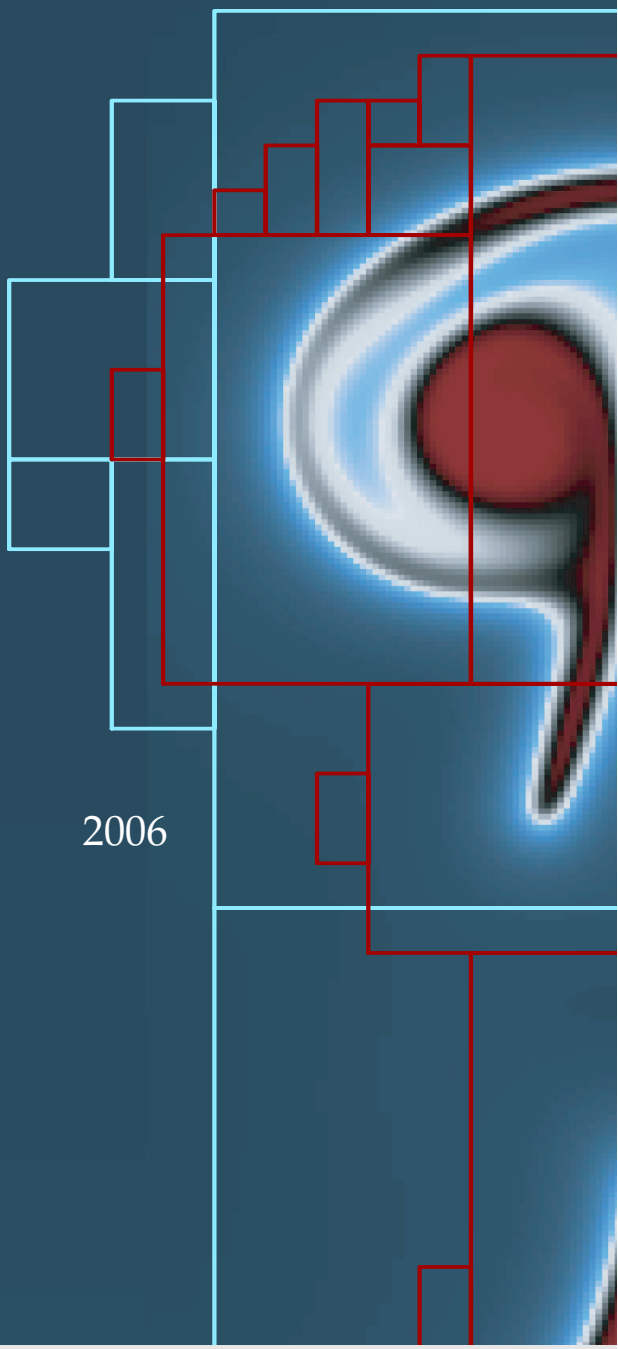
Communications in Applied Mathematics and Computational Science

Volume 1 No. 1 2006



A COMPARISON OF THE EXTENDED FINITE ELEMENT METHOD WITH THE IMMERSED INTERFACE METHOD FOR ELLIPTIC EQUATIONS WITH DISCONTINUOUS COEFFICIENTS AND SINGULAR SOURCES

BENJAMIN LEROY VAUGHAN, JR.,
BRYAN GERARD SMITH AND DAVID L. CHOPP

mathematical sciences publishers

COMM. APP. MATH. AND COMP. SCI.
Vol. 1, No. 1, 2006

A COMPARISON OF THE EXTENDED FINITE ELEMENT METHOD WITH THE IMMERSED INTERFACE METHOD FOR ELLIPTIC EQUATIONS WITH DISCONTINUOUS COEFFICIENTS AND SINGULAR SOURCES

BENJAMIN LEROY VAUGHAN, JR.,
BRYAN GERARD SMITH AND DAVID L. CHOPP

We compare the Immersed Interface Method (IIM) with the Extended Finite Element Method (X-FEM) for elliptic equations with singular sources and discontinuous coefficients. The IIM has been compared favorably with a number of other competing methods. These methods are of particular interest because they allow for the solution of elliptic equations with internal boundaries on nonconforming meshes. In the context of moving interface problems, the emphasis in this paper is placed on accuracy of solutions and their normal derivatives on the interface. These methods are briefly described and the results for benchmark problems are compared.

1. Introduction

Consider the elliptic equation

$$\nabla \cdot (\beta \nabla u) + \kappa u = f \tag{1}$$

in a domain Ω in two dimensions. Embedded within Ω, there is an interface Γ_I (see Figure 1). The coefficients β, κ, and f may be discontinuous across Γ_I and jump conditions are given on the interface.

This type of problem arises in a broad spectrum of mathematical models and hence, a wide range of numerical methods have been devised to solve it. Often, the location of Γ_I varies in time. As a result, methods which are easily adapted to an arbitrary Γ_I are important. Of particular note in this area is the Immersed Interface Method (IIM) [7], which has been shown to perform very well against competing

MSC2000: 65N06, 65N30, 65N50.

