Computational and Applied Mathematics (2021) 40:15
https://doi.org/10.1007/s40314-020-01400-7

RBF collocation and hybrid-LHI methods for Stokes systems and its application to controllability problems

Louis Breton[1] · Pedro González-Casanova[2] · Cristhian Montoya[3]

Received: 30 May 2020 / Revised: 30 May 2020 / Accepted: 11 December 2020

Abstract
The purpose of this article is to introduce radial basis function (RBF) methods for solving both direct Stokes equations and controllability problems for the Stokes system with few internal scalar controls. In both cases, Dirichlet or Navier-slip boundary conditions are considered. We introduce two radial basis function solvers, one global and the other local, to solve Stokes equations. These methods are used to discretize the primal and adjoint systems related to the controllability problems. Both techniques are based on divergence-free global RBFs. A global colocation technique based on Div-free inverse multi-quadrics is formulated and analyzed. A generalization of scalar hybrid kernels to a vector divergence-free hybrid RBFs setting is defined. Based on these kernels, the local Hermite interpolation (LHI) method in vector form is introduced. Due to the properties of the hybrid kernel, we show that due to the properties of the hybrid kernel this local method, can reduce up to double precision, the value of the condition number of the local Gram matrices. Simultaneously, it is proved that the real components of the eigenvalues corresponding to the global LHI matrix are negative and that consequently backward difference formulas are stable for time integration. The conjugate gradient algorithm is adapted to the radial basis function setting to solve the controllability problems. Several benchmarks problems in two dimensions with a non-convex domain (a star shape) are numerically solved by these RBFs methods to display and compare their feasibility. The solutions to these problems are also implemented by finite element techniques to study their relative performance.

Keywords Stokes system · Controllability · Navier-slip boundary conditions · Radial basis functions · Local Hermite interpolation method · Hybrid kernels

Mathematics Subject Classification 65D12 · 93B05 · 76D07

1 Introduction

The main goal of this article is to introduce radial basis function (RBFs) methods to solve the direct Stokes equations and control problems for the Stokes system with few internal scalar

Communicated by Baisheng Yan.

Extended author information available on the last page of the article

Published online: 08 January 2021

controls. In both problems, Dirichlet or Navier-slip boundary conditions are considered. The theoretical setting of the control problem is first reviewed. To solve this problem, it is necessary to solve two couple Stoke evolutionary systems called primal and adjoint equations. This paper has two major parts: in the first part, we introduce RBFs algorithms for the Stokes equations with Dirichlet or Navier-slip boundary conditions which will be used to solve the direct primal and adjoint problems. In the second part, using these techniques, an iterative algorithm is formulated to obtain the solution of the control problem.

Stationary and evolutionary Stokes equations have been recently solved using analytically divergence-free and curl-free matrix-valued, positive definite kernels in Wendland (2009), Fuselier et al. (2016) and Keim and Wendland (2016). In particular, Wendland (2009) formulates for first time a general frame which establish applications to partial differential equations exploiting the fact that the approximation space is analytically divergence free.

On the other hand, Fuselier et al. (2016) solve the Stokes problem using a divergency and rotational free matrix-valued positive definite kernels, based on global scalar RBFs and using Hodge decomposition to decouple the Stokes problem into an equation for the velocity and an equation for the pressure. In all these works, based on global collocation techniques, numerical examples for different Stokes benchmark problems, with Dirichlet boundary conditions on disk domains, were successfully solved.

Regarding RBF colocation and RBF-finite difference methods, it is well known that as the fill distance or the shape parameter tends to zero, the condition number of the corresponding Gram matrices grows. The solution to this problem has been studied by different techniques, see Flyer et al. (2016), Mishra et al. (2017), Fornberg and Flyer (2015b) and references therein. On the other hand for scalar evolutionary problems discretized by collocation, it has been observed that the corresponding Gram matrix can have eigenvalues with positive real components. This implies that discrete time integration do not converge under iterations, and a solution to this problem has been formulated in Fornberg and Flyer (2015a) by including a hyperviscosity term.

In what follows we describe the main contributions of this work.

1. We introduce two types of radial basis function solvers for the direct Stokes problems, one global and the other local. These methods are used to discretized the primal and adjoint systems related to the control problem. Our approach use central ideas from Wendland (2009) general frame adapted and modified in several points to be described below.
   Direct global solvers for the Stokes problem are built using Hermite interpolation technique based on divergence-free IMQ-RBFs. This allows to satisfy the incompressibility condition, at a discrete level, and to easily incorporate Navier-slip or Dirichlet boundary conditions. The first contribution of our article is that, as far as we know, we use for the first time the Navier-slip boundary conditions for the Stokes equations.
2. Unlike the benchmarks problems of the articles cited above, which used disks domains, a second contribution of our paper is that we use a star shape, thus non-convex domains in all the numerical experiments. We first analyze the numerical behavior of Div-free global collocation for these two new conditions, namely for star-shape domains and slip boundary conditions.
3. The evolutionary Stokes problem (in a non-convex domain) is first solved using global collocation Div-free RBF based on inverse multi-quadrics, IMQ, and techniques on backward difference formulas (BDFs). Besides, a stability analysis is performed for this global method. We numerically prove that the real components of the eigenvalues corresponding to the Gram matrix are all negative. This implies that the BDFs converges under iterations. Indeed, we numerically prove that the order of convergence is exponential in

space and quadratic in time.
As it is well known, the condition number of global collocation methods increases as the number of nodes grows, i.e., the so-called trade-off principle is satisfied; thus, this global method is limited to a relatively small number of nodes even in quad precision.

4. To solve direct Stokes problems with larger number of nodes, a local Hermite interpolation (LHI) method based on div-free RBFs is introduced.
   We first solved the stationary Stokes problem with the LHI method using div-free RBFs inverse multi-quadrics (IMQ) with extended precision. Consistently with the scalar case, we find that the exponential convergence can be attained as the fill distance decreases. However, we also find that:

   (a) As expected, the condition number of the local matrices grows exponentially as the number of local nodes increases and
   (b) Although the condition number of the global matrix is lower than the condition number of the local matrices, the real component of its eigenvalues can be positive. Thus, for the evolutionary Stokes equation, the method does not converges for BDF schemes.

5. To deal with these two problems, we generalize to the LHI vectorial Div-free setting a recent formulation for scalar problems, see Pankaj et al. (2019), Zhang (2019). These works introduce a new RBF basis called hybrid kernels, where hybrid scalar kernels are a convex combination of the sum of a Gaussian and Poly-harmonic odd scalar RBF. This generalization is the fifth contribution of our paper.
   To be precise, in this article, we introduce and define Div-free Hybrid kernels. By formulating the vectorial LHI method in this new basis we find that the eigenvalues of the corresponding global matrix have all negative real components. Thus, the method is stable for BDF schemes. Moreover, the condition numbers of the corresponding local matrices are considerable lower than the condition numbers of the local matrices related to inverse multi-quadrics kernels. It is interesting to note that although the real components of the eigenvalues of the global collocation scheme, discretized by inverse multi-quadrics, can be positive no mater which shape parameter we choose, this is not true for the global matrix of the LHI method with hybrid kernels.
6. We mention that we only find few articles in control theory problems using the radial basis function literature. Up to our knowledge, only some works on optimal control by RBF techniques have been reported in this field (Pearson 2013; González-Casanova et al. 2019; Mirinejad and Inanc 2017; Kishida and Braatz 2009; Rad et al. 2014). On the other hand and up to our knowledge, no work on controllability solved by RBFs has appeared in the literature. Thus, the sixth and final contribution of this article is to solve approximate controllability problems for the Stokes equations. In all cases, numerical experiments are performed to study and analyze these methods.

The paper is organized as follows. In Sect. 2, we give the continuous description of the approximate control problem. The divergence-free global collocation method and stability results are presented in Sect. 3. In Sect. 4, we recall the basic notation and concepts of the LHI scalar method and we introduce the vectorial RBF-LHI approach for solving the Stokes system. We solve the stationary Stokes problem by the inverse multi-quadrics Div-free technique. Divergence-free hybrid kernels are introduced and the evolutionary Stokes problem is numerically solved. Finally, in Sect. 5 we conclude the paper by applying these RBFs methods to the controllability problems with few scalar controls whith either Dirichlet or

Navier-slip boundary conditions. These results are compared to the equivalent finite element results. Conclusions and final remarks are included.

2 Control problem formulation

In this section, we introduce the notation and the continuous setting of the Stokes control problem that will be numerically solved in this article.

Let us first introduce some notation. Let Ω be a nonempty bounded connected open subset of $\mathbb{R}^d$ ($d = 2$ or $d = 3$) of class C^∞. Let $T > 0$ and let $\omega \subset \Omega$ be a (small) nonempty open subset which is the control domain. Furthermore, $Q := \Omega \times (0, T)$, $\Sigma := \partial\Omega \times (0, T)$, $\boldsymbol{\nu}(x)$ is the outward unit normal vector to Ω at the point $x \in \partial\Omega$. Moreover, let

$$H := \left\{ \mathbf{u} \in L^2(\Omega)^d : \nabla \cdot \mathbf{u} = 0 \text{ in } \Omega,\ \ \mathbf{u} \cdot \boldsymbol{\nu} = 0 \text{ on } \partial\Omega \right\}$$

and

$$V := \left\{ \mathbf{u} \in H_0^1(\Omega)^d : \nabla \cdot \mathbf{u} = 0 \text{ in } \Omega \right\}.$$

The continuous approximate control problem for the Stokes system with either Dirichlet or Navier-slip boundary conditions, that we are interested in, is defined as follows:

Approximate control. Given an initial data $\mathbf{y}_0$, we are looking for a control function $\mathbf{v} = \mathbf{v}(x, t)$ acting in $\omega \times (0, T)$ with *supp* $\mathbf{v} \subset \omega \times (0, T)$, such that the solution of the problem

$$\begin{cases} \mathbf{y}_t - \mu\Delta\mathbf{y} + \nabla p = \mathbf{v}1_\omega & \text{in} \quad Q, \\ \nabla \cdot \mathbf{y} = 0 & \text{in} \quad Q, \\ +\text{BC} & \text{on} \quad \Sigma, \\ \mathbf{y}(\cdot, 0) = \mathbf{y}_0(\cdot) & \text{in} \quad \Omega, \end{cases} \tag{1}$$

satisfies

$$\mathbf{y}(\cdot, T) \leq \varepsilon \quad \text{in} \quad \Omega, \qquad \text{(approximate control to zero)} \tag{2}$$

for every $\varepsilon > 0$.

In (1), $\mu > 0$ is the viscosity coefficient and p is the pressure. We focus our work in two types of boundary conditions on Σ, namely:

$$\underbrace{\mathbf{y} = \mathbf{g}}_{(a)\text{ Dirichlet}} \qquad \text{or} \qquad \underbrace{\mathbf{y} \cdot \boldsymbol{\nu} = 0, \quad (\sigma(\mathbf{y}, p) \cdot \boldsymbol{\nu})_{tg} = \mathbf{g},}_{(b)\text{ Navier-slip}} \tag{3}$$

where $\sigma(\mathbf{y}, p) := -pId + 2\mu D\mathbf{y}$ is the stress tensor, where D denotes the symmetrized gradient of $\mathbf{y}$, and tg stands for the tangential component of the corresponding vector field, i.e.,

$$\mathbf{y}_{tg} = \mathbf{y} - (\mathbf{y} \cdot \boldsymbol{\nu})\boldsymbol{\nu}.$$

From a physical point of view, the Navier-slip boundary condition arises from the interaction between wall and fluid and when the temperature is high. This behavior involves a movement on the boundary (slip), loosing energy, which do not penetrate the boundary (impermeable boundary), among other factors. The use of these slip conditions allows to describe phenomena observed in nature and remove un-physical singularities, see, for instance, (Cebeci 2012; He and Wang 2009) and references therein for more details. Now, from a mathematical point of view, such boundary conditions say that the tangential component of the stress tensor is proportionality to the tangential component of the velocity (Navier 1823)

$$\mathbf{y} \cdot \boldsymbol{\nu} = 0, \quad (\sigma(\mathbf{y}, p) \cdot \boldsymbol{\nu})_{tg} + k\mathbf{y}_{tg} = 0,$$

where k is a function that measures the local viscous coupling fluid–solid. We highlight that this proportionality factor can depend on the velocity as well as on the pressure, which complicate both the theoretical analysis and numerical solutions.

We now characterize the control problem in terms of the optimal or minimum value of a quadratic convex functional in $(L^2(Q))^2$ in the sense of García et al. (2017). Namely, for $\mathbf{y}_0 \in H$, we aim to obtain the control $\mathbf{v}$ with one vanishing component (jth component, $j \in \{1, 2\}$) such that it minimizes the functional J defined by

$$J(\mathbf{v}) := \frac{1}{2} \iint\limits_{\omega\times(0,T)} |\mathbf{v}|^2 \,\mathrm{d}x\,\mathrm{d}t + \frac{1}{2\beta_1} \|\mathbf{y}(\cdot, T)\|^2_{L^2(\Omega)} \,\mathrm{d}x + \frac{1}{2\beta_2} \iint\limits_{\omega\times(0,T)} |v_j|^2 \,\mathrm{d}x\,\mathrm{d}t, \tag{4}$$

where $\mathbf{y}$ is solution of the Stokes system (1), β_1, β_2 are arbitrary positive numbers associated, respectively, with the final condition $\mathbf{y}(\cdot, T) \le \varepsilon$ and with the control function $\mathbf{v}$.

The optimal value of the control $\mathbf{v}$ can be obtained by determining the Fréchet derivative of J with respect to $\mathbf{v}$. It is easy to verify that this corresponds to the following expressions:

$$\frac{\partial J}{\partial \mathbf{v}}(\mathbf{v}) = v_i - w_i \text{ if } i \neq j \quad \text{and} \quad \frac{\partial J}{\partial \mathbf{v}}(\mathbf{v}) = \frac{1}{2\beta_2} v_j - w_j, \quad \text{in } \omega \times (0, T), \tag{5}$$

where $\mathbf{w} \in V$ is the solution of the adjoint system of (1):

$$\begin{cases} -\mathbf{w}_t - \mu \Delta \mathbf{w} + \nabla q = 0 & \text{in} \quad Q, \\ \nabla \cdot \mathbf{w} = 0 & \text{in} \quad Q, \\ +\text{BC} & \text{on} \quad \Sigma, \\ \mathbf{w}(\cdot, T) = -\dfrac{1}{\beta_1} \mathbf{y}(\cdot, T) & \text{in} \quad \Omega. \end{cases} \tag{6}$$

In García et al. (2017), the authors proved that for every $\beta_1 > 0$, $\beta_2 > 0$, there exists a unique minimal control $\mathbf{v}$ associated to (4) which is characterized by (1), (5) and (6). We underline that only Dirichlet boundary conditions were treated in García et al. (2017).

From an abstract point of view, the control theory for the Stokes system with internal controls has been studied intensively for the mathematical community. The interested reader can see for instance (Guerrero and Montoya 2018) and references therein.

Meanwhile, from a computational point of view, we only know the recent paper by Fernández-Cara et al. (2017), whose numerical experiments are developed in two dimension for the heat, Stokes and Navier–Stokes with Dirichlet boundary conditions. The implemented methodology in Fernández-Cara et al. (2017) for the fluid equations is based on the so-called Fursikov–Imanuvilov formulation (Fursikov and Imanuvilov 1996) and Lagrangian approximation throughout mixed finite elements. Another approximation scheme to the approximate control problem is given in Fernández-Cara et al. (2015) for a turbulence model and also using Dirichlet boundary conditions.

