacks the corresponding advantages of methods based on variational forms, e.g., accuracy of solutions in the sense of total energy and some of the well-developed mathematical framework for analyzing error behavior. In depth discussion about the stability of the meshfree method can be found in [37].

The purpose of this paper is to solve higher-order PDEs using a strong form-based meshfree collocation and present some results about the error behavior of this method. First, the meshfree point collocation formulations for the second-order Stommel model and the fourth-order Stommel–Munk model are presented, as well as a strategy for implementing these formulations to solve both linear models. Second, accuracy and robustness of the proposed formulations are verified via convergence studies using several benchmark problems for both uniform and random arrangements of collocation points. In addition, the method is extended to a more realistic simulation by modeling wind-driven ocean circulation in the Mediterranean sea, a region with complex coastal boundaries.

The remainder of the paper is organized as follows. In Section 2, we present the meshfree point collocation formulations for the Stommel–Munk model and the Stommel model. In Section 3, these formulations are tested by solving three simple benchmark problems. Motivated by the numerical results, a closer look at some internal implementation parameters and their effect on error behavior is given in Section 3.3. In the last part of Section 3, we solve the Stommel–Munk model on a polygon representing the Mediterranean Sea. Finally, a summary and concluding remarks are given in Section 4.

2. The strong form based meshfree collocation method

2.1. Pointwise meshfree approximation

The meshfree point collocation method presented here allows the construction of numerical derivative operators that are used to discretize the Stommel–Munk model (11) and the Stommel model (13). We consider a plane domain

Table 1
Entries of $\mathbf{a}(\mathbf{y})$ and $\mathbf{p}(\mathbf{x};\mathbf{y})$ for $m=4$, $n=2$, and $L=15$ ($0\le i\le m$ and $0\le k\le L$).

i	k	$\boldsymbol{\alpha}_k$	Entries of $\mathbf{a}(\mathbf{y})$	Entries of $\mathbf{p}(\mathbf{x};\mathbf{y})$
0	1	(0, 0)	$u(\mathbf{y})$	1
1	2	(1, 0)	$u_{y_1}(\mathbf{y})$	x_1-y_1
	3	(0, 1)	$u_{y_2}(\mathbf{y})$	x_2-y_2
2	4	(2, 0)	$u_{y_1y_1}(\mathbf{y})$	$(x_1-y_1)^2/2$
	5	(1, 1)	$u_{y_1y_2}(\mathbf{y})$	$(x_1-y_1)(x_2-y_2)$
	6	(0, 2)	$u_{y_2y_2}(\mathbf{y})$	$(x_2-y_2)^2/2$
3	7	(3, 0)	$u_{y_1y_1y_1}(\mathbf{y})$	$(x_1-y_1)^3/6$
	8	(2, 1)	$u_{y_1y_1y_2}(\mathbf{y})$	$(x_1-y_1)^2(x_2-y_2)/2$
	9	(1, 2)	$u_{y_1y_2y_2}(\mathbf{y})$	$(x_1-y_1)(x_2-y_2)^2/2$
	10	(0, 3)	$u_{y_2y_2y_2}(\mathbf{y})$	$(x_2-y_2)^3/6$
4	11	(4, 0)	$u_{y_1y_1y_1y_1}(\mathbf{y})$	$(x_1-y_1)^4/24$
	12	(3, 1)	$u_{y_1y_1y_1y_2}(\mathbf{y})$	$(x_1-y_1)^3(x_2-y_2)/6$
	13	(2, 2)	$u_{y_1y_1y_2y_2}(\mathbf{y})$	$(x_1-y_1)^2(x_2-y_2)^2/4$
	14	(1, 3)	$u_{y_1y_2y_2y_2}(\mathbf{y})$	$(x_1-y_1)(x_2-y_2)^3/6$
	15	(0, 4)	$u_{y_2y_2y_2y_2}(\mathbf{y})$	$(x_2-y_2)^4/24$

Ω in n-space. Let $\boldsymbol{\alpha}=\{\alpha_1,\alpha_2,\ldots,\alpha_n\}$ be an n-dimensional array of nonnegative integers and define $|\boldsymbol{\alpha}|\cong\Sigma_{i=1}^{n}\alpha_i$. Let $\mathbf{x}=\{x_1,x_2,\ldots,x_n\}^T$ be an n-dimensional vector. Then, we write

$$\mathbf{x}^{\boldsymbol{\alpha}}=\prod_{i=1}^{n}x_i^{\alpha_i}\quad\text{and}\quad\boldsymbol{\alpha}!=\prod_{i=1}^{n}\alpha_i!. \tag{1}$$

For convenience, such an array of nonnegative integers $\boldsymbol{\alpha}$ will be referred to as an exponent array from this point forward.

Given a function $u(\mathbf{x})\in\mathcal{C}^m(\Omega)$ for a nonnegative integer m, we write the αth derivative of $u(\mathbf{x})$ as

$$D_{\mathbf{x}}^{\boldsymbol{\alpha}}u=\frac{\partial^{|\boldsymbol{\alpha}|}u}{\partial x_1^{\alpha_1}\partial x_2^{\alpha_2}\cdots\partial x_n^{\alpha_n}} \tag{2}$$

for $|\boldsymbol{\alpha}|\le m$. With this notation, the mth-order Taylor expansion P_m of u, centered at $\mathbf{y}$ and evaluated at $\mathbf{x}$, is given by

$$P_m(\mathbf{x};\mathbf{y})=\sum_{|\boldsymbol{\alpha}|\le m}\frac{(\mathbf{x}-\mathbf{y})^{\boldsymbol{\alpha}}}{\boldsymbol{\alpha}!}D_{\mathbf{x}}^{\boldsymbol{\alpha}}u(\mathbf{y}). \tag{3}$$

Recall that we seek a way to construct numerical derivative operators. The first step is to order all derivatives of $u(\mathbf{y})$ of up to order m in n variables and arrange them in a vector $\mathbf{a}(\mathbf{y})$. Note that there are $L=(m+n)!/(m!n!)$ such derivatives. This vector is constructed by placing all derivatives of each degree in lexicographical order, and then ordering these partial lists in order of increasing degree.

Let $\boldsymbol{\alpha}_k$ be an exponent array like the one characterized by (1), with the property that the jth entry of $\boldsymbol{\alpha}_k$ equals the number of derivatives of $u(\mathbf{y})$ with respect to y_j taken when evaluating the kth element of $\mathbf{a}(\mathbf{y})$. In other words, $\boldsymbol{\alpha}_k$ is the exponent array that makes the kth entry of $\mathbf{a}(\mathbf{y})$ equal to $D_{\mathbf{y}}^{\boldsymbol{\alpha}_k}u$. For a given m, this exponent array may be used to construct a polynomial vector $\mathbf{p}(\mathbf{x};\mathbf{y})$ with kth entry given by

$$p_k(\mathbf{x};\mathbf{y})=\frac{(\mathbf{x}-\mathbf{y})^{\boldsymbol{\alpha}_k}}{\boldsymbol{\alpha}_k!} \tag{4}$$

and corresponding to the kth entry of $\mathbf{a}(\mathbf{y})$ so that the Taylor polynomial in (3) may be rewritten as

$$P_m(\mathbf{x};\mathbf{y})=\mathbf{p}(\mathbf{x};\mathbf{y})^T\mathbf{a}(\mathbf{y}). \tag{5}$$

For better understanding, the exponent arrays and entries of $\mathbf{p}(\mathbf{x};\mathbf{y})$ and $\mathbf{a}(\mathbf{y})$ for $m=4$ and $n=2$ are given in Table 1.

Next, the method of moving least squares is used in conjunction with the Taylor expansion above to formulate discrete derivative operators. For a set of N points $\mathbf{x}_J$ ($J=1,2,\ldots,N$) in Ω, we seek to minimize the discrete

weighted residual functional

$$\mathcal{F}[\mathbf{a}(\mathbf{y})] = \sum_{J=1}^{N} w\left(\frac{\mathbf{x}_J - \mathbf{y}}{\rho(\mathbf{y})}\right)\left[\mathbf{p}(\mathbf{x}_J;\mathbf{y})^T\mathbf{a}(\mathbf{y}) - u_J\right]^2 \tag{6}$$

with respect to $\mathbf{a}(\mathbf{y})$. Here, w is a weight function and $\rho(\mathbf{y})$, termed the dilation parameter, is the radius of a ball around $\mathbf{y}$ called the compact support. Outside of this radius, the weight function is set to zero.