Keywords: cartesian grids, discontinuous coefficient, elliptic equation, extended finite element method, finite difference methods, finite element methods, immersed interface method, immersed boundary method, irregular domain, level set methods, singular source term.

Vaughan's work was supported in part by a grant from the NIH under contract #R01-GM067248. Smith's work was supported in part by the DoD NDSEG Fellowship Program and the Chicago Chapter of the ARCS Foundation.

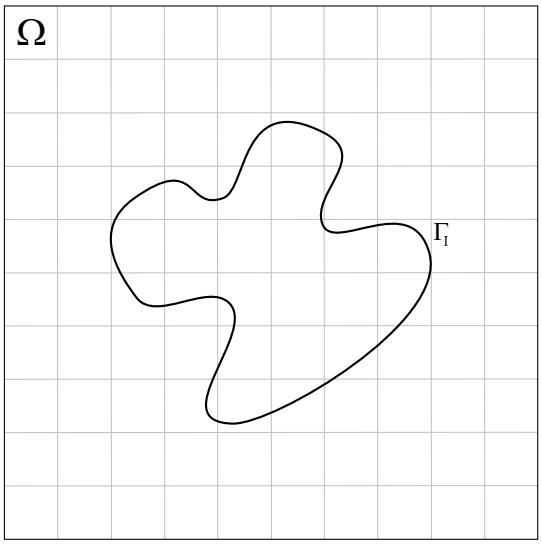


Figure 1. Domain Ω with interface Γ_I.

algorithms [7; 9]. It is representative of a class of methods that are constructed to be globally second order but locally first order on the interface.

In this paper, we compare the Extended Finite Element Method (X-FEM) [11; 3] and the IIM. The X-FEM is a variation on the partition of unity method [10] and has been used for the solution of crack growth problems [11; 2; 17; 15], arbitrary fixed material interfaces and voids [16], solidification problems [5; 4], and modeling rigid particles in Stokes flow [18].

These two methods offer similar advantages in that they both produce accurate solutions without the need for a conforming mesh. This makes them particularly attractive for coupling to methods for moving interfaces, e.g. the level set method [12].

This paper is organized as follows: Sections 2 and 3 discuss the IIM and X-FEM, respectively. A comparison of the numerical results for various types of problems is given in Section 4. Finally, Section 5 gives a summary and concluding remarks.

2. The immersed interface method

The Immersed Interface Method is a finite difference method for approximating the solution to (1). It was introduced in [7] and a detailed overview can be found in [9].

The method solves (1) with singular sources and discontinuous coefficients as well as jump conditions given on the interface by using a regular cartesian grid that does not conform to the interface. For grid points away from the interface, the standard five-point finite difference stencil is used. As a result, the method is

second order away from the interface. For grid points near the interface, a six-point stencil and correction terms are added to the right hand side in order to maintain global second order accuracy.

2.1. ***Stencil generation.*** For simplicity, suppose the domain Ω is a square with space step of length h in both the x and y directions, and let the grid points be located at points (x_i, y_j). In general, the goal is to develop a finite difference equation of the form

$$\gamma_{1,0}u_{i+1,j} + \gamma_{-1,0}u_{i-1,j} + \gamma_{0,1}u_{i,j+1} + \gamma_{0,-1}u_{i,j-1} + \gamma_{0,0}u_{i,j} + \gamma_{\pm1,\pm1}u_{i\pm1,j\pm1} + \kappa_{i,j}u_{i,j} = f_{i,j} + C_{i,j}$$

for the grid point at $\left(x_i, y_j\right)$. Here, only one combination of ±1 is used in the subscripts above which corresponds to the extra point in the stencil as described below.

For points away from the interface, i.e. a point where the interface does not come between any points in the standard five-point stencil, the standard five-point stencil

$$\frac{1}{h}\left(\left(\beta_{i+1/2,j}\frac{u_{i+1,j}-u_{i,j}}{h} - \beta_{i-1/2,j}\frac{u_{i,j}-u_{i-1,j}}{h}\right) + \left(\beta_{i,j+1/2}\frac{u_{i,j+1/2}-u_{i,j}}{h} - \beta_{i,j-1/2}\frac{u_{i,j}-u_{i,j-1}}{h}\right)\right) + \kappa_{i,j}u_{i,j} = f_{i,j},$$

with $C_{i,j} = \gamma_{\pm1,\pm1} = 0$, is used.

For a grid point which bounds a square cut by the interface, the finite difference equation is generated by using a first order expansion of the equation about some point (x^*, y^*) on the interface. The point is chosen to be the point on the interface closest to the grid point (x_i, y_i) as shown in Figure 2. To achieve global second order accuracy, a set of equations is solved to generate the coefficients $\gamma_{k,\ell}$ and $C_{i,j}$.

