DISCRETE AND CONTINUOUS
DYNAMICAL SYSTEMS SERIES B
Volume **17**, Number **4**, June **2012**

doi:10.3934/dcdsb.2012.17.1185

pp. **1185–1203**

ERROR ESTIMATION FOR IMMERSED INTERFACE SOLUTIONS

Ben A. Vanderlei

Department of Mathematics, University of British Columbia
1984 Mathematics Road
Vancouver, BC V6T 1Z2, Canada

Matthew M. Hopkins

Nanoscale and Reactive Processes
Sandia National Laboratories
Albuquerque, NM 87185-0836, USA

Lisa J. Fauci

Department of Mathematics, Tulane University
6823 St. Charles Ave.
New Orleans, LA 70118, USA

Abstract. We present an error estimation method for immersed interface solutions of elliptic boundary value problems. As opposed to an asymptotic rate that indicates how the errors in the numerical method converge to zero, we seek *a posteriori* estimates of the errors, and their spatial distribution, for a given solution. Our estimate is based upon the classical idea of defect corrections, which requires the application of a higher-order discretization operator to a solution achieved with a lower-order discretization. Our model problem will be an elliptic boundary value problem in which the coefficients are discontinuous across an internal boundary.

1. **Introduction.** As computing technology continues to progress, the scale and complexity of systems that can be simulated continues to increase. Modern advances in the numerical methods for partial differential equations allow for the solution of nonlinear problems with discontinuities across interfaces. In some settings, computational simulations have begun to replace physical experiments. When important decision-making is based on simulations, validation and verification is critical [9]. While validation provides assurance that a given system is being modeled correctly (solving the "correct equations"), verification ensures that the equations are being "solved correctly". Within the verification framework, one must address aspects such as code verification, solution verification, and error estimation [9, 12]. Error estimation techniques seek to quantify the numerical error associated with a given solution. While error estimates may be used to improve a computed solution, providing error estimates of a portion of a large multiphysics code is still an important component of verification.

2000 *Mathematics Subject Classification.* Primary: 65N22, 65N06; Secondary: 82B24.

Key words and phrases. Immersed interface, error estimation, defect correction, discontinuous coefficients.

Here, we develop an error estimation method for numerical solutions of elliptic equations based upon the classical idea of defect corrections [15, 8]. The basic ingredient of the defect correction approach is to apply a higher-order discretization of the differential operator to a solution obtained using a lower-order discretization, and to use this information to correct the lower-order solution. Although this idea is not new, we demonstrate that it may be used effectively to estimate errors in solutions of elliptic problems with discontinuous coefficients computed using the immersed interface method [7, 2]. The immersed interface method embeds the interface in a regular domain that is discretized by a Cartesian grid. The defining feature of the immersed interface method is the sharp enforcement of jumps at the interface by modifying finite difference discretizations at grid points that fall near the interface. These modifications rely on knowledge of the solution behavior at the interface (i.e., jump conditions). Away from the interface standard discretization schemes are used. Hence, in the absence of an interface, the immersed interface method reduces to a standard numerical method for a given problem.

Interface problems are central to many scientific applications such as simulation of composite materials and fluid-structure interactions. Such physical systems naturally lead to equations whose parameters are discontinuous across an interface, and may also have singular source terms supported along the interface. In the past decades, numerical approaches such as immersed boundary methods (e.g. [10]) boundary integral methods (e.g. [11]) and immersed interface methods (e.g. [7]) have been successfully developed for these interface problems. These approaches are now typically used as components of large-scale simulations that, for instance, couple fluid flow, chemical transport and reaction to other relevant phenomena. In the interest of verification, it is important to provide tools that can be used in conjunction with these large-scale simulations that monitor the error associated with each component.

We present an error estimation method that, unlike Richardson extrapolation, does not require solutions computed on different grids. Our work was motivated by a related single-grid discretization error estimator, the *method of nearby problems*, which relies upon accurate curve fitting to the numerical solution [13, 14]. Instead, we rely upon the application of a higher-order discretization to our lower-order computed solution. We present an outline of this method, and, for a very simple one-dimensional problem, examine why the error estimate is valid. We then apply the error estimate to the numerical solution of a 2D elliptic problem with continuous coefficients. Finally, we demonstrate the effectiveness of this error estimate, within the context of the immersed interface method, on a model 2D elliptic problem with discontinuous coefficients across an interface. Although we focus on a model problem, we view this work as a starting point for error estimation of immersed interface solutions used in broader contexts, such as transient flow problems with moving interfaces governed by the Navier-Stokes equations [5, 16].

2. **Defect correction estimate.**

Consider the linear elliptic boundary value problem:

$$Lu = f \tag{1}$$

in some domain Ω with appropriate boundary conditions, and the finite difference discretization:

$$L_h u_h = f_h \tag{2}$$

which is solved in some manner to obtain a numerical approximation u_h to the true solution u_R restricted to the grid.

We outline the error estimation procedure:

1. Solve the discretized problem $L_h u_h = f_h$.
2. Construct a second discretization of the operator $\tilde{L}_h$.
3. Apply $\tilde{L}_h$ to the computed solution u_h to get a source term $s_h = \tilde{L}_h u_h - f_h$.
4. Solve the problem $L_h v_h = f_h + s_h$.
5. Estimate the discretization error $e_h = u_R - u_h$ by the quantity $\hat{e}_h = u_h - v_h$.

The goal of this procedure is to produce an error estimate, but we point out that this estimate, available at each grid point, may be used to improve the computed solution. In order to examine why this is an error estimate, we define the local truncation errors τ_h and $\tilde{\tau}_h$ of L_h and $\tilde{L}_h$ respectively:

$$L_h u_R = f_h + \tau_h \tag{3}$$

$$\tilde{L}_h u_R = f_h + \tilde{\tau}_h \tag{4}$$

We see that:

$$\begin{aligned}
L_h v_h &= f_h + s_h \\
&= \tilde{L}_h u_h && \text{(definition of } s_h) \\
&= \tilde{L}_h L_h^{-1} f_h && (u_h \text{ by } 2) \\
&= \tilde{L}_h (u_R - L_h^{-1} \tau_h) && \text{(by 3)} \\
&= f_h + \tilde{\tau}_h - \tilde{L}_h L_h^{-1} \tau_h && \text{(by 4)} \\
&= L_h u_R - \tau_h + \tilde{\tau}_h - \tilde{L}_h L_h^{-1} \tau_h && \text{(by 3)}
\end{aligned}$$

so that

$$v_h = u_R - L_h^{-1} \tau_h + L_h^{-1} \tilde{\tau}_h - L_h^{-1} \tilde{L}_h L_h^{-1} \tau_h$$

The estimate of the error can then be written:

$$\begin{aligned}
\hat{e}_h = u_h - v_h &= u_h - (u_R - L_h^{-1} \tau_h + L_h^{-1} \tilde{\tau}_h - L_h^{-1} \tilde{L}_h L_h^{-1} \tau_h) \\
&= -(u_R - u_h) + L_h^{-1} \tau_h - L_h^{-1} \tilde{\tau}_h + L_h^{-1} \tilde{L}_h L_h^{-1} \tau_h \\
&= L_h^{-1} \tilde{L}_h L_h^{-1} \tau_h - L_h^{-1} \tilde{\tau}_h \\
&= L_h^{-1} \tilde{L}_h (u_R - u_h) - L_h^{-1} \tilde{\tau}_h \\
&= L_h^{-1} \tilde{L}_h (e_h) - L_h^{-1} \tilde{\tau}_h
\end{aligned}$$

In this calculation we have used the fact that $L_h(e_h) = L_h(u_R - u_h) = \tau_h$.