For a fixed $\mathbf{y}$, $\mathcal{F}[\mathbf{a}(\mathbf{y})]$ is minimized for a unique $\mathbf{a}(\mathbf{y})$ for which $\partial\mathcal{F}/\partial\mathbf{a} = 0$ because $\mathcal{F}$ is nonnegative and quadratic in $\mathbf{a}$. If the local center is moved (hence the term moving least squares) to each point $\mathbf{x}$ at which a set of derivative operators is desired, then the local $\mathcal{F}[\mathbf{a}(\mathbf{x})]$ is minimized by the $\mathbf{a}(\mathbf{x})$ for which $\partial\mathcal{F}/\partial\mathbf{a} = 0$. In other words, minimizing (6) and taking $\mathbf{y} \to \mathbf{x}$ results in

$$\mathbf{a}(\mathbf{x}) = \mathbf{M}^{-1}(\mathbf{x})\mathbf{B}(\mathbf{x})\mathbf{u} \tag{7}$$

where $\mathbf{u} = \{u_1, u_2, \ldots, u_N\}^T$ is the nodal solution vector for the N points in Ω, and $\mathbf{M}$ and $\mathbf{B}$ are given by

$$\mathbf{M}(\mathbf{x}) = \sum_{J=1}^{N}\left(\mathbf{p}(\mathbf{x}_J;\mathbf{x})w\left(\frac{\mathbf{x}_J - \mathbf{x}}{\rho(\mathbf{x})}\right)\mathbf{p}^T(\mathbf{x}_J;\mathbf{x})\right) \tag{8}$$

and

$$\mathbf{B}(\mathbf{x}) = \left[w\left(\frac{\mathbf{x}_1 - \mathbf{x}}{\rho(\mathbf{x})}\right)\mathbf{p}(\mathbf{x}_1;\mathbf{x}),\, w\left(\frac{\mathbf{x}_2 - \mathbf{x}}{\rho(\mathbf{x})}\right)\mathbf{p}(\mathbf{x}_2;\mathbf{x}),\, \ldots,\, w\left(\frac{\mathbf{x}_N - \mathbf{x}}{\rho(\mathbf{x})}\right)\mathbf{p}(\mathbf{x}_N;\mathbf{x})\right]. \tag{9}$$

Since the method uses a diffuse derivative approximation, computation of these derivative operators does not require exact differentiation of shape functions or of the weight function, so there is no differentiability requirement for either. Furthermore, the cost of computing derivatives of shape functions is replaced by that of inverting N $\mathbf{M}$ matrices, each of which is only $L \times L$.

An important consequence of this result is that the kth entry of $\mathbf{a}(\mathbf{x}) = \mathbf{M}^{-1}(\mathbf{x})\mathbf{B}(\mathbf{x})\mathbf{u}$ gives approximation for the $\boldsymbol{\alpha}_k$th derivative of u at $\mathbf{x}$. Thus, for each $\mathbf{x}_I \in \mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N$, we can define differential operators $\boldsymbol{\Phi}_I^{\boldsymbol{\alpha}_k}$ defined to be the kth row of $\mathbf{M}^{-1}(\mathbf{x}_I)\mathbf{B}(\mathbf{x}_I)$, i.e.,

$$\boldsymbol{\Phi}_I^{\boldsymbol{\alpha}_k}\mathbf{u} \approx D_{\mathbf{x}}^{\boldsymbol{\alpha}_k}u(\mathbf{x}_I) \tag{10}$$

Using (10), all differential operators in the Stommel and Stommel–Munk models and their boundary conditions can be discretized for any set of N collocation points in $\bar{\Omega}$.

2.2. Strong form and discretized equations

In this section, the Stommel model and the Stommel–Munk model are discretized using the meshfree collocation described above. While the Stommel model is a second-order PDE with respect to the streamfunction ψ, the Stommel–Munk model is a fourth-order PDE. Similar to the QGE, the Stommel–Munk model contains the biharmonic term $\Delta^2\psi$, the wind forcing term F, and the rotational term $\partial\psi/\partial x$ to include the effect of an asymmetry in the east–west direction. The Stommel–Munk model involves a Laplacian term $\Delta\psi$ instead of the nonlinear Jacobian term.

We consider a plane domain Ω with boundary Γ. The Stommel–Munk model [2] is given by

$$\begin{aligned} &-\epsilon_s\Delta\psi + \epsilon_m\Delta^2\psi - \frac{\partial\psi}{\partial x} = F \quad \text{in} \quad \Omega, \\ &\psi = 0 \quad \text{and} \quad \nabla\psi\cdot\mathbf{n} = 0 \quad \text{on} \quad \Gamma \end{aligned} \tag{11}$$

where $\mathbf{n}$ is the outward unit normal vector on the boundary Γ. For the wind-driven ocean circulation in an enclosed mid-latitude basin, let ψ and F denote the velocity streamfunction and the wind forcing, respectively. The parameters ϵ_s and ϵ_m are the non-dimensional Stommel and Munk numbers, respectively, which are defined by

$$\epsilon_s = \frac{\gamma}{\beta L} \quad \text{and} \quad \epsilon_m = \frac{A}{\beta L^3}. \tag{12}$$

Here, γ is the coefficient of the linear drag (or the Rayleigh friction) as might be generated by a bottom Ekman layer, β is the coefficient multiplying the y-coordinate in the β-plane approximation, A is the eddy viscosity parametrization, and L is the characteristic length scale.

The Stommel model (Vallis [2]) is given by

$$\begin{aligned} -\epsilon_{\mathrm{s}} \Delta\psi - \frac{\partial\psi}{\partial x} &= F \quad \text{in} \quad \Omega, \\ \psi &= 0 \quad \text{on} \quad \Gamma \end{aligned} \tag{13}$$

where ϵ_{s} is the Stommel number defined in (12). Unlike the Stommel–Munk model, the Stommel model is a second-order PDE and only $\psi = 0$ is imposed on the boundary.

For convenience, we define $\Lambda = \Lambda_i \cup \Lambda_b$ where Λ_i is the set of N_{i} interior nodes and Λ_b is the set of N_{b} boundary nodes, with $N_{\mathrm{i}} + N_{\mathrm{b}} = N$ total nodes. If $\Phi_{IJ}^{\boldsymbol{\alpha}_k}$ is used to denote the Jth entry of the $\boldsymbol{\alpha}_k$th derivative operator at $\mathbf{x}_I$, then substituting (10) into (11) yields the discrete form of the Stommel–Munk PDE:

$$\begin{aligned} \sum_{J=1}^{N} \{ &-\epsilon_{\mathrm{s}} [\Phi_{IJ}^{(2,0)} + \Phi_{IJ}^{(0,2)}] \\ &+ \epsilon_{\mathrm{m}} [\Phi_{IJ}^{(4,0)} + 2\Phi_{IJ}^{(2,2)} + \Phi_{IJ}^{(0,4)}] - \Phi_{IJ}^{(1,0)} \} \psi_J = F(\mathbf{x}_I) \end{aligned} \tag{14}$$

for each $\mathbf{x}_I \in \Lambda_i$. The boundary conditions are similarly discretized as

$$\sum_{J=1}^{N} \Phi_{IJ}^{(0,0)} \psi_J = 0, \qquad \sum_{J=1}^{N} [\Phi_{IJ}^{(1,0)} n_1 + \Phi_{IJ}^{(0,1)} n_2] \psi_J = 0 \tag{15}$$

for each $\mathbf{x}_I \in \Lambda_b$. If we define $\mathbf{F}^{\mathrm{i}} \equiv \{F(\mathbf{x}_1), F(\mathbf{x}_2), \ldots, F(\mathbf{x}_{N_i})\}^{\mathrm{T}}$, then these discretized equations may be assembled into an $(N_i + 2N_b) \times N$ system

$$\mathbf{K}\boldsymbol{\psi} = \mathbf{F} \tag{16}$$

where

$$\mathbf{K} = \begin{bmatrix} \mathbf{K}^{\mathrm{i}} \\ \mathbf{K}^{\mathrm{d}} \\ \mathbf{K}^{\mathrm{n}} \end{bmatrix} \qquad \text{and} \qquad \mathbf{F} = \begin{bmatrix} \mathbf{F}^{\mathrm{i}} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}. \tag{17}$$

In (17), the (I, J) entry of the $N_i \times N$ block K^{i} is given by

$$\begin{aligned} K_{IJ}^{\mathrm{int}} = &-\epsilon_{\mathrm{s}} [\Phi_{IJ}^{(2,0)} + \Phi_{IJ}^{(0,2)}] + \\ &\epsilon_{\mathrm{m}} [\Phi_{IJ}^{(4,0)} + 2\Phi_{IJ}^{(2,2)} + \Phi_{IJ}^{(0,4)}] - \Phi_{IJ}^{(1,0)} \end{aligned} \tag{18}$$

