

MESHFREE FINITE DIFFERENCES FOR VECTOR POISSON AND PRESSURE POISSON EQUATIONS WITH ELECTRIC BOUNDARY CONDITIONS

DONG ZHOU, BENJAMIN SEIBOLD, DAVID SHIROKOFF, PRINCE CHIDYAGWAI,
AND RODOLFO RUBEN ROSALES

ABSTRACT. We demonstrate how meshfree finite difference methods can be applied to solve vector Poisson problems with electric boundary conditions. In these, the tangential velocity and the incompressibility of the vector field are prescribed at the boundary. Even on irregular domains with only convex corners, canonical nodal-based finite elements may converge to the wrong solution due to a version of the Babuška paradox. In turn, straightforward meshfree finite differences converge to the true solution, and even high-order accuracy can be achieved in a simple fashion. The methodology is then extended to a specific pressure Poisson equation reformulation of the Navier-Stokes equations that possesses the same type of boundary conditions. The resulting numerical approach is second order accurate and allows for a simple switching between an explicit and implicit treatment of the viscosity terms.

1. INTRODUCTION

The numerical approximation of vector fields that are incompressible is often times challenging because incompressibility, $\nabla \cdot \mathbf{u} = 0$, is a global constraint that may not fit within the framework of simple discretization approaches of the complete problem. The instationary incompressible Navier-Stokes equations (NSE) represent a prime example, in which the time-evolution of the velocity field is given, but not of the pressure (which is a Lagrange multiplier associated with $\nabla \cdot \mathbf{u} = 0$). As a consequence, there is no single canonical way to advance the NSE forward in time. Similarly, in electrostatics the electric field and the magnetic potential are solutions to vector Poisson equations with divergence constraints.

One methodology to circumvent the incompressibility constraint inside the computational domain is to formulate a different problem that imposes $\nabla \cdot \mathbf{u} = 0$ as a boundary condition instead. Under certain circumstances, this new problem has the same solution as the original problem, while at the same time giving rise to new numerical approximation methods. For electrostatic problems (see §2) this approach is employed and analyzed in [23, 16], and in the context of incompressible fluid flows (see §3) it has been proposed in [30]. The specific boundary conditions for these problems consist of enforcing the tangential component(s) of the solution, together with the condition $\nabla \cdot \mathbf{u} = 0$. Due to their occurrence in electrostatics, they are often called (perfect) *electric boundary conditions* (EBC), a terminology that we follow in this paper.

A fundamental question is what are simple numerical approaches to approximate the solutions to vector-valued problems with EBC. In [30] an immersed boundary staggered grid

2000 *Mathematics Subject Classification.* 65M06; 65N06; 76M20; 35Q35.

Key words and phrases. meshfree, finite-differences, Navier-Stokes, incompressible, vector Poisson equation, pressure Poisson equation, reformulation, manufactured solution, high-order.

approach has been proposed. While convergent, this approach is definitely not a simple or canonical method. A seemingly more natural approach for problems on irregular domains is the standard nodal finite element method (FEM). Interestingly, for the class of problems at hand, nodal FEM can exhibit a Babuška paradox, i.e., for domains with curved boundaries the sequence of FEM approximations can converge to a wrong solution (see §2.2). Domains with re-entrant corners (see §2.1) pose additional challenges; however, this last aspect is not the focus of this paper. The convergence of the FEM can be recovered by converting to a mixed FEM formulation, as conducted in [5]. However, the simplicity of nodal FEM does not carry over to mixed FEM.

In this paper, we present yet another alternative for problems with EBC, namely meshfree finite differences (FD). These are generalizations of traditional grid-based FD that apply to clouds of points without any connectivity between them. The meshfree FD methodology is presented in §2.3 and §2.4, and its application to the vector Poisson equation with EBC is shown in §2.5. Structurally, the approximation of a general PDE boundary value problem via meshfree FD is very straightforward: any differential operator, whether in the domain's interior or on the boundary, is approximated via a meshfree FD stencil. Thus, the problem is directly transformed into a finite dimensional system, in which each individual equation corresponds to the governing condition that holds at a particular point of the cloud.

We demonstrate (in §2.5) that for the vector Poisson equation with EBC, meshfree finite differences do not exhibit the Babuška paradox, and furthermore that there is no conceptual problem to obtain higher-order accuracy (we test the method up to third order convergence).

We then move on to time-dependent problems. First, the meshfree FD method is extended to the vector heat equation with EBC (see §3.2), where an explicit or an implicit time-stepping can be conducted. Then, by adding nonlinear convective terms and a pressure, the approach is further extended to the incompressible Navier-Stokes equations (NSE). Specifically, we consider a pressure Poisson equation (PPE) reformulation of the Navier-Stokes equations. The idea of PPE reformulations (see §3) is that an operator function $p = P(\mathbf{u})$ is formulated that yields (via the solution of a Poisson equation) the pressure p to any given velocity field \mathbf{u} that solves the NSE. Here, we focus on a specific PPE reformulation, proposed in [30], which prescribes EBC for the fluid velocity, the motivation for which is outlined in §3.1. We demonstrate how a meshfree FD approximation for the full PPE reformulation can be constructed (see §3.3), and show computational results for a resulting numerical scheme that is second order accurate in space and time, and that allows for a choice of an explicit or an implicit treatment of the viscosity (see §3.4).

2. VECTOR POISSON EQUATION

The vector Poisson equation (VPE) arises, for instance, in problems in electrostatics. The electric field satisfies $\nabla \cdot \mathbf{E} = \rho$, where $\rho = \rho(x)$ is the (normalized) charge density. Using the fact that $\nabla \times \mathbf{E} = 0$, this implies the VPE $\Delta \mathbf{E} = \nabla \rho$. Moreover, if the boundaries of the domain are perfect conductors, then the vector field is perpendicular to the boundary, i.e., $\mathbf{n} \times \mathbf{E} = 0$, where \mathbf{n} is the outer surface normal vector. Another example is the magnetic potential, which satisfies the VPE $\Delta \mathbf{A} = -\mathbf{J}$, where $\mathbf{J} = \mathbf{J}(x)$ is the (normalized) electric current density. The Coulomb gauge yields $\nabla \cdot \mathbf{A} = 0$, and the boundary condition $\mathbf{n} \times \mathbf{A} = 0$ represents a zero magnetic field (see [9] for more details). Motivated by the structure

of these examples, we here consider the VPE

$$\begin{cases} -\Delta \mathbf{u} = \mathbf{f} & \text{in } \Omega \\ \nabla \cdot \mathbf{u} = 0 & \text{on } \Omega \\ \mathbf{n} \times \mathbf{u} = \mathbf{n} \times \mathbf{g} & \text{on } \partial\Omega. \end{cases} \quad (1)$$

As motivated in §1, it can be desirable to remove the divergence condition that holds in the whole domain. In the following, we outline how this can be achieved.

2.1. Electric Boundary Conditions. Let Ω be a bounded, simply connected domain with Lipschitz boundary $\partial\Omega$. Moreover, in this paper we restrict to domains with boundaries $\partial\Omega$ which are piecewise C^2 and convex (see Remark 1). We denote by \mathbf{n} the outward unit normal vector along the boundary (that is defined almost everywhere). The vector Poisson equation (VPE) with electric boundary conditions (EBC) takes the form

$$\begin{cases} -\Delta \mathbf{u} = \mathbf{f} & \text{in } \Omega \\ \nabla \cdot \mathbf{u} = 0 & \text{on } \partial\Omega \\ \mathbf{n} \times \mathbf{u} = \mathbf{n} \times \mathbf{g} & \text{on } \partial\Omega \end{cases} \quad (2)$$

where the source is incompressible, i.e., $\nabla \cdot \mathbf{f} = 0$. Note that in contrast to problem (1), problem (2) possesses no source-free condition in the domain's interior. Instead, $\nabla \cdot \mathbf{u} = 0$ is specified as an additional boundary condition. Clearly, any solution of (1) is also a solution of (2). Moreover...

Lemma 1. *If the solution to (2) is in $H^2(\Omega)$, then it is also a solution to (1).*

Proof. Define $\phi = \nabla \cdot \mathbf{u}$. Then ϕ is a (weak) solution of the problem

$$\begin{cases} \Delta \phi = 0 & \text{in } \Omega \\ \phi = 0 & \text{on } \partial\Omega, \end{cases}$$

which has the unique solution $\phi \equiv 0$. Hence $\nabla \cdot \mathbf{u} = 0$ in Ω . □

Remark 1. As shown in [18], the assumption of Lemma 1 is satisfied for the domains considered in this paper. However, it is not satisfied if the domain Ω has re-entrant (i.e., non-convex) corners. In such a case, the physically relevant (i.e., source-free) solution to (1) is not in H^1 , while problem (2) possesses a solution in H^1 , however, one that does not satisfy $\nabla \cdot \mathbf{u} = 0$ inside Ω . In this paper we exclude this possibility, and for the domains considered here (see above) the problems (1) and (2) are in fact equivalent (see [18] for a proof).

2.2. Nodal Finite Elements and Babuška Paradox. Among possible approaches to numerically approximate problem (2) on an irregular domain, standard nodal-based finite elements (FE) are one of the first ideas that would come to a numerical analyst's mind. Below, we derive two possible variational formulations (§2.2.1), and then use these to prove the possibility of the Babuška paradox (§2.2.2). Its actual occurrence is then demonstrated via a numerical example (§2.2.3).