Therefore, $\hat{e}_h = u_h - v_h$ will be a good approximation of the true error provided that $||L_h^{-1} \tilde{\tau}_h||$ is small and $L_h^{-1} \tilde{L}_h e_h \approx e_h$. It is natural to choose $\tilde{L}_h$ to be a higher-order discretization of L, and we do so from this point forward.

In the discussion of the quality of these error estimates, we concern ourselves with the difference between the true error and the error estimate:

$$\hat{e}_h - e_h = \underbrace{L_h^{-1} (\tilde{L}_h L_h^{-1} - I) \tau_h}_{E_1} - \underbrace{L_h^{-1} \tilde{\tau}_h}_{E_2}. \tag{5}$$

We separate the above into two terms, E_1 and E_2, because for a simple one-dimensional test problem, we will analyze these separately. If $||E_1||$ and $||E_2||$ are small relative to $||e_h||$, the estimate $\hat{e}_h$ will be useful. For the one-dimensional test problem presented below, we can bound these quantities analytically. For elliptic boundary value problems in two dimensions, we examine the difference $\hat{e}_h - e_h$ computationally. Our calculations demonstrate that, for smooth problems, the order of convergence of $||\hat{e}_h - e_h||$ to zero is higher than the order of convergence of $||e_h||$ to zero. For problems with a discontinuous coefficient, we find that $||\hat{e}_h - e_h||$ approaches zero at the same rate as $||e_h||$.

One-dimensional example. We consider now a simple one-dimensional problem where the error in the error estimate in (5) may be explicitly analyzed. We choose the problem $u_{xx} = f$ for $x \in (0, 1)$ with boundary data $u(0)$ and $u(1)$ given. We use as L_h the standard second-order discretization:

$$L_h = \frac{1}{h^2}\begin{pmatrix} -2 & 1 & & & \\ 1 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2 & 1 \\ & & & 1 & -2 \end{pmatrix}$$

For $\tilde{L}_h$ we choose the fourth-order discretization:

$$\tilde{L}_h = \frac{1}{12h^2}\begin{pmatrix} -20 & 6 & 4 & -1 & & & & \\ 16 & -30 & 16 & -1 & & & & \\ -1 & 16 & -30 & 16 & -1 & & & \\ & \ddots & \ddots & \ddots & \ddots & \ddots & & \\ & & \ddots & \ddots & \ddots & \ddots & \ddots & \\ & & & -1 & 16 & -30 & 16 & -1 \\ & & & & -1 & 16 & -30 & 16 \\ & & & & -1 & 4 & 6 & -20 \end{pmatrix}$$

For this simple problem, L_h^{-1} can be explicitly computed. The columns of the inverse are solutions g_j to the equation:

$$L_h g_j = e_j \tag{6}$$

where e_j is the jth basis vector. This problem can be interpreted as a discretization of the continuous problem for the Green's function [3]. Let us recall the definition the Green's function for this one-dimensional boundary value problem.

$$G(x; x_0)'' = \delta(x - x_0) \tag{7}$$

where $\delta(x - x_0)$ is the Dirac delta function centered at x_0 and G is the Green's function, with zero boundary conditions at the endpoints of the interval:

$$G(x; x_0) = \begin{cases} (x_0 - 1)x & x \le x_0 \\ (x - 1)x_0 & x > x_0 \end{cases} \tag{8}$$

If we wish to approximate the solution to (7) numerically, we could replace the delta function with an approximation, such as the piecewise linear function shown

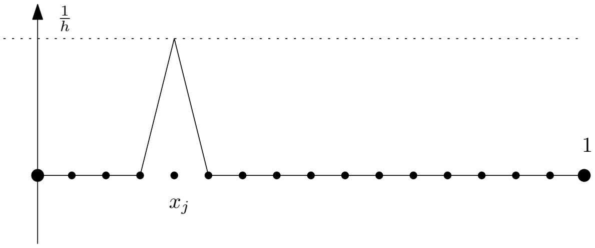


FIGURE 1. The piecewise linear “hat” function is a common choice of functions to replace the Dirac delta function when solving a problem numerically.

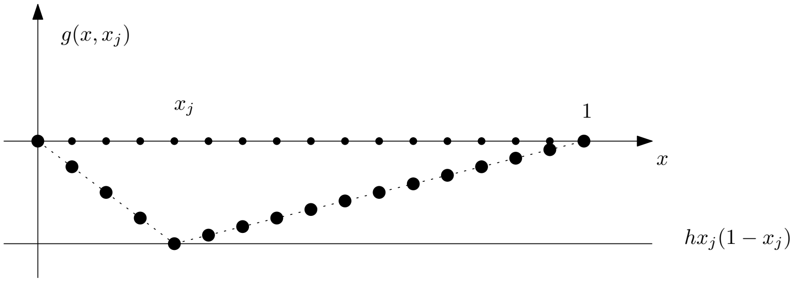


FIGURE 2. The solution to (6) is the Green’s function for the continuous problem, restricted to the grid and scaled by h.

in Figure 2. Note that the integral of this function is one, and it is zero at all the grid points except x_j, where it takes the value $\frac{1}{h}$. This gives us the system

$$L_h g_h = \frac{1}{h} e_j \tag{9}$$

The solution to this problem is *exactly* $G(x; x_j)$ evaluated at the grid points. That is, $G_R - g_h = 0$, where G_R is $G(x; x_j)$ restricted to the grid. The solution g_j to (6) then is hG_R as shown in Figure 2.

$$(L_h^{-1})_j = g_j = \begin{cases} h(x_j - 1)x_i & x_i \le x_j \\ h(x_i - 1)x_j & x_i > x_j \end{cases} \tag{10}$$

We can now use this explicit computation of L_h^{-1} to show that $||E_1||$ and $||E_2||$ in (5) are small compared to the error $||e_h||$. We first consider $||E_2|| = ||L_h^{-1}\tilde{\tau}_h|| \le ||L_h^{-1}||\, ||\tilde{\tau}_h||$. Since each component of the matrix L_h^{-1} is $O(h)$, and the dimension of this matrix is $N \times N$, where $N = h^{-1}$, the infinity norm $||L_h^{-1}||_\infty$ is O(1). Given that $\tilde{\tau}_h$ is the $O(h^4)$ truncation error of the higher-order scheme, we have $||E_2||_\infty = O(h^4)$.