We close this section by pointing out that, as far as we know, it does not exists a numerical approximation through RBFs for the Stokes problem with Navier-slip boundary conditions. In the following sections, we fill this gap and show its application for solving the approximate control problem for the Stokes system with few scalar controls described above.

3 Global divergence-free RBF methods for evolutionary Stokes problems

In this section, we present a RBF method for solving the evolutionary Stokes problem. The spatial discretization of the Stokes equation is similar to the compact support divergence-free RBF approximation of Wendland (2009), except that here we use global IMQ-RBFs. We incorporate both Dirichlet and Navier-slip boundary onto a non-convex domain, namely, a star shape. Extended precision is used in all numerical examples. Besides, a stability analysis of the corresponding Gram matrix is performed. We numerically prove that the real components of its eigenvalues are all negative, which shows that the backward differentiation formulas converge under iteration, more precisely, exponential convergence in space and algebraic convergence in time. Those behaviors are illustrated through numerical examples.

Let us define $L(\mathbf{y}, p) := -\mu \Delta \mathbf{y} + \nabla p$ and consider the system

$$\begin{cases} \mathbf{y}_t + L(\mathbf{y}, p) = \mathbf{f} & \text{in} \quad Q, \\ \nabla \cdot \mathbf{y} = 0 & \text{in} \quad Q, \\ \mathcal{B}\mathbf{y} = \mathbf{g} & \text{on} \quad \Sigma, \\ \mathbf{y}(\cdot, 0) = \mathbf{y}_0(\cdot) & \text{in} \quad Q, \end{cases} \tag{7}$$

where $\mathcal{B}$ is one of the boundary operators defined in (3).

3.1 Div-free-RBF collocation method and backward differentiation formula

The goal is to build a PDE operator via a finite difference approximation for the temporal derivative; meanwhile, a divergence-free RBF is used for the spatial operators. We first define the Div-free RBF as follows.

Definition 1 The divergence-free matrix-valued kernel is defined by

$$\mathbf{\Phi}_{\text{Div}} = \nabla \times \Delta \times \psi = \left\{ -\Delta I + \nabla \nabla^T \right\} \psi,$$

where I is the identity matrix and $\psi : \mathbb{R}^d \to \mathbb{R}$ is a smooth positive definite RBF.

The time scheme follows some ideas of Stevens et al. (2011). To illustrate this method, we use backward finite differences techniques, which are appropriate for the Stokes equations. The scheme at any time step for the system (7) is given by

$$\begin{cases} \mathbf{y}^{n+s} + \Delta t \, \beta_s \, L(\mathbf{y}^{n+s}, p^{n+s}) = \Delta t \, \beta_s \, \mathbf{f}^{n+s} + \sum\limits_{k=0}^{s-1} \sigma_k \mathbf{y}^{n+k} & \text{in} \quad Q, \\ \nabla \cdot \mathbf{y}^{n+s} = 0 & \text{in} \quad Q, \\ \mathcal{B}\mathbf{y}^{n+s} = \mathbf{g}^{n+s} & \text{on} \quad \Sigma, \end{cases} \tag{8}$$

where β_s, σ_k are known parameters defined by the BDF techniques. Thus, in each step, we solve the following PDE

$$\begin{cases} \bar{L}(\mathbf{y}^{n+s}, p^{n+s}) = \boldsymbol{F}^{n+s} & \text{in} \quad Q, \\ \nabla \cdot \mathbf{y}^{n+s} = 0 & \text{in} \quad Q, \\ \mathcal{B}\mathbf{y}^{n+s} = \mathbf{g}^{n+s} & \text{on} \quad \Sigma, \end{cases} \tag{9}$$

where $\bar{L}, \ \boldsymbol{F}^{n+s}$ are defined by

$$\bar{L}(\mathbf{y}^{n+s}, p^{n+s}) := \mathbf{y}^{n+s} + \Delta t \, \beta_s \, L(\mathbf{y}^{n+s}, p^{n+s}), \quad \boldsymbol{F}^{n+s} := \Delta t \, \beta_s \, \mathbf{f}^{n+s} + \sum_{k=0}^{s-1} \sigma_k \mathbf{y}^{n+k}. \tag{10}$$

The field $\mathbf{y}^{n+s}$ is then approximate by a linear combination of the divergence-free matrix-valued kernel $\mathbf{\Phi}_{\mathrm{Div}}$ (see Definition 1). Thus, the velocity–pressure vector $(\mathbf{y}^{n+1}, p^{n+1})$ is a linear combination of

$$\mathbf{\Phi} = \begin{bmatrix} \mathbf{\Phi}_{\mathrm{Div}} & 0 \\ 0 & \phi \end{bmatrix} : \mathbb{R}^d \to \mathbb{R}^{(d+1)\times(d+1)},$$

where $\phi : \mathbb{R}^d \to \mathbb{R}$ is a positive definite RBF.

On the other hand, using the generalized interpolation collocation method, see (Wendland 2004), the RBF ansatz is given by

$$\begin{aligned}(\hat{\boldsymbol{y}}^{n+s}, \hat{p}^{n+s})(\boldsymbol{x}) &= \sum_{i=1}^{d}\sum_{j=1}^{N_b} \mathcal{B}_i^{\xi}\,\mathbf{\Phi}(\boldsymbol{x} - \boldsymbol{\xi}_{\boldsymbol{j}})\alpha^{n+s}_{(i-1)N_b+j} \\ &\quad + \sum_{i=1}^{d}\sum_{j=1}^{N_{in}} \overline{L}_i^{\xi}\,\mathbf{\Phi}(\boldsymbol{x} - \boldsymbol{\xi}_{\boldsymbol{N_{in}+j}})\beta^{n+s}_{(i-1)N_b+j} \qquad (11)\end{aligned}$$

where $\mathcal{B}$ is the boundary condition operator; $(\alpha, \beta) \in \mathbb{R}^{d(N_b+N_{in})}$, N_b, N_{in} are the total number of boundary and interior nodes, respectively, and $\xi^b_{1..N_b}$, $\xi^{in}_{N_b+1....N_b+N_{in}} \in \mathbb{R}^d$, are the boundary and interior centers. On the other hand, $\mathcal{B}_i^{\xi}\,\mathbf{\Phi}$, $\overline{L}_i^{\xi}\,\mathbf{\Phi}$ are vector-valued functions from $\mathbb{R}^d \times \mathbb{R}^d$ to $\mathbb{R}^{d+1}$ defined as the application of the operator $\mathcal{B}_i^{\xi}$, $\overline{L}_i^{\xi}$ to each row of the kernel $\mathbf{\Phi}$.

We then replace the ansatz (11) in (9) and obtain the discrete systems

$$H_+ \begin{pmatrix} \boldsymbol{\alpha}^{n+s} \\ \boldsymbol{\beta}^{n+s} \end{pmatrix} = \begin{pmatrix} \boldsymbol{F}^{n+s} \\ \mathbf{g}^{n+s} \end{pmatrix}, \qquad (12)$$

where the collocation matrix H_+ is given by

$$\begin{pmatrix} \overline{L_1}^x \mathcal{B}_1^{\xi}\mathbf{\Phi}_{\mathbf{1}}(\boldsymbol{x}_{N_b+1}) & \dots & \overline{L_1}^x \mathcal{B}_d^{\xi}\mathbf{\Phi}_{N_b}(\boldsymbol{x}_{N_b+1}) & \overline{L_1}^x L_1^{\xi}\mathbf{\Phi}_{N_b+1}(\boldsymbol{x}_{N_b+1}) & \dots & \overline{L_1}^x L_d^{\xi}\mathbf{\Phi}_{N_b+N_{in}}(\boldsymbol{x}_{N_b+N_{in}}) \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \overline{L_d}^x \mathcal{B}_d^{\xi}\mathbf{\Phi}_{\mathbf{1}}(\boldsymbol{x}_{N_b+N_{in}}) & \dots & \overline{L_d}^x \mathcal{B}_d^{\xi}\mathbf{\Phi}_{N_b}(\boldsymbol{x}_{N_b+N_{in}}) & \overline{L_d}^x L_1^{\xi}\mathbf{\Phi}_{N_b+1}(\boldsymbol{x}_{N_b+N_{in}}) & \dots & \overline{L_d}^x L_d^{\xi}\mathbf{\Phi}_{N_b+N_{in}}(\boldsymbol{x}_{N_b+N_{in}}) \\ \mathcal{B}_1^x \mathcal{B}_1^{\xi}\mathbf{\Phi}_{\mathbf{1}}(\boldsymbol{x}_{\mathbf{1}}) & \dots & \mathcal{B}_1^x \mathcal{B}_d^{\xi}\mathbf{\Phi}_{N_b}(\boldsymbol{x}_{\mathbf{1}}) & \mathcal{B}_1^x L_1^{\xi}\mathbf{\Phi}_{N_b+1}(\boldsymbol{x}_{\mathbf{1}}) & \dots & \mathcal{B}_1^x L_d^{\xi}\mathbf{\Phi}_{N_b+N_{in}}(\boldsymbol{x}_{\mathbf{1}}) \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \mathcal{B}_d^x \mathcal{B}_1^{\xi}\mathbf{\Phi}_{\mathbf{1}}(\boldsymbol{x}_{N_b}) & \dots & \mathcal{B}_d^x \mathcal{B}_d^{\xi}\mathbf{\Phi}_{N_b}(\boldsymbol{x}_{N_b}) & \mathcal{B}_d^x L_1^{\xi}\mathbf{\Phi}_{N_b+1}(\boldsymbol{x}_{N_b}) & \dots & \mathcal{B}_d^x L_d^{\xi}\mathbf{\Phi}_{N_b+N_{in}}(\boldsymbol{x}_{N_b}) \end{pmatrix},$$

with $\mathbf{\Phi}_{\boldsymbol{j}}(\boldsymbol{x}) = \mathbf{\Phi}(\boldsymbol{x} - \boldsymbol{\xi}_{\boldsymbol{j}})$.

3.2 Stability analysis for BDF schemes

In this subsection, we present the stability analysis related to the previous scheme by using a matrix method similar to the procedure developed in Chinchapatnam et al. (2006). Our purpose here is to establish a condition to estimate the spectral radius of the gram Matrix. First, using (11), we define the interpolation matrix $A \in \mathbb{R}^{(N_{in}+N_b)\times(N_{in}+N_b)}$ such that

$$A\bar{\alpha}^{n+s} = \begin{pmatrix} M_{\phi_{in}} \\ M_{\phi_b} \end{pmatrix} \bar{\alpha}^{n+s} = \left(y^{n+s}(x_1^{in}), \dots, y^{n+s}(x^{in}_{N_{in}}), y^{n+s}(x_1^b), \dots, y^{n+s}(x^b_{N_b})\right)^T, \qquad (13)$$

where $M_{\phi_{in}} \in \mathbb{R}^{N_{in}\times(N_{in}+N_b)}$, $M_{\phi_b} \in \mathbb{R}^{N_b\times(N_{in}+N_b)}$.

Following Chinchapatnam et al. (2006), we define $H_- := \begin{pmatrix} M_{\phi_{in}} \\ 0 \end{pmatrix}$ and combining (13) and (12), we obtain

$$H_+\bar{\alpha}^{n+s} = H_- \sum_{k=0}^{s-1} \sigma_k \bar{\alpha}^{n+k} + \begin{pmatrix} \Delta t\, \boldsymbol{f}^{n+s} \\ \boldsymbol{g}^{n+s} \end{pmatrix}.$$

Thus, it follows that

$$\boldsymbol{y}^{n+s} = AH_+^{-1}H_-A^{-1}\sum_{k=0}^{s-1} \sigma_k \boldsymbol{y}^{n+s} + AH_+^{-1} \begin{pmatrix} \Delta t\, \boldsymbol{f}^{n+s} \\ \boldsymbol{g}^{n+s} \end{pmatrix}.$$

Denoting by $\mathbf{y}^n$ the exact solution and by $\hat{\mathbf{y}}^n$ the numerical solution, the error $\mathbf{e}^n = \mathbf{y}^n - \hat{\mathbf{y}}^n$ satisfies the equation

$$\mathbf{e}^{n+s} = K \sum_{k=0}^{s-1} \sigma_k \mathbf{e}^{n+k} + E_{n+s},$$

where E_{n+s} is the local error in the scheme (12) and $K = AH_+^{-1}H_-A^{-1}$. Besides, since E_{n+s} is small and therefore bounded, the error analysis can be analyzed using the identity

$$\mathbf{e}^{n+s} = K \sum_{k=0}^{s-1} \sigma_k \mathbf{e}^{n+k}. \tag{14}$$

By assuming that K is diagonalizable, i.e., $K = D^{-1}\Lambda D$, we can define $\mathbf{z}^n := D\mathbf{e}^n$ and therefore (14) is equivalent to

$$\mathbf{z}^{n+s} = \Lambda \sum_{k=0}^{s-1} \sigma_k \mathbf{z}^{n+k}.$$

Since Λ is a diagonal matrix, for every $j = 1, \ldots, d(n_b + N_{in})$ we have that $z_j^{n+s} = \sum_{k=0}^{s-1} \lambda_j \sigma_k z_j^{n+k}$, and whose solution is given by $z_j^n = \sum_{k=0}^{s-1} C_k^j r_k^n$, where C_k^j are arbitrary complex constants and r_k are the roots of the associated polynomial with the finite difference equation.

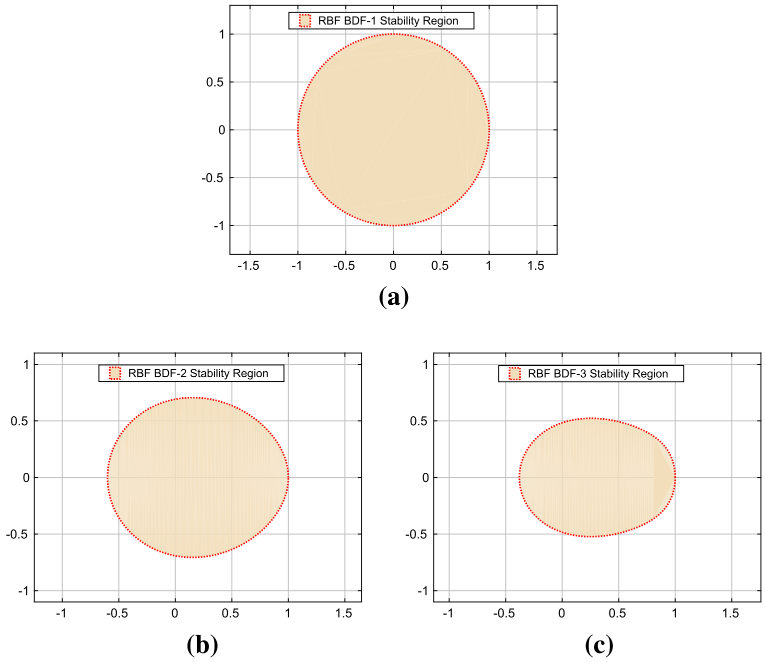
Finally, since $\|\mathbf{e}^n\|$ goes to zero iff $\|\mathbf{z}^n\|$ tends to zero, the method will be stable as long as the eigenvalues of K belong to the stability region of

$$\pi(r, \lambda) = r^s - \sum_{k=0}^{s-1} \lambda \sigma_k r^i. \tag{15}$$

As a consequence of the boundary locus technique (Lambert 1991), some stability regions are displayed in Fig. 1.