First, a new transformed coordinate system is introduced. Let θ be the angle between the x-axis and the normal direction as shown in Figure 2. The transformation is:

$$\xi = \left(x - x_i^*\right)\cos\theta + \left(y - y_j^*\right)\sin\theta$$
$$\eta = -\left(x - x_i^*\right)\sin\theta + \left(y - y_j^*\right)\cos\theta$$

After the transform, the truncation error is of the form

$$T_{i,j} = a_1u^- + a_2u^+ + a_3u_\xi^- + a_4u_\xi^+ + a_5u_\eta^- + a_6u_\eta^+ + a_7u_{\xi\xi}^- + a_8u_{\xi\xi}^+ + a_9u_{\eta\eta}^- + a_{10}u_{\eta\eta}^+ + a_{11}u_{\xi\eta}^- + a_{12}u_{\xi\eta}^+ + \kappa^-u^- - f^- - C_{i,j} + O(h)$$

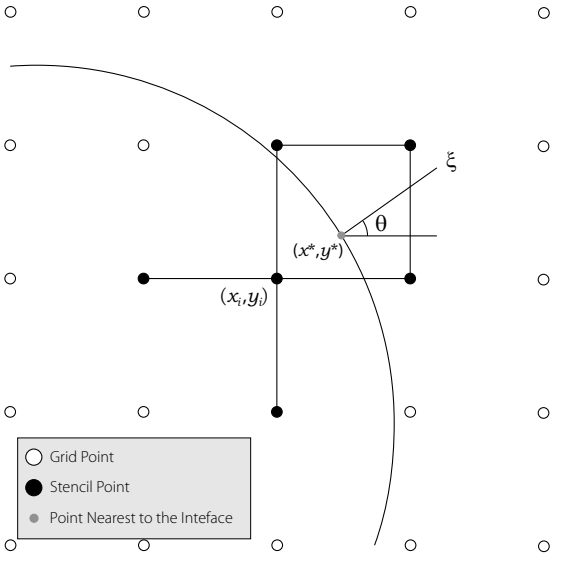


Figure 2. Geometry at a grid point (i, j) near the interface.

where a_j is given by

$$a_1 = \sum_{k \in K^-} \gamma_k \qquad a_2 = \sum_{k \in K^+} \gamma_k$$

$$a_3 = \sum_{k \in K^-} \xi_k \gamma_k \qquad a_4 = \sum_{k \in K^+} \xi_k \gamma_k$$

$$a_5 = \sum_{k \in K^-} \eta_k \gamma_k \qquad a_6 = \sum_{k \in K^+} \eta_k \gamma_k$$

$$a_7 = \frac{1}{2} \sum_{k \in K^-} \xi_k^2 \gamma_k \qquad a_8 = \frac{1}{2} \sum_{k \in K^+} \xi_k^2 \gamma_k$$

$$a_9 = \frac{1}{2} \sum_{k \in K^-} \eta_k^2 \gamma_k \qquad a_{10} = \frac{1}{2} \sum_{k \in K^+} \eta_k^2 \gamma_k$$

$$a_{11} = \sum_{k \in K^-} \xi_k \eta_k \gamma_k \qquad a_{12} = \sum_{k \in K^+} \xi_k \eta_k \gamma_k$$

and the sets K^+ and K^- are defined as

$$K^{\pm} = \{k : (\xi_k, \eta_k) \text{ is on the } \pm \text{ side of } \Gamma_I\}$$

In order to ensure $T_{i,j} = O(h)$, the coefficients of u^-, u^+, u^-_ξ, u^-_η, $u^-_{\xi\xi}$, $u^-_{\xi\eta}$, and $u^-_{\eta\eta}$ must vanish as well as the constant terms. This gives the following six

equations for the unknowns $\gamma_k, \cdots, \gamma_k$:

$$a_1 + a_2 - a_8 [\kappa] / \beta^+ = 0 \tag{2}$$

$$a_3 + \rho a_4 + a_8 \left(\beta_\xi^- - \rho \beta_\xi^+ - [\beta] \chi'' \right) / \beta^+ + a_{10} [\beta] \chi'' / \beta^+ + a_{12} \left(\beta_\eta^- - \rho \beta_\eta^+ \right) / \beta^+ = \beta_\eta^- \tag{3}$$

$$a_5 + a_6 - a_8 \left[\beta_\eta \right] / \beta^+ + a_{12} (1 - \rho) \chi'' = \beta_\eta^- \tag{4}$$

$$a_7 + a_8 \rho = \beta^- \tag{5}$$

$$a_9 + a_{10} + a_8 (\rho - 1) = \beta^- \tag{6}$$

$$a_{11} + a_{12} \rho = 0 \tag{7}$$

where $\rho = \beta^- / \beta^+$ and χ'' is the curvature of the interface at (x^*, y^*).