2.2.1. Variational formulations. In order to conduct a FE approximation, a variational formulation of the VPE (2) must be introduced. It is natural to work with the affine Hilbert space of vector-valued H^1 functions that satisfy the tangential boundary condition in (2),

$$H_{gt}^1(\Omega)^N = \{\mathbf{u} \in H^1(\Omega)^N : \mathbf{n} \times (\mathbf{u} - \mathbf{g})|_{\partial\Omega} = 0\}.$$

Moreover, let $H_{0t}^1(\Omega)^N$ denote the associated homogeneous (i.e., $\mathbf{g} = 0$) Hilbert space. There are then two equivalent weak formulations of (2). To obtain the first formulation, we use the identity $-\Delta \mathbf{u} = \nabla \times (\nabla \times \mathbf{u}) - \nabla(\nabla \cdot \mathbf{u})$ and follow the standard procedure of multiplying the first equation in (2) by a test function $\mathbf{v} \in H_{0t}(\Omega)^N$, integrating by parts, and applying the boundary conditions to the boundary integral to obtain

$$\langle \mathbf{f}, \mathbf{v} \rangle = \int_{\Omega} -\Delta \mathbf{u} \cdot \mathbf{v} \, dx = a(\mathbf{u}, \mathbf{v}) - \int_{\partial\Omega} (\nabla \cdot \mathbf{u})(\mathbf{n} \cdot \mathbf{v}) \, dS, \quad (3)$$

where the bilinear form is

$$a(\mathbf{u}, \mathbf{v}) = \int_{\Omega} (\nabla \times \mathbf{u}) \cdot (\nabla \times \mathbf{v}) + (\nabla \cdot \mathbf{u})(\nabla \cdot \mathbf{v}) \, dx.$$

Based on this, the first variational formulation of (2) reads as: Given $\mathbf{f} \in L^2(\Omega)^N$ with $\nabla \cdot \mathbf{f} = 0$, find $\mathbf{u} \in H_{gt}(\Omega)^N$ such that for each $\mathbf{v} \in H_{0t}(\Omega)^N$

$$\text{(VP1)} \quad a(\mathbf{u}, \mathbf{v}) = \langle \mathbf{f}, \mathbf{v} \rangle.$$

Note that due to (3), the condition $(\nabla \cdot \mathbf{u})_{\partial\Omega} = 0$ arises as a natural boundary condition. It is this formulation **(VP1)** that we implement in the numerical test in §2.2.3.

In obtaining the second variational formulation, we restrict the derivation to the case $\mathbf{g} = 0$, because this case is enough to show that the Babuška paradox can arise. Assume for a moment that $\mathbf{u} \in H^2(\Omega)^N$. Then, using the fact that $\Delta \mathbf{u} = \nabla \cdot (\nabla \mathbf{u})$, we multiply the left hand side of (2) by $\mathbf{v} \in H_{0t}(\Omega)^N$ and integrate by parts to obtain

$$\langle \mathbf{f}, \mathbf{v} \rangle = \int_{\Omega} -\Delta \mathbf{u} \cdot \mathbf{v} \, dx = \int_{\Omega} \nabla \mathbf{u} \cdot \nabla \mathbf{v} \, dx - \int_{\partial\Omega} \mathbf{v} \cdot \frac{d\mathbf{u}}{d\mathbf{n}} \, dS. \quad (4)$$

Combining (3) and (4), and using that $\mathbf{n} \times \mathbf{v} = 0$ on $\partial\Omega$, we can rewrite $a(\mathbf{u}, \mathbf{v})$ as a new bilinear form

$$b(\mathbf{u}, \mathbf{v}) = \int_{\Omega} \nabla \mathbf{u} \cdot \nabla \mathbf{v} \, dx + \int_{\partial\Omega} (\nabla \cdot \mathbf{u} - \mathbf{n} \cdot \frac{d\mathbf{u}}{d\mathbf{n}})(\mathbf{n} \cdot \mathbf{v}) \, dS.$$

Moreover, since $\mathbf{n} \times \mathbf{u} = 0$ on $\partial\Omega$, we can expand the divergence on the boundary as

$$\nabla \cdot \mathbf{u} = \mathbf{n} \cdot \frac{d\mathbf{u}}{d\mathbf{n}} + \kappa \mathbf{n} \cdot \mathbf{u} \quad \text{on } \partial\Omega, \quad (5)$$

where κ is the local curvature which is defined almost everywhere. Using (5), we can write $b(\mathbf{u}, \mathbf{v})$ as

$$b(\mathbf{u}, \mathbf{v}) = \langle \nabla \mathbf{u}, \nabla \mathbf{v} \rangle + \int_{\partial\Omega} \kappa \mathbf{u} \cdot \mathbf{v} \, dS,$$

thus giving rise to a different variational formulation: Given $\mathbf{f} \in L^2(\Omega)^N$ with $\nabla \cdot \mathbf{f} = 0$, find $\mathbf{u} \in H_{0t}(\Omega)^N$ such that for each $\mathbf{v} \in H_{0t}(\Omega)^N$

$$\text{(VP2)} \quad b(\mathbf{u}, \mathbf{v}) = \langle \mathbf{f}, \mathbf{v} \rangle.$$

Clearly, by construction the bilinear forms are equal, $a(\mathbf{u}, \mathbf{v}) = b(\mathbf{u}, \mathbf{v})$, for functions in $H^2(\Omega)^N$. In fact, as shown in [18], the equality also holds if the functions are in $H^1(\Omega)^N$. Moreover, the standard theory shows that the bilinear forms are coercive and continuous on $H_{0t}(\Omega)^N$ (since Ω is simply connected) so that by the Lax-Milgram theorem there is a unique solution to **(VP1)** and **(VP2)**. Moreover the variational problems **(VP1)** and **(VP2)** have the same solution.

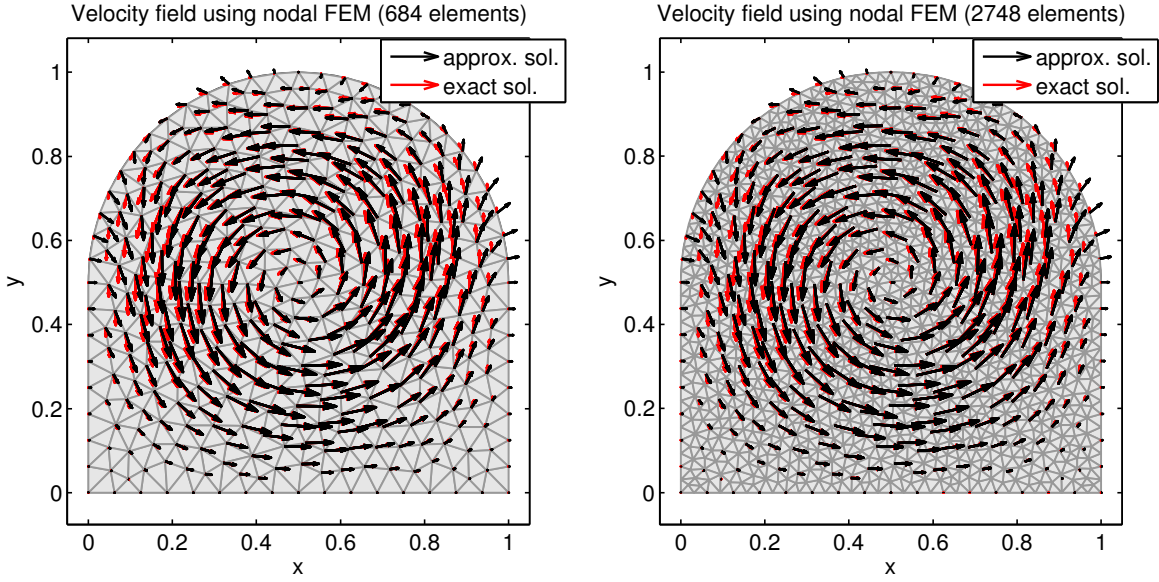


Figure 1. True vector field (red arrows) and numerical approximations (black arrows) obtained via nodal FEM. The numerical solutions obtained with two different mesh resolutions are almost identical, thus the numerics are essentially converged. Yet, the FEM solution differs from the true solution.

2.2.2. *Babuška paradox.* Using the just derived weak formulations, we prove the possible occurrence of the Babuška paradox. Note that other proofs have been provided before, such as in [32].

Theorem 2. *When solving the vector Poisson equation (2) using a nodal FEM implementation of (VP1), one may encounter the Babuška paradox.*

Proof. Suppose that \mathbf{u}_h solves (VP1) using nodal elements, a triangular mesh, and a regular polygonal domain Ω_h . Here h denotes the diameter of the largest mesh element, so that $\Omega_h \rightarrow \Omega$ (in the appropriate sup-norm sense) as $h \rightarrow 0$. Now, under the current assumptions on the domain Ω , the problems (VP1) and (VP2) have the same weak solution. Then by the equivalence of the two problems, \mathbf{u}_h also solves (VP2). For any given mesh, however, the boundary of Ω_h has flat sides with $\kappa = 0$. Consequently, the weak solution \mathbf{u}_h solves (VP2) with $\kappa = 0$, i.e.,

$$\int_{\Omega_h} \nabla \mathbf{u} \cdot \nabla \mathbf{v} \, dx = \int_{\Omega_h} \mathbf{f} \cdot \mathbf{v} \, dx$$

for each test function \mathbf{v}_h . Therefore the solutions \mathbf{u}_h converge to the function \mathbf{u}^* which solves the limit problem

$$\int_{\Omega} \nabla \mathbf{u}^* \cdot \nabla \mathbf{v} \, dx = \langle \mathbf{f}, \mathbf{v} \rangle$$

for each $\mathbf{v} \in H_{0t}(\Omega)^N$. In other words, the nodal FEM solutions \mathbf{u}_h converge to a solution where κ is artificially set to zero, or equivalently to a problem where one replaces the boundary condition $\nabla \cdot \mathbf{u} = 0$ with $\frac{d\mathbf{u}}{dn} = 0$. Hence, for an arbitrary domain (with boundaries that are at least partially curved), generally \mathbf{u}^* does not equal the true solution of the problem (VP1). \square

2.2.3. Manufactured solution test case. We now demonstrate the Babuška paradox via a numerical example. On the 2D domain $\Omega = \{(x - 0.5)^2 + (y - 0.5)^2 < 0.5^2\} \cup (0, 1) \times (0, 0.5)$, we consider the VPE (2). We employ the method of manufactured solutions, i.e., we prescribe the incompressible solution

$$\mathbf{u}(x, y) = \begin{pmatrix} \pi \sin(2\pi y) \sin^2(\pi x) \\ -\pi \sin(2\pi x) \sin^2(\pi y) \end{pmatrix}, \quad (6)$$

and set the forcing $\mathbf{f} = -\Delta \mathbf{u}$ and the boundary velocity $\mathbf{g} = \mathbf{u}$, so that the solution of the VPE (2) recovers the prescribed solution (6).