The (I, J) entry of the $N_b \times N$ block K^{d} is given by

$$K_{IJ}^{\mathrm{d}} = \Phi_{IJ}^{(0,0)} \tag{19}$$

Finally, the (I, J) entry of the $N_b \times N$ block K^{n} is given by

$$K_{IJ}^{\mathrm{n}} = \Phi_{IJ}^{(1,0)} n_1 + \Phi_{IJ}^{(0,1)} n_2 \tag{20}$$

Similarly, by substituting (10) into (13), the discrete form of the Stommel equation and its boundary conditions can be obtained as

$$\begin{aligned} &\sum_{J=1}^{N} \{-\epsilon_{\mathrm{s}} [\Phi_{IJ}^{(2,0)} + \Phi_{IJ}^{(0,2)}] - \Phi_{IJ}^{(1,0)} \} \psi_J = F(\mathbf{x}_I), \quad \text{for } \mathbf{x}_I \in \Lambda_i \\ &\qquad \text{and} \qquad \sum_{J=1}^{N} \Phi_{IJ}^{(0,0)} \psi_J = 0, \quad \text{for } \mathbf{x}_I \in \Lambda_d. \end{aligned} \tag{21}$$

These equations can also be assembled into a system (16), this time $N \times N$, with

$$\mathbf{K} = \begin{bmatrix} \mathbf{K}^{\mathrm{i}} \\ \mathbf{K}^{\mathrm{d}} \end{bmatrix} \qquad \text{and} \qquad \mathbf{F} = \begin{bmatrix} \mathbf{F}^{\mathrm{i}} \\ \mathbf{0} \end{bmatrix}. \tag{22}$$

where the (I, J) entries of the $N_i \times N$ block K^{i} and the $N_b \times N$ block K^{d} are given by

$$K^{\mathrm{i}}_{IJ} = -\epsilon_{\mathrm{s}}[\Phi^{(2,0)}_{IJ} + \Phi^{(0,2)}_{IJ}] - \Phi^{(1,0)}_{IJ}, \quad K^{\mathrm{d}}_{IJ} = \Phi^{(0,0)}_{IJ}. \tag{23}$$

3. Numerical results

To verify our method, we perform numerical studies on three benchmark problems commonly used in geophysical fluid dynamics [2]. The first two problems involve the Stommel model on a rectangular domain with a strong Western boundary layer and the Stommel–Munk model without a strong Western boundary layer, presented in 3.1 and 3.2. The results from these studies motivate an examination of the effect of numerical parameters of the proposed method on error behavior in the context of these benchmark problems, presented in 3.3. After this investigation, results from the third benchmark problem, the Stommel–Munk equation on a rectangular domain with a strong Western boundary layer, are discussed in 3.4. This includes results from applying a local refinement of the discretization. Finally, the proposed method is used to solve the Stommel–Munk equation on a polygonal domain representing the Mediterranean Sea in 3.5.

In the subsequent studies, we analyze convergence and error behavior by defining the following discrete L^2-norm and L^∞-norm errors:

$$\|e\|_2 \equiv \sqrt{\frac{\Sigma_{i=1}^N (\psi_i^h - \psi_i)^2}{\Sigma_{i=1}^N (\psi_i)^2}} \tag{24}$$

and

$$\|e\|_\infty \equiv \frac{\max|\psi_i^h - \psi_i|}{\max|\psi_i|}, \tag{25}$$

respectively, where ψ^h is the approximation of the exact solution ψ. In (6), the weight function does not need to be differentiable. Hence, all simulations are performed using the weight function

$$w\left(\frac{\mathbf{x}_I - \mathbf{y}}{\rho(\mathbf{y})}\right) = \left(1 - \left\|\frac{\mathbf{x}_I - \mathbf{y}}{\rho(\mathbf{y})}\right\|\right)^4. \tag{26}$$

The dilation parameter ρ, which represents the radius of the domain of influence of each collocation point, should be chosen so that the resulting differential operators are sparse to save computational cost. On the other hand, ρ should be large enough so that the $\mathbf{M}$ matrices are nonsingular and the solution is accurate. In general, there is some minimum number of points in a domain of influence such that all $\mathbf{M}$ matrices (8) are nonsingular [23]. This number of points dictates the radius of influence of each collocation point based on the spacing of nodes and the geometry of the domain. More discussion of the dilation parameter related to a specific example can be found in Section 3.3.

3.1. Stommel model

We consider the test problem (13) with the exact solution

$$\psi(x, y) = \frac{\sin(\pi y)}{\pi(1 + 4\pi^2\epsilon_{\mathrm{s}}^2)}\left[2\pi\epsilon_{\mathrm{s}}\sin(\pi x) + \cos(\pi x) + \frac{(1 + e^{R_2})e^{R_1 x} - (1 + e^{R_1})e^{R_2 x}}{e^{R_1} - e^{R_2}}\right] \tag{27}$$

over the domain $\Omega = [0, 1] \times [0, 1]$, where R_1 and R_2 are given by

$$R_1 = \frac{-1 + \sqrt{1 + 4\pi^2\epsilon_{\mathrm{s}}^2}}{2\epsilon_{\mathrm{s}}} \quad \text{and} \quad R_2 = \frac{-1 - \sqrt{1 + 4\pi^2\epsilon_{\mathrm{s}}^2}}{2\epsilon_{\mathrm{s}}}. \tag{28}$$

This example was used for the test of an algorithm by Myers and Weaver [7], Foster et al. [10], and Kim et al. [14]. Upon taking $\epsilon_{\mathrm{s}} = 0.05$, we work in a setting identical to that considered in these references. The forcing term F is chosen to match with that given by the exact solution (27). As shown in Figs. 1 and 2, a rectangular ocean is chosen as a computational domain. With the origin of a Cartesian coordinate system at the southwest corner, the x- and y-axis point eastward and northward, respectively, and the boundaries of the computational domain are the shores of the ocean.

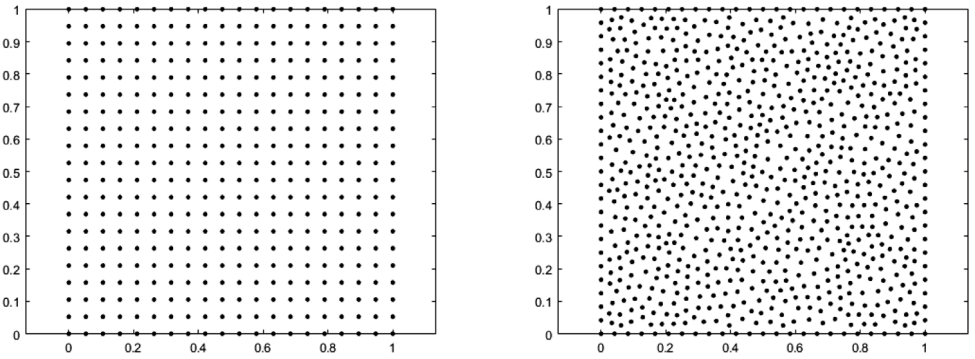

Fig. 1. Examples of (a) a uniform distribution of 400 collocation points and (b) a random distribution of 821 collocation points.

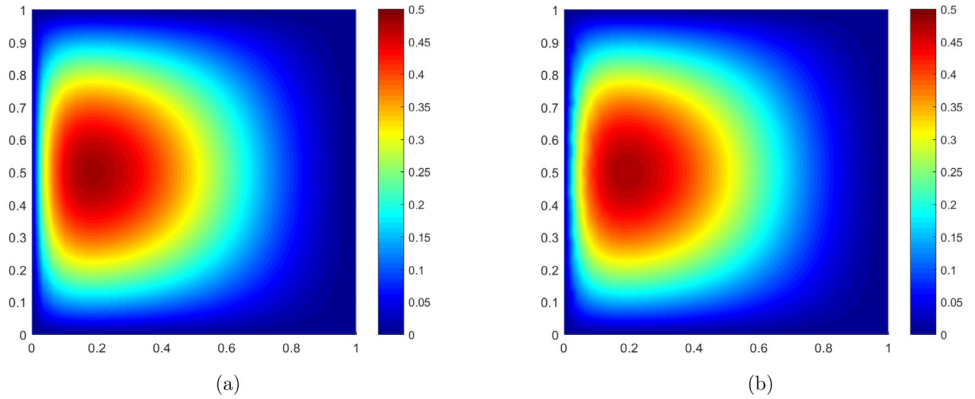

Fig. 2. Heat map of the streamfunction for the Stommel model: (a) the numerical solution and (b) the exact solution given by (27).

In Fig. 1, we display uniform and random distributions of collocation points used for this study. In Fig. 2, we compare the streamlines of the numerical solution using the randomly distributed collocation points with the exact solution. Both solutions are qualitatively indistinguishable from the exact solution. Notice that the similar solution using the uniform distribution of collocation points is observed.

To study convergence rates as a function of average nodal distance h, $\|e\|_2$ and $\|e\|_\infty$ are measured for approximately $h = \{0.020, 0.017, 0.014, 0.012\}$. Our method and the finite-difference method (FDM) both use a Taylor expansion to approximate a solution variable. Thus, based on such a similarity with the FDM, we expect that our approach also has an optimal order of convergence equal to the order of the Taylor expansion used. In other words, the optimal convergence rates for our method would be quadratic or quartic for the second-order and fourth-order polynomials, respectively. Theoretical study for the error analysis of the proposed method remains as the authors' future work while some theoretical works for similar types of strong form meshfree methods can be found in [23,37]. For the second-order polynomial, Tables 2 and 3 show the convergence rates for uniformly and randomly distributed collocation points in both L^2- and L^∞-norm errors. The plots of convergence rate in L^2-norm for both second-order and fourth-order polynomials are displayed in Figs. 3 and 4. Note that the presented convergence analysis results, such as Fig. 3, adopted log–log plots, although it is sometimes hard to recognize this due to the small range of values on the horizontal axis.

