INTERNATIONAL JOURNAL OF
NUMERICAL ANALYSIS AND MODELING
Volume 21, Number 3, Pages 353–374

doi: 10.4208/ijnam2024-1014

A DIRECT METHOD FOR SOLVING THREE-DIMENSIONAL ELLIPTIC INTERFACE PROBLEMS

KUMUDU GAMAGE[1,*], YAN PENG[2], AND ZHILIN LI[3]

Abstract. This paper presents a direct method for efficiently solving three-dimensional elliptic interface problems featuring piecewise constant coefficients with a finite jump across the interface. A key advantage of our approach lies in its avoidance of augmented variables, distinguishing it from traditional methods. The computational framework relies on a finite difference scheme implemented on a uniform Cartesian grid system. By utilizing a seven-point Laplacian for grid points away from the interface, our method only requires coefficient modifications for grid points located near or on the interface. Numerical experiments validate our method's effectiveness. Generally, it achieves second-order accuracy for both the solution and its gradient, measured in the maximum norm, particularly effective in scenarios with moderate coefficient jumps. Extending and building upon the recent work of [1] on 1D and 2D elliptic interfaces, our approach successfully introduces a simpler method for extension into three dimensions. Notably, our proposed method not only offers efficiency and accuracy but also enhances the simplicity of implementation, making it accessible to non-experts in the field.

Key words. Piecewise constant coefficients with a finite jump, elliptic interface problems, finite difference scheme.

1. Intrdoction

Interface problems have gained significant attention due to their extensive practical applications. Real-world examples of interface problems span diverse domains, including fluid dynamics where phenomena such as bubble formation have been studied [2], electromigration of voids [3], glacier prediction [4], growth of internal blood clots [5], and thermodynamics encompassing heat propagation in distinct materials. Additionally, these problems extend to areas related to Stefan problems, crystal growth [6], and various other applications.

In this paper, we consider the elliptic interface problem of the form,

$$\nabla \cdot (\beta(\mathbf{x})\nabla u(\mathbf{x})) = f(\mathbf{x}), \quad \mathbf{x} \in \Omega \backslash \Gamma, \tag{1}$$

$$[u](\mathbf{X}) = w(\mathbf{X}), \quad \mathbf{X} \in \Gamma, \tag{2}$$

$$[\beta u_{\boldsymbol{n}}](\mathbf{X}) = v(\mathbf{X}), \quad \mathbf{X} \in \Gamma, \tag{3}$$

with given boundary conditions on $\partial\Omega$, where, Γ is a smooth interface in the domain Ω and interface Γ divides the domain Ω into two subdomains Ω^+ and Ω^- and therefore, $\Omega = \Omega^+ \cup \Omega^- \cup \Gamma$. See Figure 1 for an illustration. $\mathbf{X}$ is a point on the interface Γ, $\mathbf{x}$ is a point in Ω and $\boldsymbol{n}$ is the unit outward normal vector to the interface at the point $\mathbf{X}$. The superscript $+$ or $-$ denotes the limiting value of a function from one side or the other of the interface. Here, $[u] = [u](\mathbf{X}) = u^+(\mathbf{X}) - u^-(\mathbf{X})$ is the jump in the solution at $\mathbf{X}$ and $u_{\boldsymbol{n}} = \boldsymbol{n}.\nabla u = \frac{\partial u}{\partial \boldsymbol{n}}$ is the normal derivative of the solution u. In many applications, the coefficient β often takes the form of a piecewise constant value, while the source term f may have discontinuities across the interface Γ. The jumps in the solution (2) and the flux (3), along with

Received by the editors on August 28, 2023 and, accepted on February 22, 2024.
2000 *Mathematics Subject Classification.* 35R35, 49J40, 60G40.
*Corresponding author.

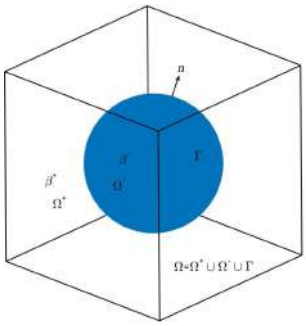

FIGURE 1. A diagram of a cubic domain Ω with a smooth interface Γ, where $\mathbf{n}$ represents the unit outward normal vector to the interface Γ.

the boundary conditions on $\partial\Omega$, are typically guided by the underlying physical principles.

Over time, substantial progress has been made in the development of numerical methods for solving interface problems [7, 8, 9, 10, 11, 12, 13, 14]. Strategies encompass the use of body-fitted grids [15, 16, 17], or the more favored Cartesian grids [12, 18, 19]. The latter choice gains prominence due to its simplified grid generation process, which is especially crucial when interfaces undergo frequent shape changes, often encountered in various physical phenomena. The preference for Cartesian grids is also amplified by the availability of versatile software tools such as fast Poisson solvers [20], Clawpack [21], Amrclawpack [22], the level set method [23, 24, 25], structured multigrid solvers [26, 27], and the immersed boundary method [10], along with others [28].

In Cartesian grid systems, interfaces are often embedded within a rectangular domain (in 2D) or a cube (in 3D). The immersed interface method (IIM), pioneered by LeVeque and Li [29], has emerged as a popular approach among the numerical community. Being the first second-order method for solving interface problems [30], IIM has been successfully applied to diverse linear and nonlinear problems, including hyperbolic elliptic systems [31], elasticity systems [32], [33], Hele-Shaw flow [34], traffic flow [35], glacier prediction [4], simulations of porosity evolution in chemical vapor infiltration [36], and shape identification in inverse problems [37].

However, while the IIM capably captures the solution and its gradient in the L^∞ norm for elliptic interface problems featuring variable coefficient β in various applications, it faces challenges when dealing with numerical examples characterized by significant jump discontinuities in the coefficient β. In such cases, the resulting linear system often becomes ill-conditioned, leading to potential non-convergence or inaccurate outcomes [38].

To address the aforementioned convergence issues, a fast immersed interface method (FIIM), also known as an augmented method, was introduced in [19]. This method involves a preconditioning step for the elliptic equation before applying the original IIM. Furthermore, it introduces an intermediate function to account for the jump in the normal derivative across the interface, enhancing the utilization of fast Poisson solvers. Although FIIM enhances accuracy, its implementation comes with complexities, notably involving establishing a Schur complement system, which adds computational overhead. Consequently, a novel direct IIM [1]