We consider two triangulations (of different resolutions) that approximate the domain Ω via polygons with straight edges. To define normal vectors at the boundary vertices, we adopt the method introduced in [10], which obtains normal vectors as suitable averages of normal vectors at the edges connecting to the boundary vertex. Using standard nodal-based finite elements, or Lagrange FE [7], we implement the tangential boundary condition in an essential fashion (by choosing the solution space $H_{\mathbf{g}t}^1(\Omega)^N$), and leave the normal component unprescribed, with the idea that $\nabla \cdot \mathbf{u} = 0$ follows naturally (see above). We use quadratic C^0 elements.

The results of the two FEM approximations are shown in Fig. 1. The true vector field (6) is shown by red arrows, and the approximate FEM solution is given by black arrows. In each case, the mesh that is used to conduct the computation is shown in the background. The right panel shows a computation with a mesh that is twice as fine as the one in the left panel. The two numerical solutions are almost identical, and hence they can be interpreted as converged (in the eye-norm) to the limit ($h \rightarrow 0$) solution \mathbf{u}^* of the nodal FEM. Clearly the FEM solution \mathbf{u}^* is different from the true solution \mathbf{u} , thus confirming the occurrence of the Babuška paradox.

Remark 2. Within the framework of FEM, the Babuška paradox can be overcome by moving to a mixed FEM formulation (cf. [27, 32, 2, 1]). The idea of mixed FEM for the VPE (2) is to introduce an additional variable, $\sigma = \nabla \times \mathbf{u}$, and transform $-\Delta \mathbf{u} = \mathbf{f}$ into two equations: $\sigma = \nabla \times \mathbf{u}$ and $\nabla \times \sigma - \nabla(\nabla \cdot \mathbf{u}) = \mathbf{f}$. This framework allows one to use Raviart-Thomas elements [27] for the approximate vector field \mathbf{u}^h (and standard nodal elements (2D) or Nédélec elements [24] (3D) for σ^h), and to incorporate the tangential velocity boundary condition $\mathbf{n} \times (\mathbf{u} - \mathbf{g}) = 0$ as boundary integrals into the weak formulation, rather than into the solution space. In a companion paper [5], we apply high-order mixed FEM to the pressure Poisson equation reformulations of the Navier-Stokes equations devised in [30]. While mixed FEM overcome the Babuška paradox, this framework is clearly not as simple as nodal FEM, or as meshfree FD, described below.

2.3. Meshfree Finite Difference Method. Meshfree finite differences (FD) generalize classical FD that are defined on regular grids: at a given point, a differential operator of a smooth function is approximated via a combination of function values at nearby points. The selection of points and the corresponding weights are called the *stencil*. In the same way as grid-based FD, meshfree FD can be derived in two ways: as derivatives of suitable local interpolants of the data (cf. [22, 8, 21]), or via Taylor expansion of the solution (cf. [29]). Here we outline the second methodology.

Consider a *point cloud* that consists of interior points (inside Ω) and boundary points (on $\partial\Omega$); see Fig. 2 for an example. For a point x_i , let a neighborhood B_i be defined. Here we employ circular neighborhoods, i.e., $B_i = \{j : \|x_j - x_i\| \leq r\}$, where r is an appropriately

chosen radius (see below). However, many other types of neighborhoods are possible [28]. Now define the relative coordinates $\bar{x}_{ij} = x_j - x_i$ and Taylor-expand the solution $u(x)$ around x_i :

$$u(x_j) = u(x_i) + \nabla u(x_i) \cdot \bar{x}_{ij} + \frac{1}{2} \nabla^2 u(x_i) : (\bar{x}_{ij} \cdot \bar{x}_{ij}^T) + \text{h.o.t.}$$

Note that in the quadratic term, the matrix scalar product $A : C = \sum_{i,j} A_{ij} C_{ij}$ and the outer product of \bar{x}_{ij} with itself are used. While here we stop at the quadratic term, the expansion can of course be carried out further (or less far). A linear combination (with weights a_{ij}) of nearby solution values yields

$$\sum_{j \in B_i} a_{ij} u(x_j) = u(x_i) \sum_{j \in B_i} a_{ij} + \nabla u(x_i) \cdot \sum_{j \in B_i} a_{ij} \bar{x}_{ij} + \nabla^2 u(x_i) : \frac{1}{2} \sum_{j \in B_i} a_{ij} (\bar{x}_{ij} \cdot \bar{x}_{ij}^T) + \text{h.o.t.} \quad (7)$$

If (7) is supposed to approximate a given differential operator applied to the solution, then the stencil weights a_{ij} must satisfy certain constraints. For instance, for (7) to approximate $\Delta u(x_i)$, it is required that

$$\sum_{j \in B_i} a_{ij} = 0, \quad \sum_{j \in B_i} \bar{x}_{ij} a_{ij} = 0, \quad \text{and} \quad \sum_{j \in B_i} (\bar{x}_{ij} \cdot \bar{x}_{ij}^T) a_{ij} = 2I,$$

which in 2D gives rise to the linear system of constraints

$$\underbrace{\begin{pmatrix} \bar{x}_{i,j_1} & \cdots & \bar{x}_{i,j_{m_i}} \\ \bar{y}_{i,j_1} & \cdots & \bar{y}_{i,j_{m_i}} \\ \bar{x}_{i,j_1}^2 & \cdots & \bar{x}_{i,j_{m_i}}^2 \\ \bar{x}_{i,j_1} \bar{y}_{i,j_1} & \cdots & \bar{x}_{i,j_{m_i}} \bar{y}_{i,j_{m_i}} \\ \bar{y}_{i,j_1}^2 & \cdots & \bar{y}_{i,j_{m_i}}^2 \end{pmatrix}}_{V_i} \cdot \underbrace{\begin{pmatrix} a_{i,j_1} \\ \vdots \\ a_{i,j_{m_i}} \end{pmatrix}}_{\mathbf{a}_i} = \underbrace{\begin{pmatrix} 0 \\ 0 \\ 2 \\ 0 \\ 2 \end{pmatrix}}_{\mathbf{b}}. \quad (8)$$

Here, the stencil vector \mathbf{a}_i does not contain the diagonal entry a_{ii} . Its value is obtained as $a_{ii} = -\sum_{j \in B_i \setminus \{i\}} a_{ij}$. Moreover, $m_i = |B_i| - 1$ is the number of neighbors of x_i . If the radius r is chosen large enough that $m_i \geq 5 \forall i$, and if the point cloud generation (see §2.4) ensures that no pathological point configurations arise (see [28] for examples), then system (8) always has a solution, and the resulting approximation is (at least) first order accurate.

If $m_i > 5$, system (8) in general has infinitely many solutions. One way (employed here) to single out a unique solution is via a weighted least-squares (WLSQ) minimization problem

$$\min \sum_{j \in B_i \setminus \{i\}} \frac{a_{ij}^2}{w_{ij}}, \quad \text{s.t. } V_i \cdot \mathbf{a}_i = \mathbf{b} \quad (9)$$

where the weights are decreasing with the distance, $w_{ij} = \|x_j - x_i\|_2^{-\beta}$ (here we choose $\beta = 2$). The solution of (9) is

$$\mathbf{a}_i = W_i V_i^T (V_i W_i V_i^T)^{-1} \cdot \mathbf{b}.$$

where $W = \text{diag}(w_{i,1}, \dots, w_{i,j_{m_i}})$. Note that an alternative approach (not employed here) would be to solve system (8) in an ℓ^1 sense, i.e.,

$$\min \sum_{j \in B_i \setminus \{i\}} \frac{a_{ij}}{w_{ij}}, \quad \text{s.t. } V_i \cdot \mathbf{a}_i = \mathbf{b}, \quad \mathbf{a}_i \geq 0,$$

which would generate optimally sparse stencils [29].

