

ARTICLE TYPE

An inverse problem formulation of the immersed boundary method

Jianfeng Yan | Jason E. Hicken*

¹Department of Mechanical, Aerospace, and Nuclear Engineering, Rensselaer Polytechnic Institute, New York, United States of America

Correspondence

*Jason E. Hicken, Department of Mechanical, Aerospace, and Nuclear Engineering, Rensselaer Polytechnic Institute, Troy, New York, 12180, United States. Email: hickej2@rpi.edu

Abstract

We formulate the immersed-boundary method (IBM) as an inverse problem. A control variable is introduced on the boundary of a larger domain that encompasses the target domain. The optimal control is the one that minimizes the mismatch between the state and the desired boundary value along the immersed target-domain boundary. We begin by investigating a naïve problem formulation that we show is ill-posed: in the case of the Laplace equation, we prove that the solution is unique but it fails to depend continuously on the data; for the linear advection equation, even solution uniqueness fails to hold. These issues are addressed by two complimentary strategies. The first strategy is to ensure that the enclosing domain tends to the true domain as the mesh is refined. The second strategy is to include a specialized parameter-free regularization that is based on penalizing the difference between the control and the state on the boundary. The proposed inverse IBM is applied to the diffusion, advection, and advection-diffusion equations using a high-order discontinuous Galerkin discretization. The numerical experiments demonstrate that the regularized scheme achieves optimal rates of convergence and that the reduced Hessian of the optimization problem has a bounded condition number as the mesh is refined.

KEYWORDS:

immersed-boundary method; inverse problem; PDE-constrained optimization

1 | INTRODUCTION

Mesh generation remains a significant bottleneck for many aerospace applications of computational fluid dynamics (CFD) [1] and the numerical solution of partial differential equations (PDEs) more generally. This bottleneck is particularly acute during the generation and adaptation of curved-element, anisotropic meshes around complex geometries, which has limited the adoption of high-order methods in industry.

One way to address the meshing bottleneck is to develop a high-order discretization that does not require a conforming mesh. The immersed boundary method (IBM) offers a potential framework for constructing such a discretization and forms the basis for the method presented herein.

Before proceeding, we should distinguish the IBM from immersed-interface, or cut-cell, methods. IBMs [2] impose the boundary conditions indirectly through a body force or modified boundary flux. Immersed-interface methods, on the other hand, modify the cells, elements, or stencil near the boundary such that the boundary condition can be applied directly [3, 4, 5, 6, 7, 8, 9, 10, 11, 12]. The present work is more closely related to classical IBMs that introduce a body force, or penalty, that imposes

the boundary conditions indirectly. The IBM framework encompasses a wide range of approaches, so a comprehensive review is beyond the scope of this work. For reviews of the IBM in the context of finite-difference/volume methods see [13] and Sections 3 and 4.1 of [14]. A review of penalty-based IBMs used in finite-element methods can be found in Section 2.2 of [9].

IBMs are popular in some CFD applications, but they have not found widespread use for the steady, advection-dominated problems common in the aerospace industry. We believe this is primarily due to their limited accuracy; most IBMs are first-order accurate in practice. That said, some high-order IBMs have been proposed and we mention a few here.

Mayo [15] solved the Poisson's and biharmonic equations on irregular domains by constructing a discontinuous extension of the solution onto a regular (e.g. square) domain. The discontinuities in the solution and its derivatives could then be determined by solving an integral equation, and these jumps could subsequently be introduced into the Poisson's or biharmonic equations to solve the PDEs on the regular domain using fast solvers. Mayo later extended this technique to 4th order in [16].

Marques *et al.* [17] and Marques [18] generalized the ghost-fluid method [19, 20, 21, 22] to high-order by defining a correction function that smoothly extends the solution on either side of an interface or boundary. Once the correction function is determined, it is included on the right-hand side of the discretized equations. The correction is independent of the solution for linear problems with Dirichlet boundary conditions, so conditioning of the left-hand side is not affected in this case; however, more generally, the correction function depends on the solution for nonlinear problems and problems with more general boundary conditions.

Finally, IBMs that impose boundary conditions using Nitsche's method have shown considerable promise [23, 24]. The penalty parameter in Nitsche-based methods must be chosen carefully to maintain coercivity and avoid ill-conditioning, and this choice can be challenging since it depends on how the exact boundary intersects with the computational mesh. However, this challenge can be alleviated with stabilization introduced by penalizing the jumps in the normal derivatives [25]. Although high-order Nitsche-based IBMs have been applied to several different types of physics, their application to advection-dominated problems is lacking, to the best of our knowledge.

In this paper we present the preliminary investigation of a novel, high-order immersed boundary method. Our approach is similar in spirit to the fictitious domain method of Glowinski and He [26] and the optimization-based interface treatment in [27]. The key idea is to formulate the IBM as an inverse problem in which unknown boundary fluxes, defined on an approximate easy-to-generate boundary, are used to satisfy the desired boundary conditions on the true boundary. One strength of this high-order inverse-problem formulation of the IBM is that it is straightforward both conceptually and in its implementation.

The remainder of the paper is organized as follows. Section 2 describes the basic inverse IBM formulation for both the Laplace and linear advection equations. This section also discusses the ill-posedness of the basic formulation. Section 3 presents an investigation of a model problem, which we use to better understand the nature of ill-conditioning in the discrete setting. Regularization of the inverse IBM is discussed in Section 4, including a parameter-free regularization based on penalizing the control against the state. The proposed method is demonstrated in Section 5 using on a discontinuous Galerkin discretization. A summary and discussion are provided in Section 6.

2 | INVERSE IBM PROBLEM FORMULATION AND ILL-POSEDNESS

This section presents the basic formulation of the proposed IBM in the context of the Laplace and linear advection equations. In both cases, we show that the basic formulation is ill-posed and requires regularization.

2.1 | Application to the Laplace equation

Consider the Laplace equation on the open, bounded domain $\Omega \in \mathbb{R}^N$, with Dirichlet boundary conditions applied on the boundary $\Gamma \equiv \partial\Omega$:

$$\nabla^2 u = 0, \quad \forall x \in \Omega, \quad (1a)$$

$$u = u_\Gamma, \quad \forall x \in \Gamma. \quad (1b)$$

Suppose Ω is a geometrically complex domain for which we do not want to generate a conforming mesh. Consequently, we introduce a geometrically simpler, bounded domain $\tilde{\Omega} \supseteq \Omega$, with boundary $\tilde{\Gamma} \equiv \partial\tilde{\Omega}$. Figure 1 depicts an example of the domain Ω , an encompassing domain $\tilde{\Omega}$, and their respective boundaries.

Having described this generic geometric setup, we now turn to the key question: how do we select the boundary conditions on $\tilde{\Gamma}$ such that the conditions (1b) are satisfied, or at least approximately satisfied, on Γ ? Our tentative answer to this question

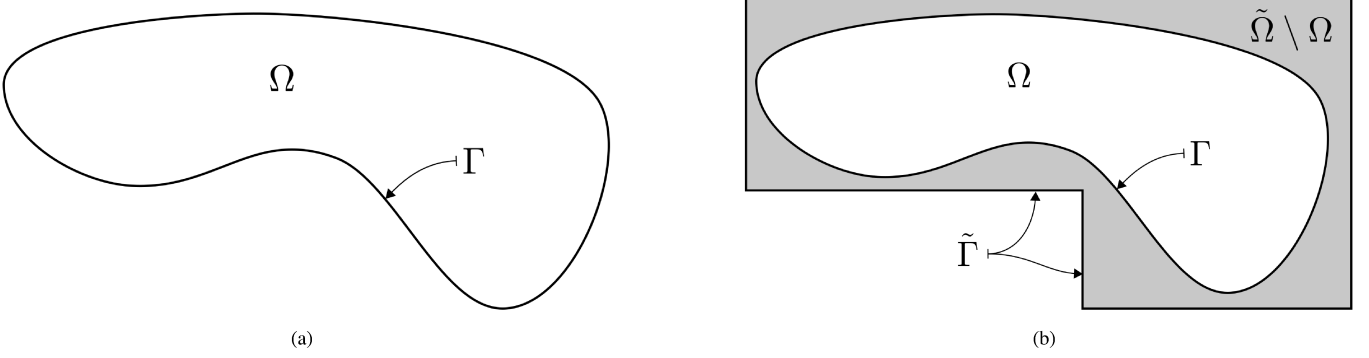


FIGURE 1 Example domain and boundary for the Laplace equation (left) and the corresponding inverse IBM domain (right): $\tilde{\Omega}$ consists of the union of the white and gray regions.

is to define the following PDE-constrained optimization problem, which is given in variational form. Please refer to Remark 1 regarding the role of the variable c .

$$\begin{aligned}
 \inf_{c,u} \quad & J(u, c) \equiv \frac{1}{2} \int_{\Gamma} (u - u_r)^2 \, d\Gamma, \\
 \text{s.t.} \quad & \mathfrak{b}(u, v) \equiv \int_{\tilde{\Omega}} \nabla u \cdot \nabla v \, d\Omega = 0, \quad \forall v \in H_0^1(\tilde{\Omega}), \\
 & c \in H^{\frac{1}{2}}(\tilde{\Gamma}), \quad \text{and} \quad u \in \{v \in H^1(\tilde{\Omega}) \mid v|_{\tilde{\Gamma}} = c\}.
 \end{aligned} \tag{2}$$

Remark 1. Notice that the (variational) Laplace equation is posed on the larger domain $\tilde{\Omega}$. Furthermore, the boundary condition imposed on $\tilde{\Gamma}$ is defined by c , rather than u_r . The variable c is the control variable used to minimize the cost functional $J(u, c)$ in order to (indirectly) satisfy the boundary condition on Γ .

Remark 2. One could pose the optimization problem without c ; that is, we could seek $u \in H^1(\tilde{\Omega})$ that minimizes $J(u, c)$ subject to the Laplace equation, $\mathfrak{b}(u, v) = 0$, $\forall v \in H^1(\tilde{\Omega})$, where the test function space is the same as the trial space. We prefer to include the boundary control c because, for a fixed c , the PDE constraint is well-posed. In practice, this means the stiffness matrix is nonsingular, which allows us to leverage existing solution algorithms for PDE-constrained optimization (see Section 5.2). Furthermore, without the boundary control, some form of stabilization is necessary to cope with the rank deficient PDE Jacobian.

Remark 3. Formulation (2) defines a boundary-control problem, and it is the prototype method that we will explore further in the rest of the paper; however, in some cases it is possible to define an inverse IBM using distributed control instead of boundary control. For example, if $\tilde{\Omega}$ is defined such that the immersed geometry is bounded and coincides with $\tilde{\Omega} \setminus \Omega$, then a distributed control can be defined over the region interior to the immersed geometry. A distributed-control approach is the basis of the fictitious domain method in [26].

2.1.1 | Ill-posedness in the context of the Laplace equation

To establish the well-posedness of (2) in the sense of Hadamard, we would need to prove that the problem has a unique solution that depends continuously on the data (e.g., u_r); unfortunately, as we show below, there exists inverse IBM problems for which the solution does not depend continuously on the data. The practical implication of this ill-posedness is that the discretized problem may not be solvable either.

Consider an inverse IBM problem whose target domain is the unit disc, $\Omega = \{(r, \theta) \mid r < 1, \theta \in [0, 2\pi)\}$, and let the boundary condition on Γ be given by the following sinusoid:

$$u_{r,n}(\theta) = \frac{1}{n} \sin(n\theta), \quad \forall \theta \in [0, 2\pi),$$

where n is a positive integer. The function $u_{r,n}(\theta)$ represents the data in this context, and we will now define an inverse IBM problem whose solution does not depend continuously on this function. Specifically, suppose the larger, enclosing domain is

also a disc, but one with radius $R > 1$; thus, $\tilde{\Omega} = \{(r, \theta) \mid r < R, \theta \in [0, 2\pi)\}$. It is easy to verify that the solution to the Laplace equation on $\tilde{\Omega}$ that satisfies $u(1, \theta) = u_{r,n}(\theta)$ is given by

$$u_n(r, \theta) = \frac{r^n}{n} \sin(n\theta), \quad \forall (r, \theta) \in \tilde{\Omega}.$$

Consequently, the corresponding (unique) boundary control that solves the inverse IBM problem (2) is the trace of $u_n(r, \theta)$ on $\tilde{\Gamma}$:

$$c_n(\theta) = \frac{R^n}{n} \sin(n\theta), \quad \forall \theta \in [0, 2\pi).$$

Now, consider the solution of the inverse IBM as the mapping $T : H^{\frac{1}{2}}(\Gamma) \rightarrow H^{\frac{1}{2}}(\tilde{\Gamma})$ that transforms the boundary data u_r into c , that is $T(u_r) = c$. Then, for the present example we find

$$\|T(u_r)\|_{H^{\frac{1}{2}}(\tilde{\Gamma})}^2 = \|c_n\|_{H^{\frac{1}{2}}(\tilde{\Gamma})}^2 \geq C_{\tilde{\Gamma}} \|c_n\|_{L^2(\tilde{\Gamma})}^2 = \frac{C_{\tilde{\Gamma}} R^n}{n} \left[\pi - \frac{\sin(4\pi n)}{4n} \right],$$

where $C_{\tilde{\Gamma}}$ is a constant. Since the right-hand side of the above inequality is unbounded as $n \rightarrow \infty$, while u_r remains bounded, it follows that T itself is not bounded, in general. We conclude that the solution does not depend continuously on the data and the inverse IBM, *at least in the form (2)*, is not well-posed for the Laplace equation.

2.2 | Application to the linear advection equation

We now turn our attention to the linear advection equation, which is an important model problem for inviscid fluid flows and hyperbolic PDEs more generally. As with the Laplace equation, we will see that the basic inverse IBM formulation is ill-posed when applied to the advection equation.