3.3 Numerical results: IMQ-Div-Free RBF global method for evolutionary problems

In the sequel, we evaluate the accuracy of the previous scheme through BDF2. The objective is to test the feasibility of the schemes by considering either non-homogeneous Dirichlet conditions or non-homogeneous Navier-slip conditions on the system (7). To generate the divergence-free kernel, see Definition 1, we use the scalar inverse multi-quadric (IMQ), $\psi(\mathbf{r}) = \sqrt{(}\mathbf{r} + c)^{-1/2}$ with shape parameter $c = 1.5$. Since this type of kernel is very ill



(a)

(b)

(c)

Fig. 1 Stability regions using backward finite differences: **a** one level, **b** two levels and **c** three levels

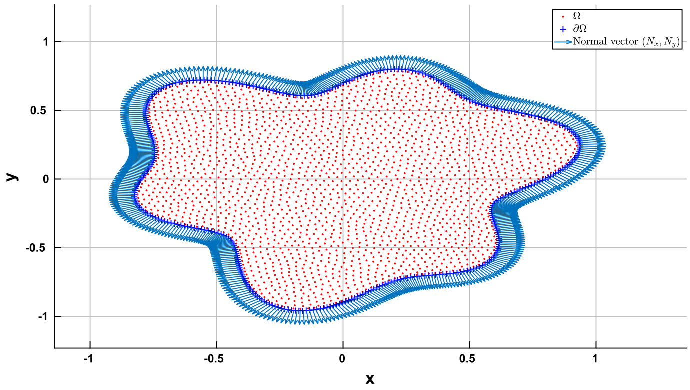


Fig. 2 Domain $\Omega \subset \mathbb{R}^2$. The boundary is defined by the parametrization given in (16)

conditioned, we have used the Matlab package ADVANPIX for multi-precision calculus and set the number of digits to 50. Henceforth, all the computations are done in the programming languages Matlab and FreeFemm++. If the reader is interested in the codes we used, these can be download from Breton (2020).

Henceforth, we consider a non-convex domain $\Omega \subset \mathbb{R}^2$ in star shape and whose boundary is parametrized by the curve (see Fig 2)

$$C = \left\{(\theta, \rho(\theta)) \in \mathbb{R}^2 : \rho(\theta) = 0.8 + \sin(6\theta) + \sin(3\theta), \quad \theta \in [0, 2\pi)\right\}. \qquad (16)$$

For every type of boundary condition (Dirichlet or Navier-slip), we consider the following exact solution of (7):

$$\begin{aligned} y_1(x,y,t) &= -y\ \sin\left((x^2+y^2)\sin(t^2+1)\right), \\ y_2(x,y,t) &= x\ \sin\left((x^2+y^2)\sin(t^2+1)\right), \\ p(x,y,t) &= \sin(x-y+t). \end{aligned}$$

We compare the error of the velocity and pressure in the L^∞-norm between the exact and numerical solutions for several time steps Δt. The errors are denoted by $\boldsymbol{\epsilon_y} = \mathbf{y}_{\text{exact}} - \mathbf{y}_{\text{aprox}}$ and $\boldsymbol{\nabla\epsilon_p} = \nabla p_{\text{exact}} - \nabla p_{\text{aprox}}$, respectively. The results are presented in Tables 1 and 2.

As expected, the error decreases as the number of nodes increases. Also, the rate of convergence of the velocity is higher than the rate of convergence of the gradient of the pressure as expected. It can be appreciated that we obtain excellent results both for Dirichlet and Navier-slip boundary conditions independently of the value of μ. Also, it is clear that the eigenvalues of the Gram matrix are inside the stability region of the BDF schemes, otherwise, the solution would not converge. On the other hand, we verify by performing numerical experiments, that this is not true for small values of the shape parameter, namely, for values of the order of $c \leq 10^{-2}$.

4 Local divergence-free RBF methods for Stokes problems: LHI technique

In this section an alternative method to the global method is introduced for evolutionary Stokes system, namely, a RBF-LHI vectorial technique, which corresponds to a generalization of the local Hermite interpolation (LHI) scalar method, see, for instance, (Stevens et al. 2011). Also, a generalization to the vectorial setting of a recent scalar method based on a new type of RBFs called hybrid kernels, see Mishra et al. (2018), will be introduced.

This section is organized as follows: in Sect. 4.1 we first recall some concepts and notation for the scalar LHI method, see González-Casanova et al. (2019) for a similar description. After that, Sect. 4.2 contains the numerical algorithm for the steady Stokes system, meanwhile Sect. 4.3 displays numerical examples with every type of boundary condition. We show that for Div-free IMQ kernels, this local method is used to efficiently solve benchmark problems for a large number of nodes, namely for more than 23,000 nodes. On the other hand, the eigenvalues for this matrix, for Div-free IMQ kernels, can have positive real components and, thus, cannot be used for evolutionary problems. To solve this problem, we introduce vectorial Div-free hybrid kernels and formulate the evolutionary Stokes LHI method in Sect. 4.4. With respect to the temporal discretization, a implicit discretization scheme is described. Finally, the numerical experiments concerning the evolutionary problems are presented in Sect. 4.5.

4.1 A reminder of the scalar LHI method: notation and preliminary remarks

In what follows, and for the sake of completeness, we first briefly recall the scalar Local Hermite Interpolation (LHI) method, introduced by Stevens et al. (2011) (see González-Casanova et al. 2019 for a similar review). This scalar setting will be used later in this paper to formulate the generalized vectorial technique to solve the Stokes problem.

Table 1 Global Hermite collocation-IMQ error for the Stokes system with Navier-slip boundary conditions. Here, the shape parameter is $c = 1.5$

	$\mu = 1$								
$N\backslash\Delta t$	1.00e−02		5.00e−03		1.00e−03		5.00e−04		Max Cond
	$\|\|e_y\|\|_\infty$	$\|\|e_{\nabla p}\|\|_\infty$	$\|\|e_y\|\|_\infty$	$\|\|e_{\nabla p}\|\|_\infty$	$\|\|e_y\|\|_\infty$	$\|\|e_{\nabla p}\|\|_\infty$	$\|\|e_y\|\|_\infty$	$\|\|e_{\nabla p}\|\|_\infty$	
117	5.94e−03	6.32e−01	6.47e−03	6.44e−01	7.87e−03	5.52e−01	8.17e−03	4.86e−01	1.55e+21
290	1.34e−03	4.39e−01	1.38e−03	3.90e−01	2.54e−03	5.41e−01	3.38e−03	7.23e−01	1.71e+21
576	2.08e−03	5.31e−01	5.72e−04	1.45e−01	4.62e−05	2.10e−02	4.46e−05	2.06e−02	9.35e+22
	$\mu = 1e\text{-}06$								
$N\backslash\Delta t$	1.00e−02		5.00e−03		1.00e−03		5.00e−04		Max Cond
	$\|\|e_y\|\|_\infty$	$\|\|e_{\nabla p}\|\|_\infty$	$\|\|e_y\|\|_\infty$	$\|\|e_{\nabla p}\|\|_\infty$	$\|\|e_y\|\|_\infty$	$\|\|e_{\nabla p}\|\|_\infty$	$\|\|e_y\|\|_\infty$	$\|\|e_{\nabla p}\|\|_\infty$	
117	5.22e−02	2.96e−01	4.15e−02	2.09e−01	1.88e−02	2.16e−01	2.32e−02	3.29e−01	6.22e+28
290	2.06e−03	1.43e−02	1.49e−03	1.36e−02	9.33e−04	6.22e−03	8.68e−04	4.40e−03	8.99e+28
576	2.33e−03	1.76e−02	5.80e−04	5.45e−03	2.88e−05	2.39e−04	9.25e−06	1.54e−04	3.34e+29

Table 2 Global Hermite collocation-IMQ error for the Stokes system with Dirichlet boundary conditions. Here, the shape parameter is $c = 1.5$

$N \backslash \Delta t$	$\mu = 1$ 1.00e−02		5.00e−03		1.00e−03		5.00e−04		Max Cond
	$\|\|e_y\|\|_\infty$	$\|\|e_{\nabla p}\|\|_\infty$	$\|\|e_y\|\|_\infty$	$\|\|e_{\nabla p}\|\|_\infty$	$\|\|e_y\|\|_\infty$	$\|\|e_{\nabla p}\|\|_\infty$	$\|\|e_y\|\|_\infty$	$\|\|e_{\nabla p}\|\|_\infty$	
117	7.04e−03	8.31e−01	6.67e−03	7.66e−01	5.36e−03	5.26e−01	5.16e−03	5.80e−01	3.94e+19
290	9.63e−04	3.57e−01	9.83e−04	3.61e−01	9.71e−04	3.47e−01	9.29e−04	3.25e−01	3.85e+19
576	5.82e−04	1.83e−01	1.65e−04	6.26e−02	4.22e−05	1.76e−02	4.02e−05	1.67e−02	7.48e+20
$N \backslash \Delta t$	$\mu = 1e$-06 1.00e−02		5.00e−03		1.00e−03		5.00e−04		Max Cond
	$\|\|e_y\|\|_\infty$	$\|\|e_{\nabla p}\|\|_\infty$	$\|\|e_y\|\|_\infty$	$\|\|e_{\nabla p}\|\|_\infty$	$\|\|e_y\|\|_\infty$	$\|\|e_{\nabla p}\|\|_\infty$	$\|\|e_y\|\|_\infty$	$\|\|e_{\nabla p}\|\|_\infty$	
117	2.40e−02	2.34e−01	2.39e−02	1.95e−01	2.98e−02	4.12e−01	3.52e−02	5.60e−01	8.11e+19
290	1.77e−03	1.01e−02	1.05e−03	8.88e−03	7.99e−04	3.69e−03	7.84e−04	3.30e−03	1.68e+20
576	1.90e−03	1.35e−02	4.77e−04	4.00e−03	2.26e−05	1.92e−04	1.47e−05	1.31e−04	1.65e+21

In the LHI scalar approach, we aim to obtain the RBF approximation of the analytic solution u of a linear steady partial differential well posed problem

$$\begin{cases} \mathcal{L}u(x) = f(x) & \text{in} \quad \Omega, \\ \mathcal{B}u(x) = g(x) & \text{on} \quad \partial\Omega, \end{cases} \tag{17}$$

where $\Omega \subset \mathbb{R}^d$ represents the spatial domain, the right-hand sides $f : \Omega \to \mathbb{R}$ and $g : \partial\Omega \to \mathbb{R}$ are given, $\mathcal{L}$ and $\mathcal{B}$ are linear partial differential operators in the domain Ω and on the contour $\partial\Omega$, which are locally approximated in the following way:

First, let $\Omega_n \subset \overline{\Omega}$ be a set of n_t total number of scattered nodes. Consider now the following subsets of Ω_n. Let $\Omega_c \subset \Omega_n$ be a subset of n_c nodes called centers, see Fig. 3.

Let now D^k be a disk, of variable radius, with center at the kth node of Ω_c, (recall that Ω_c is the set of centers of the disks), and consider the set of $n^{(k)}$ fixed number of nodes $\Omega_n \cap D^k$, see again Fig. 3. To perform the local discretization we introduce the following notation:

$\Omega_{\text{sc}} = \{x_1^{(k)}, \ldots, x_{N_{\text{sc}}^k}\} \subset \Omega_c$ be a set of N_{sc}^k nodes (called solution centers).
$\partial\Omega_{\text{fc}} = \{x_{N_{\text{sc}}^k+1}, \ldots, x_{N_{\text{sc}}^k+N_{\text{fc}}^k}\} \subset \partial\Omega$ the boundary nodes.
$\Omega_{\text{pdec}} = \{x_{N_{\text{sc}}^k+N_{\text{fc}}^k+1}, \ldots, x_{N_{\text{sc}}^k+N_{\text{fc}}^k+N_{\text{pdec}}^k}\} \subset \Omega$ interior nodes related $\mathcal{L}$.

For simplicity, we shall denote the node $x_1^{(k)}$ as the center of the disk D^k for each k. Then, every disk D^k defines a local sub-system as follows:

$$\mathcal{L}u(x) = f(x) \quad x \in \Omega_{\text{pdec}}, \tag{18}$$

$$\mathcal{B}u(x) = g(x) \quad x \in \partial\Omega_{\text{fc}}, \tag{19}$$

$$u(x_i) = h_i \quad x_i \in \Omega_{\text{sc}}, \tag{20}$$

where h_i are the unknown values.

This procedure generates a set of local linear systems given by

$$A^{(k)}\beta^{(k)} = d^{(k)}, \tag{21}$$

which are obtained by substituting in (18)–(20) the following radial ansatz for the local domains displayed (in circles) in Fig. 3

$$\hat{u}^{(k)}(x) = \sum_{j=1}^{N_{\text{sc}}^k} \beta_j^{(k)}\phi_j(r) + \sum_{j=N_{\text{sc}}^k+1}^{N_{\text{sc}}^k+N_{\text{fc}}^k} \beta_j^{(k)}\mathcal{B}^{\xi}\phi_j(r) + \sum_{j=N_{\text{sc}}^k+N_{\text{fc}}^k+1}^{N_{\text{sc}}^k+N_{\text{fc}}^k+N_{\text{pdec}}^k} \beta_j^{(k)}\mathcal{L}^{\xi}\phi_j(r) + p_k^m. \tag{22}$$

Here, k is the local system index, N_{sc}^k the number of solution centers in the local system, N_{pdec}^k denotes the number of PDE centers in the local system and N_{fc}^k is the number of boundary centers in the local system. Besides, $\mathcal{L}^{\xi}\phi_j(r) := \mathcal{L}\phi(\|x - \xi\|)|_{\xi=\xi_j}$, $\mathcal{B}^{\xi}\phi_j(r) :=$