Other differential operators are approximated in an analogous fashion. For instance, a first order approximation to $\partial_x u$ is obtained by setting

$$V_i = \begin{pmatrix} \bar{x}_{i,j_1} & \cdots & \bar{x}_{i,j_{m_i}} \\ \bar{y}_{i,j_1} & \cdots & \bar{y}_{i,j_{m_i}} \end{pmatrix}, \quad \mathbf{b} = \begin{pmatrix} 1 \\ 0 \end{pmatrix},$$

and a second order approximation to $\partial_x u$ is obtained by setting

$$V_i = \begin{pmatrix} \bar{x}_{i,j_1} & \cdots & \bar{x}_{i,j_{m_i}} \\ \bar{y}_{i,j_1} & \cdots & \bar{y}_{i,j_{m_i}} \\ \bar{x}_{i,j_1}^2 & \cdots & \bar{x}_{i,j_{m_i}}^2 \\ \bar{x}_{i,j_1} \bar{y}_{i,j_1} & \cdots & \bar{x}_{i,j_{m_i}} \bar{y}_{i,j_{m_i}} \\ \bar{y}_{i,j_1}^2 & \cdots & \bar{y}_{i,j_{m_i}}^2 \end{pmatrix}, \quad \mathbf{b} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}.$$

Applying this procedure to each equation (at all interior points) and boundary condition (at all boundary points) of the vector Poisson equation (2) leads (here is 2D) to the linear system

$$\begin{array}{|c|c|c|c|} \hline * & * & * & * \\ * & * & * & * \\ * & * & * & * \\ * & * & * & * \\ * & * & * & * \\ \hline & & * & * \\ & & * & * \\ & & * & * \\ & & * & * \\ & & * & * \\ \hline * & * & * & * \\ * & * & * & * \\ * & * & * & * \\ * & * & * & * \\ * & * & * & * \\ \hline & & * & * \\ & & * & * \\ & & * & * \\ & & * & * \\ \hline \end{array} \cdot \begin{array}{|c|} \hline u_1^x \\ \vdots \\ u_{N_i}^x \\ \hline u_1^y \\ \vdots \\ u_{N_i}^y \\ \hline u_{N_i+1}^x \\ \vdots \\ u_N^x \\ \hline u_{N_i+1}^y \\ \vdots \\ u_N^y \\ \hline \end{array} = \begin{array}{|c|} \hline f_1^x \\ \vdots \\ f_{N_i}^x \\ \hline f_1^y \\ \vdots \\ f_{N_i}^y \\ \hline 0 \\ \vdots \\ 0 \\ \hline g_{N_i+1} \\ \vdots \\ g_N \\ \hline \end{array} \left. \begin{array}{l} \right\} \Delta u^x = f^x \quad \text{in } \Omega \\ \left. \begin{array}{l} \right\} \Delta u^y = f^y \quad \text{in } \Omega \\ \left. \begin{array}{l} \right\} \partial_x u^x + \partial_y u^y = 0 \quad \text{on } \partial\Omega \\ \left. \begin{array}{l} \right\} n^x u^y - n^y u^x = g \quad \text{on } \partial\Omega \end{array} \right. \quad (10)$$

In this system, $\mathbf{u} = (u^x, u^y)$, $\mathbf{f} = (f^x, f^y)$, and the function $g = \mathbf{n} \times \mathbf{g}$ at the boundary, and $g_i = \mathbf{n}_i \times \mathbf{g}_i$. Moreover, the total number of points is N , and the number of interior points is N_i . In the sparse block matrix, the first two rows of blocks correspond to the two components of the Poisson equation at the interior points; the third block row encodes the $\nabla \cdot \mathbf{u} = 0$ boundary condition; and the fourth block row represents the $\mathbf{n} \times (\mathbf{u} - \mathbf{g}) = 0$ condition. The first two block columns corresponds to the two vector field components at the interior points x_1, \dots, x_{N_i} ; and likewise the last two block columns correspond to $\mathbf{u} = (u^x, u^y)$ at the boundary points x_{N_i}, \dots, x_N . Each empty block is devoid of nonzero entries, and each block with stars possesses some nonzero entries, corresponding to the neighboring points of each central point.

2.4. Point Cloud Generation. The generation of point clouds is conducted similarly to the mesh generation code DistMesh [25]. The domain Ω is specified via a level set function ϕ , chosen so that $\Omega = \{x : \phi(x) < 0\}$. Hence, ϕ also defines the boundary $\partial\Omega = \{x : \phi(x) = 0\}$ and surface normal vectors $\mathbf{n} = \nabla\phi/|\nabla\phi|$. The access to the level set function allows an immediate check whether a given location/point is inside or outside of the domain. Moreover, if ϕ is a signed distance function (i.e., $|\nabla\phi| = 1$ a.e.), one even has immediate access to a location's distance to the domain boundary.

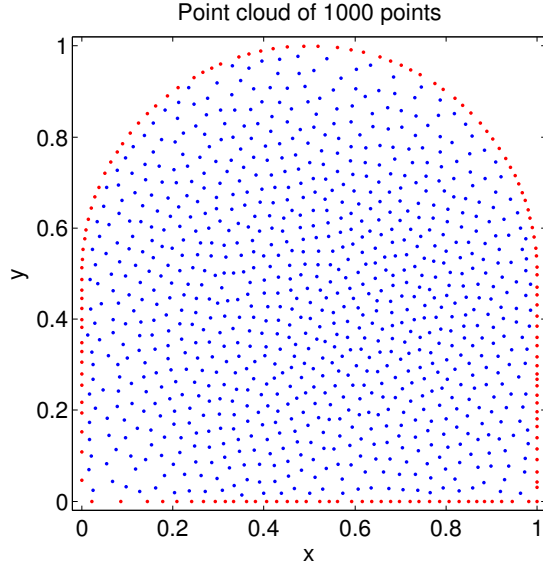


Figure 2. Point cloud for the computational domain Ω with 1000 points. The boundary points are shown in red and the interior points in blue.

The generation of a cloud of N points is initiated by placing N points randomly inside Ω . After that, the points are moved according to repulsive forces experienced from nearby points, i.e., the point x_i moves according to

$$\dot{x}_i = \sum_{j \in B_i \setminus \{i\}} \min\{\|x_j - x_i\|_2^{-2}, v_{\max}\}, \quad (11)$$

where B_i is a circular neighborhood around x_i , and v_{\max} is some upper bound on the repulsion. The law of motion (11) is further constrained by $x_i \in \bar{\Omega}$, i.e., no point can ever move beyond the domain boundary to leave the domain. A simple way to implement this constraint is to allow points to slightly leave $\bar{\Omega}$, but then to immediately project them back onto $\partial\Omega$, using the normal $\nabla\phi/|\nabla\phi|$ that is defined also outside of Ω . Finally, points that are inside Ω but too close to the boundary are also projected onto $\partial\Omega$, thus preventing interior points from being too close to the boundary (see [29] for why this would be undesirable). The law of motion (11) is then applied to all points until the amount of motion has fallen below a given threshold. The resulting point clouds are unstructured, and tend to be quite uniform (i.e., the ratio between the minimum distance between points and the radius of the largest ball that contains no points (cf. [21]) is quite large). An example of a point cloud associated with the domain defined in §2.2.3 is shown in Fig. 2.

For the definition of a “mesh” resolution h of a given point cloud, there is a variety of possible choices [21]. Here we use a simple averaged concept of resolution, defined as follows. One type of configuration in which nearby points are equidistant is an optimal sphere packing, which in 2D is a hexagonal lattice, i.e., adjacent points form equilateral triangles. Each point is a corner of six triangles, and each triangle is shared by three points. One can therefore associate to each point $1/3$ of each of the six triangles, resulting in an area per point of $A = \frac{\sqrt{3}}{2}h^2$, if the spacing between points is h . Counting the area of the N_i interior points full,

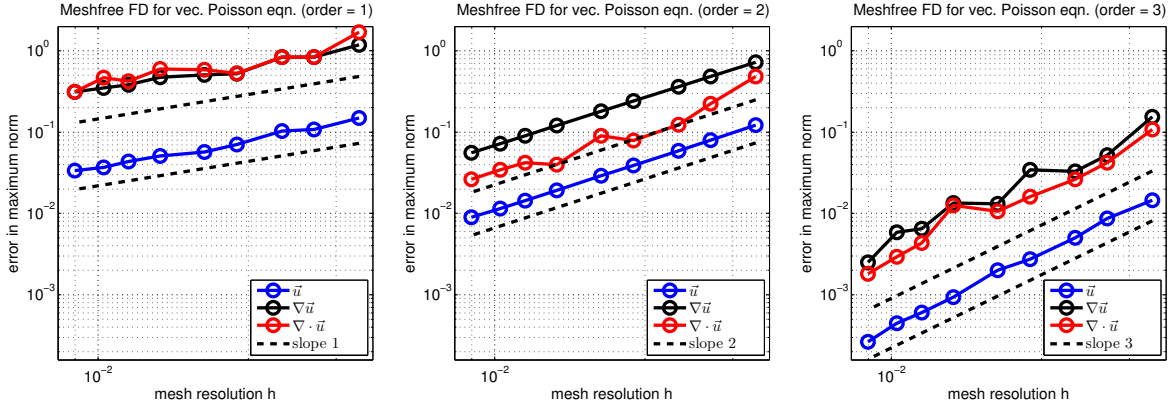


Figure 3. Error convergence for first (left), second (middle), and third (right) order meshfree FD approximations. The errors are measured in the maximum norm. The results show that k^{th} order meshfree FD stencils result in k^{th} order convergence rates for the solution of the VPE, and its derivatives.

and of the N_b boundary points half, and equating the total “point area” with the area of the domain, $\lambda(\Omega)$, we obtain the expression $h = \sqrt{\frac{4\lambda(\Omega)}{\sqrt{3}(2N_i + N_b)}}$ for the resolution.

2.5. Numerical Results. We consider the same manufactured solution test problem as studied in §2.2.3. On the domain $\Omega = \{(x - 0.5)^2 + (y - 0.5)^2 < 0.5^2\} \cup (0, 1) \times (0, 0.5)$ (see Figs. 1 and 2), point clouds of various numbers of points are generated, so that convergence studies can be conducted. For each point cloud, the vector Poisson equation (2) is discretized via the procedure described in §2.3. We conduct the meshfree FD approximation for three different orders: first order accuracy, i.e., the Taylor expansion in (7) is carried out up to the quadratic term for $\Delta \mathbf{u}$ and up to the linear term for $\nabla \cdot \mathbf{u}$; and second and third order accuracies, for which the Taylor expansions in (7) are carried out further accordingly.