The linear advection equation on an open, bounded domain $\Omega \subset \mathbb{R}^N$, with smooth boundary Γ , is given by

$$\nabla \cdot (\lambda u) = 0, \quad \forall x \in \Omega, \quad (3a)$$

$$u = u_r, \quad \forall x \in \Gamma^-, \quad (3b)$$

where $\lambda \in [H^1(\Omega)]^N$ is the vector-valued velocity field. Boundary conditions are imposed on the inflow boundary $\Gamma^- \equiv \{x \in \Gamma \mid \lambda_n < 0\}$, where $\lambda_n \equiv \lambda \cdot \hat{n}$ is the velocity component normal to the boundary and $\hat{n}$ is the unit, outward-pointing normal vector on Γ .

Remark 4. We will assume that (3) is well-posed for the given data and geometry. Furthermore, in order to apply the inverse IBM formulation, we will assume that the velocity field can be extended to the larger computational domain $\tilde{\Omega}$. That is, we assume that there exists $\tilde{\lambda} \in [H^1(\tilde{\Omega})]^N$ such that $\tilde{\lambda}(x) = \lambda(x)$ for all $x \in \tilde{\Omega}$.

Next, we formulate an inverse IBM statement, analogous to (2), for the linear advection equation. Specifically, we replace the variational Laplace equation with the variational advection equation in (2) and obtain the following problem:

$$\begin{aligned} \inf_{c, u} \quad & J(u, c) \equiv \frac{1}{2} \int_{\Gamma^-} (u - u_r)^2 d\Gamma, \\ \text{s.t.} \quad & \mathfrak{b}(u, v) \equiv \int_{\tilde{\Omega}} v \nabla \cdot (\tilde{\lambda} u) d\Omega = 0, \quad \forall v \in L^2(\tilde{\Omega}), \\ & c \in H^{\frac{1}{2}}(\tilde{\Gamma}^-), \quad \text{and} \quad u \in \{v \in L^2(\tilde{\Omega}) \mid \nabla \cdot (\tilde{\lambda} v) \in L^2(\tilde{\Omega}), v|_{\tilde{\Gamma}^-} = c|_{\tilde{\Gamma}^-}\}, \end{aligned} \quad (4)$$

where $\tilde{\Gamma}^- \equiv \{x \in \tilde{\Gamma} \mid \tilde{\lambda}_n < 0\}$ and $\tilde{\Gamma}^+ \equiv \tilde{\Gamma} \setminus \tilde{\Gamma}^-$. Figure 2 provides an example of the domains and boundaries that appear in problem (4). We will also refer to this figure when explaining the ill-posedness that affects the inverse IBM in the context of the advection equation.

2.2.1 | Ill-posedness in the context of the advection equation

Consider the point labeled $x_1 \in \tilde{\Gamma}^-$ in Figure 2. A compact control perturbation centered at x_1 determines the value of u along the characteristics downstream of x_1 via the boundary condition on $\tilde{\Gamma}^-$. However, if the perturbation is sufficiently local, then the corresponding characteristics will not pass through Γ and the perturbation will have no impact on the objective. Consequently, such control perturbations are non-unique in the context of (4), i.e. the inverse IBM problem is ill-posed for linear advection.

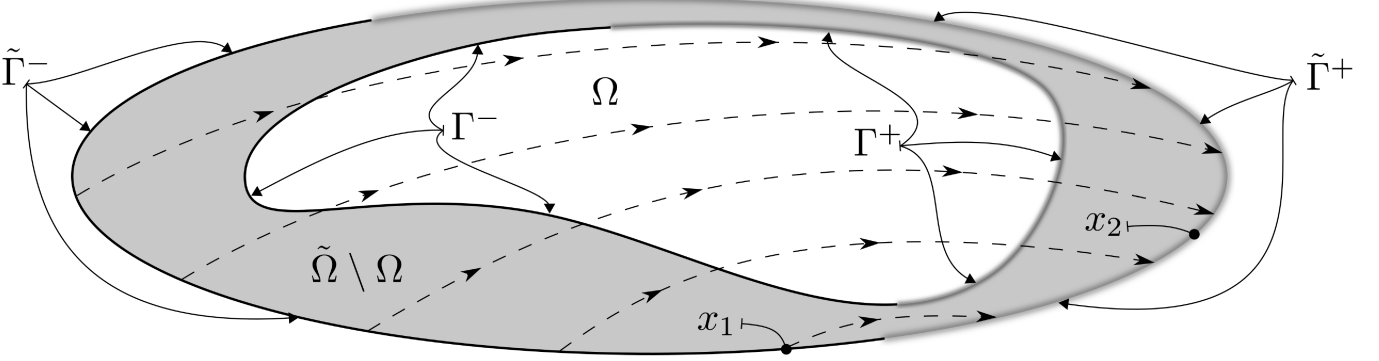


FIGURE 2 Example domains and boundaries for the inverse IBM problem applied to linear advection.

To make matters worse, nonlinear hyperbolic PDEs present an additional difficulty for the proposed inverse IBM formulation. Namely, the flow field $\tilde{\lambda}$ itself often depends on the solution, so the definition of $\tilde{\Gamma}^-$ and $\tilde{\Gamma}^+$ is not necessarily known a priori. For the discretized version of (4), this uncertainty in the set $\tilde{\Gamma}^-$ means that the dimension of the boundary control $c \in H^{\frac{1}{2}}(\tilde{\Gamma}^-)$ would need to change dynamically during the solution process.

Remark 5. This practical difficulty is exhibited by the Euler equations of gas dynamics. For example, if Γ is a no-penetration wall in the Euler equations, it will have a known number of incoming and outgoing characteristics (one each); however, the surface $\tilde{\Gamma}$ may require additional incoming/outgoing characteristics in order to satisfy the no-penetration condition.

Our solution to this practical issue is to define the control on the entire boundary, as was done for the Laplace equation. On the one hand, this introduces another source of nonuniqueness: the value of the control on $\tilde{\Gamma}^+$ — for example, $x_2 \in \tilde{\Gamma}^+$ in Figure 2 — has no influence on the state, and, therefore, no influence either satisfying the constraint or minimizing the objective in (4). On the other hand, this form of nonuniqueness is easily eliminated with the regularization we propose in Section 4.

3 | AN INVESTIGATION OF CONDITIONING OF THE DISCRETE PROBLEM

It is clear that the basic inverse IBM formulation must be modified to be useful in practice; however, before we discuss potential regularizations in Section 4, we want to better understand the sources of ill-conditioning in the finite-dimensional case. To this end, we will model the discretization of the Laplace inverse IBM on a particular domain, and we use this model to investigate the conditioning of the finite-dimensional version of (2).

3.1 | Description of the model problem

We consider a model problem similar to the one described in Section 2.1.1. The computational domain is the unit disk, $\tilde{\Omega} = \{(r, \theta) \mid r < 1, \theta \in [0, 2\pi)\}$, and the immersed boundary is an ellipse with semi-major and semi-minor lengths a and b , respectively. The model geometry is illustrated in Figure 3. We have chosen a circle for $\tilde{\Omega}$ because it allows us to take advantage of Green's function theory.

Our goal is to construct a model for the reduced Hessian of the discretized problem. To this end, we will first derive the reduced Hessian for the continuous problem, and then discretize the control.

Consider the reduced-space formulation of (2), where the state u is considered a function of the control c and the objective is

$$J(c) = J(u(c), c) = \frac{1}{2} \int_{\Gamma} u^2(c) d\Gamma - \int_{\Gamma} u(c) u_r d\Gamma + J_0,$$

where $J_0 = \frac{1}{2} \int_{\Gamma} u_r^2 d\Gamma$.

In order to differentiate $J(c)$ to obtain the Hessian, we will need the derivative of the state with respect to the control. For the Laplace equation, one can show that the state is a linear and continuous function of the control. Consequently, the derivative of u with respect to c in the direction $d \in H^{\frac{1}{2}}(\tilde{\Gamma})$ is simply $u(d)$; that is, the directional derivative is simply the state obtained by

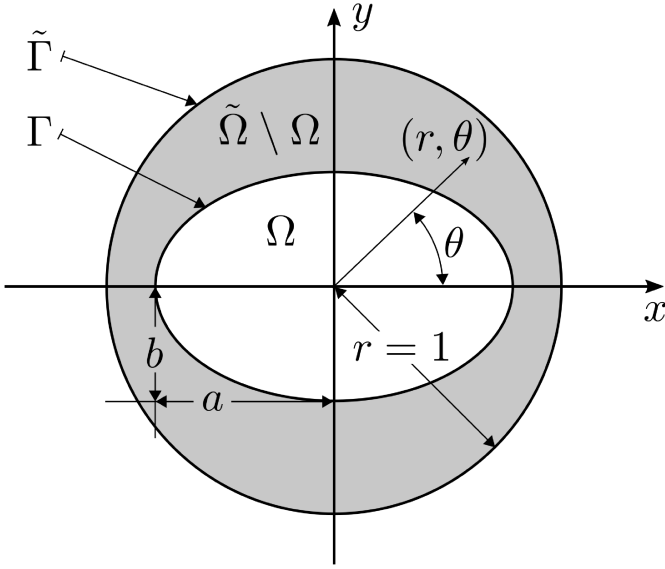


FIGURE 3 The geometry for the model, finite-dimensional Hessian.

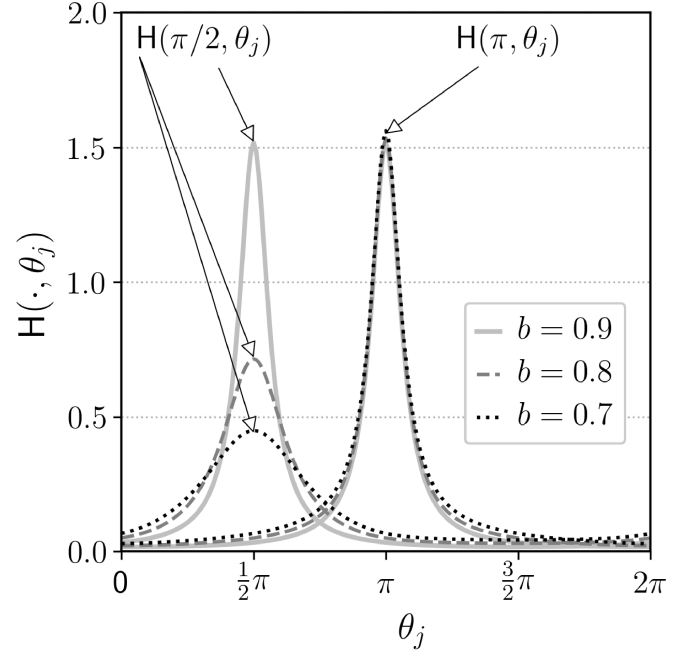


FIGURE 4 The model Hessian for the domains shown in Figure 3. Three elliptical shapes and two values of θ_i are shown. The semi-major axis is $a = 0.9$ in all cases.

substituting d for c . Furthermore, because u is a linear function of c , the second derivative of the state vanishes: $u''(d, e) = 0$, where $d, e \in H^{\frac{1}{2}}(\tilde{\Gamma})$ are arbitrary. Using these facts and differentiating J twice, we find that the Hessian of the continuous-problem objective is

$$J''(d, e) = \int_{\Gamma} u(d)u(e) d\Gamma, \quad \forall d, e \in H^{\frac{1}{2}}(\tilde{\Gamma}).$$

Next, we parameterize c to obtain a finite-dimensional control; the exact form of the parameterization is described later. Let $c_i \in \mathbb{R}^K$ denote the control that is zero everywhere except in the i th entry where it is one, i.e. $(c_i)_k = \delta_{ik}$. Then, taking $d = c_i$ and $e = c_j$ in $J''(d, e)$, we obtain the entries in the finite-dimensional Hessian:

$$H_{ij} \equiv \frac{\partial^2 J}{\partial c_i \partial c_j} = J''(c_i, c_j) = \int_{\Gamma} u(c_i)u(c_j) d\Gamma, \quad \forall i, j = 1, 2, \dots, K. \quad (5)$$

To find an explicit expression for H_{ij} , we need an explicit expression for $u(c_i)$. Using the appropriate Green's function for the disk [28, §7.1.2], we obtain the following expression for $u(c(\theta))$, where $c(\theta)$ is the control value on $\tilde{\Gamma}$:

$$u(c) = u(r, \theta) = \frac{1}{2\pi} \int_0^{2\pi} \frac{(1-r^2)c(\tilde{\theta})}{1+r^2-2r\cos(\tilde{\theta}-\theta)} d\tilde{\theta}. \quad (6)$$

The state given by (6) is for arbitrary $c(\theta)$. To simplify $u(c_i)$ for the Hessian expression, we use Dirac-delta control variations c_i that are parameterized based on uniformly-spaced points on the unit circle:

$$c_i = c(\theta_i) = \delta(\theta - \theta_i), \quad \text{where } \theta_i = 2\pi i/K, \quad i = 1, 2, \dots, K.$$

Substituting the distributions for c_i and c_j into (6), and using the resulting $u(c_i)$ and $u(c_j)$ in (5), we obtain the following expression for H_{ij} :

$$H_{ij} = H(\theta_i, \theta_j) = \frac{1}{(2\pi)^2} \int_{\Gamma} \frac{(1-r^2)}{[1+r^2-2r\cos(\theta_i-\theta)]} \frac{(1-r^2)}{[1+r^2-2r\cos(\theta_j-\theta)]} d\Gamma. \quad (7)$$

Note that, while both r and θ appear in the integrand above, the point (r, θ) is constrained to lie on the ellipse Γ .