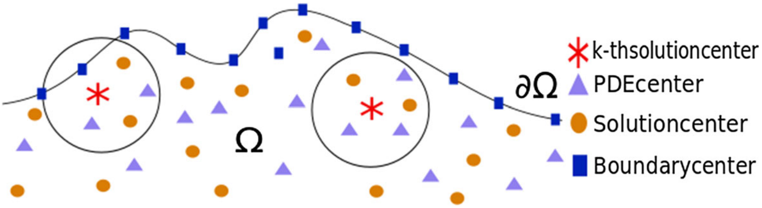


Fig. 3 Centers and local subdomains for the LHI method

$\mathcal{B}\phi(\|x-\xi\|)|_{\xi=\xi_j}$, being $\phi(r)$ the inverse multi-quadric and p_k^m a polynomial in $\mathbb{R}^d$ of degree m, which is an element of the null space of (17). The momentum condition is also required in this step, see Stevens et al. (2011). Thus, the local linear system (21) can be expressed in vectorial notation by defining $A^{(k)}$ (called Gram matrix) and the right-hand vector $d^{(k)}$ as follows

$$A^{(k)} = \begin{bmatrix} \Phi_{ij} & \mathcal{B}^{\xi}[\Phi_{ij}] & \mathcal{L}^{\xi}[\Phi_{ij}] & P_{ij} \\ \mathcal{B}^{x}[\Phi_{ij}] & \mathcal{B}^{x}\mathcal{B}^{\xi}[\Phi_{ij}] & \mathcal{B}^{x}\mathcal{L}^{\xi}[\Phi_{ij}] & \mathcal{B}^{x}[P_{ij}] \\ \mathcal{L}^{x}[\Phi_{ij}] & \mathcal{L}^{x}\mathcal{B}_{,}[\Phi_{ij}] & \mathcal{L}^{x}\mathcal{L}_{,}[\Phi_{ij}] & \mathcal{L}^{x}[P_{ij}] \\ P_{ji} & \mathcal{B}^{\xi}[P_{ji}] & \mathcal{L}^{\xi}[P_{ji}] & 0 \end{bmatrix} \quad \text{and} \quad d^{(k)} = \begin{bmatrix} h_i \\ g_i \\ f_i \\ 0 \end{bmatrix},$$

which is well known to be invertible, see Wendland (2004). Thus, we have that $\beta^{(k)} = (A^{(k)})^{-1}d^{(k)}$, and using (22), $\hat{u}^{(k)}(x)$ can be rewritten by

$$\hat{u}^{(k)}(x) = H(x)\beta^{(k)} = H(x)(A^{(k)})^{-1}d^{(k)} = W^{(k)}d^{(k)}, \tag{23}$$

where

$$H^{(k)}(x) = \left[\phi(\|x-\xi\|) \quad \mathcal{B}^{\xi}\phi(x-\xi) \quad \mathcal{L}^{\xi}\phi(\|x-\xi\|) \quad p^m(x)\right]$$

and $W^{(k)} := H(x)(A^{(k)})^{-1}$. $W^{(k)}$ is known as the vector of weights (Wendland 2004). Now, if $\mathcal{J}$ is an arbitrary differential operator given, we can compute its value at $\hat{u}^{(k)}$ by

$$\mathcal{J}\hat{u}^{(k)}(x) = \mathcal{J}H(x)(A^{(k)})^{-1}d^{(k)} = \mathcal{J}(W^{(k)})(x)d^{(k)}.$$

Let $u_c = \left[u(x_1^{(k)})\right]_{k=1}^{n_c^{(k)}}$ be the set of values of the exact solution at the centers of each disk D^k, which are unknown values (they belong to the vector $d^{(k)}$). In order to obtain these unknown values, we consider the following system of equations

$$h\left(x_1^{(k)}\right) = W_{\mathcal{J}}^{(k)}\left(x_1^{(k)}\right)d^{(k)}, \quad k = 1, \ldots, n_c, \tag{24}$$

where $W_{\mathcal{J}}^{(k)} = \mathcal{J}(W^{(k)})$ and $(h, \mathcal{J})$ is defined by $(f, \mathcal{L})$.

We shall denote by $Su_c = b$ the linear system (24), whose variables are the values at the solution centers Ω_c. Recall that each row of the matrix S is composed by zero elements except for the weights, (centers), corresponding to each disk D^k, that is, each row of size n_c, has only $n_c^{(k)}$ elements different form zero. Moreover, since $n_c \gg n_c^{(k)}$, the matrix S is sparse. To built the matrix S efficiently, i.e., to compute the weights, we can solve the following equations

$$A^{(k)}W_{\mathcal{J}}^{(k)}\left(x_1^{(k)}\right) = \mathcal{J}H^{(k)}(x_1^{(k)}), \quad k = 1, \ldots, n_c. \tag{25}$$

Since the matrix S is sparse standard solvers and preconditioners can be used. Besides, it is worth pointing out that using the method of lines and a proper numerical time integrator, non-stationary linear PDEs problems can be solve by the LHI method. To review exhaustively the LHI method, the interested reader can see (Stevens et al. 2011; Fasshauer 2007) and references therein.

4.2 Steady-state problems: Div-free RBF, LHI method

In this subsection, we study the stationary Stokes problem using divergence-free RBF. We describe the vectorial LHI algorithm for the system

$$\begin{cases} -\mu \Delta \mathbf{y} + \nabla p = \mathbf{F} & \text{in} \quad Q, \\ \nabla \cdot \mathbf{y} = 0 & \text{in} \quad Q, \\ \mathcal{B}(\mathbf{y}) = \mathbf{g} & \text{on} \quad \Sigma, \end{cases} \tag{26}$$

where $\mathbf{F}, \mathbf{g}$ are known and $\mathcal{B}$ indicates the boundary condition operator (see (3)), such that $\mathcal{B}(\mathbf{y})$ is a vector value function from $\mathbb{R}^d$ to $\mathbb{R}^d$. Let us define a vector $\mathbf{u} := (\mathbf{u}_n, u_{n+1}) = (\mathbf{y}, p) \in \mathbb{R}^{d+1}$ and the operators $\boldsymbol{\mathcal{L}}, \boldsymbol{\mathcal{B}}^D, \boldsymbol{\mathcal{B}}^{Ns}$ such that the left-hand side of (26) is given by

$$\boldsymbol{\mathcal{L}}\mathbf{u} := -{}^{-}\Delta \mathbf{u}_n + \nabla u_{n+1} = (-\mu\Delta, \nabla) \cdot (\mathbf{u}_n, u_{n+1}); \quad \nabla \cdot \mathbf{u}_n = 0$$

and

$$\underbrace{\boldsymbol{\mathcal{B}}^D \mathbf{u} := \mathbf{u}_n}_{\text{Dirichlet}} \quad \text{or} \quad \boldsymbol{\mathcal{B}}^{Ns}\mathbf{u} : = (\boldsymbol{\mathcal{B}}^1\mathbf{u}, \boldsymbol{\mathcal{B}}^2\mathbf{u}), \tag{27}$$

where

$$\underbrace{\boldsymbol{\mathcal{B}}^1\mathbf{u} = \mathbf{u}_n \cdot \boldsymbol{\nu}; \quad \boldsymbol{\mathcal{B}}^2\mathbf{u} := (\sigma(\mathbf{u}_n, u_{n+1})\boldsymbol{\nu})_{tg} = (\sigma(\mathbf{u})\boldsymbol{\nu})_{tg};}_{\text{Navier-slip}}$$

As mentioned in Sect. 4.1, for each disk D^k with center x_1^k, we must define a local system

$$\begin{cases} \mathbf{u_n}(x_i) = \overline{h_i} & \text{in} \quad \Omega_{\text{sc}}^k \subset \Omega_{\text{sc}} \bigcap D^k, \\ \\ \boldsymbol{\mathcal{L}}\mathbf{u} = \mathbf{F} & \text{in} \quad \Omega_{\text{pdec}}^k \subset \Omega_{\text{pdec}} \bigcap D^k, \\ \\ (27) & \text{on} \quad \Omega_{\text{fc}}^k \subset \Omega_{\text{fc}} \bigcap D^k. \end{cases} \tag{28}$$

To solve the systems (28), we first define the matrix-valued kernel

$$\boldsymbol{\Phi} = \begin{bmatrix} \boldsymbol{\Phi}_{\text{Div}} & 0 \\ 0 & \psi \end{bmatrix} : \mathbb{R}^d \to \mathbb{R}^{(d+1)\times(d+1)},$$

where $\boldsymbol{\Phi}_{\text{Div}} = \Delta \times \nabla \times \psi = \{-\Delta I + \nabla\nabla^T\}\psi(\mathbf{x}) : \mathbb{R}^d \to \mathbb{R}^{d+1}$ is a divergence-free positive definite kernel, Δ is the Laplace operator, I the identity matrix, and ψ is a global C^∞ positive definite scalar RBF.

Since we are choosing a free divergence radial kernel, the incompressibility equation is missing, and therefore, we lack the corresponding differential operator. This is the reason why in (28), we do not have the pressure as an unknown, never the less we will explain how it is possible compute it. Defining the canonical projection operator $I_j : \mathbb{R}^{(d+1)} \to \mathbb{R}$ as $I_j(u) = u_j$, and the operator $L_j(u) = I_j(\boldsymbol{\mathcal{L}u})$, we have that system (28) can be written by

$$\begin{cases} I_j(\mathbf{u_n}(x_i)) = I_j(\overline{h_i}) & j = 1...d, \ x_i \in \Omega_{\text{sc}}^k \subset \Omega_{\text{sc}} \bigcap D^k, \\ \\ L_j(\mathbf{u})(x_i) = \mathbf{F_j}(x_i) & j = 1...d, \ x_i \in \Omega_{\text{pdec}}^k \subset \Omega_{\text{pdec}} \bigcap D^k, \\ \\ (27) & \text{on} \qquad \Omega_{\text{fc}}^k \subset \Omega_{\text{fc}} \bigcap D^k. \end{cases} \tag{29}$$

Therefore, using the generalized interpolation collocation method (Wendland 2004), the ansatz for the Stokes equation is given by

$$\begin{aligned}(\hat{\mathbf{y}}^{(k)}, \hat{p}^{(k)})(\boldsymbol{x}) &= \sum_{i=1}^{d}\sum_{j=1}^{N_{\text{sc}}^k} I_i^{\xi}\boldsymbol{\Phi}(\boldsymbol{x}-\boldsymbol{\xi}_{\boldsymbol{j}}^{\mathbf{sc}(\boldsymbol{k})})\alpha^k_{(i-1)N^k_{\text{sc}}+j}\\ &+\sum_{i=1}^{d}\sum_{j=1}^{N_{\text{fc}}^k} B_i^{\xi}\boldsymbol{\Phi}(\boldsymbol{x}-\boldsymbol{\xi}_{\boldsymbol{j}}^{\mathbf{fc}(\boldsymbol{k})})\beta^k_{(i-1)N^k_{\text{fc}}+j}\\ &+\sum_{i=1}^{d}\sum_{j=1}^{N_{\text{pdec}}^k} L_i^{\xi}\boldsymbol{\Phi}(\boldsymbol{x}-\boldsymbol{\xi}^{\mathbf{pdec}(\boldsymbol{k})})\beta^k_{(i-1)N^k_{\text{pdec}}+j},\end{aligned}\tag{30}$$

where $(\gamma, \alpha, \beta) \in \mathbb{R}^{d(N^k_{\text{sc}}+N^k_{\text{fc}}+N^k_{\text{pdec}})}$, $N_{\text{sc}},\ N_{\text{fc}},\ N_{\text{pdec}}$ are the total numbers of local boundary, solution and pde centers respectively.

The terms $I_i^{\xi}\boldsymbol{\Phi}$, $\mathcal{B}_i^{\xi}\boldsymbol{\Phi}$, $L_i^{\xi}\boldsymbol{\Phi}$ are vector-valued functions from $\mathbb{R}^d \times \mathbb{R}^d$ to $\mathbb{R}^{d+1}$ defined by the application of the operators I_i^{ξ}, $\mathcal{B}_i^{\xi}$, L_i^{ξ} to each row of the kernel $\boldsymbol{\Phi}$, respectively.

Putting the ansatz (30) into the local Stokes system (29), we obtain the following local Gram matrix

$$A^k = \begin{pmatrix} I_1^x I_1^{\xi}\boldsymbol{\Phi} & \ldots & I_1^x I_d^{\xi}\boldsymbol{\Phi} & I_1^x \mathcal{B}_1^{\xi}\boldsymbol{\Phi} & \ldots & I_1^x \mathcal{B}_d^{\xi}\boldsymbol{\Phi} & I_1^x L_1^{\xi}\boldsymbol{\Phi} & \ldots & I_1^x L_d^{\xi}\boldsymbol{\Phi}\\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots\\ L_1^x I_1^{\xi}\boldsymbol{\Phi} & \ldots & L_1^x I_d^{\xi}\boldsymbol{\Phi} & L_1^x \mathcal{B}_1^{\xi}\boldsymbol{\Phi} & \ldots & L_1^x \mathcal{B}_d^{\xi}\boldsymbol{\Phi} & L_1^x L_1^{\xi}\boldsymbol{\Phi} & \ldots & L_1^x L_d^{\xi}\boldsymbol{\Phi}\\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots\\ \mathcal{B}_1^x I_1^{\xi}\boldsymbol{\Phi} & \ldots & \mathcal{B}_1^x I_d^{\xi}\boldsymbol{\Phi} & \mathcal{B}_1^x \mathcal{B}_1^{\xi}\boldsymbol{\Phi} & \ldots & \mathcal{B}_1^x \mathcal{B}_d^{\xi}\boldsymbol{\Phi} & \mathcal{B}_1^x L_1^{\xi}\boldsymbol{\Phi} & \ldots & \mathcal{B}_1^x L_d^{\xi}\boldsymbol{\Phi}\\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots\\ \mathcal{B}_d^x I_1^{\xi}\boldsymbol{\Phi} & \ldots & \mathcal{B}_d^x I_d^{\xi}\boldsymbol{\Phi} & \mathcal{B}_d^x \mathcal{B}_1^{\xi}\boldsymbol{\Phi} & \ldots & \mathcal{B}_d^x \mathcal{B}_d^{\xi}\boldsymbol{\Phi} & \mathcal{B}_d^x L_1^{\xi}\boldsymbol{\Phi} & \ldots & \mathcal{B}_d^x L_d^{\xi}\boldsymbol{\Phi} \end{pmatrix}, \tag{31}$$

which in turn let us to compute the weights of the differential operator L_j by solving the following systems

$$A^{(k)} W^{(k)}_{L_j}\left(x_1^{\text{sc}(k)}\right) = L_j H^{(k)}(x_1^{\text{sc}(k)}), \quad k = 1, ..., N_{\text{sc}}, \tag{32}$$

where $H^k(x)$ is given by

$$H^{(k)}(x) = \left(I_1^{\xi}\boldsymbol{\Phi}(\|\boldsymbol{x}-\boldsymbol{\xi}^{\boldsymbol{sc}(\boldsymbol{k})}\|)\ldots\mathcal{B}_1^{\xi}\boldsymbol{\Phi}(\|\boldsymbol{x}-\boldsymbol{\xi}^{\mathbf{fc}(\boldsymbol{k})}\|)\ldots L_d^{\xi}\boldsymbol{\Phi}(\|\boldsymbol{x}-\boldsymbol{\xi}^{\mathbf{pdec}(\boldsymbol{k})}\|)\right)^T. \tag{33}$$

Once the weights are known, we can build the sparse global matrix from the following equations

$$W^{(k)}_{L_j}\left(x_1^{\text{sc}(k)}\right) d^{(k)} = F_j(x_1^{\text{sc}(k)}) \quad k = 1, .., N_{\text{sc}}, \quad j = 1, \ldots, d, \tag{34}$$

where

$$d^{(k)} = \left(y_1(x^{\text{sc}(k)}), \ldots, y_d(x^{\text{sc}(k)}), g_1(x^{\text{fc}(k)}), ..., g_d(x^{\text{fc}(k)}), F_1(x^{\text{pdec}(k)}), \ldots, F_d(x^{\text{pdec}(k)})\right)^T.$$

Defining the unknown values of the vector field as

$$\mathbf{y}(x^{\text{sc}}) = \left(y_1(x_1^{\text{sc}}), \ldots, y_1(x^{\text{sc}}_{N_{\text{sc}}}), ..., y_d(x^{\text{sc}}_{N_{\text{sc}}})\right)^T$$

and the known values as $\mathbf{g}(x^{\mathrm{fc}}) \in \mathbb{R}^{dN_{\mathrm{fc}}}$, $\boldsymbol{F}(x^{\mathrm{pde}}) \in \mathbb{R}^{dN_{\mathrm{pde}}}$, respectively.