In Table 4, we summarize the convergence rates for the second-order and fourth-order polynomials for the Stommel model with the exact solution given by (27). The results show higher than the expected convergence rates in both L^2-norm and L^∞-norm.

Table 2
Convergence rates in L^2-norm $\|e\|_2$ and L^∞-norm $\|e\|_\infty$ with a second-order polynomial using uniform distributions of collocation points.

# of collocation points	$\|e\|_2$	$\|e\|_\infty$
2500	5.20e−03	1.27e−02
3600	3.38e−03	7.86e−03
4900	2.36e−03	5.21e−03
6400	1.73e−03	3.63e−03
Convergence rate	2.31	2.61

Table 3
Convergence rates in $\|e\|_2$ and $\|e\|_\infty$ with a second-order polynomial using random distributions of collocation points.

# of collocation points	$\|e\|_2$	$\|e\|_\infty$
2212	7.16e−03	2.24e−02
3297	4.52e−03	1.31e−02
4705	2.92e−03	7.57e−03
6465	2.02e−03	5.03e−03
Convergence rate	2.37	2.82

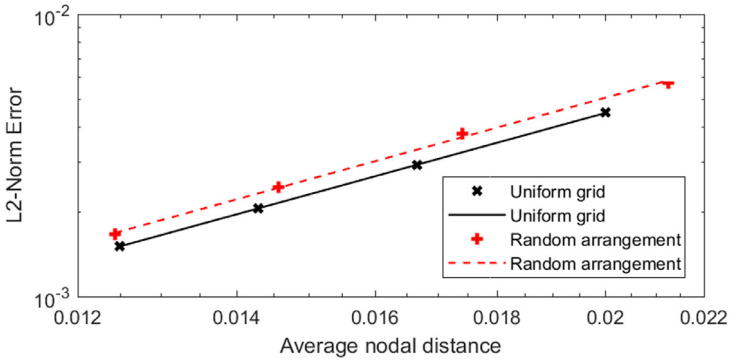

Fig. 3. Convergence plot in $\|e\|_2$ for the Stommel model using the 2nd-order polynomial approximation. The slopes of the regression lines are 2.31 (uniform) and 2.37 (random).

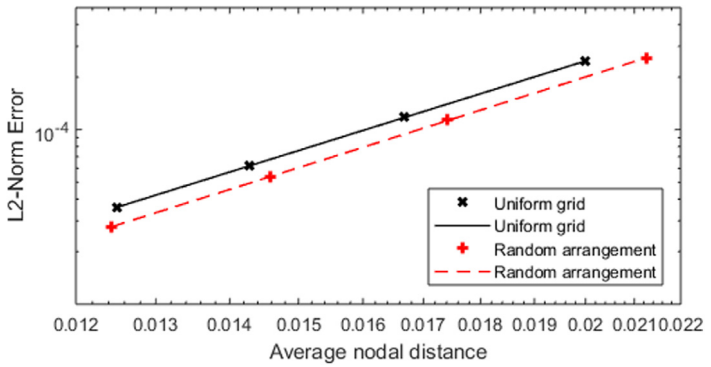

Fig. 4. Convergence plot in $\|e\|_2$ for the Stommel model using the 4th-order polynomial approximation. The slopes of the regression lines are 4.11 (uniform) and 4.16 (random).

Table 4
Convergence rates in $\|e\|_2$ and $\|e\|_\infty$ with a second-order polynomial using random distributions of collocation points.

Nodal arr.	2nd order		4th order	
	$\|e\|_2$	$\|e\|_\infty$	$\|e\|_2$	$\|e\|_\infty$
Uniform grid	2.31	2.61	4.11	3.99
Random	2.37	2.82	4.16	4.07

3.2. The Stommel–Munk model without western boundary layer

In contrast to the Stommel model, a challenge in solving the fourth-order Stommel–Munk model is the need to apply two Dirichlet boundary conditions $\psi = \nabla\psi \cdot \mathbf{n} = 0$ on the entire boundary. In applying the proposed method to a second-order PDE with only one boundary condition, nodal solutions are found by solving a linear system in which each equation is a discretized version of (11) (for each of N_i interior collocation points) or of the boundary condition (for N_b boundary nodes). In this way, boundary conditions are applied directly, and the resulting system of equations is square. However, for the Stommel–Munk model, to apply both boundary conditions at each boundary node, two separate equations must be written for each boundary node as shown in (15). This leads to a total of $N_\mathrm{i} + 2N_\mathrm{b}$ equations in $N_\mathrm{i} + N_\mathrm{b}$ unknown nodal values of ψ, i.e., an overdetermined system of equations.

Our approach is to solve this overdetermined system by the method of weighted least squares. Introducing a diagonal matrix $\mathbf{D}$ containing positive weight values along its diagonal into the system of equations (16), the solution $\boldsymbol{\psi}$ that minimizes the norm $(\mathbf{K}\boldsymbol{\psi} - \mathbf{F})^T\mathbf{D}^2(\mathbf{K}\boldsymbol{\psi} - \mathbf{F})$ is obtained by solving

$$\mathbf{K}^T\mathbf{D}^2\mathbf{K}\boldsymbol{\psi} = \mathbf{K}^T\mathbf{D}^2\mathbf{F}. \tag{29}$$

Why might it be advantageous to use some weight values in $\mathbf{D}$ other than 1.0 (standard least squares)? As an example, consider a problem in which the error between the numerical solution and the analytical solution is highest at the boundary when standard least squares is used. Then, increasing the weight values applied to the boundary condition equations penalizes this higher error at the boundary more heavily than does standard least squares, potentially resulting in lower maximum or L^2-error of the numerical solution. A discussion of how weights are chosen for the problems in this study may be found in Section 3.3.

Using this weighted least squares strategy, we test the Stommel–Munk model on a rectangular ocean in similar fashion to the Stommel model. We consider a benchmark example that is commonly used to test a finite-element algorithm [8,14]. This example has a forcing F corresponding to the exact solution given by

$$\psi(x, y) = \sin^2(\pi x/3)\sin^2(\pi y) \quad \text{in } \Omega = [0, 3] \times [0, 1]. \tag{30}$$

The Stommel and Munk numbers are chosen to be $\epsilon_\mathrm{s} = 0.05$ and $\epsilon_\mathrm{m} = 6.0 \times 10^{-5}$, respectively.

To approximate the fourth-order derivative operators, at least a fourth-order polynomial is required. As a result, we use fourth-order polynomials in this study. When assembling the overdetermined system of equations in the solution of this problem, weights of 1.0 are chosen for the rows corresponding to interior collocation points, whereas weights of 0.2 are used for the rows corresponding to boundary points. A more thorough discussion of this choice of relative weights may be found in Section 3.3.

In Fig. 5, we display the streamlines of both numerical and exact solutions for the fourth-order polynomial. Both results are qualitatively indistinguishable. In Fig. 6, the rates of convergence in L^2-norm for both uniform and random distributions of collocation points are presented. Moreover, convergence rates in both L^2 and L^∞-norms are summarized in Table 5. While the order of convergence in L^2-norm is quartic, slightly lower convergence rate is observed in L^∞-norm.

3.3. Study of error behavior for numerical parameters

While obtaining results for the Stommel–Munk model above with the exact solution given by (30), it became apparent that the error behavior of the numerical solution is sensitive to the least squares weight values chosen for the diagonal matrix $\mathbf{D}$ in (29) and the number of neighbors chosen for the compact support of each collocation

Table 5
Convergence rates in $\|e\|_2$ and $\|e\|_\infty$ with a fourth-order polynomial using uniform and random distributions of collocation points for the test problem with the exact solution (30).

Nodal arr.	$\|e\|_2$	$\|e\|_\infty$
Uniform grid	4.20	3.87
Random	4.06	3.74

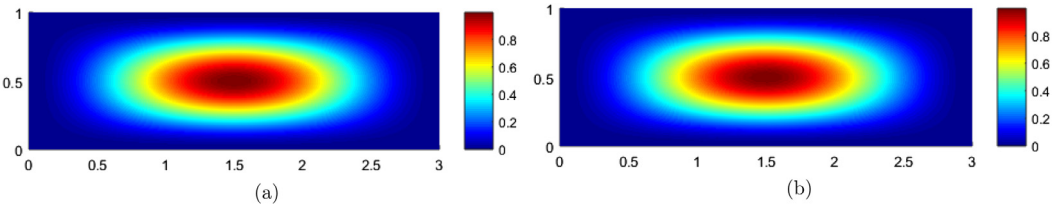

Fig. 5. Comparison between (a) numerical and (b) exact solutions of the Stommel–Munk model for the test problem with the exact solution (30).