Next we examine $||E_1||_\infty$. We note that the jth column of $\tilde{L}_h L_h^{-1}$ is just the application of the higher-order finite difference formula for the second derivative applied to the function given in (10). Since the finite difference formula reflects a five-point stencil, and this function is piecewise linear, the resulting product is tridiagonal, apart from the extra entries due to boundary corrections in the first and last rows. We have:

$$\tilde{L}_h L_h^{-1} - I = \frac{-1}{12}\begin{pmatrix} 1 & -2 & 1 & & & \\ 1 & -2 & 1 & & & \\ & \ddots & \ddots & \ddots & & \\ & & 1 & -2 & 1 \\ & & 1 & -2 & 1 \end{pmatrix}$$

Note that:

$$\frac{-12}{h^2}(\tilde{L}_h L_h^{-1} - I) = L_h + B$$

where

$$B = \frac{1}{h^2}\begin{pmatrix} 3 & -3 & 1 & & \\ 0 & 0 & 0 & & \\ & \ddots & \ddots & \ddots & \\ & & 0 & 0 & 0 \\ & & 1 & -3 & 3 \end{pmatrix}$$

We see that $E_1 = \frac{-h^2}{12}(\tau_h + L_h^{-1}B\tau_h)$. We know that $\tau_h = O(h^2)$ so the final detail is to determine the order of $L_h^{-1}B\tau_h$. Although the entries in the matrix B are of order $O(\frac{1}{h^2})$, only the first and last rows are nonzero. The product $B\tau_h$ will be a vector of zeros except for the first and last entry, which are $O(1)$ since $\tau_h = O(h^2)$. Since the entries of L_h^{-1} are $O(h)$, the entries of $L_h^{-1}B\tau_h$ are $O(h)$. Hence, $||E_1||_\infty = O(h^3)$.

In summary, we have shown that for this simple problem, the difference between the true error and the error estimate $||\hat{e}_h - e_h||_\infty$ is $O(h^3)$.

3. Application to Poisson problem.

Consider the Poisson problem:

$$\Delta u = f \qquad (11)$$

on the square domain $\Omega = [\text{-}1,1] \text{ x } [\text{-}1,1]$ with Dirichlet boundary conditions. For the discrete operator L_h we use the standard second-order, 5-point difference scheme whose stencil is shown in Figure 3(a) applied on a regular NxN grid. For the higher-order $\tilde{L}_h$ we use the stencil and coefficients shown in Figure 3(b), but for grid points adjacent to the boundary, we use the stencil shown in Figure 3(c). While the discretization away from these points is fourth order, the local truncation error at these boundary grid points is $O(h^3)$.

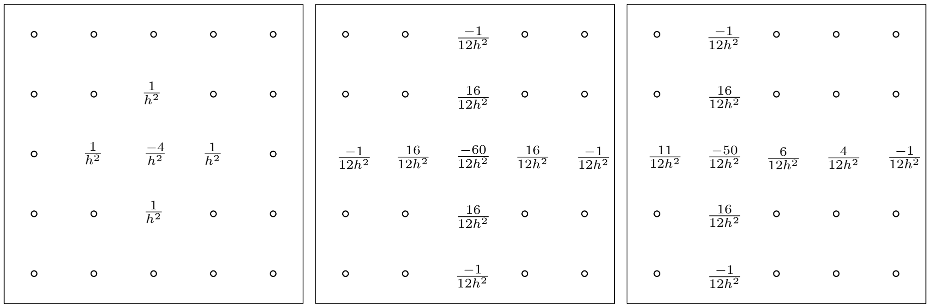
We choose as our test problem the function:

$$f(x,y) = -\pi^2\cos(\pi r) - \frac{\pi\sin(\pi r)}{r}$$

$$r = \sqrt{x^2 + y^2}$$

The boundary data on the square is specified so that the true the solution is $u = \cos(\pi r)$.

Figure 4 shows norms of the errors $||e_h||_\infty$ in the solution to the Poisson problem computed using the second order operator L_h for different values of the grid size N. Also shown are the norms of the error estimates $||u_h - v_h||_\infty = ||\hat{e}||_\infty$, as well as the difference between the true error and the estimated error. These are plotted on a log-log scale, and the slopes of the linear fits to these points (shown on the figure) indicate the order of convergence. We see that, indeed, the errors e_h using the discretization L_h are second order. We also see that the difference between the


$\frac{1}{h^2}$
$\frac{1}{h^2}$ $\frac{-4}{h^2}$ $\frac{1}{h^2}$
$\frac{1}{h^2}$
$\frac{-1}{12h^2}$
$\frac{16}{12h^2}$
$\frac{-1}{12h^2}$ $\frac{16}{12h^2}$ $\frac{-60}{12h^2}$ $\frac{16}{12h^2}$ $\frac{-1}{12h^2}$
$\frac{16}{12h^2}$
$\frac{-1}{12h^2}$
$\frac{-1}{12h^2}$
$\frac{16}{12h^2}$
$\frac{11}{12h^2}$ $\frac{-50}{12h^2}$ $\frac{6}{12h^2}$ $\frac{4}{12h^2}$ $\frac{-1}{12h^2}$
$\frac{16}{12h^2}$
$\frac{-1}{12h^2}$

(a) standard 5 point stencil (b) wide 9 point stencil (c) boundary stencil

FIGURE 3. Finite difference formulas used to solve the Poisson problem numerically and produce an estimate of the error in this solution.

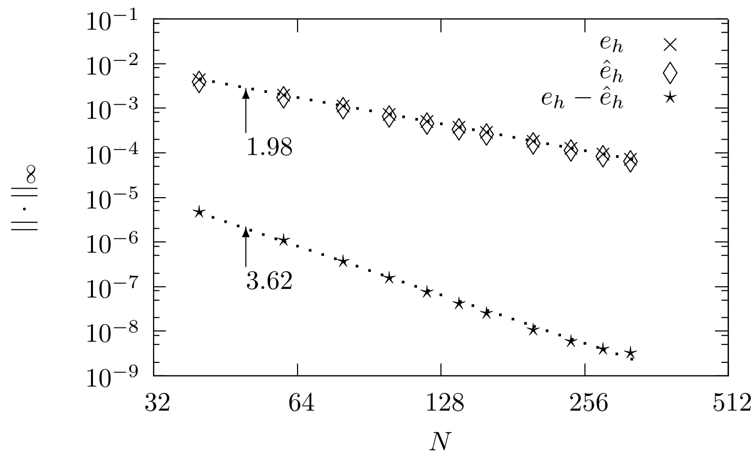


FIGURE 4. Log-log plot of the error in the numerical solution of the Poisson problem $e_h = u_h - u_R$, the error estimate $\hat{e}_h = u_h - v_h$, and the difference between them. The slopes of the linear fits to these values are shown, indicating convergence rate. We see that this difference converges faster than the error.

true error and the error estimate $||e_h - \hat{e}_h||$ converges to zero faster than the error itself. In fact, the convergence here is better than $O(h^3)$, as was predicted in the simple one-dimensional example above. It is also important to note that the data shown is the maximum norm of the quantities. This method does not just give a scalar bound for the error, but a spatial distribution of the error estimate at each grid point.