The numerical approximations obtained for the different orders and mesh resolutions are then compared to the true solution in the maximum norm, taken over all points. We consider the errors in the vector field \mathbf{u} itself, as well as its Jacobian $\nabla \mathbf{u}$ (which is important for calculating stresses at the boundary when \mathbf{u} represents a velocity field) and its divergence $\nabla \cdot \mathbf{u}$ (which by the equivalence of problems (1) and (2) should be close to zero). All derivative quantities are obtained from the vector field \mathbf{u} via meshfree FD stencils of fourth order. Hence, if an order of less than 4 is observed, we know that this is the true accuracy of the numerical result.

The error convergence of these quantities is shown in Fig. 3, for the approximation orders 1 (left panel), 2 (middle panel), and 3 (right panel). The results show that all approaches converge as $h \rightarrow 0$, and the convergence orders equal the local approximation orders. In particular, the convergence orders of the derivative quantities are the same as those of the vector field itself. This is an important property that finite difference methods commonly exhibit, and that is in contrast to finite element methods that frequently lose an order of accuracy when a derivative quantity is evaluated (in the sense of functions).

3. PRESSURE POISSON EQUATION REFORMULATION OF THE NAVIER-STOKES EQUATIONS

We consider the time-dependent incompressible Navier-Stokes equations (NSE)

$$\begin{cases} \partial_t \mathbf{u} + (\mathbf{u} \cdot \nabla) \mathbf{u} = -\nabla p + \nu \Delta \mathbf{u} + \mathbf{f} & \text{in } \Omega \times (0, T) \\ \mathbf{u} = \mathbf{g} & \text{on } \partial\Omega \times (0, T) \\ \mathbf{u} = \dot{\mathbf{u}} & \text{on } \Omega \times \{t = 0\}, \end{cases} \quad (12)$$

with compatibility conditions

$$\begin{aligned} \dot{\mathbf{u}} &= \mathbf{g} && \text{on } \partial\Omega \times \{t = 0\} && \text{(continuity between i.c. and b.c.)} \\ \nabla \cdot \dot{\mathbf{u}} &= 0 && \text{in } \Omega && \text{(incompressible i.c.)} \\ \int_{\partial\Omega} \mathbf{n} \cdot \mathbf{g} \, dx &= 0. && && \text{(inflow = outflow)} \end{aligned}$$

Due to the lack of a time evolution of the pressure, there is no single canonical way to numerically advance (12) forward in time. One class of approaches to do so is based on approximating the time derivative in the momentum equation, and solving for \mathbf{u} and p in a fully coupled fashion. This methodology is accurate, but also costly, because a large system must be solved that possesses a saddle point structure. An alternative class of approaches decouples the pressure solve from the velocity update. This methodology was first proposed in the form of projection methods [6, 31], and later employed in approaches based on pressure Poisson equation (PPE) reformulations of the NSE [13, 14, 17, 15, 30].

A fundamental difference between projection methods (see [12] for an overview) and PPE reformulations is that projection methods are based on a fractional step approach in which the time-evolution of the velocity field and its projection onto the space of incompressible fields are alternated. In contrast, PPE reformulations derive an equation for the pressure that replaces the incompressibility constraint in (12) by a global pressure function $p = P(\mathbf{u})$ that is designed so that the solutions of the PPE reformulation are identical to the solutions of the original NSE. As a consequence, numerical methods based on PPE reformulations are structurally easy to extend to high order accuracy in time. In addition, they do not suffer from poor spatial accuracy near boundaries. A difficulty of PPE reformulations is that the Poisson equation for the pressure can involve complicated expressions, whose interaction with the velocity field equation is not always easy to understand and analyze. Another important property of PPE reformulations is that, unlike the original NSE, they are also defined if the initial conditions are not incompressible, see §3.1.

3.1. PPE Reformulation with Electric Boundary Conditions. In this paper, we are concerned with the particular PPE reformulation of the NSE proposed in [30]. Its fundamental difference from previously proposed PPE reformulations [13, 14, 17] is that the velocity field satisfies electric boundary conditions, i.e., incompressibility and the tangential flow are prescribed at the boundary. In turn, the normal velocity is enforced via a relaxation term in the pressure equation (see [30] for a discussion on the choice of λ). The PPE reformulation consists of the momentum equation

$$\begin{cases} \partial_t \mathbf{u} + (\mathbf{u} \cdot \nabla) \mathbf{u} = -\nabla P(\mathbf{u}) + \nu \Delta \mathbf{u} + \mathbf{f} & \text{in } \Omega \times (0, T) \\ \nabla \cdot \mathbf{u} = 0 & \text{on } \partial\Omega \times (0, T) \\ \mathbf{n} \times \mathbf{u} = \mathbf{n} \times \mathbf{g} & \text{on } \partial\Omega \times (0, T) \\ \mathbf{u} = \dot{\mathbf{u}} & \text{on } \Omega \times \{t = 0\}, \end{cases} \quad (13)$$

where $P(\mathbf{u})$ is a solution of the associated pressure Poisson equation

$$\begin{cases} \Delta p = \nabla \cdot (\mathbf{f} - (\mathbf{u} \cdot \nabla)\mathbf{u}) & \text{in } \Omega \\ \frac{\partial p}{\partial \mathbf{n}} = \mathbf{n} \cdot (\mathbf{f} - \partial_t \mathbf{g} + \nu \Delta \mathbf{u} - (\mathbf{u} \cdot \nabla)\mathbf{u}) + \lambda \mathbf{n} \cdot (\mathbf{u} - \mathbf{g}) & \text{on } \partial\Omega. \end{cases} \quad (14)$$

While a variety of modifications and additions can be applied to these equations (cf. [30, 5]), here we study the equations exactly in the given form (with one small caveat regarding the compatibility of the pressure boundary conditions, see below).

If the initial conditions are not incompressible, i.e., $\nabla \cdot \dot{\mathbf{u}} \neq 0$, then the solution of (13) relaxes towards a solutions of (12), for the following reason. Let $\phi = \nabla \cdot \mathbf{u}$. Then the application of $\nabla \cdot$ to the momentum equation in (12), and the use of the first equation in (14) yields that ϕ satisfies the heat equation with homogeneous Dirichlet boundary conditions

$$\begin{cases} \partial_t \phi = \nu \Delta \phi & \text{in } \Omega \times (0, T) \\ \phi = 0 & \text{on } \partial\Omega \times (0, T) \\ \phi = \nabla \cdot \dot{\mathbf{u}} & \text{on } \Omega \times \{t = 0\}. \end{cases} \quad (15)$$

This property is of great relevance. It means that in PPE reformulations, there is no need to impose a discrete incompressibility principle. If, due to numerical approximation errors, the numerical solution starts to drift away from the $\nabla \cdot \mathbf{u} = 0$ manifold, equation (15) ensures that it is pulled back towards incompressibility.

At the same time, the fact that the numerical solution may not be exactly incompressible, implies that the compatibility condition in the pressure Poisson equation (14) may be violated. Specifically, (14) has a solution if

$$\int_{\Omega} (\lambda + \nu \Delta) \phi \, dx - \int_{\partial\Omega} (\partial_t + \lambda) \mathbf{g} \cdot \mathbf{n} \, dS = 0, \quad (16)$$

and due to numerical approximation errors (or because a problem with $\nabla \cdot \dot{\mathbf{u}} \neq 0$ is considered) this condition may be violated. However, whenever this occurs, the solution of the augmented system (22), described in §3.3 projects the right hand side of (14) in a way that the solvability condition is satisfied.

Below, we first generalize the meshfree finite difference methods developed in §2 for the vector Poisson equation to the vector heat equation (§3.2). Then, we extend the methodology to the PPE reformulation (§3.3).

3.2. Meshfree Finite Differences for the Vector Heat Equation. Before moving to the full PPE reformulation (13), we first generalize the numerical scheme developed in §2.3 for the vector Poisson equation (2) to the vector heat equation (VHE) that describes the evolution of a vector field $\mathbf{u}(x, t)$ via the system

$$\begin{cases} \partial_t \mathbf{u} = \nu \Delta \mathbf{u} + \mathbf{f} & \text{in } \Omega \times (0, T) \\ \nabla \cdot \mathbf{u} = 0 & \text{on } \partial\Omega \times (0, T) \\ \mathbf{n} \times \mathbf{u} = \mathbf{n} \times \mathbf{g} & \text{on } \partial\Omega \times (0, T) \\ \mathbf{u} = \dot{\mathbf{u}} & \text{on } \Omega \times \{t = 0\}. \end{cases} \quad (17)$$

Here the forcing $\mathbf{f}(x, t)$ and the initial data $\dot{\mathbf{u}}(x)$ are incompressible, i.e., $\nabla \cdot \mathbf{f} = 0$ and $\nabla \cdot \dot{\mathbf{u}} = 0$. We discretize (17) in time via ImEx (Implicit-Explicit) schemes [3]. Specifically, we use the first-order scheme

$$\frac{1}{\Delta t} (\mathbf{u}(t + \Delta t) - \mathbf{u}(t)) = \mathbf{R}(\mathbf{u}(t + \Delta t)) + \mathbf{Q}(\mathbf{u}(t)) \quad (18)$$

and the second-order two-stage Runge-Kutta scheme

$$\begin{aligned} \frac{1}{\Delta t} (\mathbf{u}^* - \mathbf{u}(t)) &= \gamma \mathbf{R}(\mathbf{u}^*) + \gamma \mathbf{Q}(\mathbf{u}(t)) \\ \frac{1}{\Delta t} (\mathbf{u}(t + \Delta t) - \mathbf{u}(t)) &= \gamma \mathbf{R}(\mathbf{u}(t + \Delta t)) + (1 - \gamma) \mathbf{R}(\mathbf{u}^*) + \delta \mathbf{Q}(\mathbf{u}(t)) + (1 - \delta) \mathbf{Q}(\mathbf{u}^*), \end{aligned} \quad (19)$$

where $\gamma = 1 - \frac{1}{2}\sqrt{2}$ and $\delta = 1 - \frac{1}{2\gamma}$. For the VHE (17), an explicit first-order scheme (forward Euler) is obtained by setting $\mathbf{R}(\mathbf{u}) = 0$ and $\mathbf{Q}(\mathbf{u}) = \nu \Delta \mathbf{u} + \mathbf{f}$ in (18). In turn, setting $\mathbf{R}(\mathbf{u}) = \nu \Delta \mathbf{u}$ and $\mathbf{Q}(\mathbf{u}) = \mathbf{f}$ yields semi-implicit schemes of first order via (18), and of second order via (19). Moreover, the boundary conditions are always treated implicitly, so that they are satisfied by the new state (at the end of each Runge-Kutta stage).