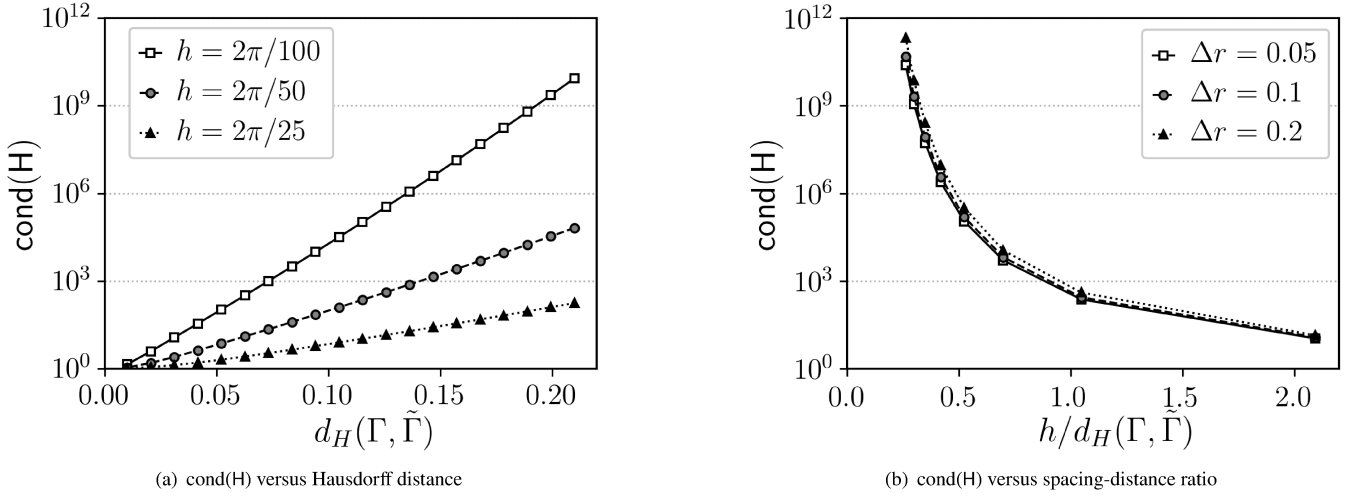


FIGURE 5 Trends in the reduced Hessian condition number for a circular Γ (i.e. $a = b$).

Before we use (7) to explore the condition number of the discrete Hessian, it is worthwhile to plot $H_{ij} = H(\theta_i, \theta_j)$ versus θ_j for fixed θ_i , because these plots will help explain the trends in the condition number. Two such θ_i rows are shown in Figure 4: one corresponding to $\theta_i = \pi/2$ and one corresponding to $\theta_i = \pi$. Each H_{ij} entry is evaluated for three different immersed ellipse boundaries with a fixed semi-major axis of $a = 0.9$ and progressively smaller semi-minor axes b . We have approximated the integral in (7) using the trapezoid rule with $n = 300$ intervals.

The trend is clear from the figure. There is little impact on the Hessian row near the fixed, semi-major axis at $\theta_i = \pi$. In contrast, decreasing the semi-minor axis stretches $H(\pi/2, \theta)$ horizontally and compresses it vertically. The spreading of this row of the Hessian hints at conditioning issues on the horizon, since relatively “flat” rows are difficult to distinguish from one another, i.e. the rows are nearly linearly dependent.

3.2 | Circular immersed boundary

Consider a circular immersed boundary Γ , with $a = b = r < 1$. Figure 5(a) plots the condition number of the reduced Hessian H versus the Hausdorff distance between Γ and $\tilde{\Gamma}$ for three different control-mesh spacings $h = \Delta\theta$. In the present case, where both Γ and $\tilde{\Gamma}$ are circles, the Hausdorff distance is given by $d_H(\Gamma, \tilde{\Gamma}) = 1 - r$.

For fixed h , the condition number grows exponentially with the distance between Γ and $\tilde{\Gamma}$. This growth is due to the diffusing of the analytical Hessian as the immersed boundary gets further from the domain boundary, as described earlier. Note that, for a fixed $d_H(\Gamma, \tilde{\Gamma})$, we also observe a strong dependence on mesh spacing.

Figure 5(a) suggests that it is possible to keep the condition number of H fixed as we refine the mesh, provided $d_H(\Gamma, \tilde{\Gamma})$ is also reduced. This relationship is clearly illustrated in Figure 5(b), which plots the condition number of H versus the ratio $h/d_H(\Gamma, \tilde{\Gamma})$. This figure is important because it shows that different Hausdorff distances essentially collapse onto the same curve, and it is the nondimensional ratio $h/d_H(\Gamma, \tilde{\Gamma})$ that determines the condition number. Our hypothesis is that, as long as $\lim_{h \rightarrow 0} h/d_H(\Gamma, \tilde{\Gamma})$ is bounded, the condition number of the reduced Hessian should also remain bounded. In particular, this means that the computational domain $\tilde{\Omega}$ must converge to Ω , in some sense, as the mesh is refined.

3.3 | Elliptical immersed boundary

For our next investigation of the model problem, we consider the influence of differing semi-major and semi-minor axis lengths. Figure 6(a) plots the reduced Hessian condition number versus $1 - a$ for three different fixed values of b . In all cases the mesh spacing is fixed at $h = 2\pi/50$. In general, the condition number remains stable as $1 - a$ increases until $a < b$, i.e. the semi-minor and semi-major axes switch, at which point the exponential growth is similar to that observed for circular Γ . This behavior

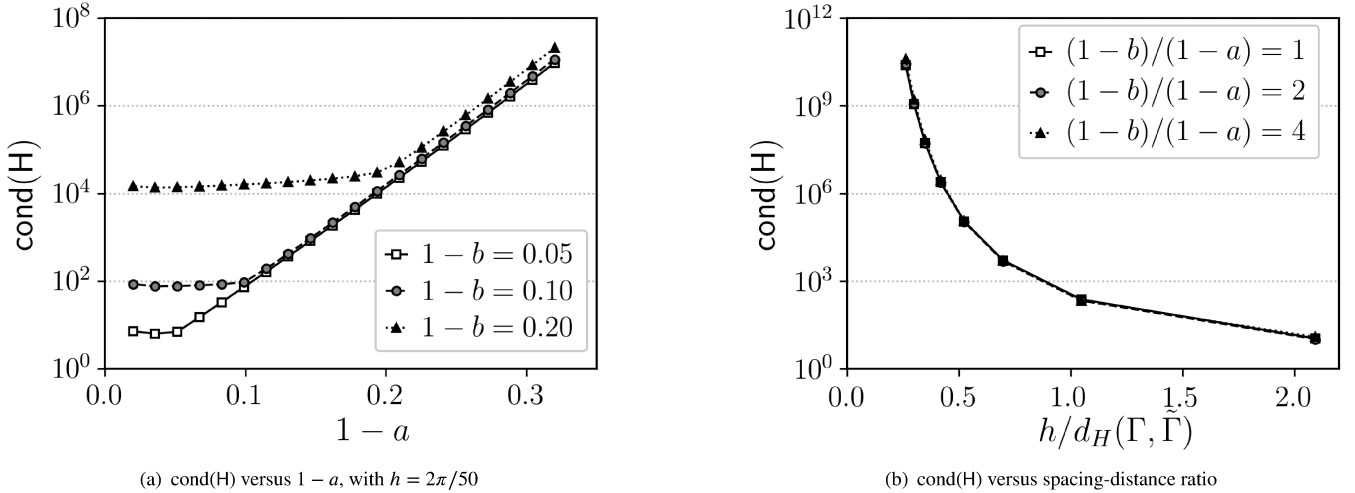


FIGURE 6 Trends in the reduced Hessian condition number for elliptical Γ .

is consistent with the Hausdorff distance determining the condition number for fixed h : for $a > b$ the Hausdorff distance is $d_H(\Gamma, \tilde{\Gamma}) = 1 - b$, which is constant for fixed b .

Figure 6(b) plots the condition number of H versus the nondimensional mesh spacing $h/d_H(\Gamma, \tilde{\Gamma})$. We have included curves corresponding to several different major-minor offset ratios, $(1 - b)/(1 - a)$. We observe that $h/d_H(\Gamma, \tilde{\Gamma})$ remains the key parameter that determines the condition number.

3.4 | Discussion

The results of the above studies suggest that keeping the ratio $h/d_H(\Gamma, \tilde{\Gamma})$ bounded is critical to the conditioning of the discretized inverse IBM problem. This is not surprising, since bounding this ratio implies that $\lim_{h \rightarrow 0} \tilde{\Gamma} = \Gamma$, so the inverse IBM problem becomes essentially equivalent to the direct problem with $c = u_r$.

However, there are good reasons why bounding $h/d_H(\Gamma, \tilde{\Gamma})$ alone may not be sufficient in practice.

- The model problem does not reflect the behavior of more difficult geometries, e.g., non-convex domains. Indeed, in the results section we consider one problem for which the basic formulation of the inverse IBM works (the reduced Hessian is invertible), and one problem for which the reduced Hessian is singular.
- For a desired mesh resolution h , it may be challenging to generate a mesh for which the distance between Γ and $\tilde{\Gamma}$ is sufficiently small to avoid large condition numbers; indeed, in the limit as $d_H(\Gamma, \tilde{\Gamma}) \rightarrow 0$, we are back to generating a conformal mesh.
- The non-unique solutions that the inverse IBM admits for linear advection are not excluded by bounding $h/d_H(\Gamma, \tilde{\Gamma})$. Thus, without further intervention, the discretized IBM will produce a singular system when applied to the linear advection equation.

For the reasons listed above, we need to consider some form of regularization for the inverse IBM.

4 | REGULARIZATION

We have seen that the basic formulation of the inverse IBM is ill-posed for both the Laplace and linear advection equations. Furthermore, while bounding the ratio $h/d_H(\Gamma, \tilde{\Gamma})$ seems to help in some cases, this strategy does not suffice on its own for all problems. Therefore, the basic formulation must be modified if it is to be useful in practice.

Regularized formulations are commonly used to ensure inverse problems are both well-posed and well-conditioned, so this is an obvious strategy to examine for the inverse IBM. In this section, we briefly review two popular regularizations used for inverse problems and argue why they are poorly suited for the inverse IBM. We then introduce a novel regularization for the inverse IBM that is well suited for discretizations that use weakly imposed boundary conditions.

4.1 | Review of regularization approaches for inverse problems

The popular regularizations that we review here are the Tikhonov and total variation (TV) diminishing regularizations. Both methods modify the objective function $J(u, c)$ by adding a (scaled) convex term, which we denote by S . Thus, the modified objective is given by

$$J_{\text{reg}}(u, c) = J(u, c) + \alpha S(u, c), \quad (8)$$

where $\alpha > 0$ is a regularization parameter.

Remark 6. The regularized optimization problem based on J_{reg} corresponds to a bi-objective problem in which α implicitly determines a particular Pareto optimal solution. Relatively small values of α select solutions that emphasize the boundary-condition accuracy at the cost of ill-conditioning, while large α produce better conditioned problems whose solutions may have significant mismatch in the boundary condition.

In the case of Tikhonov regularization applied to the inverse IBM, we have

$$S_{\text{Tik}}(c) \equiv \frac{1}{2} \int_{\tilde{\Gamma}} (c - c_0)^2 d\Gamma, \quad (9)$$

where $c_0 \in H^{\frac{1}{2}}(\tilde{\Gamma})$ is a constant function that can be used to reduce the detrimental effect that the regularization has on solution accuracy. The TV-diminishing regularization takes the form

$$S_{\text{TV}}(c) \equiv \int_{\tilde{\Gamma}} \sqrt{\nabla_{\tilde{\Gamma}} c \cdot \nabla_{\tilde{\Gamma}} c + \epsilon} d\Gamma,$$

where $c \in H^{\frac{3}{2}}(\tilde{\Gamma})$ and $\nabla_{\tilde{\Gamma}} c$ denotes the gradient of the control with respect to a parameterization of $\tilde{\Gamma}$. The constant $\epsilon > 0$ is a small parameter that is included to ensure the regularization is differentiable at points where $\nabla_{\tilde{\Gamma}} c = 0$. TV-based regularization was originally introduced in digital image processing for noise removal [29].

The Tikhonov term S_{Tik} is quadratic, strongly convex, and it tends to produce smooth solutions. In contrast, the TV-diminishing term tends to produce piecewise constant solutions that may arise, for example, in the presence of discontinuous physical properties. Note that the TV regularization is not strongly convex — any constant c minimizes S_{TV} — so it may not be sufficient to guarantee well-posedness, in general. Furthermore, the TV regularization is not quadratic, so iterative solution methods are required even for linear state equations.

For both regularizations, choosing an appropriate value for the parameter α can be challenging: it should be sufficiently large to stabilize the problem, but not too large to significantly impact solution accuracy; see [30] for an error estimation of the solution with respect to α . Some common techniques to select α include the L -curve method [31] and the Morozov discrepancy principle [32].

In the case of the inverse IBM, selecting the parameter α is further complicated by the requirements of solution convergence as $h \rightarrow 0$. To explain this complication, we consider a conforming finite-element discretization of the Laplace inverse IBM problem (2) with Tikhonov regularization: find $c_h \in V_h(\tilde{\Gamma})$ and $u_h \in V_h(\tilde{\Omega})$ that satisfy

$$\begin{aligned} \min_{c_h, u_h} \quad & J_{\text{reg}}(u_h, c_h) \equiv \frac{1}{2} \int_{\Gamma} (u_h - u_r)^2 d\Gamma + \frac{\alpha}{2} \int_{\tilde{\Gamma}} c_h^2 d\Gamma \\ \text{s.t.} \quad & \mathfrak{b}_h(u_h, v_h) = \int_{\tilde{\Omega}} \nabla u_h \cdot \nabla v_h d\Omega = 0, \quad \forall v_h \in V'_h(\tilde{\Omega}), \end{aligned} \quad (10)$$

where $V_h(\tilde{\Gamma})$, $V_h(\tilde{\Omega})$ and $V'_h(\tilde{\Omega})$ are appropriate finite-element spaces.