Once the γ_j's are computed, $C_{i,j}$ can be obtained from

$$C_{i,j} = a_2 w + a_{12} \frac{v'}{\beta^+} + \left(a_6 - a_8 \frac{\beta_\xi^+}{\beta^+} + a_{12} \chi'' \right) w' + a_{10} w'' + \frac{1}{\beta^+} \left(a_4 + a_8 \left(\chi'' - \frac{\beta_\xi^+}{\beta^+} \right) - a_{10} \chi'' - a_{12} \frac{\beta_\eta^+}{\beta^+} \right) v + a_8 \left(\frac{[f]}{\beta^+} - \frac{\kappa^+}{\beta^+} w - w'' \right) \tag{8}$$

where w and v are defined from the jump conditions on the interface:

$$w(\eta) = u^+ - u^-$$

$$v(\eta) = \beta^+ \frac{\partial u}{\partial \hat{n}}^+ - \beta^- \frac{\partial u}{\partial \hat{n}}^-$$

For a detailed derivation of these equations, see [7].

To summarize, using (2)–(7) to solve for γ_k and (8) for $C_{i,j}$, the stencil and the right hand side corrections are obtained. For continuous coefficients β and κ, the five-point stencil is obtained while a six-point stencil is needed if they are discontinuous. These stencils and the correction term, $C_{i,j}$, is used to assemble the linear system to solve for the values $u_{i,j}$ at the grid points.

3. The extended finite element method

The second method used in this paper is the Extended Finite Element Method (X-FEM). Like the Immersed Interface Method, the X-FEM can use a regular cartesian mesh that does not conform to the interface. Note that the X-FEM can also be used on arbitrary triangulated meshes as well. Since there is no comparable

review article discussing this method in detail, we provide here a little more detail on its implementation.

In contrast to finite element meshes, where the mesh conforms to the interface, the X-FEM uses a fixed mesh which does not need to conform to the interface. This is done by extending the standard finite element approximation with extra basis functions on certain "enriched" nodes that capture the behavior of the solution near the interface. This is particularly useful for problems involving moving interfaces where the mesh would otherwise require regeneration every time step. We present here a summary of the method described in [2; 3; 11] with some slight modifications. While the discussion here will focus on 2D problems, it should be noted that this method can be readily applied to 3D as well.

Consider solving (1) on a rectangular domain Ω in two dimensions with Dirichlet boundary conditions applied to the domain boundary $\partial\Omega$. The X-FEM approximation of u is

$$u^h(x, y) = \sum_{n_i \in N} \phi_i(x, y)\, u_i + \sum_{n_j \in N_E} \phi_j(x, y)\, \psi(\varphi)\, a_j \tag{9}$$

where n_i and n_j are the i-th and j-th nodes of their respective sets, N is the set of all nodes in the domain, N_E is the set of enriched nodes, ϕ is a standard finite element basis function (i.e., bilinear or biquadratic), ψ is the enrichment function (described in Section 3.1), and φ is the signed distance function from the interface. The variables u_i and a_j are the unenriched and enriched degrees of freedom, respectively. Also, multiple enrichment functions can be used in the same X-FEM approximation while in this paper, only one is used at a time.

The domain Ω may be meshed by an arbitrary finite element mesh, but in this paper it is meshed with regular rectangular elements independent of the interface. The interface Γ_I is represented by a signed distance function φ and within each element cut by the interface, Γ_I is interpolated as a single line segment.

3.1. ***Enrichments.*** To include the interface's effect, enrichment functions are added to the standard finite element approximation for each element cut by the interface (Figure 3). The choice of enrichment function is based on the behavior of the solution near the interface. In this paper, two enrichment functions are used: a discontinuous, generalized Heaviside function or step function [17] and a continuous ramp function [5]. More application specific enrichment functions can also be used, e.g., a square root singularity function around crack tips [1]. Each of these is a function of the signed distance from the interface given as

$$\varphi(\boldsymbol{x}) = \pm \min_{\boldsymbol{X} \in \Gamma_I} ||\boldsymbol{x} - \boldsymbol{X}||$$

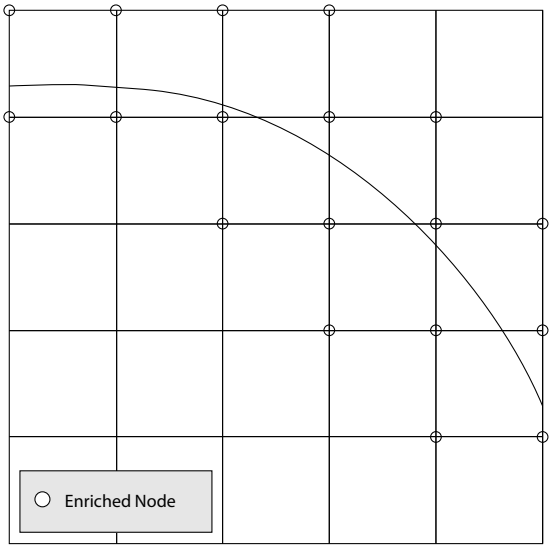


Figure 3. Enriched nodes.

where the sign is positive (negative) if $\boldsymbol{x}$ is outside (inside) the region enclosed by the interface Γ_I. For moving interface problems, the signed distance function is provided directly by the Level Set Method.