The schemes that treat $\Delta \mathbf{u}$ implicitly lead to relatively simple modifications of the linear system (10) of the VPE (17). For instance, the first-order scheme,

$$\left\{ \begin{array}{ll} \frac{1}{\Delta t} (\mathbf{u}(t + \Delta t) - \mathbf{u}(t)) = \nu \Delta \mathbf{u}(t + \Delta t) + \mathbf{f}(t) & \text{in } \Omega \\ \nabla \cdot \mathbf{u}(t + \Delta t) = 0 & \text{on } \partial \Omega \\ \mathbf{n} \times \mathbf{u}(t + \Delta t) = \mathbf{n} \times \mathbf{g}(t + \Delta t) & \text{on } \partial \Omega \\ \mathbf{u}(0) = \dot{\mathbf{u}} & \text{in } \Omega, \end{array} \right.$$

amounts to the following modifications of system (10): (i) The vector of unknowns becomes the new vector field at time $t + \Delta t$. (ii) In the right hand side vector, the function g is evaluated at time $t + \Delta t$. (iii) In the top two block rows of the system matrix, multiply all entries by ν and add $1/\Delta t$ to the diagonal entries. (iv) Add $1/\Delta t$ times the solution at time t to the top two blocks of the right hand side vector. Everything else remains unchanged.

In the forward Euler case, the update at the interior points becomes explicit, while the boundary conditions are still implicit. Hence, to advance the solution from time t to $t + \Delta t$, one first updates at each interior point explicitly

$$\mathbf{u}(x_i, t + \Delta t) = \mathbf{u}(x_i, t) + \Delta t \nu \sum_{j \in B_i} a_{ij} \mathbf{u}(x_j, t) + \Delta t \mathbf{f}(x_i, t)$$

according to (7) and (9), and after that solves a small linear system for the boundary points that results from the two bottom block rows of system (10), where the left two block columns are brought to the right hand side (using the just updated interior point values).

To study these numerical schemes, we generalize the manufactured solution from §2.2.3 to the time-dependent case. We set $\nu = 1$, prescribe the incompressible solution

$$\mathbf{u}(x, y, t) = \begin{pmatrix} \pi \cos(t) \sin(2\pi y) \sin^2(\pi x) \\ -\pi \cos(t) \sin(2\pi x) \sin^2(\pi y) \end{pmatrix}, \quad (20)$$

and calculate the forcing $\mathbf{f} = \partial_t \mathbf{u} - \nu \Delta \mathbf{u}$, the boundary velocity $\mathbf{g} = \mathbf{u}$, and the initial conditions $\dot{\mathbf{u}} = \mathbf{u}(t = 0)$ accordingly. Using this test case, we first determine, via numerical experiments, the maximum time step that the forward Euler scheme admits to be stable. We find $\Delta t \leq \frac{Ch^2}{\nu}$, where for the given point clouds, C always lies between 0.2 and 0.3. Then, we study the convergence orders of the numerical schemes. The results are shown in Fig. 4. We use a second order meshfree FD approximation in space, and conduct five kinds of time stepping: forward Euler and backward Euler with $\Delta t = 0.2h^2$ (left panel; the plots of forward vs. backward Euler are indistinguishable); backward Euler with $\Delta t = h$ (not shown; due to very small temporal errors, the convergence looks like second order); backward Euler with $\Delta t = 100h$ (middle panel; the temporal error is visible and yields the expected drop to first order); and the second-order ImEx scheme (19) (right panel). From these results,

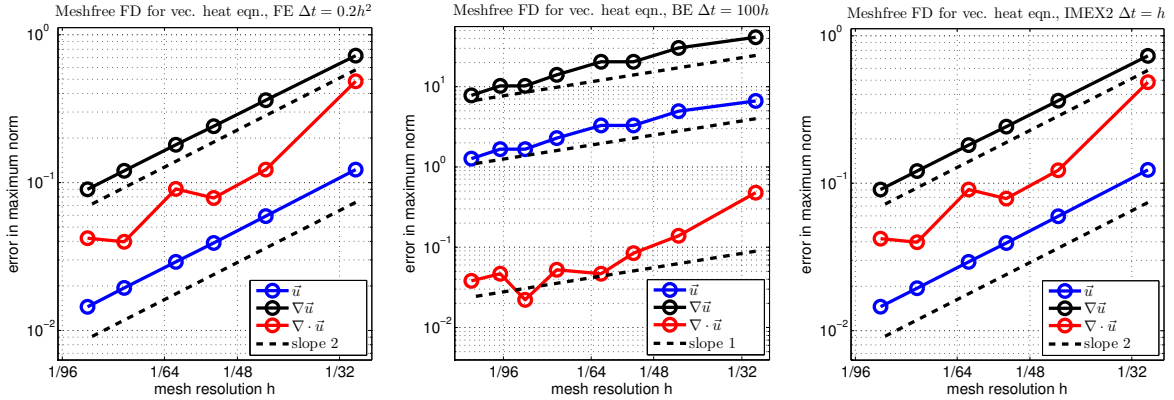


Figure 4. Error convergence for the vector heat equation, using a spatially second order meshfree finite difference discretization. Left panel: using forward Euler time stepping with $\Delta t = 0.2h^2$, confirming the expected $O(h^2)$ convergence order. Middle panel: using backward Euler time stepping with $\Delta t = 100h$, yielding the expected $O(h)$ convergence due to temporal errors. Right panel: using a second-order ImEx scheme with $\Delta t = h$, confirming the expected $O(h^2)$ convergence order.

we can see that the solution, and its derivatives, are in fact second-order accurate in space. Moreover, explicit and implicit time stepping (of first and second order) can be conducted without problems; and the temporal errors are relatively small.

3.3. Meshfree Finite Differences for the PPE Reformulation. Structurally the PPE reformulation (13) is the same as the vector heat equation (17), “just” with the nonlinear term $\mathbf{N}(\mathbf{u}) = (\mathbf{u} \cdot \nabla)\mathbf{u}$ and the pressure term $\nabla P(\mathbf{u})$ added to the time evolution. We treat both of these terms, as well as the forcing, explicitly. The first-order ImEx time-stepping (18) gives rise to the update rule

$$\frac{1}{\Delta t} (\mathbf{u}(t + \Delta t) - \mathbf{u}(t)) = -\mathbf{N}(\mathbf{u}(t)) - \nabla P(\mathbf{u}(t)) + \nu \Delta \mathbf{u}(t + \theta \Delta t) + \mathbf{f}(t), \quad (21)$$

where $\theta \in \{0, 1\}$ allows to switch between an explicit/implicit treatment of viscosity. In the forward Euler case ($\theta = 0$), stability requires $\Delta t = O(h^2)$, and thus the scheme’s accuracy is $O(h^2)$ overall. In the semi-implicit case ($\theta = 1$), one can choose larger time steps, and consequently the temporal accuracy is not sufficient. We therefore use instead the second-order ImEx time-stepping (19) with $\mathbf{R}(\mathbf{u}) = \nu \Delta \mathbf{u}$ and $\mathbf{Q}(\mathbf{u}) = \mathbf{f} - \mathbf{N}(\mathbf{u}) - \nabla P(\mathbf{u})$, which allows for time steps $\Delta t = O(h)$ and yields an $O(h^2)$ accurate scheme.

The Jacobi matrix $\nabla \mathbf{u}$ needed for the nonlinear terms is approximated very simply via point-centered meshfree finite differences, via the methodology described in §2.3. Clearly, such a centered treatment of advection is not the most effective choice for high Reynolds numbers (i.e., $\nu \ll 1$). And in fact, meshfree FD are quite easily amenable to an upwind treatment (e.g., by centering the approximation around a position $x_i - \beta \mathbf{u}(x_i)$, where β is a suitably chosen parameter). However, for the purpose of demonstrating the convergence of meshfree FD methods for the PPE reformulation (13), the central treatment of $\mathbf{N}(\mathbf{u})$ is sufficient.

The pressure $P(\mathbf{u})$ results from the solution of the pressure Poisson equation (14). We discretize this problem via the same meshfree FD method described in §2.3, with one important deviation from the standard procedure. The right hand side of the boundary conditions in (14) requires the evaluation of $\Delta \mathbf{u}$ at the boundary $\partial \Omega$. While straightforward meshfree FD

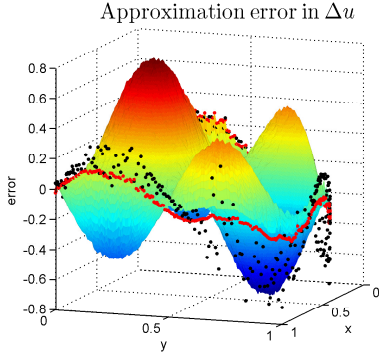


Figure 5. Error function when approximating Δu using mesh-free FD. Black dots: use FD directly at boundary points. Red dots: extrapolate function Δu from interior to the boundary.