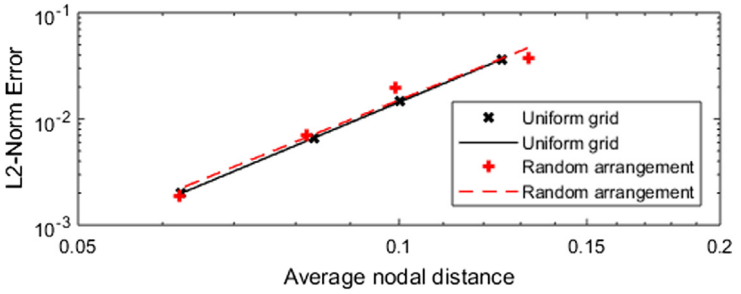

Fig. 6. Convergence in $\|e\|_2$ for the 4th-order polynomial approximation for the problem with the exact solution given by (30). The slopes of the regression lines are 4.20 (uniform) and 4.06 (random).

point. This section explores the effect of these numerical parameters on error behavior of the proposed collocation method.

First, we conduct a numerical experiment to determine the effect of weight values on the error behavior. The Stommel–Munk model with the exact solution (30) is used as a basis for this study. The number of neighbors in the compact support is held fixed at 35. Weights corresponding to the interior nodes are fixed at 1.0, while weights corresponding to the boundary nodes are varied between 0.001 and 100. We study the order of convergence in both L^2-norm and L^∞-norm for both the uniform grid and random arrangements. The order of convergence is based on the following ranges of total numbers of collocation points: For the uniform grid, error is computed for 243, 300, 432, and 675 total points; for the random arrangement, error is computed for 172, 304, 448, and 775 total points. In addition to convergence rates, the values of $\|e\|_2$ and $\|e\|_\infty$ are recorded for a fixed number of collocation points (675 in the uniform grid case and 775 in the random arrangement case). The results are displayed in Figs. 7 and 8.

For the uniform grid, the highest rate of convergence appears to occur when the boundary condition (BC) weights are less than 1.0, but this choice produces largest overall error of the solution for a fixed number of collocation points. A possible explanation for this phenomenon is the following: When BC weights are less than 1.0, the error tends to be high at the boundary and the weighted least squares method penalizes this high boundary error very little. As a consequence, for low BC weights, the overall spatial distribution of error is unbalanced with highest error at the boundary, causing the L^2-norm $\|e\|_2$ and L^∞-norm $\|e\|_\infty$ errors to be relatively high. However, increasing the density of the collocation points causes this boundary error to disappear rapidly, resulting in a high rate of convergence. Conversely, strictly penalizing boundary error with larger weight values results in lower overall error even for coarse discretizations, resulting in a lower rate of convergence. The spatial distributions of error for two BC weight values, shown in Fig. 9, bear out this hypothesis. It is noteworthy that a choice of weights around 1.0

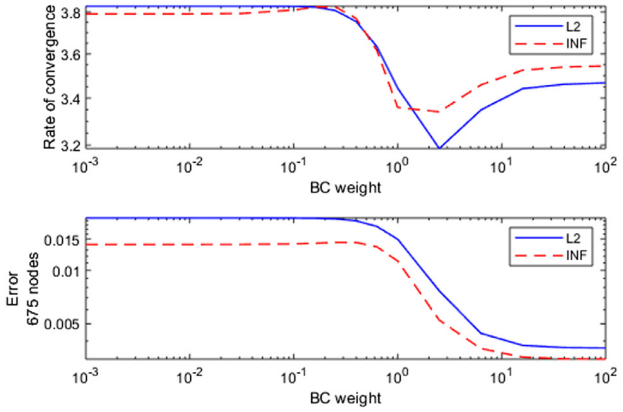

Fig. 7. Effect of BC weights on error behavior, uniform grid.

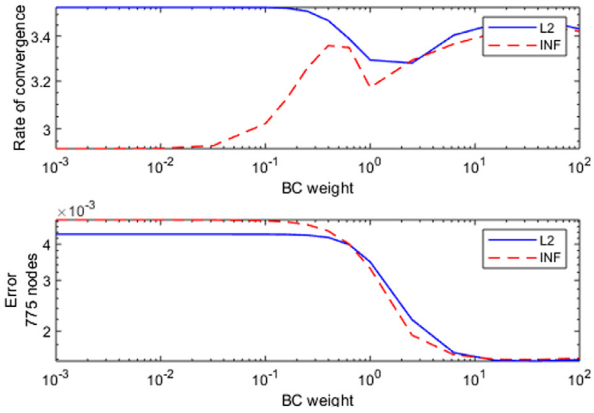

Fig. 8. Effect of BC weights on error behavior, random arrangement.

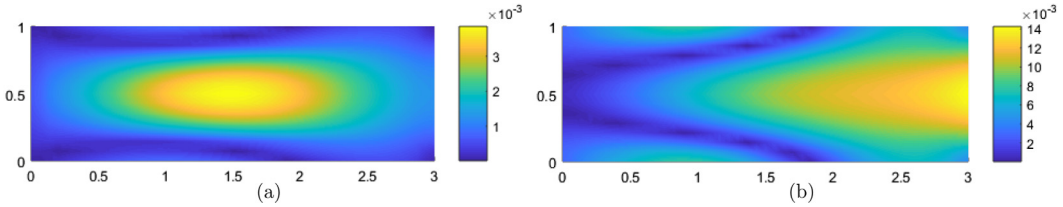

Fig. 9. Spatial distribution of absolute error for (a) BC weight = 5.0 and (b) BC weight = 0.2.

to 3.0 results in high error and in the lowest rates of convergence, suggesting that it can indeed be advantageous to vary weight values away from 1.0 for the boundary condition equations. Further research should investigate what characteristics of a problem (e.g., the forcing term) make it advantageous or disadvantageous to do so.

Similar error behavior occurs in the case of the random nodal arrangement, except with respect to the rate of convergence in the L^2-norm $\|e\|_2$ compared to the L^∞-norm $\|e\|_\infty$ errors for small values of BC weights. In this case, it is possible that the maximum value of the error is localized to a few isolated collocation points at the boundary, which would cause the maximum error to decrease slowly compared to the L^2-norm error as the density of the collocation points is increased. It is worth noting that the level of discretization and the boundary condition weights (i.e. the matrix D) both have a limited effect on the condition number of the matrix $\mathbf{K}^T\mathbf{D}^2\mathbf{K}$ in Eq. (29). However, proposed method has not exhibited a severe dependency on the condition numbers according to the authors' experience.

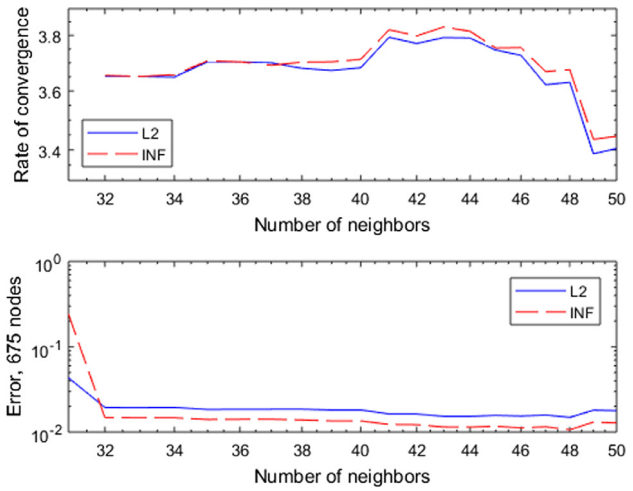

Fig. 10. Effect of the number of neighbors in compact support on error behavior, uniform grid.

We also conducted a numerical experiment to determine the effect on error behavior of the number of neighbors chosen for the compact support of each collocation point. As before, the problem with the exact solution given by (30) was used. The weights applied to the interior node equations were again fixed at 1.0, but this time the BC weights were fixed at 0.5 for all boundary nodes. The number of neighbors specified for the compact support of each collocation point was varied between 30 and 50. In each trial, the dilation parameter at each collocation point was taken to be the distance between the collocation point and the farthest of its 30–50 neighbors identified using the k-nearest neighbors (KNN) search algorithm [38]. An alternative algorithm to construct spatially varying continuous compact support function, which uses a pseudo-counting function constructed based on collocation point density, is described in [39]. Again, the convergence rates and error values were measured for each choice of number of neighbors, for a uniform grid of points. The results are shown in Fig. 10.