We highlight two salient features of this error estimation approach:

- The error estimate we produce requires that the problem be solved a second time on the same grid.
- The operator $\tilde{L}_h$ is never "inverted", it is only applied to the solution u_h. This means that we need not be concerned if the matrix representing $\tilde{L}_h$ has a large bandwidth.

Here, we offer a comparison to the error estimates produced using this defect correction methodology to those produced using a standard Richardson Extrapolation. This requires the solution on two grids of discretization sizes h and $2h$. The formula that we use for the extrapolation error estimate is $\hat{e}_h = \frac{1}{3}(u_h - u_{2h})$. Figure 5 shows a comparison of these two error estimates as a function of grid size for this Poisson test problem. We see that the two estimates are comparable at all the grid levels tested, with the error in the defect correction estimate approximately half of the error in the extrapolation estimate. The extrapolation estimate requires a second solution on a coarser grid, while the defect correction estimate requires a second solution on the same grid, and hence requires more work. However, the extrapolation estimate is only computed on the coarse grid, while the defect correction estimate is available at each point of the original grid.

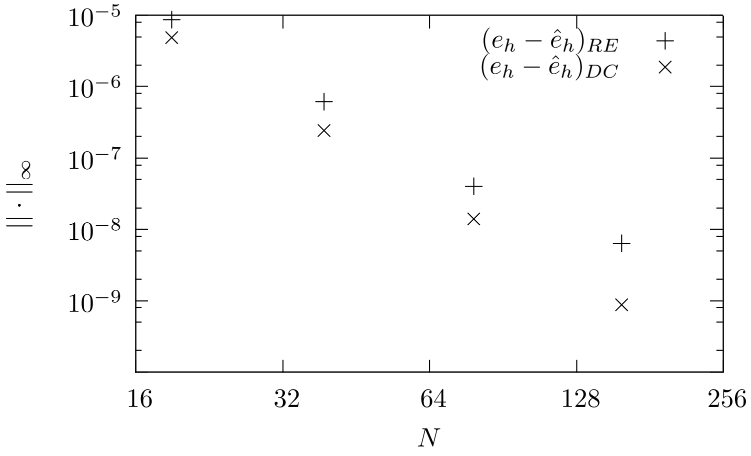


FIGURE 5. The difference between the true error and the error estimate provided by Richardson extrapolation, to be compared with the difference between the true error and the error estimate provided by our defect correction method.

4. **Interface problem and numerical solution.** Next we consider the elliptic interface problem

$$\nabla \cdot (\beta(x,y)\nabla u) = f \tag{12}$$

on the domain $\Omega =$ [-1,1] x [-1,1] with Dirichlet boundary conditions. We will take the circle given by $x^2 + y^2 = 1/4$ as the interface between the subdomains Ω^- and Ω^+. Figure 6 shows this geometry. In each of the subdomains β will be constant.

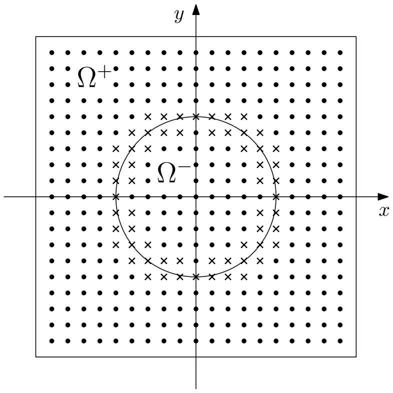


FIGURE 6. Geometry of interface problem with irregular points indicated. Note that the interface does not coincide with grid points.

The following conditions on the solution u at the interface can be derived from (12):

$$u^+ = u^- \tag{13}$$

$$(\beta u_n)^+ = (\beta u_n)^- \tag{14}$$

The superscripts (+) and (-) here and throughout the discussion are to denote limiting values of the quantities as one approaches the interface from within Ω^+ and Ω^- respectively.

The Immersed Interface Method (IIM) [2] embeds the interface on a regular Cartesian grid and uses a standard discretization of the problem for points that are not near the interface. We choose the second-order discretization of the Laplacian using the 5-point scheme (Figure 3(a)) at any point in the domain for which the stencil does not cross the interface. These points will be referred to as regular points and the coefficients in the discretization will be referred to as the standard coefficients. At the other grid points, termed irregular points, information about the problem is used to derive a special finite difference equation. The irregular points for a sample grid in Figure 6 are marked.

We treat the discretization at irregular points in a manner closely following the immersed interface implementation described by Li and Ito in [6], and refer the reader to the detailed derivation therein. At irregular grid points, we choose the 9-point stencil shown in Figure 7(a). The finite difference discretization at these irregular point (x_i, y_j) is:

$$\sum_{k=1}^{9} c_k U_k = f_{i,j} \tag{15}$$

where U_k are the unknown solution values at the grid points, c_k are the coefficients to be determined, and $f_{i,j} = f(x_i, y_j)$. In order to eliminate leading order terms in the truncation errors, the following six constraints, as presented in [2, 6], must be satisfied:

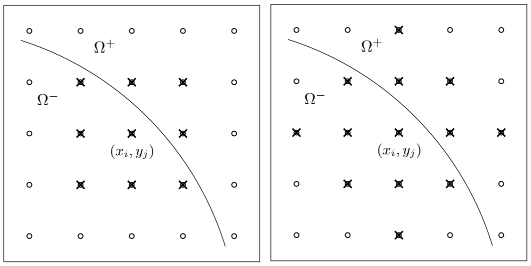
$$\begin{aligned}
a_1 + a_2 &= 0\\
a_3 + \rho a_4 + a_8(\rho - 1)\chi'' + a_{10}(1-\rho)\chi'' &= 0\\
a_5 + a_6 + a_{12}(1-\rho)\chi'' &= 0\\
a_7 + \rho a_8 &= \beta^-\\
a_9 + a_{10} + a_8(\rho - 1) &= \beta^-\\
a_{11} + \rho a_{12} &= 0
\end{aligned}$$