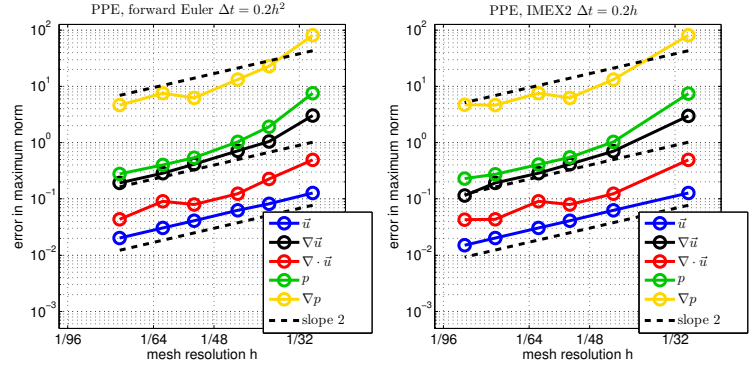


Figure 6. Error convergence of the velocity field, the pressure, and their derivatives, for the PPE reformulation of the NSE. A spatially second order meshfree FD discretization is used. Left panel: forward Euler time stepping (with $\Delta t = 0.2h^2$). Right panel: ImEx2 time stepping (with $\Delta t = 0.2h$). In both cases, the expected $O(h^2)$ convergence is confirmed.

for $\Delta \mathbf{u}$ yield accurate approximations inside the domain, it turns out that low accuracy (in the form of bounded but noticeable spatial oscillations along $\partial\Omega$) is achieved when using the same procedure at a boundary point. The reason is that the Laplacian is an operator that naturally “likes” to use data around the approximation point (cf. [29]); however, at the boundary, data in such a configuration is not accessible. We therefore employ a different approach that remedies the problem: we use the meshfree approximation of $\mathbf{w} = \Delta \mathbf{u}$ at the interior points (as calculated for the viscosity), and extrapolate this field \mathbf{w} to the boundary points, using moving least squares (MLS) interpolation [19]. This aspect is visualized in Fig. 5: the black dots are the approximation errors when approximating $\Delta \mathbf{u}$ at $\partial\Omega$ via meshfree FD; the red dots are the errors obtained when using MLS interpolation.

Since (14) is a Neumann problem, its discretization leads to a linear system $A \cdot p = r$, in which the Poisson matrix A has corank 1. In fact, because in the meshfree FD expansion (7) the first term must vanish for any differential operator, the kernel of A is $e = (1, \dots, 1)^T$. In line with the approach described in [13], we solve the augmented system

$$\begin{pmatrix} A & e \\ e^T & 0 \end{pmatrix} \cdot \begin{pmatrix} p \\ \alpha \end{pmatrix} = \begin{pmatrix} r \\ 0 \end{pmatrix}, \quad (22)$$

whose unique solution is the one satisfying $A \cdot p = r - \alpha e$, where the new right hand side is the projection of r onto the range of A . Moreover, a unique solution is singled out by the condition $e^T \cdot p = 0$. This approach in particular addresses the possibility that the PPE compatibility condition (16) may be not satisfied exactly. The gradient of the resulting pressure is then approximated via standard meshfree FD at all interior points.

The maximum admissible time step of the numerical scheme is determined by the viscosity term in the explicit case, i.e., $\Delta t = O(h^2)$, and by the nonlinear advection term in the semi-implicit case, i.e., $\Delta t = O(h)$. These stability time step restrictions are in line with those observed for the numerical method presented in [30]. However, they are different from the phenomenon observed and analyzed in [26, 15] for a different PPE reformulation and a different numerical discretization. In that study, the parabolic scaling $\Delta t = O(h^2)$ is observed to be required for stability, even if viscosity is treated implicitly.

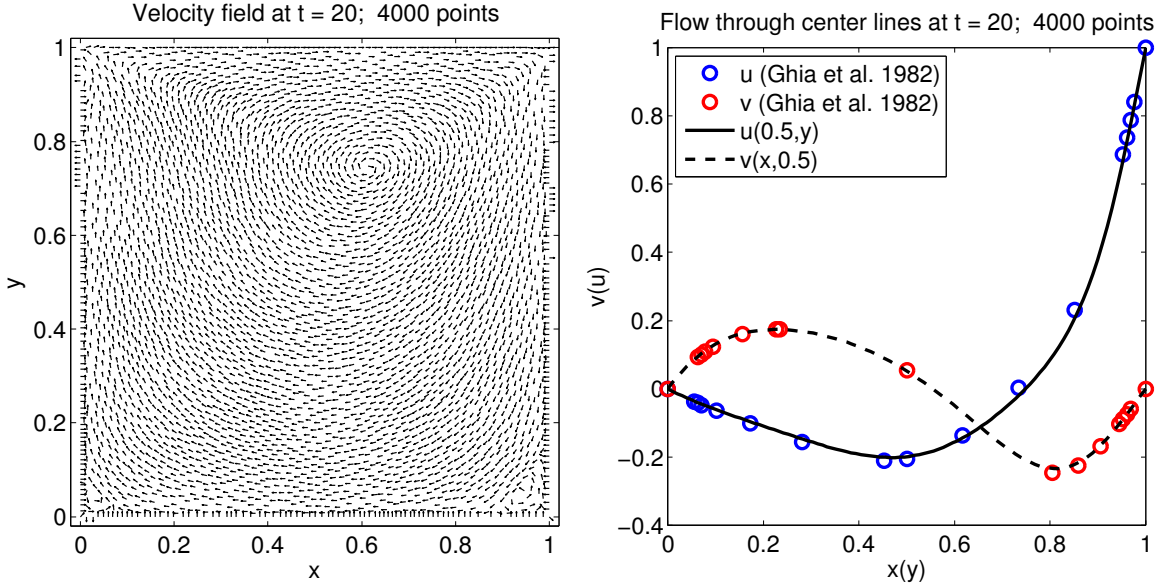


Figure 7. Lid-driven cavity test with $\text{Re}=100$ using a second-order meshfree scheme with 4000 points. Left: normalized velocity field at time $T = 20$. Right: plot of the flow normal to the two centerlines of the cavity compared with reference data [11].

3.4. Numerical Results. In order to investigate the convergence of the numerical scheme developed in §3.3, we use the same manufactured solution (20) as for the VHE, set $\nu = 1$, and calculate the pressure $p(x, y, t) = -\cos(t) \cos(\pi x) \sin(\pi y)$, the forcing $\mathbf{f} = \partial_t \mathbf{u} + (\mathbf{u} \cdot \nabla) \mathbf{u} + \nabla P(\mathbf{u}) - \nu \Delta \mathbf{u}$, and the boundary velocity $\mathbf{g} = \mathbf{u}$ accordingly. We use a spatially second order meshfree FD scheme (with the special treatment of $\Delta \mathbf{u}|_{\partial\Omega}$, see §3.3), and two types of time stepping: a) forward Euler with $\Delta t = 0.2h^2$; and second-order ImEx with $\Delta t = 0.2h$. In all cases the boundary relaxation value is chosen $\lambda = 30$. The numerical results, shown in Fig. 6, demonstrate that we obtain an overall second order convergence rate for all quantities of interest: the velocity field, its gradient, the divergence, the pressure, and the pressure gradient.

Moreover, to demonstrate the applicability of the numerical methodology, we conduct the standard benchmark lid-driven cavity test [4] for Reynolds number 100, i.e., $\nu = 0.01$. On the domain $\Omega = (0, 1)^2$, the velocity field is zero at the boundaries, except for the tangential velocity at $y = 1$, which is 1. The initial velocity field is zero everywhere except for $y = 1$, where it equals the boundary condition. The numerical approach used here is the same forward Euler-based scheme as in the manufactured solution test. The results of the steady-state profile ($t = 20$) are shown in Fig. 7. The velocity field (left panel) is depicted in the form of normalized vectors at the approximation points. The large center vortex and the two vortices in the bottom corners are captured. Moreover, a comparison (right panel) of the velocity through the centerlines of the cavity with reference data [11] shows a good agreement even on this not very highly resolved point cloud. Note that one particularity of the PPE reformulation (13) is that the flow through the boundary need not necessarily be exactly zero due to numerical approximation errors. This is why a flow through the boundaries is visible in the scaled quiver plot. However, the actual flow through the boundary is very small.

4. CONCLUSIONS AND OUTLOOK

The results in this paper show that meshfree finite differences (FD) provide a relatively straightforward methodology to approximate the solutions of vector-valued elliptic, parabolic, and fluid flow problems with electric boundary conditions (EBC), on domains without re-entrant corners. This is in contrast to finite element methods (FEM), whose simplest version, nodal-based FEM on triangular elements, fails at generating the correct solution. Instead, a Babuška paradox arises, which is shown to arise from the fact that nodal-based FEM do not capture the domain boundary’s curvature.

For the vector Poisson equation, meshfree FD lead to a linear system that discretizes the Laplace operator at interior points, and the divergence operator at boundary points, in a natural and very systematic fashion. The same methodology is shown to yield first, second, and third order convergent numerical schemes. Analogous statements hold for the vector-heat equation. Implicit time stepping is a straightforward extension of the vector Poisson case; explicit time stepping is a bit more interesting (because boundary conditions remain implicit), but poses no conceptual complication.