The global system induced by (34) can be expressed in matrix form as follows:

$$\left(W_L^y \; W_L^B \; W_L^L \right) \begin{bmatrix} \mathbf{y}(x^{\mathrm{sc}}) \\ \mathbf{g}(x^{\mathrm{fc}}) \\ \boldsymbol{F}(x^{\mathrm{pde}}) \end{bmatrix} = \left[\boldsymbol{F}(x^{\mathrm{sc}}) \right], \tag{35}$$

where $W_L^y \in \mathbb{R}^{dN_{\mathrm{sc}} \times dN_{\mathrm{sc}}}$, $W_L^B \in \mathbb{R}^{dN_{\mathrm{sc}} \times dN_{\mathrm{fc}}}$,$W_L^B \in \mathbb{R}^{dN_{\mathrm{sc}} \times dN_{\mathrm{pde}}}$.

Therefore, to compute the velocity field, we must solve the following linear system

$$W_L^y \mathbf{y}(x^{\mathrm{sc}}) = \boldsymbol{F}(x^{\mathrm{sc}}) - W_L^B \mathbf{g}(x^{\mathrm{fc}}) - W_L^L \boldsymbol{F}(x^{\mathrm{pde}}). \tag{36}$$

To compute the pressure gradient we need to obtain the weights linked to the partial derivatives of the pressure component of the local Anzats. In other words, we need to compute the weight of the operators

$$LP_i(y, p) \equiv \frac{\partial I_{d+1}(y, p)}{\partial x_i}, \quad i = 1, 2. \tag{37}$$

Again, this is performed in solving the following local systems:

$$A^{(k)} W_{LP_i}^{(k)} \left(x_1^{\mathrm{sc}(k)} \right) = LP_i H^{(k)}(x_1^{\mathrm{sc}(k)}), \quad k = 1, \ldots, N_{\mathrm{sc}}. \tag{38}$$

Once these weights are obtained and assuming that $\mathbf{y}(x^{\mathrm{sc}})$ have been computed via (36), we just have to do the following matrix multiplication

$$W_{LP} \begin{bmatrix} \mathbf{y}(x^{\mathrm{sc}}) \\ \mathbf{g}(x^{\mathrm{fc}}) \\ \boldsymbol{F}(x^{\mathrm{pde}}) \end{bmatrix} = \left[\nabla p(x^{\mathrm{sc}}) \right]. \tag{39}$$

Remark 41 It is important to highlight that in order to avoid singularity of the sparse system, we need that $x_1^{\mathrm{sc}(k)} \notin \Omega_{\mathrm{pdec}}^k$ (see Stevens et al. 2011).

4.3 Numerical results: stationary problem, LHI-Div-free IMQ method

Using the LHI Div-free IMQ-RBFs technique, we present numerical results concerning to the convergence order for Dirichlet and Navier-slip boundary condition. Also we verify that the errors, even for small number of support centers and different values of the diffusion parameter μ are excellent.

We consider the non-convex domain whose boundary was defined in (16) and the following exact solution to (26):

$$\mathbf{u}(\boldsymbol{x}) = \left(-\pi y sin(\frac{\pi}{2}(x^2 + y^2)), \pi x sin(\frac{\pi}{2}(x^2 + y^2) \right), \quad p(\boldsymbol{x}) = sin(x - y).$$

Here, the non-dimensional shape parameter is $c = 0.1$. Tables 3, 4 contains the approximation orders in the L^2 and L^∞ norms of the velocity field and pressure. We denote such an errors by $e_u := \mathbf{u}_{\mathrm{exact}} - \mathbf{u}_{\mathrm{aprox}}$ and $e_{\nabla p} := \nabla \mathbf{p}_{\mathrm{exact}} - \nabla \mathbf{p}_{\mathrm{aprox}}$. In all experiments, we have used extended precision via the mpfr++ library in c++ to overcome the ill condition Gram matrix.

From Tables 3 and 4, it can be appreciated that we obtain spectral order of convergence as the number of local nodes increases or equivalently if the fill distance decreases. Also the results are excellent both for $\mu = 1$ and $\mu = 10^{-3}$, and the error consistently decreases as the number of local nodes increases. The main point to be noted from this tables is that we can use up to 24,000 total number of nodes in the computations. This can not be done

Table 3 Div-free IMQ-LHI error table for stationary case with Dirichlet boundary condition and shape parameter $c = 1.0$

$\mu = 1$							
Total nodes	Local nodes	$\|\|e_y\|\|_{L_2}$	$\|\|e_y\|\|_\infty$	$\|\|e_{\nabla p}\|\|_{L_2}$	$\|\|e_{\nabla p}\|\|_\infty$	local cond	Sparse cond
1010	15	3.14e−06	4.96e−06	4.45e−04	2.00e−03	4.74e+21	5.05e+07
1010	20	1.49e−06	2.21e−06	3.65e−04	1.15e−03	1.24e+25	4.02e+08
1010	30	3.63e−09	1.12e−08	1.32e−06	8.83e−06	7.37e+30	1.44e+08
1010	40	4.11e−10	1.22e−09	1.66e−07	8.51e−07	1.55e+35	1.04e+09
2177	15	3.83e−06	5.91e−06	4.52e−04	3.26e−03	8.01e+23	5.57e+08
2177	20	5.40e−08	9.89e−08	1.11e−05	4.84e−05	2.96e+27	6.53e+08
2177	30	2.08e−10	9.06e−10	9.14e−08	6.22e−07	1.24e+34	1.14e+09
2177	40	2.97e−12	1.14e−11	1.25e−09	4.14e−09	1.16e+39	2.23e+09
5924	15	2.36e−06	4.99e−06	2.81e−04	1.81e−03	3.24e+26	3.49e+10
5924	20	8.23e−09	1.10e−08	1.85e−06	3.04e−05	4.29e+30	1.45e+10
5924	30	1.08e−12	3.43e−12	7.17e−10	4.19e−09	1.62e+38	1.27e+10
5924	40	5.38e−14	2.52e−13	5.74e−11	3.36e−10	1.16e+44	5.82e+10
15051	15	1.64e−07	3.12e−07	3.06e−05	2.11e−04	9.63e+28	1.76e+11
15051	20	1.83e−09	2.81e−09	7.45e−07	5.29e−06	4.45e+33	4.65e+11
15051	30	3.87e−14	9.73e−14	4.27e−11	5.27e−10	9.63e+41	2.17e+11
15051	40	1.18e−16	6.70e−16	1.41e−13	9.34e−13	3.63e+48	1.64e+11
23461	15	3.31e−07	5.92e−07	3.53e−05	1.67e−04	2.57e+30	1.67e+12
23461	20	3.67e−10	5.58e−10	2.12e−07	2.82e−06	1.02e+35	9.00e+11
23461	30	4.11e−14	6.82e−14	3.15e−11	6.81e−10	5.47e+43	2.01e+12
23461	40	1.45e−17	4.05e−17	1.44e−14	2.33e−13	2.97e+50	1.87e+12

Table 3 continued

$\mu = 1$							
Total nodes	Local nodes	$\|\|e_y\|\|_{L_2}$	$\|\|e_y\|\|_\infty$	$\|\|e_{\nabla p}\|\|_{L_2}$	$\|\|e_{\nabla p}\|\|_\infty$	local cond	Sparse cond
$\mu = 1\text{e}{-}03$							
Total nodes	Local nodes	$\|\|e_y\|\|_{L_2}$	$\|\|e_y\|\|_\infty$	$\|\|e_{\nabla p}\|\|_{L_2}$	$\|\|e_{\nabla p}\|\|_\infty$	local cond	Sparse cond
1010	15	4.29e−05	7.65e−05	1.50e−05	1.16e−04	2.89e+19	5.84e+07
1010	20	1.88e−06	3.65e−06	7.90e−07	5.53e−06	1.85e+23	9.80e+07
1010	30	7.07e−08	1.68e−07	2.41e−08	1.06e−07	3.89e+28	2.85e+08
1010	40	1.37e−09	4.56e−09	5.59e−10	3.52e−09	1.18e+33	1.43e+10
2177	15	6.58e−06	1.50e−05	2.39e−06	1.87e−05	3.32e+21	1.82e+09
2177	20	1.19e−06	2.04e−06	2.57e−07	1.29e−06	5.13e+25	1.92e+09
2177	30	1.28e−09	5.39e−09	5.96e−10	6.09e−09	7.92e+31	1.94e+09
2177	40	9.12e−12	3.16e−11	6.13e−12	3.87e−11	5.37e+36	2.31e+09
5924	15	1.62e−04	2.96e−04	1.43e−05	7.55e−05	9.33e+23	3.29e+11
5924	20	1.66e−07	3.51e−07	6.17e−08	5.75e−07	1.17e+29	6.66e+10
5924	30	6.13e−12	1.81e−11	5.59e−12	4.52e−11	5.00e+35	5.76e+09
5924	40	2.16e−13	7.80e−13	1.83e−13	2.03e−12	3.30e+41	5.50e+10
15051	15	7.24e−07	1.19e−06	1.08e−07	1.08e−06	2.94e+26	2.81e+11
15051	20	4.59e−08	6.95e−08	7.24e−09	1.46e−07	1.09e+32	1.89e+12
15051	30	3.23e−13	6.31e−13	4.48e−13	5.31e−12	5.36e+39	8.10e+10
15051	40	8.62e−15	3.07e−14	2.17e−14	1.94e−13	1.45e+46	8.37e+11
23461	15	4.99e−07	8.47e−07	1.11e−07	1.16e−06	2.86e+27	7.96e+11
23461	20	8.39e−10	1.28e−09	3.62e−10	7.71e−09	1.94e+33	1.80e+12
23461	30	6.43e−14	9.86e−14	9.15e−14	1.22e−12	2.70e+41	2.23e+11
23461	40	2.21e−16	9.69e−16	5.04e−16	7.49e−15	1.40e+48	9.00e+11

Table 4 Div-free IMQ-LHI error table for stationary case with Navier-slip boundary condition and shape parameter $c = 1.0$

$\mu = 1$

Total nodes	Local nodes	$\|\|e_y\|\|_{L_2}$	$\|\|e_y\|\|_\infty$	$\|\|e_{\nabla p}\|\|_{L_2}$	$\|\|e_{\nabla p}\|\|_\infty$	local cond	Sparse cond
1010	15	2.02e−05	3.88e−05	1.93e−03	6.68e−03	4.32e+21	4.57e+07
1010	20	3.18e−06	8.24e−06	5.00e−04	2.64e−03	9.56e+24	2.03e+08
1010	30	5.26e−08	2.12e−07	9.79e−06	4.59e−05	6.36e+30	1.50e+09
1010	40	4.19e−10	2.44e−09	1.38e−07	8.23e−07	1.45e+35	2.30e+09
2177	15	4.35e−06	9.47e−06	6.66e−04	2.69e−03	7.60e+23	4.21e+08
2177	20	1.63e−07	5.19e−07	2.02e−05	7.65e−05	2.69e+27	2.56e+09
2177	30	1.67e−09	4.79e−09	3.12e−07	8.17e−07	1.11e+34	5.76e+09
2177	40	4.13e−11	9.94e−11	7.59e−09	3.10e−08	1.31e+39	1.95e+10
5924	15	4.93e−07	9.11e−07	7.28e−05	7.66e−04	2.33e+26	4.79e+09
5924	20	1.55e−07	3.43e−07	2.20e−05	7.93e−05	4.97e+30	2.36e+11
5924	30	8.96e−11	2.20e−10	1.53e−08	1.08e−07	1.71e+38	3.33e+11
5924	40	2.37e−11	7.26e−11	7.61e−09	4.73e−08	1.17e+44	3.60e+13
15051	15	1.14e−07	1.86e−07	1.73e−05	1.26e−04	8.48e+28	2.78e+11
15051	20	2.07e−08	3.39e−08	2.22e−06	2.02e−05	3.86e+33	2.40e+13
15051	30	6.89e−11	2.32e−10	7.58e−09	4.71e−08	1.00e+42	2.62e+14
15051	40	1.98e−13	6.70e−13	2.05e−11	1.15e−10	2.69e+48	4.42e+14
23461	15	4.30e−06	8.38e−06	3.06e−04	3.47e−03	8.51e+29	7.42e+12
23461	20	6.57e−10	1.49e−09	1.57e−07	2.00e−06	7.12e+34	5.83e+12
23461	30	1.20e−10	5.91e−10	3.11e−08	4.24e−07	4.55e+43	2.69e+15
23461	40	6.94e−15	2.92e−14	2.12e−12	1.21e−11	2.73e+50	9.49e+14

Table 4 continued

$\mu = 1$

Total nodes	Local nodes	$\|\|e_y\|\|_{L_2}$	$\|\|e_y\|\|_\infty$	$\|\|e_{\nabla p}\|\|_{L_2}$	$\|\|e_{\nabla p}\|\|_\infty$	local cond	Sparse cond

$\mu = 1\text{e}{-}03$

Total nodes	Local nodes	$\|\|e_y\|\|_{L_2}$	$\|\|e_y\|\|_\infty$	$\|\|e_{\nabla p}\|\|_{L_2}$	$\|\|e_{\nabla p}\|\|_\infty$	local cond	Sparse cond
1010	15	3.29e−05	6.80e−05	6.31e−06	2.51e−05	5.09e+19	3.27e+08
1010	20	4.36e−06	1.82e−05	5.66e−07	1.98e−06	2.25e+23	7.47e+07
1010	30	6.21e−07	1.72e−06	1.48e−07	7.93e−07	6.28e+28	1.37e+09
1010	40	2.32e−09	6.91e−09	5.46e−10	2.30e−09	1.30e+33	4.40e+09
2177	15	3.66e−05	6.08e−05	4.95e−06	2.34e−05	4.33e+21	1.86e+09
2177	20	7.75e−07	2.24e−06	7.93e−08	4.06e−07	5.67e+25	1.56e+09
2177	30	5.59e−09	1.63e−08	1.25e−09	5.67e−09	7.95e+31	1.05e+10
2177	40	4.17e−10	1.53e−09	7.60e−11	4.10e−10	5.71e+36	1.08e+11
5924	15	1.15e−05	4.14e−05	2.53e−06	3.76e−05	6.94e+23	1.56e+10
5924	20	2.89e−06	6.13e−06	1.99e−07	1.04e−06	1.59e+29	3.70e+11
5924	30	1.56e−10	4.61e−10	3.07e−11	1.60e−10	9.54e+35	1.16e+12
5924	40	2.11e−12	6.49e−12	2.31e−13	1.05e−12	3.57e+41	2.15e+12
15051	15	3.18e−04	4.62e−04	3.10e−05	4.15e−04	2.16e+26	4.50e+13
15051	20	6.38e−08	1.31e−07	8.65e−09	1.03e−07	1.25e+32	1.63e+12
15051	30	1.86e−11	6.80e−11	2.26e−12	1.40e−11	4.32e+39	2.81e+12
15051	40	2.23e−14	5.64e−14	2.93e−15	2.04e−14	1.44e+46	1.46e+13
23461	15	2.13e−05	3.02e−05	1.47e−06	1.19e−05	2.54e+27	3.02e+13
23461	20	1.20e−08	2.20e−08	1.54e−09	3.37e−08	1.79e+33	6.92e+12
23461	30	1.30e−12	3.48e−12	2.09e−13	2.63e−12	2.71e+41	4.62e+12
23461	40	2.25e−15	5.92e−15	4.75e−16	3.00e−15	1.95e+48	1.62e+13

with global collocation methods due to the high value of the condition number of the Gram matrix.