The highest rates of convergence were achieved when 41–43 neighbors were used. Some of the lowest values of error for a fixed number of nodes were also achieved for this range of numbers of neighbors. These results suggest that, for a uniform grid of collocation points, there is an optimal number of neighbors to specify for the compact support. We propose the following reason for the sensitivity of the error behavior to this parameter: Depending on the number of neighbors chosen, the spatial pattern of neighbor nodes in a compact support will be one of a variety of shapes. Thus, just as the error behavior of the FDM depends on the choice of a five-point stencil versus a nine-point stencil due to the relative positions of the neighbors and their effect on the numerical solution, so too is the present method sensitive to the choice of spatial pattern of neighborhood nodes.

Based on the studies discussed in this section, values of number of neighbor nodes and BC weights were chosen so as to optimize the convergence rate for the above problem, leading to the results displayed in Table 5 from the previous section.

3.4. Stommel–Munk model with Western boundary layer

Having studied the effect of numerical parameters on error behavior, we are able to use our method to solve the Stommel–Munk model with a different forcing term. This problem has a forcing corresponding to the exact solution given by

$$\psi(x, y) = \left[(1 - x/3)(1 - e^{-20x}) \sin(\pi y)\right]^2 . \tag{31}$$

In contrast to the previous example in Section 3, this one has a strong Western boundary layer as shown in Fig. 11. This boundary layer results in a region of very high gradient on the left side of the domain, which is difficult to capture. For this reason, fourth- and fifth-order polynomials are unable to achieve reliable solutions due to instability on the left side of the domain. Thus, a sixth-order polynomial is used. Fortunately, the jump in

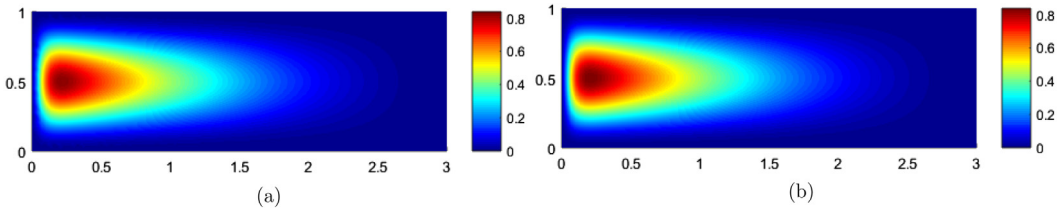

Fig. 11. Comparison between (a) numerical and (b) exact solutions of the Stommel–Munk model with the Western boundary layer.

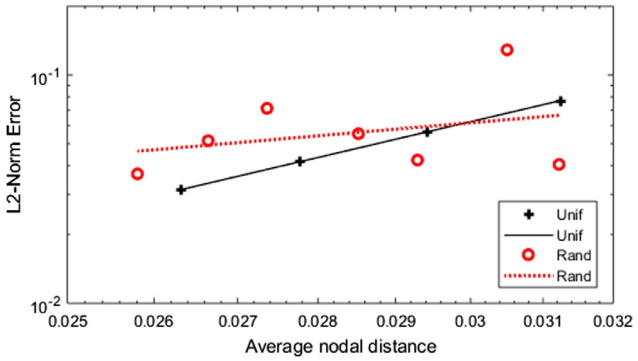

Fig. 12. Convergence in $\|e\|_2$ for the Stommel–Munk model with the western boundary layer.

computational cost due to using a sixth-order polynomial approximation rather than a fourth-order approximation is relatively small because the number of small matrices needing to be inverted while computing the differential operators remains unchanged.

A comparison of the numerical solution with the exact solution shows qualitative agreement, as shown in Fig. 11. Moreover, as shown in Table 6, for the uniform grid arrangement of collocation points, slightly lower than optimal (sixth-order) convergence rate is obtained in both $\|e\|_2$ and $\|e\|_\infty$ due to the presence of the Western boundary layer. The convergence for uniform collocation points is remarkably steady, as shown in Fig. 12. For the random arrangement of collocation points, low convergence rates and largely unpredictable error behavior are observed due to local instabilities in the solution.

To improve accuracy, local refinement is applied to the left side of the domain where the strong Western boundary layer is found. This is done for both uniform and random arrangements of collocation points. For the uniform grid, we locally refined the domain by placing an extra collocation point at the centroid of each group of four existing nodes in the left sixth of the domain. For the random arrangement, the left sixth of the domain had an average nodal distance three times that of the right side of the domain; the `gmsh` software allowed a smoother transition between the right side of the domain and the locally refined area on the left than in the case of the uniform grid. Examples of the refined nodal arrangements used are given in Fig. 14.

Fig. 13 shows the convergence plots in $\|e\|_2$ for uniform and random arrangements with and without local refinement. Table 6 contains the rates of convergence for these various schemes in $\|e\|_2$ and $\|e\|_\infty$. It should be noted that in the case of both uniform and random arrangements, the rate of convergence for the refined nodal arrangements is much better than that for the uniform-density nodal arrangements. In addition, the errors for the refined random arrangement are the lowest of all the schemes, suggesting that properly implemented refinement would improve both the accuracy and the rate of convergence of the numerical solution for a problem with a high-gradient region such as this one.

As shown in Fig. 13, instability (or suboptimal convergence) is exhibited by the proposed method for the Western boundary layer example, especially for the random arrangements of collocation points. As explained in [37], this is expected since the proposed method's formulation is based on the strong form and avoids the Galerkin framework. Currently, to the best of the authors' knowledge, there are no theoretical stability and perturbation analyses available for the proposed method other than numerical studies of the method. More numerical results

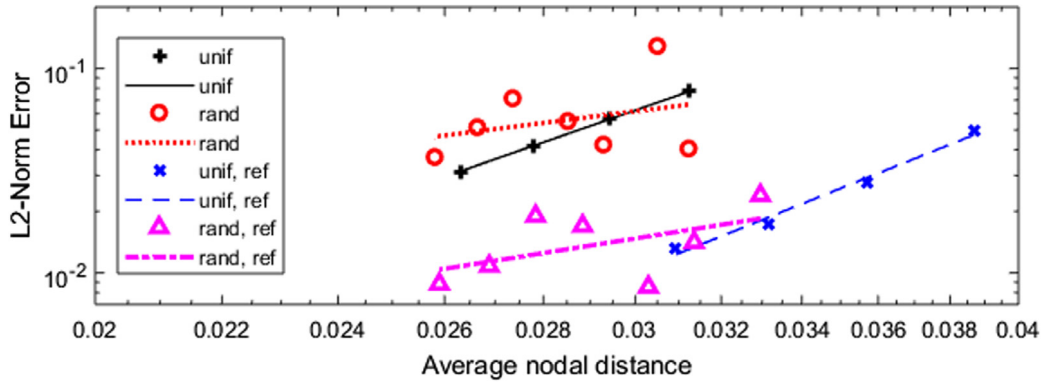

Fig. 13. Influence of local refinement for the Stommel–Munk model with the western boundary layer.

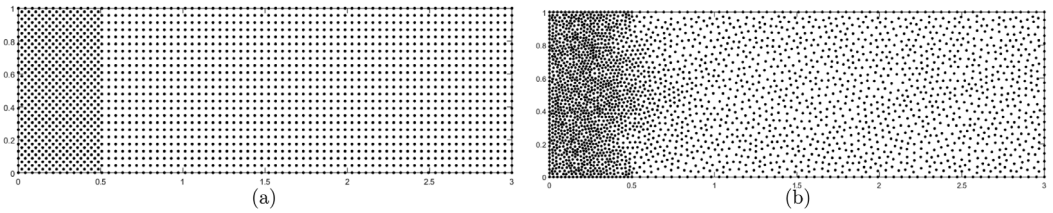

Fig. 14. Illustration of nodal arrangements for local refinement applied to (a) uniformly spaced collocation points and (b) randomly spaced collocation points.

Table 6
Rate of convergence (ROC) in $\|e\|_2$ and $\|e\|_\infty$ with local refinement for both uniform and random distributed collocation points.

Scheme	ROC in $\|e\|_2$	ROC in $\|e\|_\infty$
Unif.	5.23	5.78
Rand.	1.90	1.27
Unif. ref.	6.02	7.65
Rand. ref.	2.39	2.81

for the discretization sensitivity-induced instability of the method within similar contexts can be found in authors' previous works [32,34,36]. Theoretical study for the error and stability analysis of the proposed method remains part of the authors' future work.

3.5. Wind-driven ocean circulation in the Mediterranean Sea

To demonstrate the usefulness of the proposed method in solving real-world problems on arbitrary shaped domains, the method is used to solve the Stommel–Munk model on the interior of a polygon representing the coastlines of the Mediterranean sea. This example was studied by Foster et al. [10] and Jiang and Kim [16] for the test of their finite-element formulations of the stationary quasi-geostrophic equations. We use the forcing term $F = \sin(\pi y/4)$ arising from the derivative of the wind stress (Myers and Weaver [7]). The same values of the Stommel and Munk numbers (0.05 and 6.0×10^{-5}, respectively) are used. Notice that an analytical solution is not available, so a convergence study cannot be performed.