Here $\rho = \beta^-/\beta^+$ and χ is a local representation of the interface. The a_k are defined in terms of the coefficients:

$$\begin{aligned}
a_1 &= \sum_{k\in K^-} c_k & a_2 &= \sum_{k\in K^+} c_k\\
a_3 &= \sum_{k\in K^-} c_k\xi_k & a_4 &= \sum_{k\in K^+} c_k\xi_k\\
a_5 &= \sum_{k\in K^-} c_k\eta_k & a_6 &= \sum_{k\in K^+} c_k\eta_k\\
a_7 &= \frac{1}{2}\sum_{k\in K^-} c_k\xi_k^2 & a_8 &= \frac{1}{2}\sum_{k\in K^+} c_k\xi_k^2\\
a_9 &= \frac{1}{2}\sum_{k\in K^-} c_k\eta_k^2 & a_{10} &= \frac{1}{2}\sum_{k\in K^+} c_k\eta_k^2\\
a_{11} &= \sum_{k\in K^-} c_k\xi_k\eta_k & a_{12} &= \sum_{k\in K^+} c_k\xi_k\eta_k
\end{aligned}$$

and η_k and ξ_k are the locations of the grid points in the stencil in the local coordinate system shown in Figure 8. The sums are over the grid points in Ω^+ or Ω^- as denoted by the index sets $K^\pm = \{k : (\xi_k, \eta_k) \in \Omega^\pm\}$ These constraints make use of the conditions (13) and (14).

At each irregular point, we must compute nine coefficients, and have thus far presented six constraints. The coefficients are computed by minimizing the difference between the unknown c_k and the standard coefficients in the least squares sense, subject to the above six constraints. (The standard coefficients are those of the five-point stencil overlayed on the nine-point stencil, with zeros as the coefficients



(a) 9 point stencil (b) 13 point stencil

FIGURE 7. Stencils used for L_h and $\tilde{L_h}$ at irregular point (x_i, y_j).

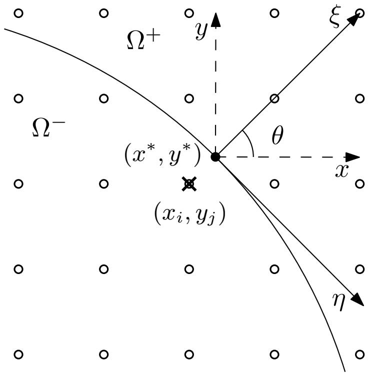


FIGURE 8. A local coordinate system is used to represent the interface and the interface conditions on the solution. $\xi = (x - x^*)\cos\theta + (y - y^*)\sin\theta$ and $\eta = -(x - x^*)\sin\theta + (y - y^*)\cos\theta$. The interface is represented locally as $\xi = \chi(\eta)$

of the extra four points). Here our immersed interface implementation departs from that described in [6] in that we do not enforce any sign restrictions on the unknown coefficients.

Before discussing error estimates, we use this immersed interface discretization on the test problem (12) on the geometry shown in Figure 6 with $f = -\pi^2\cos(\pi r) - \pi\sin(\pi r)/r$, $\beta^- = 1$, and $\beta^+ = B$, where $r = \sqrt{x^2 + y^2}$. The analytic solution is then $u^- = \cos(\pi r)$ and $u^+ = \cos(\pi r)/B$. Note that this problem reduces to the test problem in §3 when $B = 1$. Figures 9 and 10 show the errors in the solution of the above discretization for jumps of $B = 2$ and $B = 100$. Note that at regular points the discretization was second order, while at irregular points the discretization was first order. Nevertheless, we see overall second order convergence for both values of B, as was discussed in [6].

Construction of $\tilde{L}_h$. We next describe the construction of $\tilde{L}_h$, the higher-order version of the discretization just described. There has been recent progress in the development of higher-order IIM discretizations. Zhou et al. derive a related high-order matched interface and boundary method [18]. In [17], Zhong introduces new high-order interface difference formulas that are expressed in an explicit form. In both of these works, the methods are developed in one dimension and then extended to two dimensions by decomposing the discretization at each point into two related one-dimensional problems.

Here we present an alternate higher-order immersed interface discretization $\tilde{L}_h$. As discussed in §2, it is desirable for $\tilde{L}_h$ to be chosen so that $L_h^{-1}\tilde{L}_h e_h \approx e_h$. For this reason, we choose the fourth-order discretization at regular points to use the the wide 9-point stencil (Figure 3(b)). At irregular points, we will use a discretization that is $O(h^2)$, and will choose the 13-point stencil shown in Figure (7(b)).

In order to reduce the local truncation error to $O(h^2)$ at the irregular points, we must enforce another four constraints on the coefficients. We omit the calculation of further terms in the series expansion and simply provide the set of constraints used:

$$\begin{aligned}
a_1 + a_2 &= 0\\
a_3 + \rho a_4 + a_8(\rho-1)\chi'' + a_{10}(1-\rho)\chi'' + 3a_{14}(\rho-1)\chi''^2 + 3a_{20}(1-\rho)\chi''^2 &= 0\\
a_5 + a_6 + a_{12}(1-\rho)\chi'' + 3a_{16}(\rho-1)\chi''^2 + 3a_{18}(1-\rho)\chi''^2 &= 0\\
a_7 + \rho a_8 &= \beta^-\\
a_9 + a_{10} + a_8(\rho-1) + 3a_{14}(\rho-1)\chi'' + 3a_{20}(1-\rho)\chi'' &= \beta^-\\
a_{11} + \rho a_{12} + 3a_{16}(1-\rho)\chi'' + 3a_{18}(\rho-1)\chi'' &= 0\\
a_{13} + \rho a_{14} &= \beta^-\xi_1\\
a_{15} + a_{16} + a_{18}(\rho-1) &= \beta^-\eta_1\\
a_{17} + \rho a_{18} &= \beta^-\eta_1\\
a_{19} + \rho a_{20} &= \beta^-\xi_1
\end{aligned}$$

Here the a_k are as previously defined with the additional terms as follows:

$$\begin{aligned}
a_{13} &= \frac{1}{6}\sum_{k\in K^-} c_k\xi_k^3 & a_{14} &= \frac{1}{6}\sum_{k\in K^+} c_k\xi_k^3\\
a_{15} &= \frac{1}{6}\sum_{k\in K^-} c_k\eta_k^3 & a_{16} &= \frac{1}{6}\sum_{k\in K^+} c_k\eta_k^3\\
a_{17} &= \frac{1}{2}\sum_{k\in K^-} c_k\xi_k^2\eta_k & a_{18} &= \frac{1}{2}\sum_{k\in K^+} c_k\xi_k^2\eta\\
a_{19} &= \frac{1}{2}\sum_{k\in K^-} c_k\xi_k\eta_k^2 & a_{20} &= \frac{1}{2}\sum_{k\in K^+} c_k\xi_k\eta^2
\end{aligned}$$

Note that (ξ_1, η_1) are the local coordinates of the point (x_i, y_j). Also, we have neglected to write the terms involving higher order derivatives of χ since $\chi''' = 0$ for this particular geometry, but no additional difficulty arises by their inclusion.

As above, we compute the thirteen coefficients by minimizing the difference between the unknown c_k and the coefficients of the standard 9-point stencil used at regular points (Figure 3(b)), subject to the ten constraints listed above.