4.4 Evolutionary problems: hybrid-Div-free RBF, LHI method and BDF scheme

In this subsection we formulate a RBF-LHI vectorial technique for the evolutionary Stokes problem

$$\begin{cases} \mathbf{y}_t + L(\mathbf{y}, p) = \mathbf{f} & \text{in} \quad Q, \\ \nabla \cdot \mathbf{y} = 0 & \text{in} \quad Q, \\ \mathcal{B}\mathbf{y} = \mathbf{g} & \text{on} \quad \Sigma, \\ \mathbf{y}(\cdot, 0) = \mathbf{y}_0(\cdot) & \text{in} \quad Q, \end{cases} \tag{40}$$

where $L(\mathbf{y}, p) = -\mu \Delta \mathbf{y} + \nabla p$.

Observe that, for every $t \in (0, T)$, (40) can be seen as a stationary Stokes equation

$$\begin{cases} L(\mathbf{y}, p) = \mathbf{F} & \text{in} \quad \Omega, \\ \nabla \cdot \mathbf{y} = 0 & \text{in} \quad \Omega, \\ \mathcal{B}\mathbf{y} = \mathbf{g} & \text{on} \quad \partial\Omega, \end{cases} \tag{41}$$

where $\mathbf{F} = \mathbf{f} - \mathbf{y}_t$.

This implies that we can use the weight of the stationary system (see Eq. (32)) to approximate (41). Thus, using (34) we have

$$\begin{pmatrix} W_L^y \; W_L^B \; W_L^L \end{pmatrix} \begin{bmatrix} \mathbf{y}(t; x^{\mathrm{sc}}) \\ \mathbf{g}(t; x^{\mathrm{fc}}) \\ \boldsymbol{F}(t; x^{\mathrm{pde}}) \end{bmatrix} = \left[\boldsymbol{F}(t; x^{\mathrm{sc}}) \right]. \tag{42}$$

Now, assuming that $x^{\mathrm{sc}} = x^{\mathrm{pde}}$, we get the following ODE system

$$(I - W_L^L)\, \mathbf{y}_t(t; x^{\mathrm{sc}}) = -W_L^y\, \mathbf{y}(t; x^{\mathrm{sc}}) - W_L^B \mathbf{g}(t; x^{\mathrm{fc}}) + (I - W_L^L)\boldsymbol{F}(\boldsymbol{t}; \boldsymbol{x}^{\mathbf{sc}}), \tag{43}$$

where the boundary condition have been imposed in the LHI weights.

To solve the above system, we use a BDF2 scheme. Thus, in each time step we shall solve

$$\left(I - W_L^L + \frac{2\Delta t}{3} W_L^y \right) \mathbf{y}_{n+2} = \left(I - W_L^L \right) \left(\frac{2\Delta t}{3} \boldsymbol{F}_{\boldsymbol{n+2}} + \frac{4}{3} \mathbf{y}_{n+1} - \frac{1}{3} \mathbf{y}_n \right) - \frac{2\Delta t}{3} W_L^B \mathbf{g}_{n+2}. \tag{44}$$

4.5 Numerical results: evolutionary problem, Div-free Hybrid RBF-LHI method

In this subsection we use the theoretical description of the RBF-LHI method presented in Sect. 4.4, to solve the unsteady Stokes system. We also introduce the concept of Divergence free hybrid radial basis function, a generalization of the scalar hybrid RBF (see Mishra et al. (2018)), which allows us to build a global LHI matrix whose eigenvalues have negative real components. We present numerical results for different benchmark problems.

We stress that, for some cases and according to extensive numerical experimentation, no matter which parameters c, μ or h we select, it is not possible to obtain negative eigenvalues for IMQ kernel, thus most ODE solvers are unstable.

On the other hand, for hybrid kernels, the eigenvalues can always be obtained to be negative depending on the parameters we choose, thus providing the stability condition for ODE's solvers. We first define the concept of Div-free hybrid kernel.

Definition 2 Let $\mathbf{r} = \|\mathbf{x}\|$, $\psi_1(\mathbf{r}) = \exp^{-c\mathbf{r}^2}$ and $\psi_2(\mathbf{r}) = \mathbf{r}^{2n+1}$. The divergence free hybrid kernel $\mathbf{\Phi}_{\text{Div}} : \mathbb{R}^d \to \mathbb{R}^{d+1}$ is defined by

$$\mathbf{\Phi}_{\text{Div}}(\mathbf{x}) = \Delta \times \nabla \times \psi = \{-\Delta I + \nabla\nabla^T\}(\psi_1(\mathbf{x}) + \gamma_1\psi_2(\mathbf{x})).$$

where γ_1 is a positive real number.

By direct computation, we obtain

$$\mathbf{\Phi}_{\text{Div}}(\mathbf{x}) = \begin{pmatrix} -2\,c\,\mathrm{e}^{-c\,r^2}\,\left(2\,c\,(y_1-y_2)^2-1\right) & 4\,c^2\,\mathrm{e}^{-c\,r^2}\,(x_1-x_2)\,(y_1-y_2) \\ 4\,c^2\,\mathrm{e}^{-c\,r^2}\,(x_1-x_2)\,(y_1-y_2) & -2\,c\,\mathrm{e}^{-c\,r^2}\,\left(2\,c\,(x_1-x_2)^2-1\right) \end{pmatrix}$$
$$+\,\gamma_1\,(2n+1)\,r^{2n-3}\begin{pmatrix} -\left((y_1-y_2)^2\,(2n-1)+r^2\right) & (x_1-x_2)\,(y_1-y_2)\,(2n-1) \\ (x_1-x_2)\,(y_1-y_2)\,(2n-1) & -\left((x_1-x_2)^2\,(2n-1)+r^2\right) \end{pmatrix}$$

and the combined velocity–pressure kernel is given by

$$\mathbf{\Phi}(\mathbf{x}) = \begin{pmatrix} \mathbf{\Phi}_{Div}(\mathbf{x}) & 0 \\ 0 & \mathrm{e}^{-c_2\,r} + \gamma_2\,r^{2m+1} \end{pmatrix}. \tag{45}$$

Here, γ_2 is corresponding weight relative to the hybrid kernel related to the pressure.

In the following experiments and for stability reasons, we use a weight parameter γ_1, a shape parameter c_1 for the velocity and the corresponding values γ_2; c_2 for the pressure. Also we shall use the values $n = 3$ and $m = 1$ in Eq. (45), ie. r^7 for the $\mathbf{\Phi}_{Div}$ vector component and r^3 for the scalar component.

We note that the condition numbers of the local and global Gram matrices are considerable lower than for the non-hybrid, IMQ-RBF. This is in agreement with a recently, a new approach to reduce the ill-conditioning problem in RBF approximations by using a hybrid Gaussian-cubic kernel was proposed (Mishra et al. 2018). The basic idea behind such a hybridization is to obtain a kernel which utilizes the merits of two different kernels while compensating for the limitations of each and keeping the formulation as a standard RBF method.

The numerical computations of Tables 5, 6, 7 and 8 were obtained by using divergence free hybrid kernels and considering the following analytical solution to (40)

$$\begin{aligned} y_1(x, y, t) &= -\pi\ y\ \sin\left(\frac{\pi}{2}(tx^2+y^2)\right)\sin(20\pi t), \\ y_2(x, y, t) &= -\pi\ y\ \sin\left(\frac{\pi}{2}(tx^2+y^2)\right)\sin(20\pi t)t, \\ p(x, y, t) &= \sin(x-y+t). \end{aligned}$$

The total number of nodes for Tables 5, 6, 7 and 8 is 1010. To better appreciate the convergence of the method as the fill distance decrease we scale the domain by factor α ranging from 1 to 10^{-5}.

As in Sect. 3.3, we compare the velocity error in the L^2-norm between the exact and numerical solutions, i.e., $\boldsymbol{\epsilon}_{\mathbf{y}} = \mathbf{y}_{\text{exact}} - \mathbf{y}_{\text{aprox}}$.

From the tables presented in this subsection, we conclude that the local condition number for hybrid kernels are several orders of magnitude smaller than the values of tables presented in Sect. 4.3, which corresponds to the condition number for inverse multi-quadrics. We stress, however, that the error is greater for hybrids kernels than for IMQ. The values of the shape parameters were obtained by direct trial and error computation. Of course, up to now, there is no theory that tells us how to obtain these values. It is important to note that to obtain a good condition number, we only need to decrease the value γ_1 and decrease the shape parameter

Table 5 Error table for Stokes-unsteady, Dirichlet boundary condition with BDF2 LHI-semidiscret method $\mu = 1$, Pressure $\gamma_2 = 1\text{e}{-}06$, Velocity $c_1 = 0.5$, Pressure $c_2 = 5\text{e}{-}04$

Δt				2.00e−02	1.00e−02	1.00e−03	1.00e−04	1.00e−05
Fill distance	Local nodes	Velocity $-\gamma_1$	Max local cond	$\|\|e_y\|\|_{L_2}$	$\|\|e_y\|\|_{L_2}$	$\|\|e_y\|\|_{L_2}$	$\|\|e_y\|\|_{L_2}$	$\|\|e_y\|\|_{L_2}$
5.00e−02	15	2.00e−01	4.20e+11	3.89631e−01	1.37393e−01	2.28270e−02	2.26144e−02	2.26158e−02
5.00e−02	20	1.00e−01	1.62e+12	3.27511e−01	1.14382e−01	1.16019e−02	1.12192e−02	1.12195e−02
5.00e−02	30	1.00e−01	3.00e+12	3.17346e−01	1.10879e−01	1.03071e−02	9.88475e−03	9.88514e−03
5.00e−03	15	1.00e−01	1.12e+17	2.37872e−02	8.84353e−03	2.85082e−04	2.36036e−05	2.18281e−05
5.00e−03	20	1.00e−01	1.96e+17	2.37824e−02	8.84254e−03	2.84566e−04	1.62663e−05	1.35604e−05
5.00e−03	30	1.00e−01	3.98e+17	2.37821e−02	8.84248e−03	2.84493e−04	1.49392e−05	1.19358e−05
5.00e−04	15	1.00e−01	1.15e+22	2.37205e−03	8.81808e−04	2.83446e−05	8.96755e−07	3.59745e−08
5.00e−04	20	1.00e−01	1.99e+22	2.37205e−03	8.81808e−04	2.83446e−05	8.96564e−07	3.08596e−08
5.00e−04	30	1.00e−01	4.08e+22	2.37205e−03	8.81808e−04	2.83446e−05	8.96543e−07	3.02459e−08
5.00e−05	15	1.00e−01	1.15e+27	2.37201e−04	8.81791e−05	2.83440e−06	8.96463e−08	2.83496e−09
5.00e−05	20	1.00e−01	2.01e+27	2.37201e−04	8.81791e−05	2.83440e−06	8.96463e−08	2.83490e−09
5.00e−05	30	1.00e−01	4.28e+27	2.37201e−04	8.81791e−05	2.83440e−06	8.96463e−08	2.83490e−09

Table 6 Error table for Stokes-unsteady, Dirichlet boundary condition with BDF2 LHI-semidiscret method $\mu = 1\mathrm{e}{-}03$, Pressure $\gamma_2 = 1\mathrm{e}{-}06$, Velocity $c_1 = 0.5$, Pressure $c_2 = 5\mathrm{e}{-}04$

Δt				2.00e−02	1.00e−02	1.00e−03	1.00e−04	1.00e−05
Fill distance	Local nodes	Velocity $-\gamma_1$	Max local cond	$\|\|e_y\|\|_{L_2}$	$\|\|e_y\|\|_{L_2}$	$\|\|e_y\|\|_{L_2}$	$\|\|e_y\|\|_{L_2}$	$\|\|e_y\|\|_{L_2}$
5.00e−02	15	1.00e+00	1.20e+09	1.51164e+00	9.95465e−01	8.52428e−01	8.52643e−01	8.52750e−01
5.00e−02	20	1.00e−01	1.63e+10	8.20890e−01	4.66209e−01	3.80483e−01	3.80544e−01	3.80584e−01
5.00e−02	30	1.00e−01	3.26e+10	6.43212e−01	2.95525e−01	2.08385e−01	2.08408e−01	2.08428e−01
5.00e−03	15	1.00e−01	9.62e+14	2.42604e−02	8.98489e−03	7.02252e−04	6.42266e−04	6.42253e−04
5.00e−03	20	1.00e−01	1.67e+15	2.38317e−02	8.85552e−03	3.26210e−04	1.60253e−04	1.60013e−04
5.00e−03	30	1.00e−01	3.22e+15	2.37950e−02	8.84617e−03	3.05968e−04	1.13596e−04	1.13247e−04
5.00e−04	15	1.00e−01	9.51e+19	2.37205e−03	8.81808e−04	2.83447e−05	9.00079e−07	8.52535e−08
5.00e−04	20	1.00e−01	1.51e+20	2.37205e−03	8.81808e−04	2.83446e−05	8.97570e−07	5.25082e−08
5.00e−04	30	1.00e−01	3.41e+20	2.37205e−03	8.81808e−04	2.83446e−05	8.96529e−07	2.98050e−08
5.00e−05	15	1.00e−01	9.68e+24	2.37201e−04	8.81791e−05	2.83440e−06	8.96463e−08	2.83489e−09
5.00e−05	20	1.00e−01	1.54e+25	2.37201e−04	8.81791e−05	2.83440e−06	8.96463e−08	2.83487e−09
5.00e−05	30	1.00e−01	3.48e+25	2.37201e−04	8.81791e−05	2.83440e−06	8.96463e−08	2.83487e−09

Table 7 Error table for Stokes-unsteady, Navier-slip boundary condition with BDF2 LHI-semidiscret method $\mu = 1$, Pressure $\gamma_2 = 1e{-}06$,Velocity $c_1 = 1.0$,Pressure $c_2 = 5e{-}06$

Δt				2.00e−02	1.00e−02	1.00e−03	1.00e−04	1.00e−05
Fill distance	Local nodes	Velocity-γ_1	Max local cond	$\|\|e_y\|\|_{L_2}$	$\|\|e_y\|\|_{L_2}$	$\|\|e_y\|\|_{L_2}$	$\|\|e_y\|\|_{L_2}$	$\|\|e_y\|\|_{L_2}$
5.00e−02	15	2.00e−01	4.20e+11	5.34629e−01	2.03483e−01	8.57445e−02	8.56851e−02	8.56904e−02
5.00e−02	20	1.00e−01	1.62e+12	4.59891e−01	1.65007e−01	3.61239e−02	3.60060e−02	3.60083e−02
5.00e−02	30	1.00e−01	3.00e+12	4.83596e−01	1.75874e−01	4.08739e−02	4.07419e−02	4.07447e−02
5.00e−03	15	1.00e−01	1.12e+17	2.38759e−02	8.86207e−03	2.92628e−04	6.98930e−05	6.93184e−05
5.00e−03	20	1.00e−01	1.96e+17	2.38063e−02	8.84747e−03	2.86525e−04	3.70751e−05	3.59725e−05
5.00e−03	30	1.00e−01	3.98e+17	2.38141e−02	8.84912e−03	2.88693e−04	5.11912e−05	5.04004e−05
5.00e−04	15	1.00e−01	1.15e+22	2.37205e−03	8.81808e−04	2.83447e−05	8.99279e−07	7.63437e−08
5.00e−04	20	1.00e−01	2.10e+22	2.37205e−03	8.81808e−04	2.83446e−05	8.97295e−07	4.75745e−08
5.00e−04	30	1.00e−01	4.08e+22	2.37205e−03	8.81808e−04	2.83446e−05	8.97518e−07	5.16150e−08
5.00e−05	15	1.00e−01	1.15e+27	2.37201e−04	8.81791e−05	2.83440e−06	8.96463e−08	2.83576e−09
5.00e−05	20	1.00e−01	2.01e+27	2.37201e−04	8.81791e−05	2.83440e−06	8.96463e−08	2.83511e−09
5.00e−05	30	1.00e−01	4.28e+27	2.37201e−04	8.81791e−05	2.83440e−06	8.96463e−08	2.83526e−09