This polygon shown in Fig. 15 encloses a simply connected region. However, this region's concavity and various narrow subregions present challenges to solving the model on its interior. The first challenge in solving the Stommel–Munk model on this domain is that of generating an arrangement of collocation points for the Mediterranean sea region. Without an unreasonable degree of refinement, the use of a uniform tensor-product grid of points would have been insufficient to capture the geometry of the region. Even in the case of the rectangular domain with

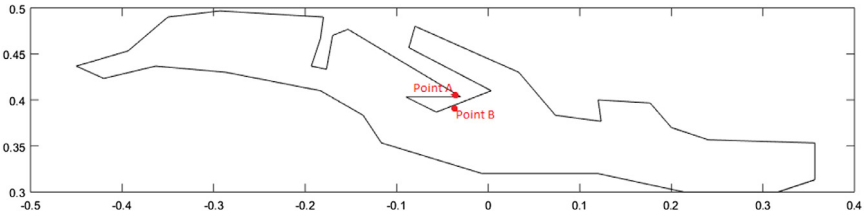

Fig. 15. Polygon representing the boundary of the Mediterranean Sea.

Western boundary layer in Section 3.4, the random arrangement with selective refinement resulted in consistently lower error than the uniform grid with no refinement, as shown in Fig. 13. The random arrangement was also less cumbersome to produce than the uniform grid with selective refinement. For domains with complicated geometry, producing a uniform grid with selective refinement is much more difficult and tedious, while producing a random arrangement with selective refinement remains straightforward using readily available meshing software. Thus, for complicated geometries, a selectively refined random arrangement of collocation points is more practical than a selectively refined uniform arrangement, but still exhibits less error for a given number of collocation points than a uniform grid with no selective refinement. Thus, generating a random arrangement of nodes is preferred and left to the open-source meshing software `gmsh` [40]. Each segment of the polygon boundary is assigned a number of boundary collocation points approximately proportional to this segment's fraction of the total length of the boundary. If this results in a number of boundary nodes that is too small for this segment (e.g., only two or three nodes), this segment is assigned five nodes. Given these specifications, `gmsh` generated the arrangement of collocation points shown in Fig. 16.

After discretizing the region, another challenge is to ensure that the domain of influence of each collocation point is reasonable despite the concavity of some parts of the domain. For example, consider points A and B in Fig. 15. Although these points are very close to each other, the solution of the PDE at point A cannot reasonably be expected to affect the solution at point B because these points are separated by land. Thus, the domain of influence of point A should exclude point B and vice-versa. To accomplish such exclusions, the domain is divided into the subregions as shown in Fig. 17. Then, in selecting the points to include in the domain of influence of a given point D, only candidate points from the subregion containing point D or adjacent subregions are considered.

Using MATLAB's `knnsearch` accomplishes this task of selecting points for the domain of influence of each point. Each domain of influence contains 18 collocation points, which proved the optimal number of points in terms of ensuring invertibility of the $\mathbf{M}$ matrix (8), stability of the solution field at the boundary, and reasonable computational time. Finally, the value chosen for the dilation parameter ρ at each collocation point D was simply the distance between point D and the farthest-away point in the domain of influence of point D. An illustration of the domain of influence for a point in this problem and related ideas is shown in Fig. 18. Once these tasks are complete, we solve the Stommel–Munk model using the point collocation method. In order to balance the errors between the boundary and the interior, we choose BC weights corresponding to the boundary condition $\mathbf{u} = 0$ as 100 times those corresponding to the zero-flux boundary condition $\nabla\mathbf{u} \cdot \mathbf{n} = 0$. A similar scheme weighting the Dirichlet boundary condition equations more heavily than the Neumann boundary condition equations was used in the weighted collocation method of Chen et al. [41]; one should refer to the guidelines presented in [41] to choose the proper values of BC weights for a general problem. In Fig. 19, we display a heat map of the streamfunction. The plot shows qualitative agreement with the one obtained using finite-element methods by Foster et al. [10] and Jiang and Kim [16]. This result verifies the capability of our method on realistic ocean circulation problems with arbitrary shaped coastal boundaries.

4. Conclusion

In this paper, a new meshfree point collocation method was introduced to directly discretize the second-order Stommel model and the fourth-order Stommel–Munk model for the large-scale wind-driven ocean circulation simulations. The robustness and accuracy of the method was investigated with three benchmark problems commonly used for the study of oceanic flows, as well as a more realistic problem related to wind-driven ocean circulation in the Mediterranean Sea.

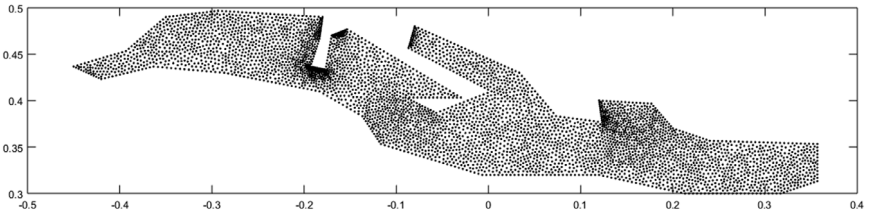

Fig. 16. Arrangement of 4834 collocation points used for the Mediterranean sea example.

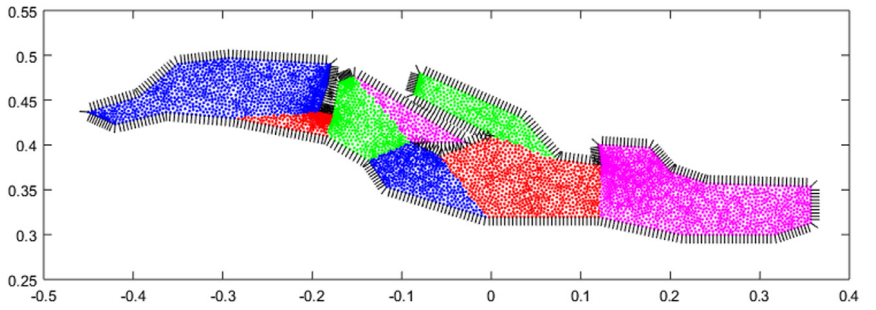

Fig. 17. Mediterranean Sea domain subdivided into cells in order to control the domain of influence of each collocation point. This plot also shows the direction of the normal vectors to the boundary.

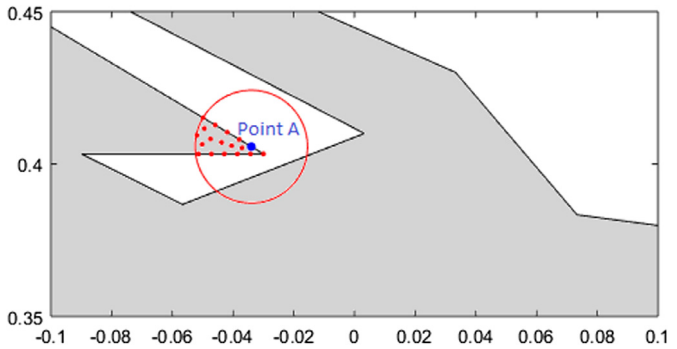

Fig. 18. Domain of influence of Point A (largest dot, in blue) from Fig. 15 containing 18 points. The radius of the circle is ρ for Point A. Note that the domain of influence does not contain points from the subregion across land from Point A.

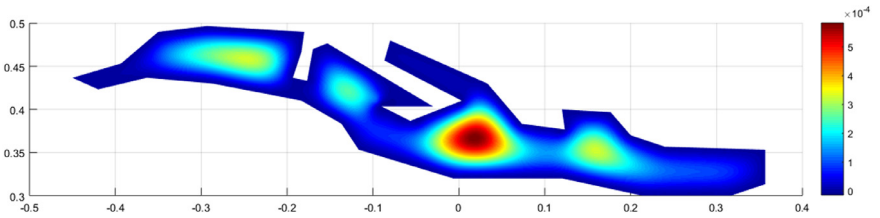

Fig. 19. Heat map of the PDM solution in the Mediterranean Sea example.

For each test problem, the numerical results exhibited near-optimal, and in some cases better than optimal, convergence. Furthermore, local refinement was shown to improve the accuracy and convergence rate for both uniform grid and random nodal arrangements in the context of a problem whose exact solution had a difficult-to-capture high-gradient region. Finally, it was demonstrated that the proposed method successfully solved the fourth-order Stommel–Munk model on a domain with complicated geometry, further implying the robustness of

the method. The results from this study strongly suggest that the proposed meshfree collocation method has the potential to predict the wind-driven ocean circulation on real complex domains with arbitrary shaped coastal lines.