Before we make use of this new discretization scheme to compute error estimates, we will evaluate it as a method of direct solution by solving $\tilde{L}_h\tilde{u}_h = f_h$. Figures 9 and 10 also show the errors in the solution of the test problem using the higher-order discretization for jumps of $B = 2$ and $B = 100$. Note that at regular points the discretization was fourth order, while at irregular points the discretization was second order. We see that, overall, the convergence is third order. Zhong [17] also reports overall third-order convergence when using a second-order discretization at irregular points in conjunction with a fourth-order discretization at regular points.

Now that we have assembled the higher-order operator $\tilde{L}_h$, we use it to estimate the error in the immersed interface solution computed using L_h. Figures 11-13 show norms of the errors $||e_h||_\infty$ in the solution to the interface problem computed using the second order operator L_h for different values of the grid size N for different jump parameters ($B = 2, 10, 100$). Also shown are the norms of the error estimates $||u_h - v_h||_\infty = ||\hat{e}||_\infty$, as well as the difference between the true error and the estimated error. We see that the errors e_h using the discretization L_h are second order, as expected. For the milder jump discontinuities ($B = 2, 10$), the estimates

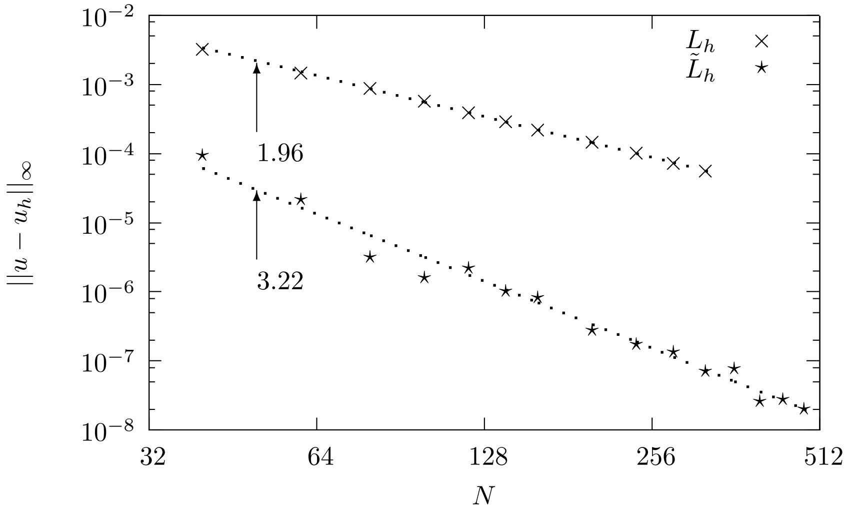


FIGURE 9. The error in the numerical solution of the model problem with $B = 2$ using low and high order schemes. The slopes of the linear fits to these values are shown, indicating convergence rate. Note that the numerical solution obtained using $\tilde{L}_h$ is not needed to construct the error estimates.

are uniformly reliable for all grids, but some oscillatory behavior is exhibited for the strong discontinuity $B = 100$. We also note that, unlike in the Poisson problem with continuous coefficients, there is no gain in the convergence rate of the estimate – the error in the estimate is converging to zero at the same rate as the true error.

Again we point out that these error estimates are grid functions and do reflect a spatial resolution of the error. Figure 14 shows surface plots of the true error, the estimated error, and the difference between these for the case $B = 100$ using a grid of size 40×40. We see that the estimate and the true error have similar profiles, and note that the error in the estimate is concentrated along the interface.

We next present some data reflecting the time required to compute these error estimates compared to the time required to find the solution. Table 1 shows the time required to compute the solution using the operator L_h for the case $B = 10$, the time required to compute the estimate, and the ratio of the estimation time to total time for solution plus estimate. All solutions were obtained using GMRES as an iterative solver with a tolerance of 10^{-12}. For the solution of the original problem we use zero as the initial iterate. However, for the second solution using the same operator L_h, we use the first solution as the initial iterate. For moderate grid resolution, the error estimation time saturates at about forty percent of total time. This is due to the improvement of the initial iterate for the linear solver.

Finally, we compare these error estimates based upon defect correction with those produced using Richardson extrapolation. Figure 15 shows a comparison of these two error estimates as a function of grid size for the test problem with $B = 10$. While the defect correction estimates are uniformly good for all grids, Richardson

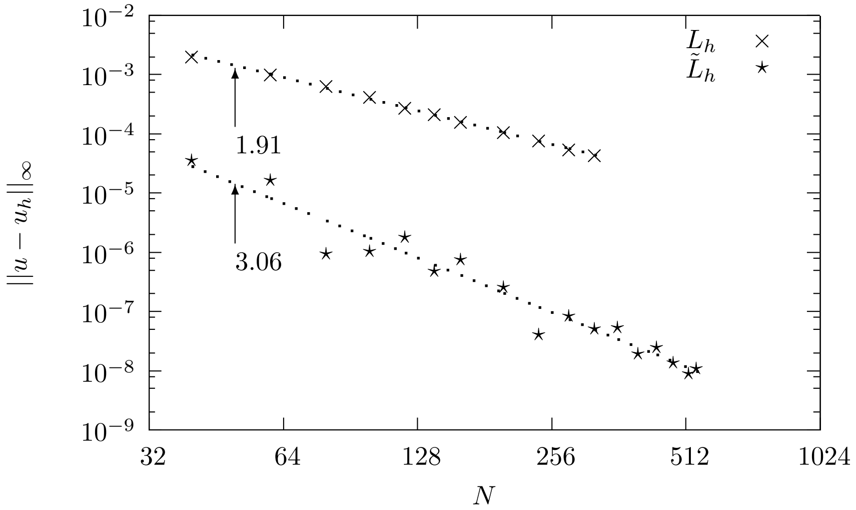


FIGURE 10. The error in the numerical solution of the model problem with $B = 100$ using low and high order schemes. The slopes of the linear fits to these values are shown, indicating convergence rate. Note that the numerical solution obtained using $\tilde{L}_h$ is not needed to construct the error estimates.

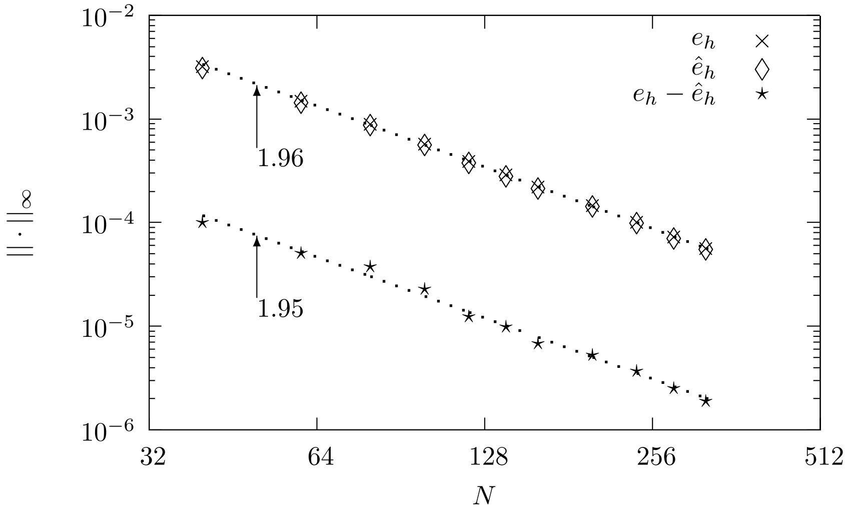


FIGURE 11. A plot of the errors and estimates for the test problem (12) with $B = 2$. The slopes of the linear fits to these values are shown, indicating convergence rate.