The step enrichment function is defined as:

$$\psi_{\text{Step}}(\varphi) = \begin{cases} 1 & \varphi > 0 \\ -1 & \varphi \leq 0 \end{cases}$$

This enrichment function can yield a continuous or discontinuous solution across the interface but requires Lagrange multipliers to apply the Dirichlet jump condition.

The ramp function is defined as:

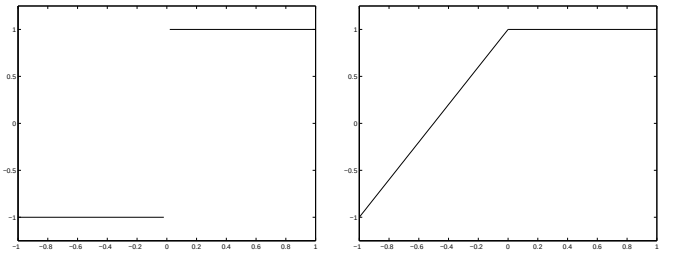
$$\psi_{\text{Ramp}}(\varphi) = \begin{cases} 1 & \varphi > 0 \\ 1 - 2\varphi & \varphi \leq 0 \end{cases}$$

This enrichment function yields only continuous solutions. The advantage is that it automatically satisfies the continuity condition $[\![u]\!] = 0$ and does not require the use of Lagrange multipliers.

3.2. ***Element matrices.*** Using the weak form of (1), there are two types of integral terms: domain and interface.

All the matrices computed from the integral terms are block matrices of the form

$$A = \begin{bmatrix} A^{UU} & A^{UA} \\ A^{AU} & A^{AA} \end{bmatrix}$$



(a) Step function (b) Ramp function

Figure 4. Enrichment functions.

where A^{UU} is the standard FEM element matrix. The A^{UA}, A^{AU}, and A^{AA} matrices are the new matrix terms that arise from the addition of the enriched degrees of freedom. Note that the enriched matrix terms only appear when an element has enriched degrees of freedom and are much smaller than the standard FEM matrix term.

The vector terms also have the same form

$$\boldsymbol{v} = \begin{bmatrix} \boldsymbol{v}^U \\ \boldsymbol{v}^A \end{bmatrix}$$

where $\boldsymbol{v}^U$ is the standard FEM element vector and $\boldsymbol{v}^A$ is the vector term from the enriched degrees of freedom.

3.2.1. *Domain integrals.* The following domain integral terms come from the Laplacian operator $\nabla \cdot (\beta \nabla u)$:

$$K_{i,j}^{UU} = -\int_{\Omega_E} \beta \left[\nabla\phi_i \cdot \nabla\phi_j\right] \partial\Omega_E$$

$$K_{i,j}^{UA} = -\int_{\Omega_E} \beta \left[\nabla\phi_i \cdot \nabla\left(\phi_j\psi_j\right)\right] \partial\Omega_E = K_{j,i}^{AU}$$

$$K_{i,j}^{AA} = -\int_{\Omega_E} \beta \left[\nabla\left(\phi_i\psi_i\right) \cdot \left(\nabla\phi_j\psi_j\right)\right] \partial\Omega_E$$

From the mass operator κu, the matrices are:

$$M_{i,j}^{UU} = \int_{\Omega_E} \kappa\phi_i\phi_j \partial\Omega_E$$

$$M_{i,j}^{UA} = \int_{\Omega_E} \kappa\phi_i\phi_j\psi_j \partial\Omega_E = M_{j,i}^{AU}$$

$$M_{i,j}^{AA} = \int_{\Omega_E} \kappa\phi_i\psi_i\phi_j\psi_j \partial\Omega_E$$

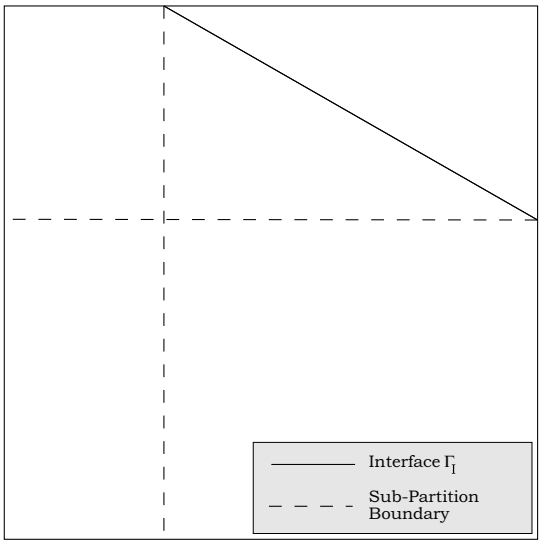


Figure 5. Element subpartitions.

and from the force operator f, the vectors:

$$f_i^U = \int_{\Omega_E} \phi_i f\,(x, y)\,\partial\Omega_E$$

$$f_i^A = \int_{\Omega_E} \phi_i \psi_i f\,(x, y)\,\partial\Omega_E$$

3.2.2. *Element integration.* Evaluating the domain integral terms requires a numerical quadrature method. Elements away from the interface are evaluated using standard Gaussian quadrature in two dimensions.