Table 8 Error table for Stokes-unsteady, Navier-slip boundary condition with BDF2 LHI-semidiscret method $\mu = 1\text{e}{-}03$, Pressure $\gamma_2 = 1\text{e}{-}06$,Velocity $c_1 = 1.0$, Pressure $c_2 = 5\text{e}{-}06$

Δt				2.00e−02	1.00e−02	1.00e−03	1.00e−04	1.00e−05
Fill distance	Local nodes	Velocity-γ_1	Max local cond	$\|\|e_y\|\|_{L_2}$	$\|\|e_y\|\|_{L_2}$	$\|\|e_y\|\|_{L_2}$	$\|\|e_y\|\|_{L_2}$	$\|\|e_y\|\|_{L_2}$
5.00e−02	15	1.00e−01	2.30e+11	1.67563e+00	1.10142e+00	9.45213e−01	9.45553e−01	9.45679e−01
5.00e−02	20	1.00e−01	2.01e+11	1.51242e+00	9.90276e−01	8.43845e−01	8.44026e−01	8.44133e−01
5.00e−02	30	1.00e−01	3.50e+11	1.45439e+00	9.47934e−01	8.06837e−01	8.06999e−01	8.07101e−01
5.00e−03	15	8.00e−01	1.58e+14	2.29486e−01	1.18803e−01	1.10860e−01	1.11219e−01	1.11242e−01
5.00e−03	20	1.00e−01	2.19e+15	3.66377e−02	1.41194e−02	5.86116e−03	5.85250e−03	5.85300e−03
5.00e−03	30	1.00e−01	4.17e+15	5.24203e−02	2.30795e−02	1.97011e−02	1.97778e−02	1.97812e−02
5.00e−04	15	5.12e+01	2.70e+17	2.88384e−03	1.01449e−03	8.02069e−05	7.48608e−05	7.48610e−05
5.00e−04	20	1.00e−01	2.08e+20	2.37255e−03	8.81927e−04	2.84960e−05	3.06396e−06	2.93023e−06
5.00e−04	30	1.00e−01	3.72e+20	2.49592e−03	9.14376e−04	5.79928e−05	5.05434e−05	5.05393e−05
5.00e−05	15	4.10e+02	3.33e+21	2.37209e−04	8.81807e−05	2.83544e−06	1.17900e−07	7.66347e−08
5.00e−05	20	1.00e−01	1.79e+25	2.37201e−04	8.81791e−05	2.83440e−06	8.96478e−08	2.88258e−09
5.00e−05	30	8.00e−01	5.21e+24	2.37201e−04	8.81791e−05	2.83440e−06	8.97063e−08	4.33499e−09

of the pressure for a fixed γ_2. Here, γ_1 is the parameter of the convex combination of the hybrid kernel related to the velocity, and γ_2 the parameter related to the pressure. This means that the algorithm is relatively stable with respect to the variation of the parameters.

5 Numerical control problem: RBF-Div-free LHI techniques and FEM method

The numerical solution of the approximate controllability problem for the two-dimensional Stokes system with few scalar controls is carried out in this section. Here, both Dirichlet and Navier-slip boundary conditions are considered. The numerical implementation follows the method formerly developed through RBF-LHI technique. The numerical results obtained by this method are compared against the finite element method (FEM) formulation for Dirichlet and Navier-slip boundary conditions.

Following Glowinski et al. (2008), the conjugate gradient method, CGM, is implemented with a stopping criteria of $\epsilon = 10^{-8}$ for solving the dual system (1), (6). Again, the domain $\Omega \subset \mathbb{R}^2$ is the star whose boundary is parametrized by the curve

$$C = \Big\{(\theta, \rho(\theta)) \in \mathbb{R}^2 : \rho(\theta) = 0.8 + \sin(6\theta) + \sin(3\theta), \quad \theta \in [0, 2\pi)\Big\}. \tag{46}$$

The observation set $\omega = \{(x, y) \in \mathbb{R}^2 : \frac{x^2}{2.5e^{-3}} + \frac{y^2}{4e^{-4}} < 1.0\}$, and $T = 1.0$.

In all cases, we use a uniform mesh of 1010 points generated with FreeFem++, the time step size is $\Delta t = 5 \times 10^{-3}$ and diffusion coefficient $\mu = 1e$-03. For the initial condition, we choose

$$(y_1^0, y_2^0) = (-10^2\pi\ y\ \cos\left(\frac{\pi}{2}(x^2+y^2)\right)^2, 10^2\pi\ x\ \cos\left(\frac{\pi}{2}(x^2+y^2)\right)^2)$$

Regarding the functional (4), we set the regularization parameters $\beta_1 = 1.0e$-03 , $\beta_2^{-1} = 0$ by having controls with both non-zeros scalar component ($\mathbf{v} = (v_1, v_2)$) and $\beta_1^{-1} = 0$, $\beta_2 = 1.0e$-03 by considering controls with one scalar control (either $\mathbf{v} = (v_1, 0)$ or $\mathbf{v} = (0, v_2)$).

For the numerical experiments, we use a triangular mesh for two specific reasons, the first one is to have a fair comparison between RBF-LHI and finite element, and the second is because CGM requires to calculate integrals over the domain and although it is possible to compute them for scattered nodes, it is more efficient to use the triangulation for LHI-RBF method to calculate integrals with $\mathbb{P}_1$-type elements.

5.1 Divergence free RBF-LHI method for the control problem

To generate the divergence free kernel for the LHI method, we use hybrid kernel with $n = 3$ and $m = 1$. In particular, for Dirichlet boundary, we used the following set of parameter: $\gamma_1 = 1e$-01,$\gamma_2 = 1.0e$-5,$c_1 = 0.5$,$c_2 = 5.e$-8. For Navier-slip boundary conditions, we use: $\gamma_1 = 1e$-03,$\gamma_2 = 1e$-08,$c_1 = 1.0$,$c_2 = 5.e$-10.

Table 9 shows the number of iterations to achieve the stopping criteria $\epsilon = 10^{-8}$ in the CGM implemented.

Table 10 and Figs. 4 and 5 show the L^2-norm of the velocity vector field for the approximate control problems as a function of time. The numerical control function $\mathbf{v}$ has all possible structures, namely, $\mathbf{v} = \mathbf{0}$, $\mathbf{v} = (v_1, v_2)$, $\mathbf{v} = (v_1, 0)$ and $\mathbf{v} = (0, v_2)$.

Table 9 Number of iterations for obtaining the convergence criteria of the CGM for Hybrid LHI-RBF

B.C	$\mathbf{v} = (v_1, v_2)$	$\mathbf{v} = (v_1, 0)$	$\mathbf{v} = (0, v_2)$
Navier-slip	116	68	99
Dirichlet	78	208	304

Table 10 Evolution in time of the L^2-norm for the solution of the approximate control problem with few scalar controls (LHI-RBF with hybrid kernel) $\mu = 1.0\text{e}{-03}$

Boundary condition = Dirichlet				
t	$\mathbf{v} = \mathbf{0}$	$\mathbf{v} = (v_1, v_2)$	$\mathbf{v} = (v_1, 0)$	$\mathbf{v} = (0, v_2)$
0.0E+00	6.103E+00	6.103E+00	6.103E+00	6.103E+00
1.0E−01	1.561E+00	1.551E+00	1.551E+00	1.550E+00
2.0E−01	7.845E−01	7.658E−01	7.651E−01	7.656E−01
3.0E−01	4.221E−01	3.988E−01	3.977E−01	3.988E−01
4.0E−01	2.318E−01	2.064E−01	2.058E−01	2.066E−01
5.0E−01	1.283E−01	1.025E−01	1.032E−01	1.028E−01
6.0E−01	7.114E−02	4.664E−02	4.943E−02	4.694E−02
7.0E−01	3.949E−02	1.797E−02	2.319E−02	1.818E−02
8.0E−01	2.193E−02	5.207E−03	1.167E−02	5.203E−03
9.0E−01	1.218E−02	1.192E−03	5.297E−03	1.118E−03
1.0E+00	6.761E−03	8.748E−05	1.093E−03	1.271E−04
Boundary condition = Navier-Slip				
t	$\mathbf{v} = \mathbf{0}$	$\mathbf{v} = (v_1, v_2)$	$\mathbf{v} = (v_1, 0)$	$\mathbf{v} = (0, v_2)$
0.0E+00	6.982E+00	6.982E+00	6.982E+00	6.982E+00
1.0E−01	3.495E+00	3.335E+00	3.385E+00	3.262E+00
2.0E−01	2.413E+00	2.112E+00	2.213E+00	2.004E+00
3.0E−01	1.713E+00	1.322E+00	1.473E+00	1.210E+00
4.0E−01	1.226E+00	7.921E−01	9.911E−01	6.993E−01
5.0E−01	8.801E−01	4.454E−01	6.824E−01	3.838E−01
6.0E−01	6.331E−01	2.313E−01	4.845E−01	2.027E−01
7.0E−01	4.559E−01	1.113E−01	3.460E−01	1.078E−01
8.0E−01	3.285E−01	5.040E−02	2.260E−01	5.858E−02
9.0E−01	2.368E−01	1.815E−02	1.146E−01	2.558E−02
1.0E+00	1.708E−01	2.149E−03	5.506E−02	6.359E−03

For Dirichlet boundary condition $\gamma_1 = 1\text{e}{-01}$,$\gamma_2 = 1\text{e}{-05}$, and for Navier-slip $\gamma_1 = 1\text{e}{-03}$,$\gamma_2 = 1\text{e}{-08}$

Note that the solution of system (1) with Dirichlet boundary conditions, namely the state variable, decay faster to zero in the L^2-norm sense than the solution corresponding to the Navier-slip homogeneous boundary conditions. Numerically, this behavior is described in Table 10.

Remark 51 The previous results are valid for homogeneous Navier-slip boundary conditions; however, to understand the balance between Dirichlet and Navier-slip conditions and the convergence order of the state associated to the approximate control problem, the friction between the fluid and the boundary should be considered (that is, non-homogeneous

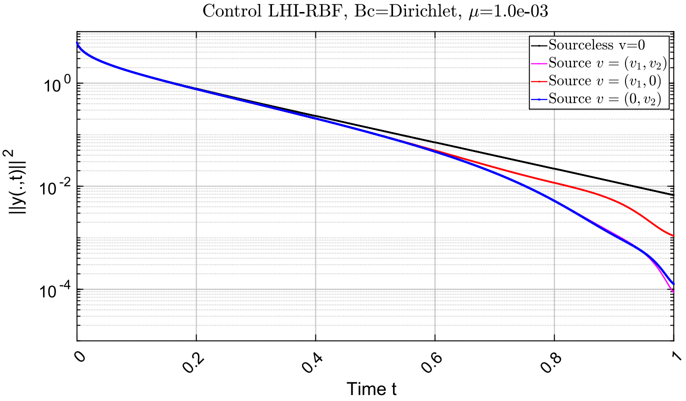


Fig. 4 L^2-norm square solution of the velocity field (as a function of time) for the approximate control problem with controls $\mathbf{v} = \mathbf{0}$ (black), $\mathbf{v} = (v_1, v_2)$ (pink), $\mathbf{v} = (v_1, 0)$ (red) and $\mathbf{v} = (0, v_2)$ with Dirichlet boundary condition. LHI-RBF hybrid kernel, with parameters $\gamma_1 = 1e{-}01$,$\gamma_2 = 1.0e{-}5$,$c_1 = 0.5$,$c_2 = 5.e{-}8$

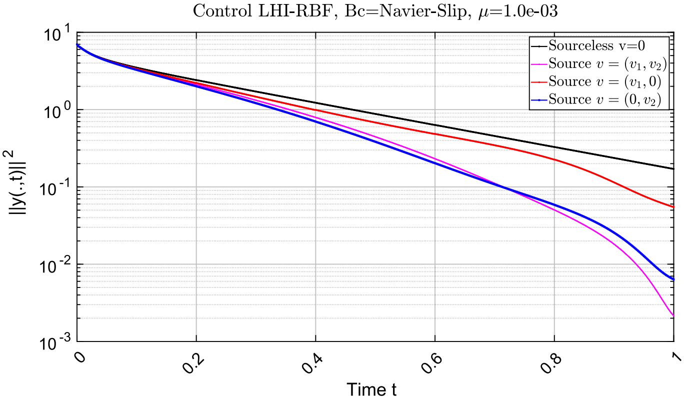


Fig. 5 L^2-norm square solution of the velocity field (as a function of time) for the approximate control problem with controls $\mathbf{v} = \mathbf{0}$ (black), $\mathbf{v} = (v_1, v_2)$ (pink), $\mathbf{v} = (v_1, 0)$ (red) and $\mathbf{v} = (0, v_2)$ with Navier-slip boundary condition. LHI-RBF hybrid kernel, with parameters $\gamma_1 = 1e{-}03$,$\gamma_2 = 1e{-}08$,$c_1 = 1.0$,$c_2 = 5.e{-}10$

Navier-slip conditions). Also, it might be important to review the stability for the Stokes system with these Navier-slip conditions (Ding et al. 2018).

5.2 Finite element method, FEM, for the control problem

Taking as starting point the classical optimal control problem for the Stokes system (Glowinski et al. 2008), we can solve the optimality system given in (1) (5) and (6) in a similar way. In our case, the time−space discretization of the coupled system (1), (6), lies in a mixed finite element formulation in space using $\mathbb{P}_2$-type elements for the velocity and $\mathbb{P}_1$-type elements for the pressure, meanwhile finite differences are used for the time discretization (see Glowinski and Pironneau 1992; Girault and Raviart 2012; Allaire 2005 for a complete review). It has to be pointed that to solve the unsteady Stokes equation with Navier-slip boundary condition, we used a penalization method given in Dione and Urquiza (2015).