In future work, the capability of the proposed method will be further explored by solving the nonlinear stationary and the time-dependent QGE. The differences between the present work and solving the QGE are the nonlinearity and time-dependent nature of the QGE. In this regard, it should be emphasized that the proposed method has been successfully verified within the similar contexts through the authors' previous works. Of course, yet another challenge lies in combining the nonlinear, time-dependent, and higher-order PDE capabilities of the proposed method to tackle the QGE. Building on this method's demonstrated ability to solve higher-order PDEs, we will also apply this method to solving other higher-order PDEs in other contexts, such as the phase field equations for solid-state chemical diffusion in an alloy at the microscale. In addition, further examinations of the mathematical properties of the method including conservative properties and error behaviors should be also conducted.

References

[1] H.A. Dijkstra, Nonlinear Physical Oceanography: A Dynamical Systems Approach to the Large Scale Ocean Circulation and El Niño, Springer–Verlag, 2005.
[2] G.K. Vallis, Atmosphere and Ocean Fluid Dynamics: Fundamentals and Large-scale Circulation, Cambridge University Press, 2006.
[3] J. Pedlosky, Geophysical Fluid Dynamics, Springer–Verlag, 1992.
[4] O. San, A.E. Staples, Z. Wang, T. Iliescu, Approximate deconvolution large eddy simulation of a barotropic ocean circulation model, Ocean Model. 40 (2011) 120–132.
[5] T.D. Ringler, J. Thuburn, J.B. Klemp, W.C. Skamarock, A unified approach to energy conservation and potential vorticity dynamics for arbitrary-structured c-grids, J. Comput. Phys. 229 (2010) 3065–3090.
[6] Q. Chen, T.D. Ringler, M. Gunzburger, A co-volume scheme for the rotating shallow water equations on conforming non-orthogonal grids, J. Comput. Phys. 240 (2013) 174–197.
[7] P.G. Myers, A.J. Weaver, A diagnostic barotropic finite-element ocean circulation model, J. Atmos. Ocean. Technol. 12 (1995) 511–526.
[8] J.M. Cascón, G.C. Garcia, R. Rodriguez, A priori and a posteriori error analysis for a large-scale ocean circulation finite element model, Comput. Methods Appl. Mech. Engrg. 192 (2003) 5305–5327.
[9] G.J. Fix, Finite element models for ocean circulation problems, SIAM J. Appl. Math. 29 (3) (1975) 371–387.
[10] E.L. Foster, T. Iliescu, Z. Wang, A finite element discretization of the streamfunction formulation of the stationary quasi-geostrophic equations of the ocean, Comput. Methods Appl. Mech. Engrg. 261 (2013) 105–117.
[11] T.-Y. Kim, E.-J. Park, D. Shin, A C^0-discontinuous Galerkin method for the stationary quasi-geostrophic equations of the ocean, Comput. Methods Appl. Mech. Engrg. 300 (2016) 225–244.
[12] D. Kim, T.-Y. Kim, E.-J. Park, D. Shin, Error estimates of B-splined based finite-element methods for the stationary quasi-geostrophic equations of the ocean, Comput. Methods Appl. Mech. Engrg. 335 (2018) 225–272.
[13] N. Rotundo, T.-Y. Kim, W.L. Heltai, E. Fried, Error analysis of a B-spline based finite-element method for modeling wind-driven ocean circulation, J. Sci. Comput. 69 (1) (2016) 430–459.
[14] T.-Y. Kim, T. Iliescu, E. Fried, B-spline based finite-element method for the stationary quasi-geostrophic equations of the ocean, Comput. Methods Appl. Mech. Engrg. 286 (2015) 168–191.
[15] I. Al Balushi, W. Jiang, G. Tsogtgerel, T.-Y. Kim, Adaptivity of a B-spline based finite-element method for modeling wind-driven ocean circulation, Comput. Methods Appl. Mech. Engrg. 332 (2018) 1–24.
[16] W. Jiang abd T.-Y. Kim, Spline-based finite-element method for the stationary quasi-geostrophic equations on arbitrary shaped coastal boundaries, Comput. Methods Appl. Mech. Engrg. 299 (2016) 144–160.
[17] T.J.R. Hughes, J.A. Cottrell, S.Y. Bazilevs, Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement, Comput. Methods Appl. Mech. Engrg. 194 (2005) 4135–4195.
[18] K. Höllig, Finite Element Methods with B-Splines, SIAM, Philadelphia, 2003.
[19] Y. Bazilevs, T.J.R. Hughes, Weak imposition of Dirichlet boundary conditions in fluid mechanics, Comput. & Fluids 36 (1) (2007) 12–26.
[20] S. Li, W.K. Liu, Synchronized reproducing kernel interpolant via multiple wavelet expansion, Comput. Mech. 21 (1998) 28–47.
[21] S. Li, W.K. Liu, Reproducing kernel hierarchical partition of unity, Part I-formulation and theory, Internat. J. Numer. Methods Engrg. 45 (1999) 251–288.
[22] S. Li, W.K. Liu, Meshfree and particle methods and their applications, Appl. Mech. Rev. 55 (2002) 1–34.
[23] D.W. Kim, Y. Kim, Point collocation method using the fast moving least-square reproducing kernel approximation, Internat. J. Numer. Methods Engrg. 56 (10) (2003) 1445–1464.
[24] M. Hillman, J.S. Chen, An accelerated, convergent, and stable nodal integration in galerkin meshfree methods for linear and nonlinear mechanics, Internat. J. Numer. Methods Engrg. 107 (2016) 603–630.
[25] T. Rabczuk, T. Belytschko, Cracking particles: a simplified meshfree method for arbitrary evolving cracks, Internat. J. Numer. Methods Engrg. 61 (2004) 2316–2343.
[26] T. Rabczuk, T. Belytschko, S.P. Xiao, Stable particle methods based on lagrangian kernels, Comput. Methods Appl. Mech. Engrg. 193 (2004) 1035–1063.
[27] C. Anitescu, Y. Jia, Y.Z. Zhang, T. Rabczuk, An isogeometric collocation method using superconvergent points, Comput. Methods Appl. Mech. Engrg. 284 (2015) 1073–1097.

[28] S.H. Lee, Y.C. Yoon, Meshfree point collocation method for elasticity and crack problems, Internat. J. Numer. Methods Engrg. 61 (1) (2004) 22–48.
[29] D.W. Kim, W.K. Liu, Y.C. Yoon, T. Belytschko, S.H. Lee, Meshfree point collocation method with intrinsic enrichment for interface problems, Comput. Mech. 40 (2007) 1037–1052.
[30] D.W. Kim, Y.C. Yoon, W.K. Liu, T. Belytschko, Extrinsic meshfree approximation using asymptotic expansion for interfacial discontinuity of derivative, J. Comput. Phys. 221 (2007) 370–394.
[31] Y.C. Yoon, J.H. Song, Extended particle difference method for weak and strong discontinuity problems: part I. Derivation of the extended particle derivative approximation for the representation of weak and strong discontinuities, Comput. Mech. 53 (6) (2014) 1087–1103.
[32] Y.C. Yoon, J.H. Song, Extended particle difference method for weak and strong discontinuity problems: part II. formulations and applications for various interfacial singularity problems, Comput. Mech. 53 (6) (2014) 1105–1128.
[33] Y.C. Yoon, J.H. Song, Extended particle difference method for moving boundary problems, Comput. Mech. 54 (3) (2014) 723–743.
[34] J.H. Song, Y. Fu, T.Y. Kim, Y.C. Yoon, J.G. Michopoulos, T. Rabczuk, Phase field simulations of coupled microstructure solidification problems via the strong form particle difference method, Int. J. Mech. Mater. Design 14 (2018) 491–509.
[35] Y. Fu, J.G. Michopoulos, J.H. Song, Bridging the multi phase-field and molecular dynamics models for the solidification of nano-crystals, J. Comput. Sci. 20 (2017) 187–197.
[36] Y.C. Yoon, P. Schaefferkoetter, T. Rabczuk, J.H. Song, New strong formulation for material nonlinear problems based on the particle difference method, Eng. Anal. Bound. Elem. 98 (2019) 310–327.
[37] H.Y. Hu, J.S. Chen, S.W. Chi, Perturbation and stability analysis of strong form collocation with reproducing kernel approximation, Internat. J. Numer. Methods Engrg. 88 (2011) 157–179.
[38] J.H. Friedman, J. Bentley, R.A. Finkel, An algorithm for finding best matches in logarithmic time, ACM Trans. Math. Softw. 3 (3) (1977) 209–226.
[39] D.W. Kim, H.K. Kim, Point collocation method based on the FMLSRK approximation for electromagnetic field analysis, IEEE Trans. Magn. 40 (2) (2004) 1029–1032.
[40] C. Geuzaine, J.-F. Remacle, Gmsh: a three-dimensional finite element mesh generator with built-in pre- and post-processing facilities, Internat. J. Numer. Methods Engrg. 79 (11) (2009) 1309–1331.
[41] J.S. Chen, W. Hu, H.Y. Hu, Reproducing kernel enhanced local radial basis collocation method, Internat. J. Numer. Methods Engrg. 75 (2007) 600–627.