Theorem 1. Assume that the data and geometry in problem (2) are such that a unique solution exists, e.g., the model problem in Section 3, and denote this unique solution by $c \in H^{\frac{1}{2}}(\tilde{\Gamma})$ and $u \in H^1(\tilde{\Omega})$. Let $c_h \in V_h(\tilde{\Gamma})$ and $u_h \in V_h(\tilde{\Omega})$ be the solution to (10), and let h denote the nominal element size. Suppose the L^2 solution error on Γ satisfies $\|u_h - u\|_{\Gamma} = O(h^{p+1})$ for some integer

$p \geq 1$ and sufficiently small h . Furthermore, suppose the discrete control is bounded below, $\|c_h\|_{\tilde{\Gamma}} > B > 0$, for sufficiently small h . Then there exists $h^* > 0$ such that the regularization parameter satisfies

$$\alpha \leq M h^{p+1}, \quad \forall h \leq h^*,$$

where $M > 0$ is independent of h .

Proof. The first-order optimality conditions for (10) in the reduced space are

$$J'_{\text{reg}}(d_h) = \int_{\Gamma} (u_h(c_h) - u_{\Gamma}) u_h(d_h) d\Gamma + \alpha \int_{\tilde{\Gamma}} c_h d_h d\Gamma = 0, \quad \forall d_h \in V_h(\tilde{\Gamma}),$$

where, as in the continuous case, the directional derivative $(\nabla_{c_h} u_h) \cdot d_h = u_h(d_h)$ follows from the linear dependence of u_h on c_h . In particular, for $d_h = c_h$, we have

$$J'_{\text{reg}}(c_h) = \int_{\Gamma} (u_h(c_h) - u_{\Gamma}) u_h(c_h) d\Gamma + \alpha \int_{\tilde{\Gamma}} c_h^2 d\Gamma = 0.$$

Rearranging the above equality and applying Cauchy-Schwarz, we find

$$\alpha \|c_h\|_{\tilde{\Gamma}}^2 \leq \|u_h(c_h) - u_{\Gamma}\|_{\Gamma} \|u_h(c_h)\|_{\Gamma}.$$

The assumptions on the solution error and $\|c_h\|_{\tilde{\Gamma}}^2$ imply that there exists an $h^* > 0$ such that both $\|c_h\|_{\tilde{\Gamma}}^2 > B^2$ and $\|u_h(c_h) - u_{\Gamma}\|_{\Gamma} \|u_h(c_h)\|_{\Gamma} \leq \Lambda h^{p+1} \|u_{\Gamma}\|_{\Gamma}$, for some $\Lambda > 0$ and all $h < h^*$. Using these bounds in the above inequality we arrive at

$$\alpha \leq \frac{\|u_h(c_h) - u_{\Gamma}\|_{\Gamma} \|u_h(c_h)\|_{\Gamma}}{\|c_h\|_{\tilde{\Gamma}}^2} \leq \frac{\Lambda h^{p+1} \|u_{\Gamma}\|_{\Gamma}}{B^2}, \quad \forall h < h^*.$$

The result follows with $M = \Lambda \|u_{\Gamma}\|_{\Gamma} / B^2$. $\square$

The implication of Theorem 1 is that, if we have optimal solution error, then the positive effect that Tikhonov regularization has on conditioning must necessarily vanish as $h \rightarrow 0$. In other words, the ill-conditioning will return as the mesh is refined. A similar conclusion can be drawn with TV-based regularization. This motivates our search for a regularization that is compatible with high-order accuracy, but whose Hessian does not vanish with mesh refinement.

Remark 7. Theorem 1 gives a necessary but not sufficient condition on α . Indeed, our numerical experiments (not reported here) indicate that in some cases optimal solution accuracy is achieved only if $\alpha = O(h^{2p})$.

4.2 | Regularization via boundary-condition penalization

Our proposed regularization for the inverse IBM is inspired by weakly imposed boundary conditions. To motivate the regularization, recall our discussion of the linear advection problem in Section 2.2. If the boundary control is defined on all of $\tilde{\Gamma}$, we explained why a variation in c on the downwind boundary $\tilde{\Gamma}^+$ does not influence the state. While the control c is intended to define the boundary value, we can reverse the causation on $\tilde{\Gamma}^+$ and insist that u determines c . More generally, for weakly imposed boundary conditions, we can seek a solution for which the difference between the discretized state and control is small *on the entire boundary* $\tilde{\Gamma}$. That is, we consider a discrete regularization of the form

$$S(u_h, c_h) = \frac{1}{2} \int_{\tilde{\Gamma}} (u_h - c_h)^2 d\Gamma. \quad (11)$$

Remark 8. The regularization (11) is not useful for the continuous inverse IBM problem, since $c = u|_{\tilde{\Gamma}}$ in that case. However, the term is nonzero when the problem is discretized with weakly imposed boundary conditions, because $c_h \neq u_h|_{\tilde{\Gamma}}$, in general.

Remark 9. The regularization (11) is analogous to a Tikhonov regularization (9) with $c_0 = u$.

Unlike Tikhonov and TV-diminishing regularizations, the discretized version of (11) tends to zero as $h \rightarrow 0$. Consequently, any (constant) choice of the parameter α in (8) is compatible with an optimal solution convergence rate. For example, consider the Laplace equation under the assumptions in Theorem 1 and suppose the optimal convergence rate is $O(h^{p+1})$. Using the

regularization (11), the first-order optimality conditions in the reduced space imply (see the proof of Theorem 1 for further details)

$$\int_{\Gamma} (u_h(c_h) - u_r) u_h(d_h) d\Gamma + \alpha \int_{\tilde{\Gamma}} (u_h(c_h) - c_h)(u_h(d_h) - d_h) d\Gamma = 0, \quad \forall d_h \in V_h(\tilde{\Gamma}).$$

Previously, with Tikhonov regularization, it was necessary (but not sufficient) to have $\alpha = O(h^{p+1})$ in order to satisfy the first-order optimality conditions as $h \rightarrow 0$. In contrast, the proposed regularization is consistent with the above, since both terms in the condition asymptotically approach zero. In particular, assuming optimal convergence of the state to the boundary value on $\tilde{\Gamma}$, we have $|\int_{\tilde{\Gamma}} (u_h - c_h)^2 d\Gamma| = O(h^{2p+2})$.

Remark 10. Based on the above discussion, any constant choice for α is consistent with optimal solution convergence. We adopt $\alpha = 1$ for all subsequent numerical experiments.

The results, presented in the following section, confirm that the regularization (11) improves the conditioning of the discretized inverse IBM problem; however, *this regularization does not prevent ill-posedness as $h \rightarrow 0$* . To ensure a well-posed problem our hypothesis remains as before: the ratio $h/d_H(\Gamma, \tilde{\Gamma})$ must be bounded as the mesh is refined. Thus, the regularization should be used in conjunction with a computational domain $\tilde{\Omega}$ that converges to Ω .

5 | A NUMERICAL INVESTIGATION OF ACCURACY AND CONDITIONING

The purpose of the following numerical experiments is investigate i) the conditioning and ii) solution accuracy of the inverse IBM. For these studies we consider the steady, constant-coefficient, advection-diffusion equation:

$$\begin{aligned} \nabla \cdot (\lambda u - \mu \nabla u) &= f, & \forall x \in \Omega, \\ u &= u_r, & \forall x \in \Gamma', \end{aligned} \tag{12}$$

where $\lambda \in \mathbb{R}^2$ is the advection velocity and $\mu \in \mathbb{R}$ is the non-negative diffusion coefficient.

In the numerical experiments, three sets of coefficients are chosen to model different physics. These sets are as follows:

- $\lambda = [1, 1]^T$ and $\mu = 0$ for a pure advection problem;
- $\lambda = [0, 0]^T$, $\mu = 1$ for a Poisson-type diffusion problem;
- $\lambda = [1, 1]^T$, $\mu = 10^{-2}$ for an advection-diffusion problem.

The boundary conditions are imposed on $\Gamma' \equiv \Gamma$ for pure diffusion and advection-diffusion, and they are imposed on $\Gamma' \equiv \Gamma^-$ for pure advection.

5.1 | Discretization of the inverse IBM formulation

5.1.1 | Discretization of the advection-diffusion equation

We use a standard discontinuous Galerkin (DG) finite-element method to discretize the advection-diffusion equation (12). The symmetric interior penalty Galerkin (SIPG) method [33, 34] is used to discretize the diffusion term while standard upwinding is used for the advection part; see, for example, [35]. For completeness, a detailed description of the discretization is provided in Appendix A.

Remark 11. While we have adopted a DG finite-element method here, we emphasize that the proposed inverse IBM formulation is agnostic to the choice of discretization with the exception of the regularization (11), which requires weakly imposed boundary conditions.

5.1.2 | Discretization of the objective function

Let us first consider the discretization of $J(u, c) = \int_{\Gamma} (u - u_r)^2 d\Gamma$. We uniformly divide the boundary contour Γ into n_r line segments. On each line segment we introduce Gauss quadrature points, and we construct an interpolation to each quadrature point based on the finite element that contains the point. We can then use the interpolated values to evaluate u_h on Γ . The value of u_r at the quadrature points can be determined from the manufactured solution, which is given by either (16) or (17).

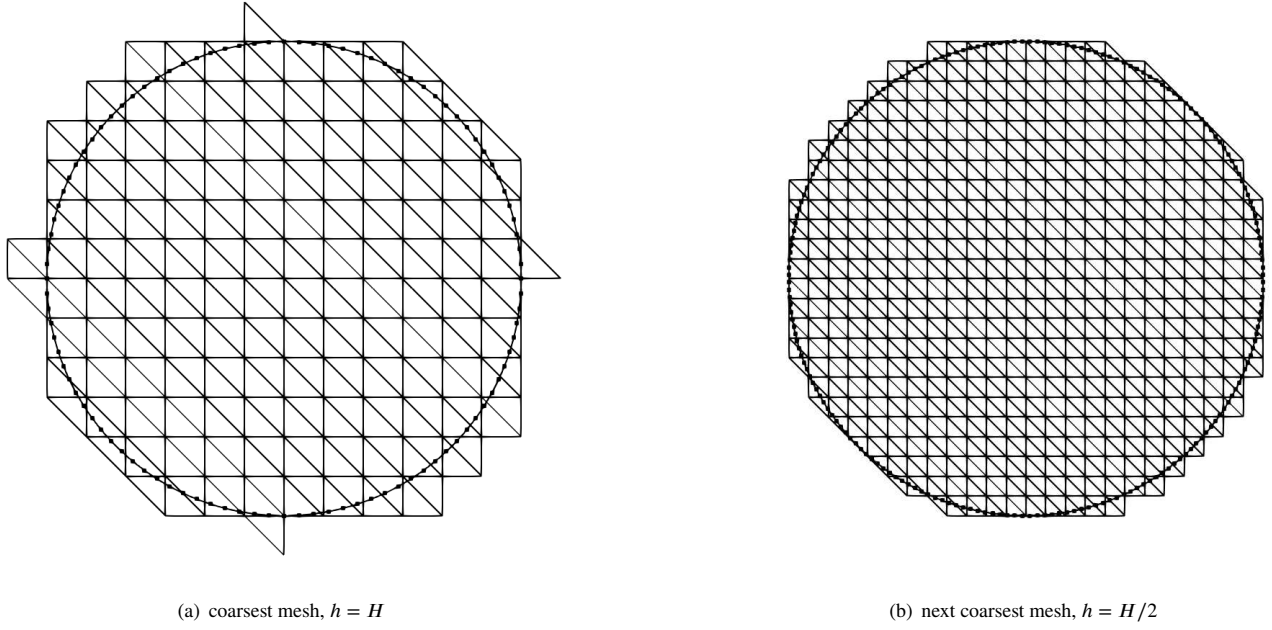


FIGURE 7 The coarsest two meshes for the unit disk domain used for the numerical experiments. The small black dots on circle denote the quadrature locations where the boundary-condition mismatch term in the objective is evaluated.

For the following results, we employ a quadrature rule that is exact for polynomials of degree p for each segment on Γ , where p is the degree of the Lagrange bases used in the discretization (A1). In addition, we consider different values of h_Γ when studying the condition number of the reduced Hessian; however, in the solution-convergence study, we choose n_Γ such that $h_\Gamma/h \approx 0.5$.

Remark 12. The size of the line segments, h_Γ , as well as the number of quadrature points on each line segment, can be varied independently of the finite-element discretization; however, as we shall see, there is a limit to how large the ratio h_Γ/h can be made, for a given quadrature rule, before it impacts the conditioning of the discrete system.

Finally, the regularization term $S(u_h, c_h) = \int_{\tilde{\Gamma}} (u_h - c_h)^2 d\Gamma$ is evaluated directly using the trace of u_h and the value of c_h on the edges $e \in \tilde{\Gamma}_h$.

5.1.3 | Definition of the expanded domain $\tilde{\Omega}$

The domain $\tilde{\Omega}$ is determined using the following process. We begin by generating a rectangular background mesh composed of uniform triangles that is of sufficient size to cover the target domain Ω . Then $\tilde{\Omega}$ is defined as the set of all elements that are either interior to Ω or are intersected by Γ . To illustrate, Figure 7 shows the coarsest two meshes for a unit disk domain and the corresponding $\tilde{\Omega}$. The nominal element size h is defined as the square root of the triangle areas.