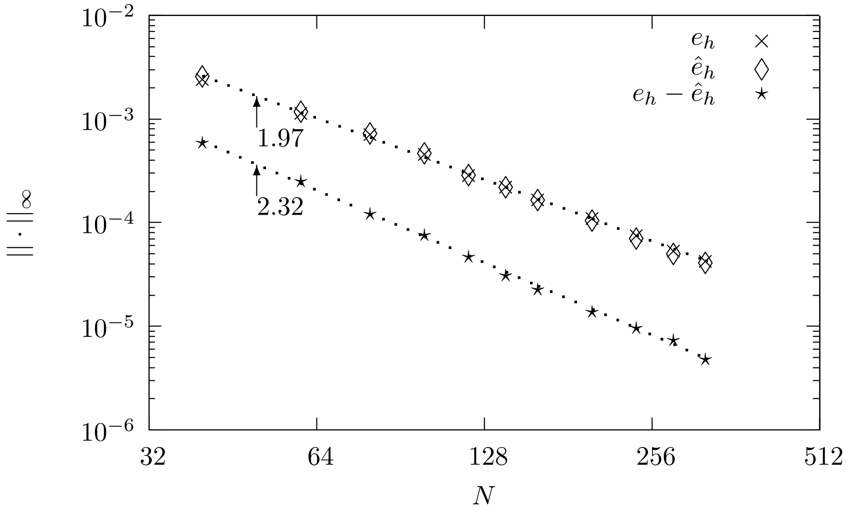


FIGURE 12. A plot of the errors and estimates for the test problem (12) with $B = 10$. The slopes of the linear fits to these values are shown, indicating convergence rate.

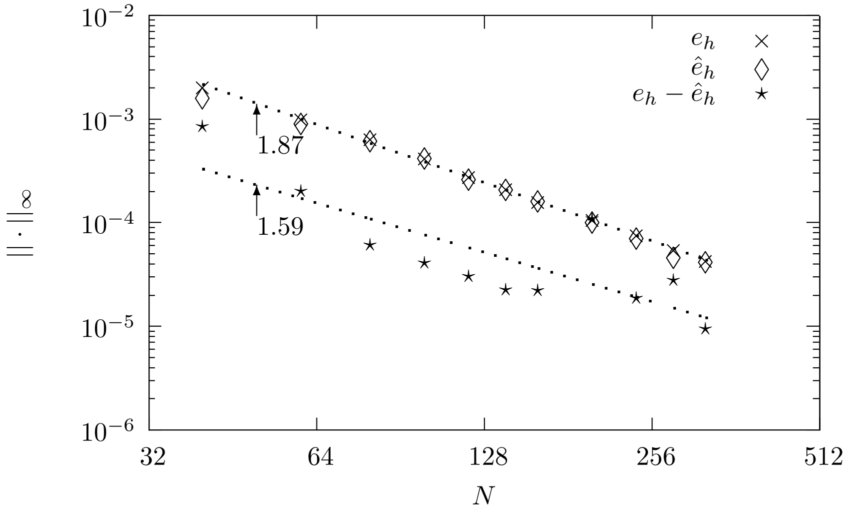
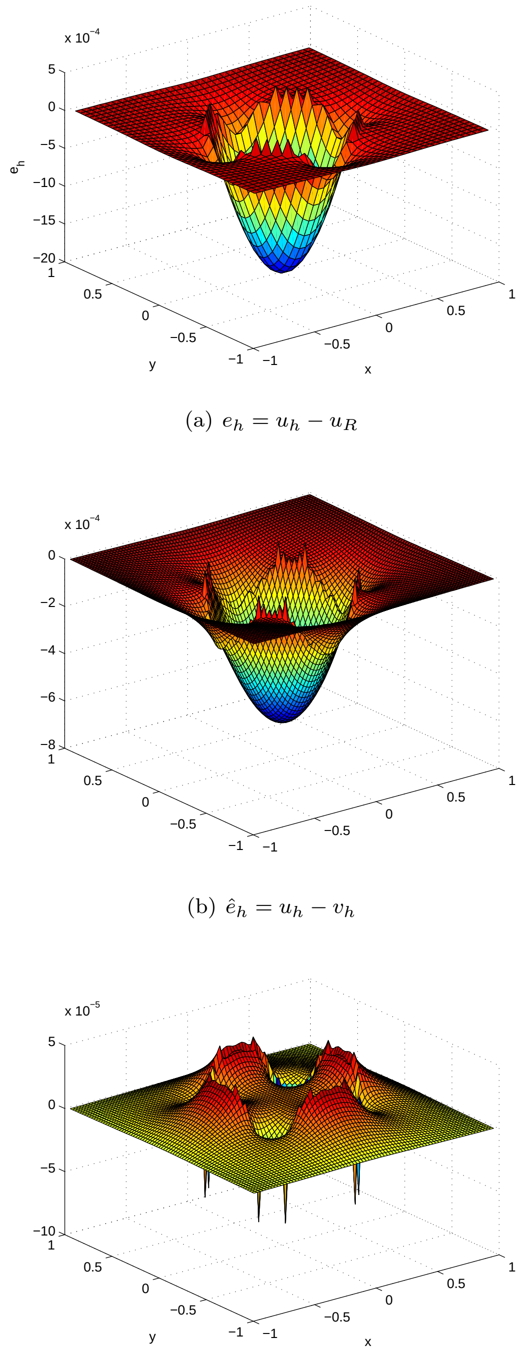


FIGURE 13. A plot of the errors and estimates for the test problem (12) with $B = 100$. The slopes of the linear fits to these values are shown, indicating convergence rate.

extrapolation is erratic. For some grid sizes, the extrapolation estimates are in line



(a) $e_h = u_h - u_R$

(b) $\hat{e}_h = u_h - v_h$

(c) $e_h - \hat{e}_h$

FIGURE 14. Spatial distribution of the error in the numerical solution e_h, the estimate of the error $\hat{e}_h = u_h - v_h$, and the difference between the two for test problem (12) with $B = 100$. These calculations were performed on a grid of size 40x40.