Table 11 Number of iterations for obtaining the convergence criteria of the CGM for FEM

B.C	$\mathbf{v} = (v_1, v_2)$	$\mathbf{v} = (v_1, 0)$	$\mathbf{v} = (0, v_2)$
Navier-slip	73	73	65
Dirichlet	56	54	49

Table 12 Evolution in time of the L^2-norm for the solution of the approximate control problem with Dirichlet boundary conditions and few scalar controls, (FEM) $\mu = 1.0\mathrm{e}{-03}$

Boundary condition = Dirichlet

t	$\mathbf{v} = \mathbf{0}$	$\mathbf{v} = (v_1, v_2)$	$\mathbf{v} = (v_1, 0)$	$\mathbf{v} = (0, v_2)$
0.0E+00	6.103E+00	6.103E+00	6.103E+00	6.103E+00
1.0E−01	1.572E+00	1.561E+00	1.560E+00	1.560E+00
2.0E−01	8.014E−01	7.814E−01	7.788E−01	7.799E−01
3.0E−01	4.351E−01	4.101E−01	4.070E−01	4.085E−01
4.0E−01	2.409E−01	2.135E−01	2.111E−01	2.121E−01
5.0E−01	1.342E−01	1.065E−01	1.059E−01	1.055E−01
6.0E−01	7.498E−02	4.864E−02	5.097E−02	4.798E−02
7.0E−01	4.192E−02	1.879E−02	2.463E−02	1.854E−02
8.0E−01	2.344E−02	5.397E−03	1.326E−02	5.418E−03
9.0E−01	1.311E−02	1.120E−03	6.087E−03	1.256E−03
1.0E+00	7.333E−03	7.110E−05	1.076E−03	1.371E−04

Boundary condition = Navier-Slip

t	$\mathbf{v} = \mathbf{0}$	$\mathbf{v} = (v_1, v_2)$	$\mathbf{v} = (v_1, 0)$	$\mathbf{v} = (0, v_2)$
0.0E+00	6.934E+00	6.934E+00	6.934E+00	6.934E+00
1.0E−01	2.496E+00	2.359E+00	2.403E+00	2.345E+00
2.0E−01	1.660E+00	1.409E+00	1.531E+00	1.388E+00
3.0E−01	1.173E+00	8.462E−01	1.048E+00	8.255E−01
4.0E−01	8.519E−01	4.890E−01	7.634E−01	4.731E−01
5.0E−01	6.277E−01	2.652E−01	5.884E−01	2.563E−01
6.0E−01	4.663E−01	1.335E−01	4.587E−01	1.319E−01
7.0E−01	3.480E−01	6.361E−02	3.254E−01	6.801E−02
8.0E−01	2.604E−01	3.009E−02	1.747E−01	3.719E−02
9.0E−01	1.952E−01	1.188E−02	5.537E−02	1.668E−02
1.0E+00	1.465E−01	1.060E−03	1.904E−02	2.582E−03

Table 11 shows the number of iterations to achieve the stopping criteria $\epsilon = 10^{-8}$ in the CGM implemented.

Table 12 and Figs. 6 and 7 display the evolution in time of the L^2-norm of the velocity vector field $\mathbf{y} = (y_1, y_2)$, which represents the solution to the approximate control problem (1), and where the control function v has different structure, namely, $\mathbf{v} = 0$, $\mathbf{v} = (v_1, v_2)$, $\mathbf{v} = (v_1, 0)$ and $\mathbf{v} = (0, v_2)$.

As we can see from Tables 10, 11 and 12, the RBF-LHI method or FEM are similar, nevertheless, RBF-LHI method has the advantage of being mesh-less and showing more accuracy using divergence free kernels. The number of iterations for the CGM necessary to

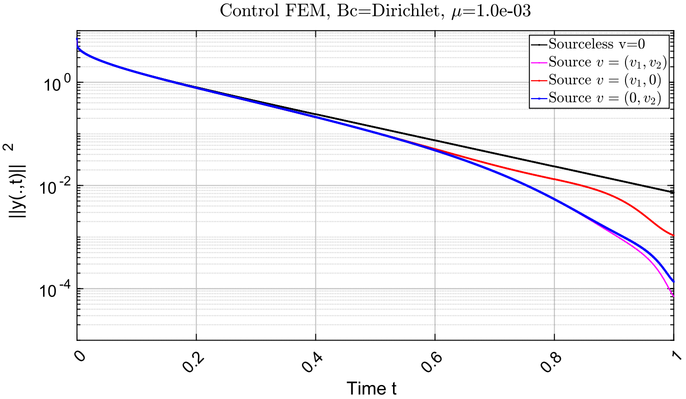


Fig. 6 Evolution in time of the L^2-norm square for the solution of the approximate control problem with Dirichlet boundary conditions and $v = 0$ (black), $v = (v_1, v_2)$ (pink), $v = (v_1, 0)$ (red) and $v = (0, v_2)$ (blue). (FEM)

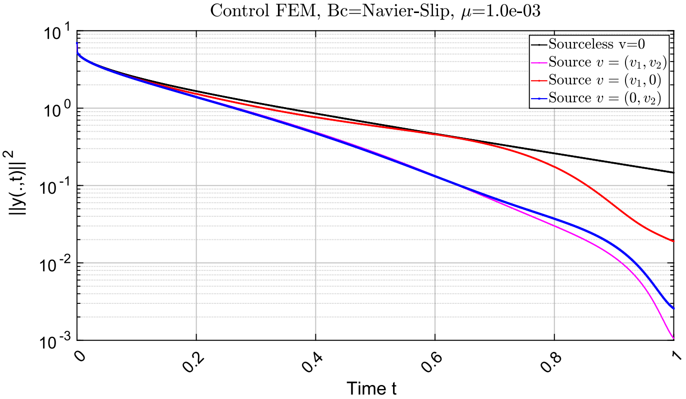


Fig. 7 Evolution in time of the L^2-norm square for the solution of the approximate control problem with Dirichlet boundary conditions and $v = 0$ (black), $v = (v_1, v_2)$ (pink), $v = (v_1, 0)$ (red) and $v = (0, v_2)$ (blue). (FEM)

converge is higher for the RBF-LHI method; however it should be noted that for RBF-LHI technique, we use $\mathbb{P}_1$-type elements to compute the integral expressions in the CGM, while for FEM we use $\mathbb{P}_2$-type elements.

6 Conclusions and final remarks

In this article, we have introduced radial basis function (RBF) methods to approximate the solution of Stokes equations and controllability problems for the Stokes system with few internal scalar controls. Dirichlet or Navier-slip boundary conditions are used. Two type of radial basis function solvers for the direct Stokes problems, one global and the other local were formulated. All the numerical experiments in his work are done in a star shape, thus a non-convex domain. Direct global solvers for the evolutionary Stokes problem are built using Hermite interpolation technique based on divergence free IMQ-RBFs. This allows to satisfy the incompressibility condition, at a discrete level, and to easily incorporate Navier-slip or Dirichlet boundary conditions.

In our article, and as far as to our knowledge, we imposed for the first time Navier-slip boundary conditions for Stokes equations. Stability analysis for this method shows that the real part of the eigenvalues are all negative if the shape parameter is properly selected. This implies that the method is stable for backward differences formulas (BDFs). Exponential convergence is numerically studied.

On the other hand, we find that the condition number, which is computed using extended precession, also grows in an exponential form; thus, the number of nodes that can be used is relatively small. To use larger number of nodes, we use a local Hermite interpolation technique, LHI. For the stationary Stokes problem, we numerically find that the convergence is exponential as the fill distance tends to zero.

We also investigated the evolutionary Stokes problem with LHI-IMQ-RBFs and we find that, by means of extensive numerical experimentation, see Sect. 4.5, it is not possible to find a suitable set of shape parameters such that the real component of the eigenvalues of the global ODE−LHI matrix are all negative. Thus, the evolutionary Stokes equations discretized by the IMQ-LHI method do not converge when integrated by BDF techniques. Moreover, for both the stationary and evolutionary Stokes problem, the condition number of the local matices grows exponentially as the fill distance decreases.

To deal with these two problems, namely, the high condition number of the local matrices and the existence of positive real eigenvalues of the global matrix, we generalized a recently formulated scalar hybrid kernels to a vectorial setting. Scalar hybrid kernels are a liner combination of Gaussians and odd Poly-harmonic splines. In our work, we introduce divergency free matrix hybrid radial basis kernels (Div-Free Hybrid). We find that the LHI Div-free Hybrid algorithm, for both Dirichlet and Navier-slip boundary conditions, reduces the condition number of the local matrices by several orders of magnitude and that the real components of the eigenvalues, when parameters are properly selected, can be all negative. The convergence of the solution gives excellent results which depends on the number of local nodes and the size of the parameters related to the hybrid kernel.

A final contribution of our work is that we solved, for first time, the approximate controllability problem for the Stokes equations using the conjugate gradient method and solving the direct and adjoint equations by LHI Div-free Hybrid RBF technique. This problem is also solved by the Finite element method. Comparative results are in excellent agreement.

For hybrid collocation of scalar stationary PDE problems, it has been reported that the condition numbers and the errors are stable, i.e., do not varies, in a wide range of values of the shape parameter (Zhang 2019; Pankaj et al. 2019) for time-dependent scalar PDEs. On the other hand, for the evolutionary vector Stokes problem teated in this work, we have not analyzed the sensibility or stability of the hybrid algorithms with respect to the variation of the parameters. Further research is in curse to study this point.

Acknowledgements The research was supported by Fondecyt Grant 3180100 (Cristhian Montoya). The authors acknowledge the CONACYT (624997), project Fordecyt 265667. This work also was supported by the National Autonomous University of México [Grant: PAPIIT, IN102116] and by Network of Mathematics and Development of CONACYT.

References

Allaire G (2005) Analyse numérique et optimisation: une introduction à la modélisation mathématique et à la simulation numérique. Editions Ecole Polytechnique

Breton L (2020) Global-divergence-free-rbf-matlab-codes. https://github.com/LDBreton/Global-Divergence-Free-RBF

Cebeci T (2012) Analysis of turbulent boundary layers, vol 15. Elsevier, Amsterdam

Chinchapatnam P, Djidjeli K, Nair P (2006) Unsymmetric and symmetric meshless schemes for the unsteady convection–diffusion equation. Comput Methods Appl Mech Eng 195(19–22):2432–2453

Ding S, Li Q, Xin Z (2018) Stability analysis for the incompressible Navier–Stokes equations with Navier boundary conditions. J Math Fluid Mech 20(2):603–629

Dione I, Urquiza JM (2015) Penalty: finite element approximation of stokes equations with slip boundary conditions. Numer Math 129(3):587–610

Fasshauer GE (2007) Meshfree approximation methods with Matlab: (with CD-ROM), vol 6. World Scientific Publishing Co Inc, Singapore

Fernández-Cara E, Limaco J, de Menezes S (2015) Theoretical and numerical local null controllability of a Ladyzhenskaya–Smagorinsky model of turbulence. J Math Fluid Mech 17(4):669–698

Fernández-Cara E, Münch A, Souza DA (2017) On the numerical controllability of the two-dimensional heat, stokes and Navier–Stokes equations. J Sci Comput 70(2):819–858

Flyer N, Fornberg B, Bayona V, Barnett GA (2016) On the role of polynomials in RBF-FD approximations: I. Interpolation and accuracy. J Comput Phys 321:21–38. https://doi.org/10.1016/j.jcp.2016.05.026

Fornberg B, Flyer N (2015a) A primer on radial basis functions with applications to the geosciences. Society for Industrial and Applied Mathematics, Philadelphia

Fornberg B, Flyer N (2015) Solving PDEs with radial basis functions. Acta Numer 24:215–258. https://doi.org/10.1017/S0962492914000130

Fursikov AV, Imanuvilov OY (1996) Controllability of evolution equations, vol 34. Seoul National University

Fuselier EJ, Shankar V, Wright GB (2016) A high-order radial basis function (RBF) Leray projection method for the solution of the incompressible unsteady stokes equations. Comput Fluids 128:41–52

García GC, Montoya C, Osses A (2017) A source reconstruction algorithm for the stokes system from incomplete velocity measurements. Inverse Probl 33(10):105003

Girault V, Raviart PA (2012) Finite element methods for Navier–Stokes equations: theory and algorithms, vol 5. Springer Science & Business Media, Berlin

Glowinski R, Pironneau O (1992) Finite element methods for Navier–Stokes equations. Annu Rev Fluid Mech 24(1):167–204

Glowinski R, Lions JL, He J (2008) Exact and approximate controllability for distributed parameter systems: a numerical approach (encyclopedia of mathematics and its applications). Cambridge University Press, Cambridge

González-Casanova P, Gout C, Zavaleta J (2019) Radial basis function methods for optimal control of the convection diffusion equation: a numerical study. Eng Anal Bound Elem 108:201–209. https://doi.org/10.1016/j.enganabound.2019.08.008

Guerrero S, Montoya C (2018) Local null controllability of the n-dimensional Navier–Stokes system with nonlinear Navier-slip boundary conditions and n-1 scalar controls. J Math Pures Appl 113:37–69

He Q, Wang XP (2009) Numerical study of the effect of Navier slip on the driven cavity flow. ZAMM J Appl Math Mech 89(10):857–868

Keim C, Wendland H (2016) A high-order, analytically divergence-free approximation method for the time-dependent stokes problem. SIAM J Numer Anal 54:1288–1312

Kishida M, Braatz R.D (2009) Rbf-based 2d optimal spatial control of the 3d reaction-convection–diffusion equation. In: 2009 European control conference (ECC), pp 1949–1954 . https://doi.org/10.23919/ECC.2009.7074689

Lambert JD (1991) Numerical methods for ordinary differential systems: the initial value problem. Wiley, New York

Mirinejad H, Inanc T (2017) An RBF collocation method for solving optimal control problems. Robot Auton Syst 87:219–225. https://doi.org/10.1016/j.robot.2016.10.015

Mishra PK, Nath SK, Kosec G, Sen MK (2017) An improved radial basis-pseudospectral method with hybrid Gaussian-cubic kernels. Eng Anal Bound Elem 80:162–171. https://doi.org/10.1016/j.enganabound.2017.03.009

Mishra P, Nath S, Sen M, Fasshauer G (2018) Hybrid Gaussian-cubic radial basis functions for scattered data interpolation. Comput Geosci 22:1203–1218. https://doi.org/10.1007/s10596-018-9747-3

Navier C (1823) Mémoire sur les lois du mouvement des fluides. Mem Acad Sci Inst Fr 6(1823):389–416

Pankaj K, Mishra Gregory E, Fasshauer MKSLL (2019) A stabilized radial basis-finite difference (RBF-FD) method with hybrid kernels. Comput Math Appl 77:2354–2368. https://doi.org/10.1016/j.camwa.2018.12.027

Pearson (2013) A radial basis function method for solving PDE-constrained optimization problems. Numer Algorithms 4:481–506

Rad JA, Kazem S, Parand K (2014) Radial basis functions approach on optimal control problems: a numerical investigation. J Vib Control 20:1394–1416

Stevens D, Power H, Lees M, Morvan H (2011) A local Hermitian RBF meshless numerical method for the solution of multi-zone problems. Numer Methods Partial Differ Equ 27(5):1201–1230

Wendland H (2004) Scattered data approximation, vol 17. Cambridge University Press, Cambridge

Wendland H (2009) Divergence-free kernel methods for approximating the Stokes problem. SIAM J Numer Anal 47(4):3158–3179

Zhang Y (2019) An accurate and stable RBF method for solving partial differential equations. Appl Math Lett 97:93–98. https://doi.org/10.1016/j.aml.2019.05.021

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Affiliations

Louis Breton[1] · Pedro González-Casanova[2] · Cristhian Montoya[3]

✉ Pedro González-Casanova
casanova@matem.unam.mx

Louis Breton
louis.breton@ciencias.unam.mx

Cristhian Montoya
cdmontoy@mat.uc.cl

[1] Facultad de Ciencias,Universidad Nacional Autónoma de México, Mexico, Mexico

[2] Instituto de Matemáticas, Universidad Nacional Autónoma de México, Circuito Exterior C.U, C.P. 04510 Mexico, Mexico

[3] Instituto de Ingeniería Matemática y Computacional, Pontificia Universidad Católica de Chile, Santiago, Chile