Elements that are cut by the interface must be treated differently due to discontinuities in the coefficients and enrichment functions. The interface is first interpolated as a line segment and the element is then divided into triangles and quadrilaterals that conform to the interface as illustrated in Figure 5. The subdivisions are for integration only and do not introduce any extra degrees of freedom. This method is slightly different than the method used in [3] in that the elements are not partitioned strictly into triangles. In this method, quadrilaterals are used with triangles transformed into quads for integration using the method given in [14].

3.3. ***Interface conditions.*** After creating the element matrices for each element, the only remaining terms arise from the interface conditions. Enforcing the Dirichlet jump conditions are discussed in Section 3.4.

The Neumann jump condition, $[\![\beta \hat{n} \cdot \nabla u]\!] = v\,(x, y)$, is enforced by introducing a line source term with strength v. The term is of the form

$$\int_{\Gamma_I} v\,(x, y)\,\delta\,(\boldsymbol{x} - \boldsymbol{X}\,(\boldsymbol{s}))\,\partial\Gamma_I \tag{10}$$

where $\boldsymbol{X}\,(\boldsymbol{s})$ is the parameterized coordinates of the interface and the direction of integration is such that the normal points from the positive domain into the negative domain. This term is only added if a source term is not already in the equation and the Neumann jump condition is an external constraint.

Integrating (10) over each element yields the vector terms

$$\gamma_i^U = \int_{\Gamma_I} \phi_i v\,(x, y)\,\partial\Gamma_I$$

$$\gamma_i^A = \psi_i\,(0^-) \int_{\Gamma_I} \phi_i v\,(x, y)\,\partial\Gamma_I$$

where $\psi_i\,(0^-)$ indicates that the enrichment function is evaluated on the negative side of the interface.

3.4. ***Lagrange multipliers.*** Since the Dirichlet jump condition on the interface has not been satisfied when using step enrichments, Lagrange multipliers are used to enforce this condition.

Equations (1) and (10) are combined and rewritten as

$$\nabla \cdot (\beta \nabla u) + \kappa u + [\![u]\!]\,\lambda = f + \int_{\Gamma_I} v\,(x, y)\,\delta\,(\boldsymbol{x} - \boldsymbol{X}\,(\boldsymbol{s}))\,\partial\Gamma_I \tag{11}$$

where $v = \left[\!\!\left[\beta \frac{\partial u}{\partial n}\right]\!\!\right]$ and λ is the Lagrange multiplier used to enforce the jump in the solution.

First, a one dimensional mesh is laid down along the interface as shown in Figure 6 by using a piecewise linear interpolation of the interface within each rectangular element. Next, the Lagrange multipliers are approximated using a 1D finite element approximation

$$\lambda^h = \sum_{m_i \in M} \theta_i \lambda_i$$

where M is the set of all Lagrange multiplier nodes [6].

The jump in the solution $[\![u]\!] = w\,(x, y)$ yields

$$C = \begin{bmatrix} 0 \\ C^A \end{bmatrix}$$

where

$$C_{i,j}^A = \int_{\Gamma_I} \theta_j \phi_i\,[\![\psi_i]\!]\,\partial\Gamma_I$$

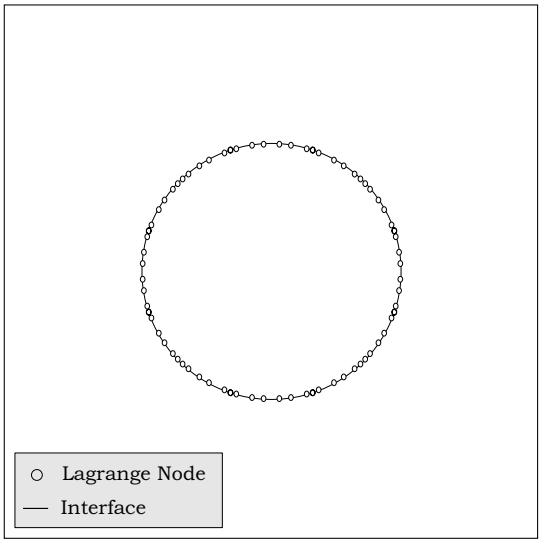


Figure 6. Lagrange multiplier mesh.