Remark 13. The proposed construction for $\tilde{\Omega}$ ensures that the ratio $h/d_H(\Gamma, \tilde{\Gamma})$ remains bounded as $h \rightarrow 0$, which is what we want in practice. On the other hand, the construction makes it impossible to vary the Hausdorff distance $d_H(\Gamma, \tilde{\Gamma})$ independently of h , as we did for the model problem in Section 3; however, we can still investigate the impact of the ratio of $d_H(\Gamma, \tilde{\Gamma})$ on the spacing of the degrees of freedom by varying the finite-element polynomial degree p . We will present results of such a study in Section 5.3.

Remark 14. Recall the ill-posedness that affects the inverse IBM applied to the linear advection equation, e.g. point x_1 in Figure 2. We hypothesize that the proposed construction of $\tilde{\Omega}$ helps avoid ill-conditioning produced by such points, since any element containing these points is necessarily coupled to elements interior to the domain Ω ; however, further study is necessary to confirm this claim.

5.2 | Solution of the discretized inverse IBM system

We now briefly discuss the solution of the discretized inverse IBM formulation, as well as the construction of the reduced Hessian matrix used to study the conditioning of the system.

Let $\tilde{\Omega}_h$ be the triangularization of $\tilde{\Omega}$, let $\{\Phi_i\}_{i=1}^n$ denote the basis for the discrete state space $V_{h,p}(\tilde{\Omega}_h)$, and let $\{\phi_i\}_{i=1}^m$ be the basis for the discretized control space $V_{h,p}(\tilde{\Gamma}_h)$; see Appendix A for definitions of $V_{h,p}(\tilde{\Omega}_h)$ and $V_{h,p}(\tilde{\Gamma}_h)$. The finite-element state and control can be represented as

$$u_h(x) = \sum_{i=1}^n (\bar{u}_h)_i \Phi_i(x), \quad \text{and} \quad c_h(x) = \sum_{i=1}^m (\bar{c}_h)_i \phi_i(x),$$

respectively, where $\bar{u}_h \in \mathbb{R}^n$ and $\bar{c}_h \in \mathbb{R}^m$ are the nodal coefficients. Substituting these expressions for u_h and c_h into the advection-diffusion trilinear form (A1) and the discretized objective, the discretized inverse IBM problem can be expressed as the following finite-dimensional optimization problem:

$$\begin{aligned} \min_{\bar{c}_h, \bar{u}_h} \quad & J_h(u_h, c_h) = \frac{1}{2} \bar{c}_h^T H_{cc} \bar{c}_h - \bar{c}_h^T H_{cu} \bar{u}_h + \frac{1}{2} \bar{u}_h^T H_{uu} \bar{u}_h - \bar{b}_h^T \bar{u}_h + J_0, \\ \text{s.t.} \quad & \mathbf{b}_h(u_h, c_h, v_h) = \bar{v}_h^T (A_u \bar{u}_h + A_c \bar{c}_h - \bar{f}_h) = 0, \quad \forall \bar{v}_h \in \mathbb{R}^n. \end{aligned} \quad (13)$$

The optimization problem (13) is a quadratic program since it has a quadratic objective and a linear constraint. Furthermore, one can show that the objective is convex in $\bar{c}_h$ and $\bar{u}_h$. Consequently, the solution to (13) is equivalent to the solution to the following saddle-point problem:

$$\begin{bmatrix} H_{uu} & -H_{cu}^T & A_u^T \\ -H_{cu} & H_{cc} & A_c^T \\ A_u & A_c & 0 \end{bmatrix} \begin{bmatrix} \bar{u}_h \\ \bar{c}_h \\ \bar{\psi}_h \end{bmatrix} = \begin{bmatrix} \bar{b}_h \\ 0 \\ \bar{f}_h \end{bmatrix}, \quad (14)$$

where $\bar{\psi}_h \in \mathbb{R}^n$ are the Lagrange multipliers associated with the constraint; in the present context, the multipliers are equivalent to the nodal coefficients of the adjoint. We solve the linear system (14) using a sparse direct solver.

Remark 15. In practice, the efficient solution of (13) or (14) would require more sophisticated algorithms, such as sparse iterative solvers with special-purpose preconditioners and Newton's method for nonlinear PDEs. The application of such algorithms to (13) is the subject of ongoing research — see, for example, the preliminary results in [36] — but is beyond the scope of the current work.

The optimization statement (13) and linear system (14) are full-space formulations, since the control, state, and multipliers are solved simultaneously; see, for example, [37, 38, 39]. Alternatively, we can eliminate the state and multipliers to arrive at the following reduced-space optimization statement:

$$\min_{\bar{c}_h} \quad J_h(\bar{c}_h) = \frac{1}{2} \bar{c}_h^T H \bar{c}_h + g_c^T \bar{c}_h + J_{0,c}$$

where $J_{0,c}$ is a constant, $g_c \in \mathbb{R}^m$ is the reduced gradient and

$$H = A_c^T A_u^{-T} H_{uu} A_u^{-1} A_c - A_c^T A_u^{-T} H_{cu}^T - H_{cu} A_u^{-1} A_c + H_{cc} \quad (15)$$

is the reduced Hessian. We will study the conditioning of the reduced Hessian in the following sections, since this provides an indication of the difficulty of solving (14) using iterative methods. Furthermore, we can use H to verify our conclusions regarding the model problem from Section 3.

5.3 | Conditioning of the reduced Hessian

This section investigates how the proposed regularization helps the conditioning of H on both convex and nonconvex geometries. We also consider the impact of the ratio h/h_Γ .

5.3.1 | Poisson equation on the unit-disk domain

We begin our investigation of the conditioning of H , defined by (15), with the Poisson-type diffusion case, that is, with $\lambda = [0, 0]^T$ and $\mu = 1$ in (12). The problem domain is the unit disk, $\Omega = \{x \in \mathbb{R}^2 \mid \|x\|_2 < 1\}$.

Recall that the immersed boundary is discretized into line n_Γ segments of length h_Γ , and that h_Γ is independent of the volume mesh size h . However, if there are m control degrees of freedom, we expect that $n_\Gamma \gtrsim m$ is necessary for the reduced Hessian

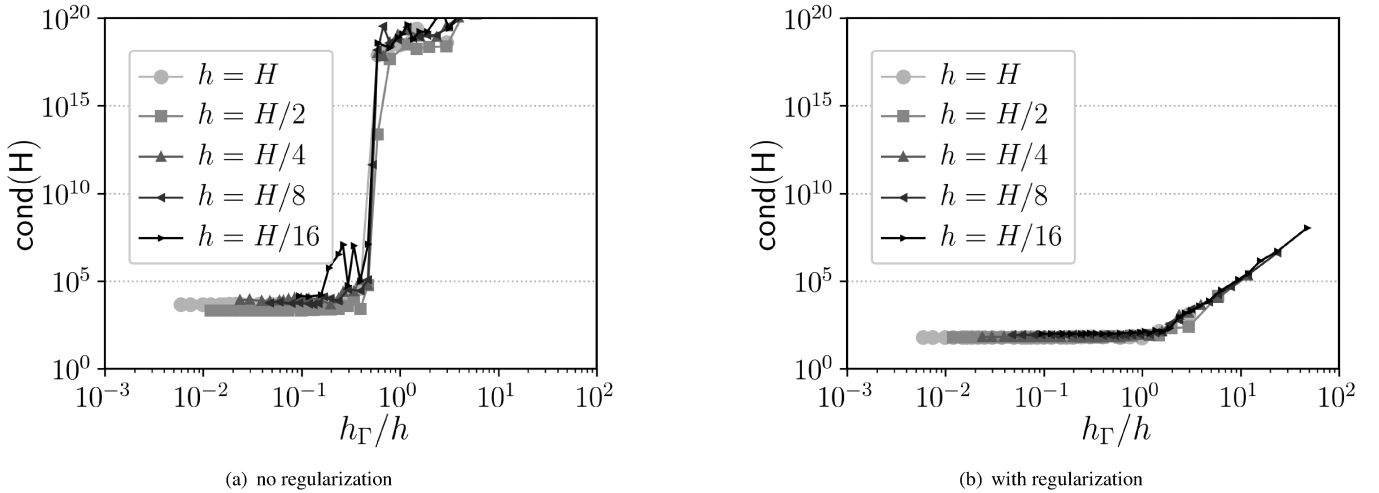


FIGURE 8 Condition number of the reduced Hessian (15) versus the ratio between the spacing on Γ to the mesh spacing, for the Poisson equation.

to be non-singular. This intuition is supported by the results in Figure 8(a), which plots the condition number of H for the basic (unregularized) inverse IBM versus the ratio h_Γ/h . Results for a range of element sizes h are shown, with $p = 1$ elements adopted in all cases. We see that the condition number is relatively constant until a threshold of $h_\Gamma/h \approx 0.5$ is reached, at which point H becomes singular to working precision.

Figure 8(b) shows the analogous results for the regularized objective. The figure highlights two benefits of the regularization for diffusive PDEs. First, rather than abruptly becoming singular, the condition number of H gradually increases when h_Γ/h passes a threshold of approximately one. Second, for values of h_Γ/h below this threshold, the condition number of H is approximately two orders of magnitude smaller than when no regularization is used.

Next, we would like to study the effect of the Hausdorff distance $d_H(\Gamma, \tilde{\Gamma})$ on the conditioning of H like we did for the model problem in Section 3. Unfortunately, we cannot vary the Hausdorff distance independently from h because of how $\tilde{\Omega}$ is constructed; see the discussion in Section 5.1.3. We can, however, mimic the study from Section 3 by varying the degree of the polynomial basis for fixed h and $d_H(\Gamma, \tilde{\Gamma})$. That is, whereas h was the control spacing in the earlier study, here we can use h/p to define the nominal control spacing.

Figure 9 plots the condition number of the reduced Hessian versus the ratio $(h/p)/d_H(\Gamma, \tilde{\Gamma})$ for both the unregularized and regularized Poisson problems. We consider $p = 1, 2$, and 3 degree finite-element basis functions and a range of mesh sizes h and a (dependent) range of Hausdorff distances. The results for the unregularized problem, plotted in Figure 9(a), behave similar to the model-problem results shown in Figures 5(b) and 6(b): as the control spacing becomes small relative to the Hausdorff distance, the condition number “blows up.” The regularized problem — see Figure 9(b) — displays qualitatively the same pattern, but the condition number is two orders of magnitude smaller for the same value of $(h/p)/d_H(\Gamma, \tilde{\Gamma})$. For both the unregularized and regularized cases, we observe no significant dependence on h itself. Overall, the results in Figure 9 confirm the conclusions drawn earlier.

5.3.2 | Linear advection equation on the unit-disk domain

Here we investigate the conditioning of the reduced Hessian of the inverse IBM in the context of the linear advection equation on the unit circle domain. We can only consider the regularized variant of the inverse IBM, because, unlike the Poisson problem, the unregularized advection problem is singular to working precision.

Figure 10(a) plots the condition number of H versus the size ratio h_Γ/h for the $p = 1$ discretization. We see that the regularized advection problem behaves more like the unregularized diffusion problem. In particular, there is a threshold, $h_\Gamma/h \approx 1$ here, above which the reduced Hessian becomes singular. Consequently, it is critical to use a sufficiently fine discretization of Γ when the inverse IBM is applied to hyperbolic PDEs.

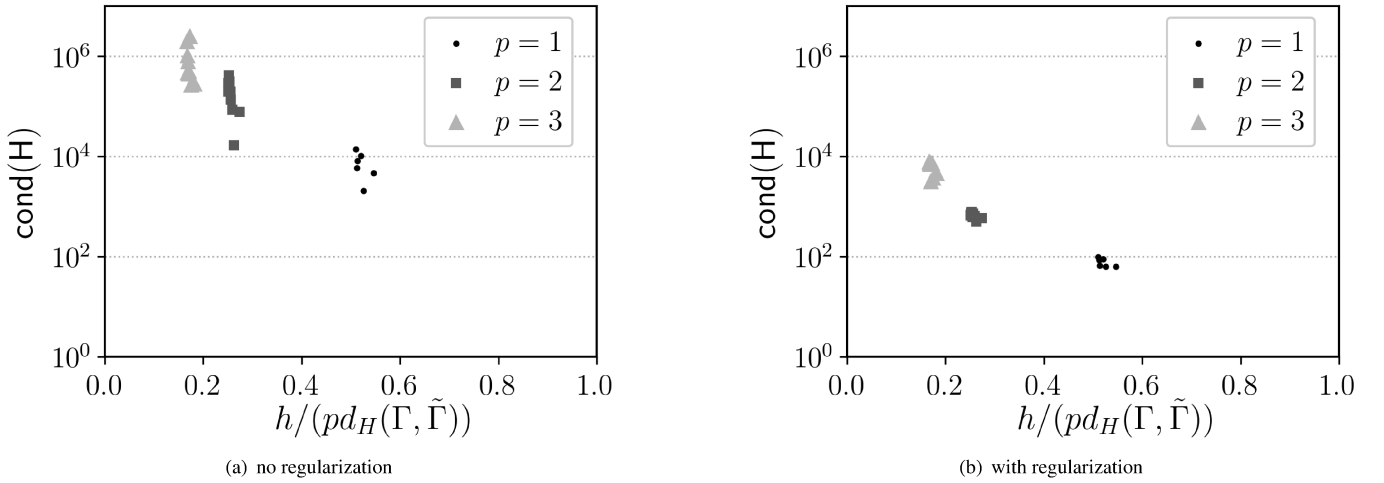


FIGURE 9 Condition number of the reduced Hessian (15) versus the ratio between the degree-of-freedom spacing, h/p , and the Hausdorff distance $d_H((\Gamma, \tilde{\Gamma}))$, for the Poisson equation.