The extension of the methods to a PPE reformulation of the Navier-Stokes equation with EBC is, again, conceptually not complicated. There is one challenge that must be overcome, namely the approximation of the Laplacian of the velocity field at the domain boundary. Once this issue has been addressed, a second-order accurate numerical scheme is obtained in which the pressure solve and the viscosity solve are decoupled. As a consequence, one can choose between an explicit and an implicit treatment of viscosity.

Being FD approaches, the numerical schemes yield the values of the velocity field \mathbf{u} at the approximation points only. However, meshfree stencils can be employed to also calculate spatial derivatives of the velocity field. For instance, the velocity gradient $\nabla\mathbf{u}$ is crucial in computing forces and stresses acting on the boundary. Our investigation of the accuracy of these derivative quantities reveals that, for all studied problems, they show no degradation in order: a k^{th} order scheme yields k^{th} order convergence in \mathbf{u} , and also in $\nabla\mathbf{u}$.

While the results demonstrate the potential of meshfree FD for these types of problems, they also give rise to further questions. One important question is whether domains with re-entrant corners can also be treated. At first glance, one would think “no”, because of the lack of smoothness of the solutions on such domains. However, FD methods are known to be able to yield correct answers even for certain problems that lack smoothness (such as hyperbolic conservation laws [20]). This work also gives rise to a number of questions regarding the high-order accurate meshfree FD approximations. First, extensions of the numerical schemes for the PPE reformulation to convergence orders higher than two are of interest. Second, to avoid excessively large stencils for higher approximation orders, it is of interest whether the numerical schemes would work equally well (or better) if the meshfree FD approximations were obtained in a different fashion, such as via radial basis functions, compact FD, or deferred correction.

ACKNOWLEDGMENTS

P. Chidyagwai, R. R. Rosales, B. Seibold, and D. Zhou wish to acknowledge support by the National Science Foundation. This work was supported through grants DMS-1115269 and DMS-1115278. Furthermore, R. R. Rosales would like to acknowledge partial support by

NSF through grant DMS–1318942, and B. Seibold would like to acknowledge partial support by NSF through grants DMS–1318641 and DMS–1318709. D. Shirokoff acknowledges partial funding by NSERC.

REFERENCES

- [1] D. N. Arnold, R. S. Falk, and J. Gopalakrishnan. Mixed finite element approximation of the vector Laplacian with Dirichlet boundary conditions. *Math. Models Methods Appl. Sci.*, 22(9):1250024, 2012.
- [2] D. N. Arnold, R. S. Falk, and R. Winther. Finite element exterior calculus: From Hodge theory to numerical stability. *Bull. Amer. Math. Soc.*, 47(2):281–354, 2010.
- [3] U. M. Ascher, S. J. Ruuth, and R. J. Spiteri. Implicit-explicit Runge-Kutta methods for time-dependent partial differential equations. *Appl. Numer. Math.*, 25(2):151–167, 1997.
- [4] O. R. Burggraf. Analytical and numerical studies of the structure of steady separated flows. *J. Fluid Mech.*, 24(1):113–151, 1966.
- [5] P. Chidyagwai, R. R. Rosales, B. Seibold, D. Shirokoff, and D. Zhou. High-order mixed finite elements for a pressure Poisson equation reformulation of the Navier-Stokes equations with electric boundary conditions. *in preparation*, 2013.
- [6] A. J. Chorin. Numerical solution of the Navier-Stokes equations. *Math. Comput.*, 22:745–762, 1968.
- [7] R. Courant. Variational methods for the solution of problems of equilibrium and vibrations. *Bull. Amer. Math. Soc.*, 49(1):1–23, 1943.
- [8] L. Duarte, T. Liszka, and W. Tworzyako. hp-meshless cloud method. *Comput. Methods Appl. Mech. Engrg.*, 139(1–7):263–288, 1996.
- [9] F. El Dabaghi and O. Pironneau. Stream vectors in three dimensional aerodynamics. *Numer. Math.*, 48(5):561–589, 1986.
- [10] M. S. Engelman and R. L. Sani. The implementation of normal and/or tangential boundary conditions in finite element codes for incompressible fluid flow. *Int. J. Numer. Methods Fluids*, 2:225–238, 1982.
- [11] U. Ghia, K. N. Ghia, and C. T. Shin. High-resolutions for incompressible flow using the Navier-Stokes equations and a multigrid method. *J. Comput. Phys.*, 48(3):387–411, 1982.
- [12] J. L. Guermond, P. Mineev, and J. Shen. An overview of projection methods for incompressible flows. *Comput. Methods in Appl. Mech. Eng.*, 195(44–47):6011–6045, 2006.
- [13] W. D. Henshaw. A fourth-order accurate method for the incompressible Navier-Stokes equations on overlapping grids. *J. Comput. Phys.*, 113(6):13–25, 1994.
- [14] W. D. Henshaw, H.-O. Kreiss, and L. G. M. Reyna. A fourth-order accurate difference approximation for the incompressible Navier-Stokes equations. *Comput. Fluids*, 23(4):575–593, 1994.
- [15] W. D. Henshaw and N. A. Petersson. A split-step scheme for the incompressible Navier-Stokes equations. In M. M. Hafez, editor, *Numerical Simulation of Incompressible Flows*, volume 2502, pages 108–125. World Scientific, 2003.
- [16] B. N. Jiang, J. Wu, and L. A. Povinelli. The origin of spurious solutions in computational electromagnetics. *J. Comput. Phys.*, 125(1):104–123, 1996.
- [17] H. Johnston and J.-G. Liu. A finite difference method for incompressible flow based on local pressure boundary conditions. *J. Comput. Phys.*, 180(1):120–154, 2002.
- [18] U. Kangro and R. Nicholaides. Divergence boundary conditions for vector Helmholtz equations with divergence constraints. *ESAIM Math. Model. Numer. Anal.*, 33(3):479–492, 1999.
- [19] P. Lancaster and K. Salkauskas. Surfaces generated by moving least squares methods. *Math. Comp.*, 37:141–158, 1981.
- [20] R. J. LeVeque. *Numerical methods for conservation laws*. Birkhäuser, second edition, 1992.
- [21] D. Levin. The approximation power of moving least-squares. *Math. Comp.*, 67:1517–1531, 1998.
- [22] T. Liszka and J. Orkisz. The finite difference method at arbitrary irregular grids and its application in applied mechanics. *Comput. & Structures*, 11:83–95, 1980.
- [23] I. D. Mayergoyz. A new point of view on the mathematical structure of Maxwell’s equations. *IEEE Trans. Magn.*, 29(2):1315–1320, 1993.
- [24] J.-C. Nédélec. Mixed finite elements in \mathbb{R}^3 . *Numer. Math.*, 35(3):315–341, 1980.
- [25] P.-O. Persson and G. Strang. A simple mesh generator in MATLAB. *SIAM Review*, 46(2):329–345, 2004.
- [26] N. A. Petersson. Stability of pressure boundary conditions for Stokes and Navier-Stokes equations. *J. Comput. Phys.*, 172(1):40–70, 2001.

- [27] P.-A. Raviart and J.-M. Thomas. A mixed finite element method for 2-nd order elliptic problems. In *Mathematical Aspects of Finite Element Methods*, volume 606 of *Lecture Notes in Math.*, pages 292–315. Springer, Berlin, 1977.
- [28] B. Seibold. *M-Matrices in meshless finite difference methods*. Dissertation, Department of Mathematics, University of Kaiserslautern, 2006.
- [29] B. Seibold. Minimal positive stencils in meshfree finite difference methods for the Poisson equation. *Comput. Methods Appl. Mech. Engrg.*, 198(3–4):592–601, 2008.
- [30] D. Shirokoff and R. R. Rosales. An efficient method for the incompressible Navier-Stokes equations on irregular domains with no-slip boundary conditions, high order up to the boundary. *J. Comput. Phys.*, 230(23):8619–8646, 2011.
- [31] R. Temam. Sur l’approximation de la solution des equations de Navier-Stokes par la methode des pas fractionnaires, II. *Arch. Ration. Mech. Anal.*, 33(3):377–385, 1969.
- [32] R. Verfürth. Mixed finite element approximation of the vector potential. *Numer. Math.*, 50(6):685–695, 1986.

(Dong Zhou) DEPARTMENT OF MATHEMATICS, TEMPLE UNIVERSITY,
1805 NORTH BROAD STREET, PHILADELPHIA, PA 19122

E-mail address: `dong.zhou@temple.edu`

URL: `http://www.math.temple.edu/~dzhou`

(Benjamin Seibold) DEPARTMENT OF MATHEMATICS, TEMPLE UNIVERSITY,
1805 NORTH BROAD STREET, PHILADELPHIA, PA 19122

E-mail address: `seibold@temple.edu`

URL: `http://www.math.temple.edu/~seibold`

(David Shirokoff) DEPARTMENT OF MATHEMATICS AND STATISTICS, MCGILL UNIVERSITY,
805 SHERBROOKE STREET WEST, MONTREAL, QUEBEC, CANADA, H3A 0B9

E-mail address: `david.shirokoff@mail.mcgill.ca`

URL: `http://www.math.mcgill.ca/dshirokoff`

(Prince Chidyagwai) DEPARTMENT OF MATHEMATICS AND STATISTICS, LOYOLA UNIVERSITY MARYLAND,
4501 N. CHARLES STREET, BALTIMORE, MD 21210

E-mail address: `pchidyagwai@loyola.edu`

URL: `http://math.loyola.edu/~chidyagg`

(Rodolfo Ruben Rosales) DEPARTMENT OF MATHEMATICS, MASSACHUSETTS INSTITUTE OF TECHNOLOGY,
77 MASSACHUSETTS AVENUE, CAMBRIDGE, MA 02139

E-mail address: `rrr@math.mit.edu`