and the vector term

$$g_i = \int_{\Gamma_I} \theta_i w \partial \Gamma_I$$

3.5. ***Linear system.*** The resulting linear system contains terms (see Section 3.2) of the form

$$K = \begin{bmatrix} K^{UU} & K^{UA} \\ K^{AU} & K^{AA} \end{bmatrix}$$

$$M = \begin{bmatrix} M^{UU} & M^{UA} \\ M^{AU} & M^{AA} \end{bmatrix}$$

$$\boldsymbol{f} = \begin{bmatrix} \boldsymbol{f}^U \\ \boldsymbol{f}^A \end{bmatrix}$$

$$\boldsymbol{\gamma} = \begin{bmatrix} \boldsymbol{\gamma}^U \\ \boldsymbol{\gamma}^A \end{bmatrix}$$

and the Lagrange multiplier terms C and $\boldsymbol{g}$.

The final assembled linear system is $A\boldsymbol{x} = \boldsymbol{b}$ where

$$A = \begin{bmatrix} K + M & C \\ (C)^T & 0 \end{bmatrix}$$

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{u} \\ \boldsymbol{a} \\ \boldsymbol{\lambda} \end{bmatrix}$$

$$\boldsymbol{b} = \begin{bmatrix} \boldsymbol{f} + \boldsymbol{\gamma} \\ \boldsymbol{g} \end{bmatrix}$$

and K, M, and C are all block matrices and $\boldsymbol{f}$ and $\boldsymbol{\gamma}$ are block vectors. When using ramp enrichments, the Lagrange multiplier terms, C and $\boldsymbol{g}$, along with the Lagrange degrees of freedom, λ, are not needed.

4. Results

In this section, the Immersed Interface Method and the X-FEM are compared on three example problems that are originally from [7]. For all the examples, a square domain is used with an embedded circular interface (Figure 7). Also, the results are confined to be on the interface since the results there are the most important for moving interface problems, and both methods become their standard counterparts away from the interface. In addition, since the solution of the linear system with the X-FEM requires very little time compared with the construction of the system, a direct linear solver is used for the example problems.

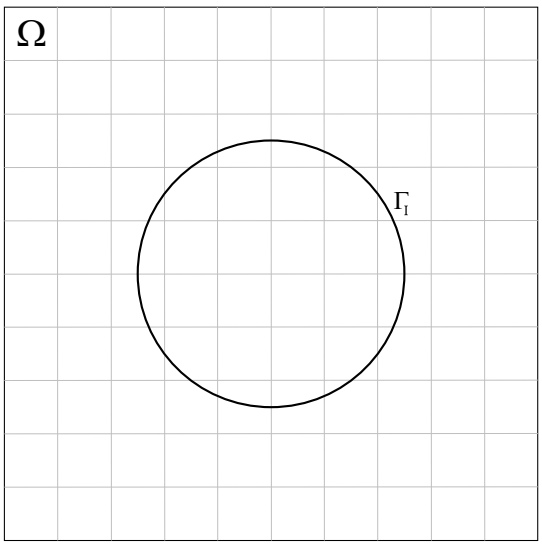


Figure 7. Domain Ω with interface Γ_I.

4.1. ***Example 1.*** The first example has a singular source on Γ_I. The differential equation is:

$$\nabla^2 u = \int_{\Gamma_I} \delta\left(r - R_I\right) \partial\Gamma_I \tag{12}$$

where Γ_I is a circle of radius $R_I = 1/2$ and δ is the Dirac delta function.

The solution to this equation is continuous, $[\![u]\!] = 0$, but the line source gives a jump in the normal

$$\left[\!\left[\frac{\partial u}{\partial n}\right]\!\right] = -2$$

The exact solution to (12) is:

$$u(x, y) = \begin{cases} 1 & r \leq \frac{1}{2} \\ 1 + \log(2r) & r > \frac{1}{2} \end{cases}$$

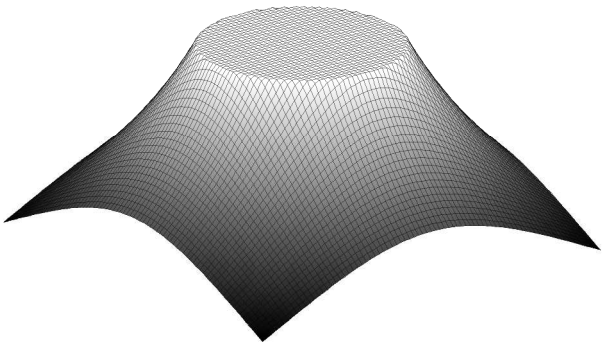


Figure 8. Solution for example 1.