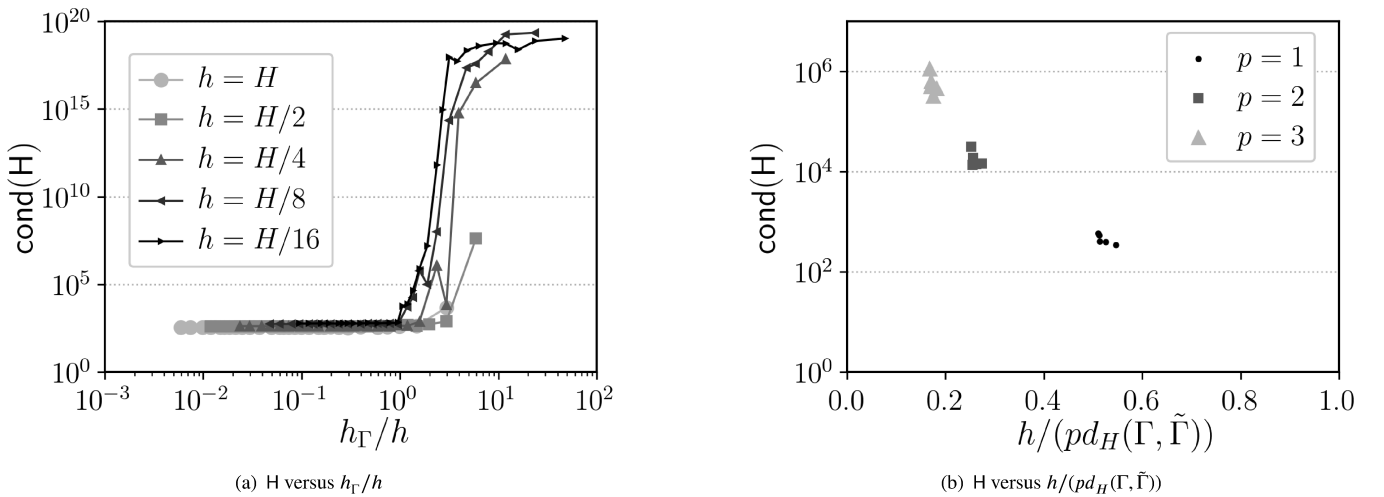


FIGURE 10 Trends in the condition number of the reduced Hessian (15) for the linear advection equation: dependence on h_Γ/h (left) and dependence on $h/(pd_H(\Gamma, \tilde{\Gamma}))$ (right). These results are based on the regularized inverse IBM formulation, since the Hessian is singular without regularization.

Figure 10(b) shows how the condition number of the reduced Hessian responds to changes in the ratio $(h/p)/d_H(\Gamma, \tilde{\Gamma})$. As with the diffusion problem, the condition number grows unbounded as the control spacing becomes small relative to the Hausdorff distance. Furthermore, the magnitude of the condition number is larger for linear advection for the same value of $(h/p)/d_H(\Gamma, \tilde{\Gamma})$. This suggests that the inverse IBM, with the proposed regularization, may be limited to modest values of p for hyperbolic problems.

5.3.3 | Conditioning on a star-shaped (nonconvex) domain

The results of the previous sections were obtained for a relatively benign domain; in particular, the interior of the unit-disk is convex and the boundary is smooth. Therefore, to see if the conclusions apply to more general domains, we repeat the numerical

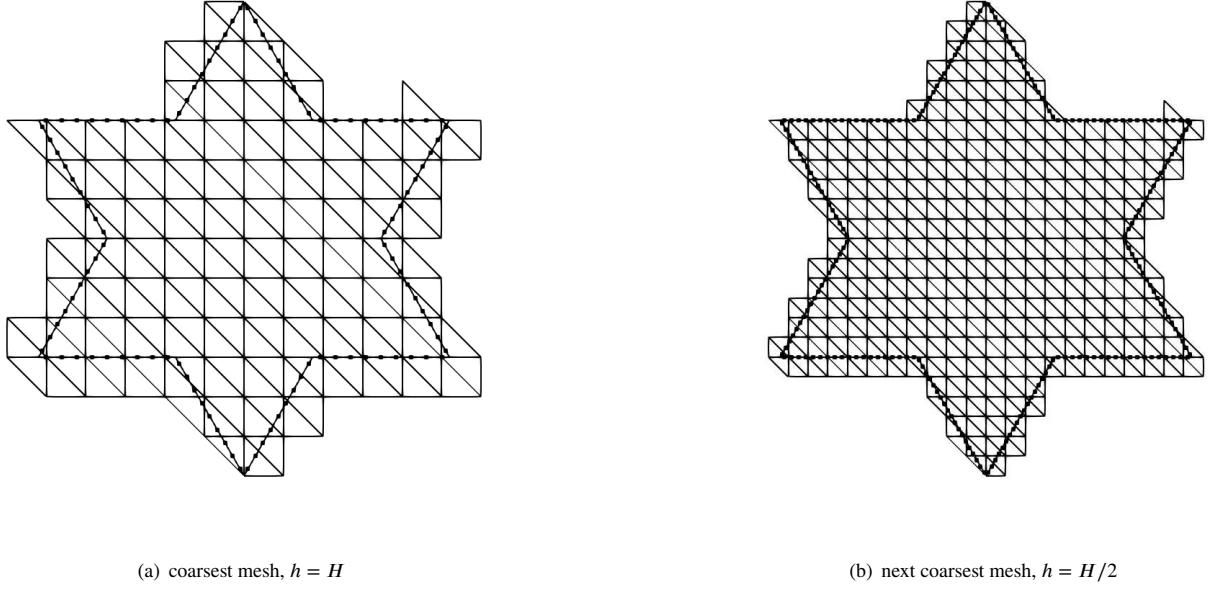


FIGURE 11 The coarsest two meshes for the star-shape domain used for the numerical experiments. The small black dots on circle denote the quadrature locations where the boundary-condition mismatch term in the objective is evaluated.

experiments on a more complicated, star-shaped domain. See Figure 11 for the coarsest two meshes used for the star-shaped domain.

Unlike for the unit-disk domain, the unregularized inverse IBM is singular for both the advection and diffusion problems. We are not certain why the star-shaped domain causes issues for the unregularized formulation applied to the diffusion problem; however, we have solved problems on square domains (geometric singularities) without difficulty, so we believe the non-convex boundary is to blame. Regardless of the source of the problem, the regularized formulation successfully resolves it, and the corresponding results are plotted in this section.

We begin with the Poisson equation on the star-shaped domain. Analogous to Figures 8 and 9(b), Figures 12(a) and 12(b) plot the dependence of condition number of the reduced Hessian on the size ratio h_Γ/h and the ratio $h/(pd_H(\Gamma, \tilde{\Gamma}))$, respectively. We can see from the figures that the condition number behaves similarly on the star-shaped and unit-disk domains, both in terms of trend and value.

Results for linear-advection are also similar between the unit-disk and star-shaped domains; see Figure 13. One noticeable difference from the results in Figure 10, is that the condition number on the mesh corresponding to $h = H/8$ is consistently larger than on other meshes, and that oscillations in the condition number for this mesh occur around $h_\Gamma/h \approx 0.5$. The source of this behavior for the $h = H/8$ mesh is unclear.

5.4 | Convergence study

We conclude the numerical experiments with a study of solution accuracy and convergence rate. To this end, the method of manufactured solutions [40] is employed, and both smooth and nonsmooth solutions are considered.

5.4.1 | Smooth solution

First a smooth exact solution defined as

$$u(x, y) = e^{x+y} \sin(\pi x) \sin(\pi y), \quad (16)$$

is selected a priori and used to determine f and u_Γ in (12). The two problem domains used in the condition-number study, the unit-disk and the star-shaped domains, are again considered.

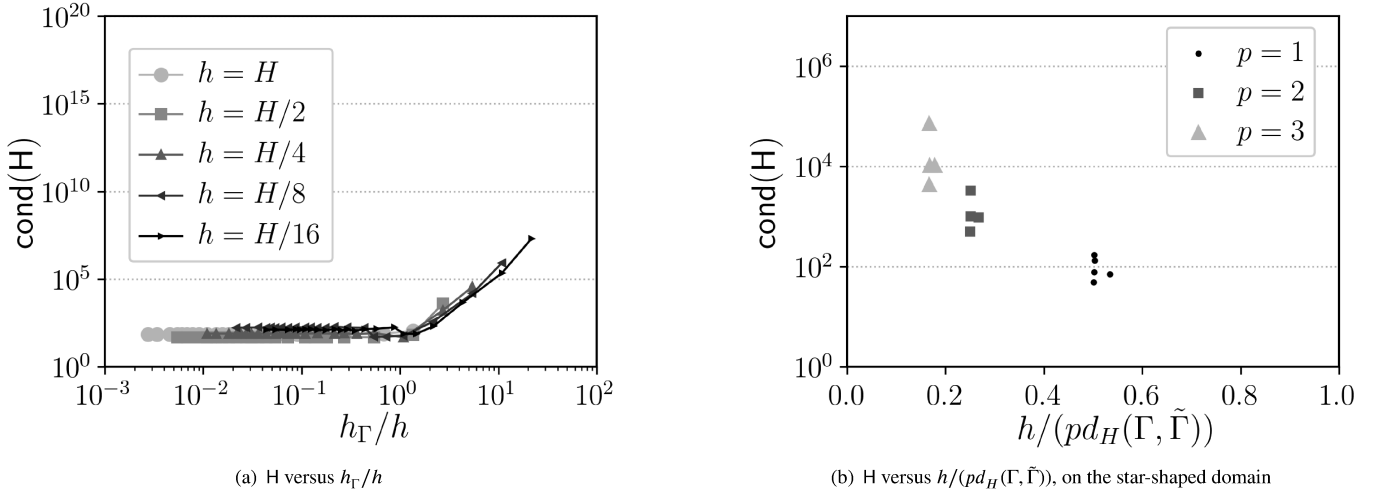


FIGURE 12 Trends in the condition number of the reduced Hessian (15) for the Poisson equation on *the star-shaped domain*: dependence on h_Γ/h (left) and dependence on $h/(pd_H(\Gamma, \tilde{\Gamma}))$ (right). The regularized inverse IBM formulation was necessary for this non-convex problem.

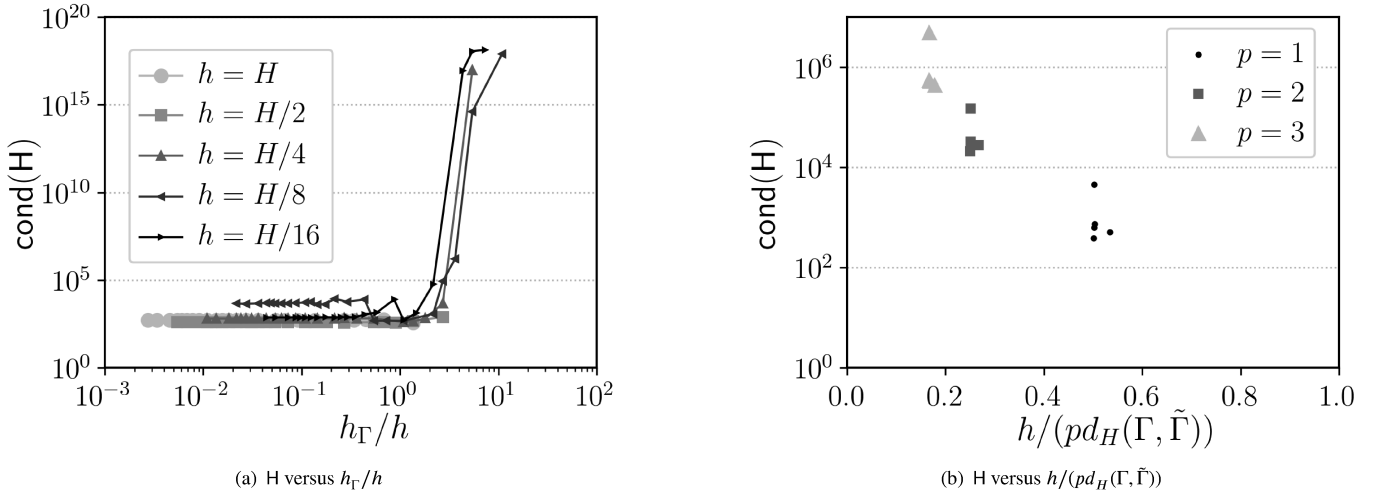


FIGURE 13 Trends in the condition number of the reduced Hessian (15) for the linear advection equation on **the star-shaped domain**: dependence on h_Γ/h (left) and dependence on $h/(pd_H(\Gamma, \tilde{\Gamma}))$ (right). As always with the linear advection equation, regularization was included with the inverse IBM.

To estimate the asymptotic convergence rate, we use a sequence of five uniformly refined triangular meshes for each problem domain. To obtain the next mesh in the sequence, each element is subdivided into four. Thus, the element sizes of the finer meshes are $H/2$, $H/4$, $H/8$ and $H/16$. The coarsest two meshes for the two domains are shown in Figure 7 and 11. We can see that the physical boundary intersects mesh elements at different locations, including vertices; in our experience, the method is robust with respect to such coincident nodes.