N	Time to compute solution	Time to compute estimate	$\frac{\text{Estimate time}}{\text{Total time}}$
40	0.45	0.47	0.51
80	6.52	4.39	0.40
160	85.25	51.4	0.37
200	175.41	127.91	0.42
250	347.62	232.01	0.40

TABLE 1. Runtimes (in seconds) and ratios of runtimes for the case $B = 10$.

with the defect correction estimates, but for other grid sizes, the estimates are on the order of the error itself.

Recall that these extrapolation estimates rely on solving the problem on a coarser grid, while the defect correction estimates rely on a second solution on the same grid. We believe that this distinction is responsible for the inconsistent error estimates by Richardson extrapolation in the presence of the interface. In fact, it was shown by Li [4] that the error in an immersed interface calculation may depend strongly upon the location of interface points with respect to the grid. If we keep in mind the goal of estimating the error on the original grid, we assert that an estimation method that uses only information on that grid will give more consistent results than one that relies on other grid calculations.

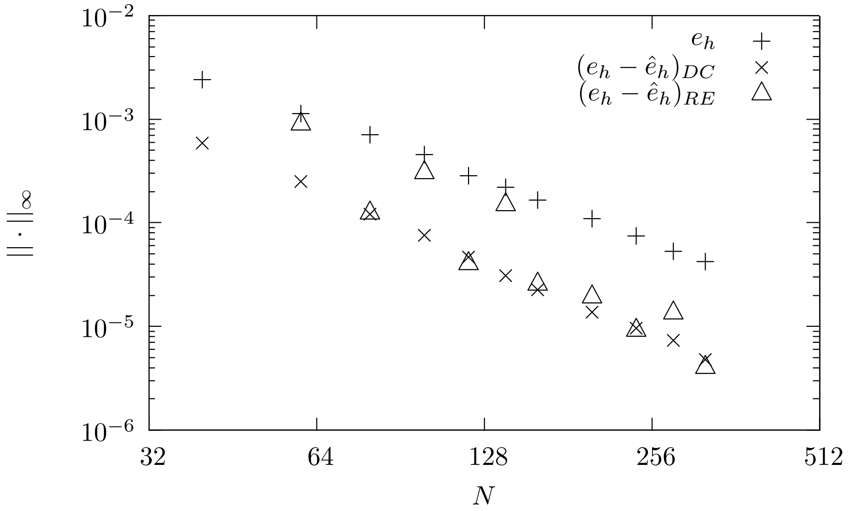


FIGURE 15. Comparison of our defect correction error estimates with Richardson extrapolation error estimates for the test problem (12) with $B = 10$.

5. **Conclusion.** We have revisited the classical idea of defect correction as a method to estimate errors associated with the immersed interface solution of elliptic boundary value problems with discontinuous coefficients. The value of this procedure is

two-fold: it can be used to improve the lower-order solution, as well as provide an error estimate for the lower-order solution. If it is used for correction, one would produce the improved solution $u_h^{new} = u_h + \hat{e}_h$. In this case, the error in the corrected solution u_h^{new} would be the difference $\hat{e}_h - e_h$. Our study here is concerned only with the estimation of the error in the lower-order solution. We believe that the construction of estimates on the original grid will be more reliable than estimates constructed on coarser grids, as in the case of Richardson extrapolation. We also view these calculations as a starting point for the development of an error estimation methodology that can be used in more general immersed interface computations. For instance, the method presented here for the elliptic problem may be readily extended to the corresponding parabolic problem to estimate the error in an implicit solution at a given time step. In addition, although our analysis of the error estimate relies upon the linearity of the model operator, ideas outlined in previous work on defect corrections for nonlinear elliptic equations (i.e. [1]) may be exploited.

Acknowledgments. Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

REFERENCES

[1] W. Auzinger, *Defect correction for nonlinear elliptic difference equations*, Numerische Mathematik, **51** (1987), 199–208.

[2] R. LeVeque and Z. Li, *The immersed interface method for elliptic equations with discontinuous coefficients and singular sources*, SIAM Journal on Numerical Analysis, **31** (1994), 1019–1044.

[3] R. LeVeque, "Finite Difference Methods for Ordinary and Partial Differential Equations. Steady-State and Time-Dependent Problems," SIAM, Philadelphia, PA, 2007.

[4] Z. Li, *A fast iterative algorithm for elliptic interface problems*, SIAM Journal on Numerical Analysis, **35** (1998), 230–254.

[5] Z. Li and M.-C. Lai, *The immersed interface method for Navier-Stokes equations with singular forces*, Journal of Computational Physics, **171** (2001), 822–842.

[6] Z. Li and K. Ito, *Maximum principle preserving schemes for interface problems with discontinuous coefficients*, SIAM Journal on Scientific Computing, **23** (2001), 339–361.

[7] Z. Li and K. Ito, "The Immersed Interface Method. Numerical Solutions of PDE's Involving Interfaces and Irregular Domains," Frontiers in Applied Mathematics, **33**, SIAM, Philadelphia, PA, 2006.

[8] B. Lindberg, *Error estimation and iterative improvement for discretization algorithms*, BIT, **20** (1980), 486–500.

[9] W. Oberkampf and C. Roy, "Verification and Validation in Scientific Computing," Cambridge University Press, Cambridge, 2010.

[10] C. Peskin, *The immersed boundary method*, Acta Numerica, **11** (2002), 479–517.

[11] C. Pozrikidis, "Boundary Integral and Singularity Methods for Linearized Viscous Flow," Cambridge Texts in Applied Mathematics, Cambridge University Press, Cambridge, 1992.

[12] P. Roache, "Verification and Validation in Computational Science and Engineering," Hermosa Publishers, Socorro, New Mexico, 1998.

[13] C. Roy, A. Raju and M. Hopkins, *Estimation of discretization errors using the method of nearby problems*, AIAA Journal, **45** (2007), 1232–1243.

[14] C. Roy and A. Sinclair, *On the generation of exact solutions for evaluating numerical schemes and estimating discretization error*, Journal of Computational Physics, **228** (2009), 1790–1802.

[15] H. Stetter, *The defect correction principle and discretization methods*, Numerische Mathematik, **29** (1977/78), 425–443.

[16] S. Xu and Z. Wang, *An immersed interface method for simulating the interaction of a fluid with moving boundaries*, Journal of Computational Physics, **216** (2006), 454–493.

[17] X. Zhong, *A new high-order immersed interface method for solving elliptic equations with imbedded interface of discontinuity*, Journal of Computational Physics, **225** (2007), 1066–1099.

[18] Y. Zhou, S. Zhou, M. Feig and G. Wei, *High order matched interface and boundary method for elliptic equations with discontinuous coefficients and singular sources*, Journal of Computational Physics, **213** (2006), 1–30.

Received September 2010; revised August 2011.

E-mail address: bvander@math.ubc.ca
E-mail address: mmhopki@sandia.gov
E-mail address: fauci@tulane.edu