The solution contours for the star-shaped domain using $p = 1$ and $p = 4$ basis functions on the coarsest mesh are compared against the exact contours in Figure 14(a) and 14(b), respectively. We can see that the discrete solution agrees well with the exact

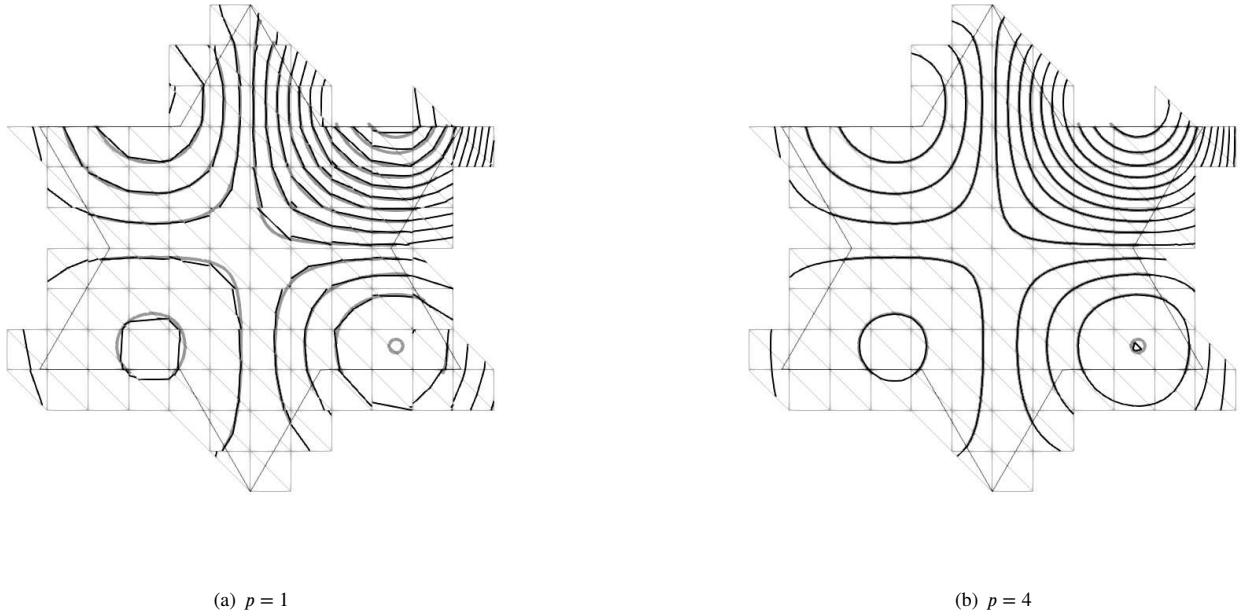


FIGURE 14 Star domain: solution contours of the advection-diffusion problem on the coarsest mesh. Exact solution: thick gray line, discrete solution: black line.

solution. Furthermore, as expected for a smooth solution, the higher-order approximation produces qualitatively better results on the same mesh.

For a quantitative assessment of accuracy, it is common to use the L^2 error to measure the accuracy of a finite-element solution u_h ; however, evaluating the L^2 norm exactly on Ω is not straightforward, because the standard quadrature rules do not apply on the elements cut by the boundary. The approach adopted in this paper is to set the solution error to zero at all quadrature points that lie in $\tilde{\Omega} \setminus \Omega$.

Figure 15 and 16 plot the solution error versus element size h for the specific advection, diffusion, and advection-diffusion problems defined earlier. For all problems, the inverse IBM achieves the optimal convergence rate of $p + 1$.

5.4.2 | Nonsmooth solution

It is well known that the solution accuracy of a (polynomial basis) finite-element method is limited by the smoothness of the solution. Therefore, since we use a standard DG finite-element method to discretize the state equation, we do not claim or expect the inverse IBM to be high-order accurate in the case of nonsmooth solutions. To show this, we consider the following exact solution for the Laplace equation on an L-shaped domain as shown in Figure 17(a):

$$u = (x^2 + y^2)^{1/3} \sin \left[\arctan \left(\frac{2x}{3y} \right) + \frac{\pi}{3} \right]. \quad (17)$$

This exact solution results in a zero source, $f = 0$, in the Poisson equation and it has a singularity at the concave corner.

The L^2 errors for degree $p = 1, 2, 3$, and 4 basis functions are plotted in Figure 17(b). The errors are obtained on the same sequence of background meshes as in Section 5.4.1. For all degrees of approximation, we obtain an asymptotic error rate of approximately 1.5 rather than the optimal rate of $p + 1$. Nevertheless, higher degree basis functions do give a smaller error on the same mesh, which is probably due to the high resolution of the discretization in the smooth region.

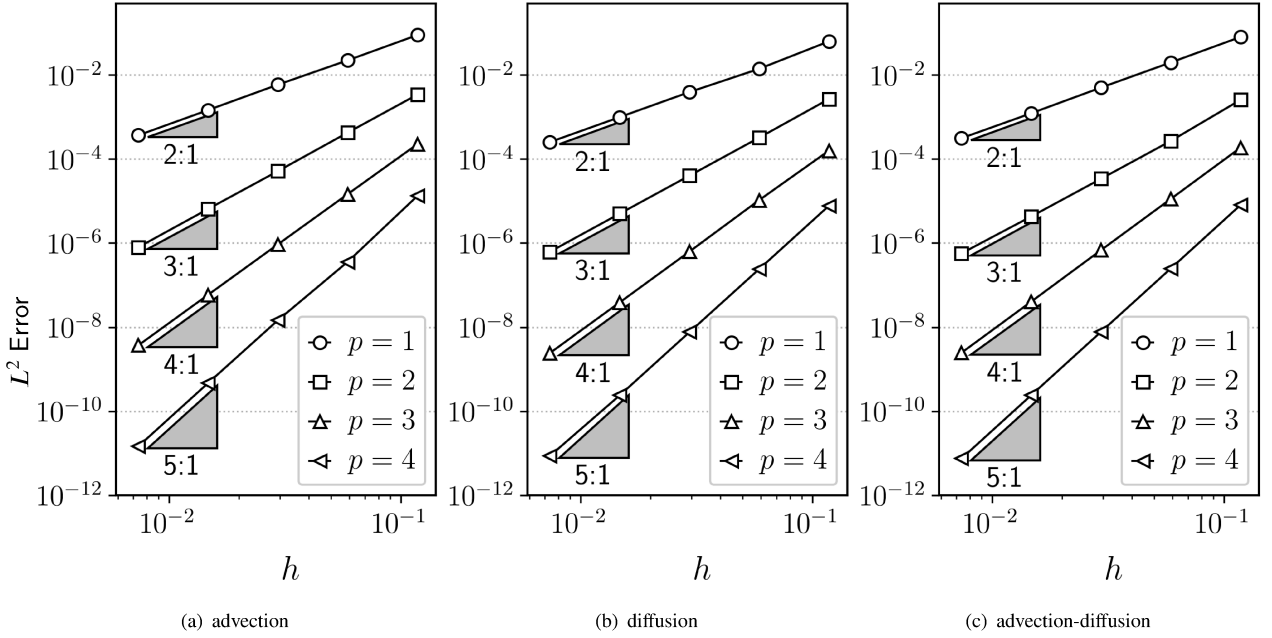


FIGURE 15 L^2 solution error versus element size h on the unit disk.

6 | SUMMARY AND DISCUSSION

First, a brief summary. The proposed inverse IBM introduces a control on the boundary of an expanded domain $\tilde{\Omega}$ that encompasses the target domain Ω . In the basic formulation, the value of the control is determined by minimizing the mismatch in the boundary condition on the true boundary Γ subject to the state satisfying the desired PDE on the expanded domain $\tilde{\Omega}$. This basic formulation is ill-posed for the Laplace and linear advection equations: for the former, the solution does not depend continuously on the data, and for the latter the solution is not uniquely determined by the data. The ill-posedness is addressed by ensuring the expanded domain tends to the target domain and by including a control-state penalty term in the objective. Applied to a discontinuous Galerkin finite-element discretization of the advection, advection-diffusion, and diffusion equations, the inverse IBM formulation achieves optimal solution convergence rates (for smooth solutions), and the reduced Hessian remains well-conditioned.

We believe the inverse IBM has a number of attractive features. The approach is agnostic to the underlying discretization, so it can be applied to finite difference, finite volume, and finite-element discretizations. Furthermore, the method is compatible with high-order discretizations as our results demonstrate. Finally, once the discretization is chosen, the implementation is straightforward in the sense that there are no “corner cases” that require special treatment; all that is required is an interpolation/projection operator from $\tilde{\Omega}$ to the boundary Γ .

There is no “free lunch,” and the attractive features of the inverse IBM come at a price. Specifically, the method trades computational cost for ease of implementation and accuracy. If there are n state variables and m control variables, the inverse IBM has $2n + m$ variables. This should be contrasted with other approaches that have only n state variables. In addition, the inverse IBM is a PDE-constrained optimization problem, which requires the solution of a linear saddle-point problem for linear PDEs and a nonlinear system in general. Nevertheless, we believe the increased computational cost is well worth the reduced human time required in mesh generation around complex configurations.

The proposed inverse IBM is promising, but there remain many challenges that must be addressed before it can be used on practical problems. Foremost among these challenges is the robust and efficient solution of the nonlinear PDE-constrained optimization problem associated with the inverse IBM. A related issue is the need for specialized preconditioners for the saddle-point systems that arise in the Newton iterations. Finally, there is the question of how the method can be extended to unsteady simulations.

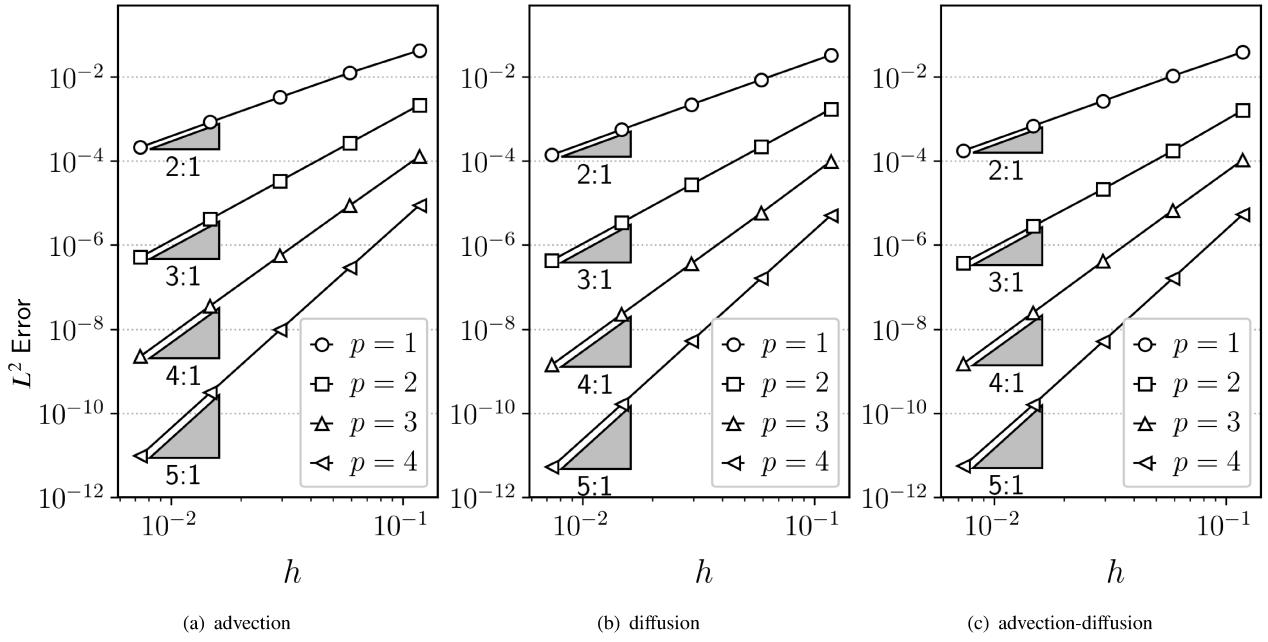


FIGURE 16 L^2 solution error versus element size h on the star-shaped domain.

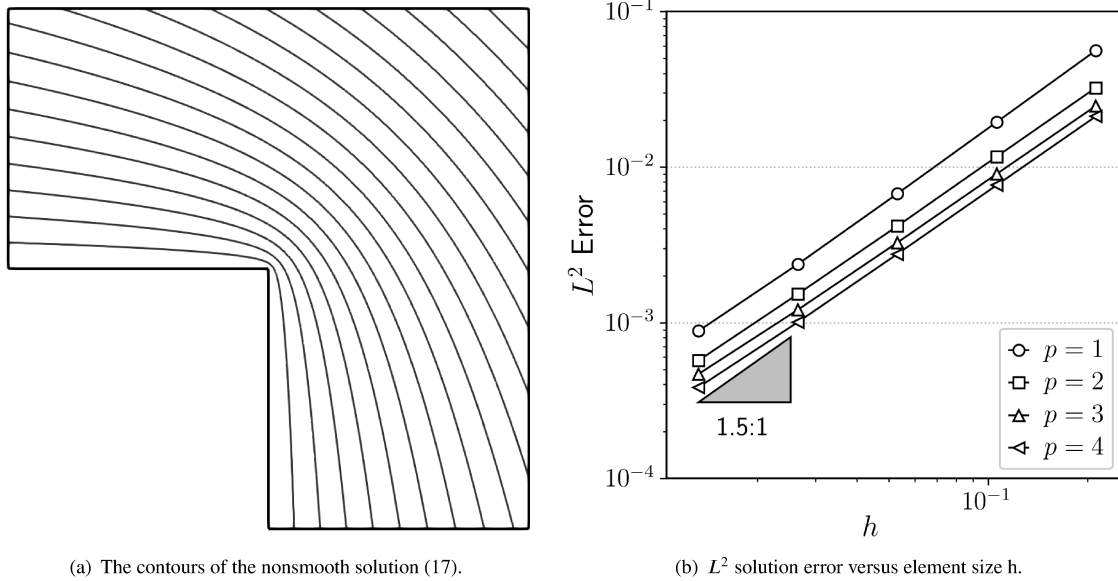


FIGURE 17 Convergence study for the Poisson equation on the L-shaped domain.



APPENDIX

A DISCRETIZATION OF THE ADVECTION-DIFFUSION EQUATION

We begin with introducing some additional notation. Let $\tilde{\Omega}_h$ be a shape-regular subdivision of $\tilde{\Omega}$ into disjoint triangular elements $K \in \tilde{\Omega}_h$; the definition of $\tilde{\Omega}$ is provided in Section 5.1.3. The space of discontinuous piecewise polynomials of total degree p on $\tilde{\Omega}_h$ is

$$V_{h,p}(\tilde{\Omega}_h) \equiv \{v_h \in L^2(\tilde{\Omega}) \mid v_h|_K \circ \sigma_K \in \mathbb{P}^p(\hat{K}), K \in \tilde{\Omega}_h\},$$

where $\mathbb{P}^p(\hat{K})$ is the space of polynomials of total degree p on the unit triangle $\hat{K}$, and σ_K denotes a push-forward operation from K to $\hat{K}$. In this work we use nodal Lagrange bases of degree one to four.

To define the discrete space for the control, we introduce $\tilde{\Gamma}_h$ as the subdivision of $\tilde{\Gamma}$ corresponding to the boundary faces of the elements in $\tilde{\Omega}_h$, i.e. $\tilde{\Gamma}_h = \{\tilde{K} \cap \tilde{\Gamma} \neq \emptyset, K \in \tilde{\Omega}_h\}$. Then, the discrete control space is the set of discontinuous piecewise polynomials of total degree p on $\tilde{\Gamma}_h$:

$$V_{h,p}(\tilde{\Gamma}_h) \equiv \{d_h \in L^2(\tilde{\Gamma}) \mid d_h|_e \circ \sigma_e \in \mathbb{P}^p(\hat{e}), e \in \tilde{\Gamma}_h\}.$$

Here, $\mathbb{P}^p(\hat{e})$ is the space of polynomials of degree p on the reference interval $\hat{e} = [-1, 1]$, and σ_e denotes push-forward operation from edge e to $\hat{e}$.

Next, we define the standard jump and average operators on the set of interior faces, Γ_I . For an interior face $e \in \Gamma_I$, the jump and average operators, applied to the scalar $u_h \in V_{h,p}(\tilde{\Omega}_h)$ and vector $q_h \in [V_{h,p}(\tilde{\Omega}_h)]^2$, are given by

$$\begin{aligned} \{u_h\} &= (u_h^+ + u_h^-)/2, & \{q_h\} &= (q_h^+ + q_h^-)/2, \\ [u_h] &= u_h^+ n^+ + u_h^- n^-, & [q_h] &= q_h^+ \cdot n^+ + q_h^- \cdot n^-, \end{aligned}$$

where n^+ and n^- are the outward unit normals with respect to ∂K^+ and ∂K^- , respectively, and u_h^+ and u_h^- are the traces of u_h along the common face from the interior of K^+ and K^- , respectively. The traces q_h^+ and q_h^- are defined similarly.

The trilinear weak form corresponding to the DG discretization of (12) is a mapping $\mathfrak{b}_h : V_{h,p}(\tilde{\Omega}_h) \times V_{h,p}(\tilde{\Gamma}_h) \times V_{h,p}(\tilde{\Omega}_h) \rightarrow \mathbb{R}$ defined by

$$\begin{aligned} \mathfrak{b}_h(u_h, c_h, v_h) &= \int_{\tilde{\Omega}_h} [\nabla v_h \cdot (\lambda u_h - \mu \nabla u_h) + v_h f_h] \, d\Omega \\ &\quad - \int_{\Gamma_I} [v_h] \cdot \hat{F}(u_h^+, u_h^-) \, d\Gamma - \int_{\tilde{\Gamma}} [v_h \hat{F}(u_h, c_h) \cdot n] \, d\Gamma \\ &\quad + \int_{\Gamma_I} [v_h] \cdot \{\mu \nabla u_h\} + \{\mu \nabla v_h\} \cdot [u_h] - \epsilon \{\mu\} [u_h] \cdot [v_h] \, d\Gamma \\ &\quad + \int_{\tilde{\Gamma}} [v_h \mu \nabla u_h \cdot n + (\mu \nabla v_h) \cdot n(u_h - c_h) - \epsilon \mu v_h (u_h - c_h)] \, d\Gamma, \end{aligned} \tag{A1}$$

where $\hat{F}$ denotes the standard upwind flux function, and ϵ is the SIPG penalty parameter based on [34]. The discrete source, f_h , is obtained by projecting the exact source f , derived from the manufactured solution (16), onto $V_{h,p}(\tilde{\Omega}_h)$.

ACKNOWLEDGEMENTS

The authors are grateful to Fengyan Li for looking over the manuscript, answering our functional analysis questions, and helping show that the mapping T was unbounded in Section 2.1.1. The authors were supported by the National Science Foundation under Grant No. 1825991 and gratefully acknowledge this support.

References

- [1] Slotnick J, Khodadoust A, Alonso J, et al. CFD vision 2030 study: a path to revolutionary computational aerosciences. tech. rep., NASA Langley Research Center; 2014. NASA/CR-2014-218178.

- [2] Peskin CS. Numerical Analysis of Blood Flow in the Heart. *Journal of Computational Physics* 1977; 25(3): 220–252. doi: 10.1016/0021-9991(77)90100-0
- [3] Purvis JW, Burkhalter JE. Prediction of critical Mach number for store configurations. *AIAA Journal* 1979; 17(11): 1170–1177. doi: 10.2514/3.7617
- [4] Berger M, Leveque R. An adaptive Cartesian mesh algorithm for the Euler equations in arbitrary geometries. In: American Institute of Aeronautics and Astronautics; 1989
- [5] LeVeque RJ, Li Z. The Immersed Interface Method for Elliptic Equations with Discontinuous Coefficients and Singular Sources. *SIAM Journal on Numerical Analysis* 1994; 31(4): 1019–1044. doi: 10.1137/0731054
- [6] Aftosmis MJ, Berger MJ, Melton JE. Robust and efficient Cartesian mesh generation for component-based geometry. *AIAA journal* 1998; 36(6): 952–960.
- [7] Hansbo A, Hansbo P. An unfitted finite element method, based on Nitsche’s method, for elliptic interface problems. *Computer Methods in Applied Mechanics and Engineering* 2002; 191(47-48): 5537–5552. doi: 10.1016/s0045-7825(02)00524-8
- [8] Fidkowski KJ, Darmofal DL. A triangular cut-cell adaptive method for high-order discretizations of the compressible Navier-Stokes equations. *Journal of Computational Physics* 2007; 225: 1653–1672.
- [9] Lew AJ, Buscaglia GC. A discontinuous-Galerkin-based immersed boundary method. *International Journal for Numerical Methods in Engineering* 2008; 76(4): 427–454. doi: 10.1002/nme.2312
- [10] Brehm C, Fasel HF. A novel concept for the design of immersed interface methods. *Journal of Computational Physics* 2013; 242: 234–267. doi: 10.1016/j.jcp.2013.01.027
- [11] Sun H. *A robust simplex cut-cell method for adaptive high-order discretizations of aerodynamics and multi-physics problems*. PhD thesis. Massachusetts Institute of Technology, 2013.
- [12] Mattsson K, Almquist M. A high-order accurate embedded boundary method for first order hyperbolic equations. *Journal of Computational Physics* 2017; 334: 255–279. doi: 10.1016/j.jcp.2016.12.034
- [13] Peskin CS. The immersed boundary method. *Acta Numerica* 2002; 11: 479–517. doi: 10.1017/s0962492902000077
- [14] Mittal R, Iaccarino G. Immersed boundary methods. *Annual Review of Fluid Mechanics* 2005; 37: 239–261.
- [15] Mayo A. The Fast Solution of Poisson’s and the Biharmonic Equations on Irregular Regions. *SIAM Journal on Numerical Analysis* 1984; 21(2): 285–299. doi: 10.1137/0721021
- [16] Mayo A. The rapid evaluation of volume integrals of potential theory on general regions. *Journal of Computational Physics* 1992; 100(2): 236–245. doi: 10.1016/0021-9991(92)90231-m
- [17] Marques AN, Nave JC, Rosales RR. A Correction Function Method for Poisson problems with interface jump conditions. *Journal of Computational Physics* 2011; 230(20): 7567–7597. doi: 10.1016/j.jcp.2011.06.014
- [18] Marques AN. *A correction function method to solve incompressible fluid flows to high accuracy with immersed geometries*. PhD thesis. Massachusetts Institute of Technology, 2012.
- [19] Fedkiw RP, Aslam T, Xu S. The Ghost Fluid Method for Deflagration and Detonation Discontinuities. *Journal of Computational Physics* 1999; 154(2): 393–427. doi: 10.1006/jcph.1999.6320
- [20] Fedkiw RP, Aslam T, Merriman B, Osher S. A Non-oscillatory Eulerian Approach to Interfaces in Multimaterial Flows (the Ghost Fluid Method). *Journal of Computational Physics* 1999; 152(2): 457–492. doi: 10.1006/jcph.1999.6236
- [21] Liu XD, Fedkiw RP, Kang M. A Boundary Condition Capturing Method for Poisson’s Equation on Irregular Domains. *Journal of Computational Physics* 2000; 160(1): 151–178. doi: 10.1006/jcph.2000.6444

- [22] Kang M, Fedkiw RP, Liu XD. A Boundary Condition Capturing Method for Multiphase Incompressible Flow. *Journal of Scientific Computing* 2000; 15(3): 323–360. doi: 10.1023/A:1011178417620
- [23] Dolbow J, Harari I. An efficient finite element method for embedded interface problems. *International Journal for Numerical Methods in Engineering* 2009; 78(2): 229–252. doi: 10.1002/nme.2486
- [24] Annavarapu C, Hautefeuille M, Dolbow JE. A robust Nitsche’s formulation for interface problems. *Computer Methods in Applied Mechanics and Engineering* 2012; 225–228: 44–54. doi: <https://doi.org/10.1016/j.cma.2012.03.008>
- [25] Burman E, Claus S, Hansbo P, Larson MG, Massing A. CutFEM: Discretizing geometry and partial differential equations. *International Journal for Numerical Methods in Engineering* 2015; 104(7): 472–501. doi: 10.1002/nme.4823
- [26] Glowinski R, He Q. A Least-Squares/Fictitious Domain Method for Linear Elliptic Problems with Robin Boundary Conditions. *Communications in Computational Physics* 2011; 9(3): 587–606. doi: 10.4208/cicp.071009.160310s
- [27] Kuberly P, Bochev P, Peterson K. An Optimization-Based Approach for Elliptic Problems with Interfaces. *SIAM Journal on Scientific Computing* 2017; 39(5): S757–S781. doi: 10.1137/16m1084547
- [28] Polyanin AD. *Handbook of linear partial differential equations for engineers and scientists*. CRC Press/C & H. first ed. 2002.
- [29] Rudin LI, Osher S, Fatemi E. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena* 1992; 60(1): 259 - 268. doi: 10.1016/0167-2789(92)90242-F
- [30] Tröltzsch F. On Finite Element Error Estimates for Optimal Control Problems with Elliptic PDEs. In: Lirkov I, Margenov S, Waśniewski J., eds. *Large-Scale Scientific Computing* Springer Berlin Heidelberg; 2010; Berlin, Heidelberg: 40–53.
- [31] Hansen P, O’Leary D. The Use of the L-Curve in the Regularization of Discrete Ill-Posed Problems. *SIAM Journal on Scientific Computing* 1993; 14(6): 1487–1503. doi: 10.1137/0914086
- [32] Scherzer O. The use of Morozov’s discrepancy principle for Tikhonov regularization for solving nonlinear ill-posed problems. *Computing* 1993; 51(1): 45–60. doi: 10.1007/BF02243828
- [33] Arnold D. An Interior Penalty Finite Element Method with Discontinuous Elements. *SIAM Journal on Numerical Analysis* 1982; 19(4): 742–760. doi: 10.1137/0719052
- [34] Shahbazi K. An explicit expression for the penalty parameter of the interior penalty method. *Journal of Computational Physics* 2005; 205(2): 401–407. doi: 10.1016/j.jcp.2004.11.017
- [35] Houston P, Schwab C, Süli E. Discontinuous hp-Finite Element Methods for Advection-Diffusion-Reaction Problems. *SIAM Journal on Numerical Analysis* 2002; 39(6): 2133–2163. doi: 10.1137/S0036142900374111
- [36] Yan J, Hicken JE. Immersed Boundary Method as an Inverse Problem. In: 2018 Fluid Dynamics Conference, AIAA AVIATION Forum. ; 2018; Atlanta, Georgia. AIAA–2018–4162
- [37] Biros G, Ghattas O. Parallel Lagrange-Newton-Krylov-Schur Methods for PDE-Constrained Optimization. Part I: The Krylov-Schur Solver. *SIAM Journal on Scientific Computing* 2005; 27(2): 687–713. doi: 10.1137/S106482750241565X
- [38] Borzi A, Schulz V. *Computational Optimization of Systems Governed by Partial Differential Equations*. Society for Industrial and Applied Mathematics . 2011
- [39] Akçelik V, Biros G, Ghattas O, Hill J, Keyes D, Bloemen Waanders vB. Parallel Algorithms for PDE-Constrained Optimization. In: Heroux MA, Raghavan P, Simon HD., eds. *Parallel Processing for Scientific Computing* Society for Industrial and Applied Mathematics. 2006 (pp. 291–322)
- [40] Roache PJ. Code Verification by the Method of Manufactured Solutions. *Journal of Fluids Engineering* 2001; 124(1): 4–10. doi: 10.1115/1.1436090