Table 1 shows the convergence results for the X-FEM using four node bilinear elements with step and ramp enrichments. Piecewise constant Lagrange multipliers are used to enforce the Dirichlet jump conditions at the interface when using step enrichments. For comparison, convergence results for the Immersed Interface Method and the Immersed Boundary Method (IBM) [13] are shown. The error values for the Immersed Boundary Method data are taken from [7]. The error given is the maximum error at the nodes defined as

$$\|T_n\|_\infty = \max_{n_i \in N} \{ \left| u(x_i, y_i) - u_i^h \right| \}$$

where n_i is the i-th node with coordinates (x_i, y_i), N is the set of all nodes, and u_i^h is the computed solution at that node. In addition, the ratio of successive errors is given as $\|T_{2n}\| / \|T_n\|$.

Note that this is one of the two error measures that are given in [7]. The second, E_n, is the measure of the error at the nodes away from the interface. In this paper,

T_n is used since the error near or on the interface is of more concern for moving interface problems. In addition, the results for our implementation of the IIM differ from the results given in [7] possibly due to the representation of the interface, a signed distance function, and the choice of the point on the interface for computing the irregular stencil. This implementation does not converge as nicely but the error values are much smaller.

n	Step Enrichment		Ramp Enrichment	
	$\|T_n\|_\infty$	ratio	$\|T_n\|_\infty$	ratio
19	3.8397×10^{-3}		7.8138×10^{-3}	
39	9.3782×10^{-4}	4.0943	3.9577×10^{-3}	1.9743
79	2.3034×10^{-4}	4.0715	1.9029×10^{-3}	2.0798
159	6.4061×10^{-5}	3.5956	9.3797×10^{-4}	2.0287
319	1.5619×10^{-5}	4.1015	4.7646×10^{-4}	1.9686
n	IIM		IBM	
	$\|T_n\|_\infty$	ratio	$\|E_n\|_\infty$	ratio
19	3.1207×10^{-2}		3.6140×10^{-1}	
39	4.3918×10^{-3}	7.1057	2.6467×10^{-2}	12.7939
79	3.2066×10^{-3}	1.3696	1.3204×10^{-2}	2.0045
159	8.9322×10^{-4}	3.5899	6.6847×10^{-3}	1.9753
319	3.4105×10^{-4}	2.6190	3.3393×10^{-3}	2.0018

Table 1. Numerical results for example 1.

From Table 1, the X-FEM is shown to be first order with ramp enrichments and second order with step enrichments coupled with bilinear elements. Ramp enrichments give accuracy comparable with IIM but are only first order. On the other hand, step enrichments show second order accuracy and an order of magnitude improvement over IIM. The first order convergence for the IIM is expected since the error measure includes all the nodes near the interface where the approximation is only first order. Away from the interface both IIM and X-FEM converge second order. As shown before in [7], the IIM outperforms the IBM and consequently, the X-FEM is more accurate than IBM.

Notice that with the X-FEM, the choice of enrichments can change the convergence rate of the method. For this example, ramp enrichments converge first order while step enrichments converge second order. The cause of this is a subject of current research but it seems that extending the region where nodes are enriched, ie enriching nodes a certain distance from the interface but whose support is not necessarily cut by the interface, can regain the second order convergence for certain enrichments.

The X-FEM does show a slight increase in the linear system size. Table 2 gives the linear system size and its sparsity. It is seen that the enrichments and Lagrange multipliers introduce only a small number of new degrees of freedom (less than 2% for a 319×319 mesh).

n	Step Enrichment		Ramp Enrichment		IIM	
	Sys. Size	% Sparse	Sys. Size	% Sparse	Sys. Size	% Sparse
19	520	2.07396%	480	2.15625%	400	1.09250%
39	1,840	0.54277%	1,760	0.55191%	1,600	0.29234%
79	6,880	0.13840%	6,720	0.13940%	6,400	0.07558%
159	26,560	0.03490%	26,240	0.03501%	25,600	0.01921%
319	104,320	0.00876%	103,680	0.00877%	102,400	0.00484%

Table 2. System sizes for example 1.

Table 3 shows the errors interpolated on the interface using (9) for the X-FEM and the method described in [8] for the IIM. The interpolated value on the interface is important if the method is to be coupled with methods for evolving interfaces where the interface velocity is tied to the value at the interface. The interface is parameterized and the errors are computed at 10,000 evenly spaced points on the interface. It is seen that the X-FEM still maintains an order of magnitude improvement over IIM when using step enrichments and both maintain their respective convergence rates.

n	Step Enrichment		Ramp Enrichment		IIM	
	$\|T_n\|_\infty$	ratio	$\|T_n\|_\infty$	ratio	$\|T_n\|_\infty$	ratio
19	5.1857×10^{-3}		2.1871×10^{-2}		6.1970×10^{-2}	
39	1.2444×10^{-3}	4.1672	1.1708×10^{-2}	1.8680	7.5111×10^{-3}	8.2505
79	3.0043×10^{-4}	4.1421	6.0996×10^{-3}	1.9482	3.3766×10^{-3}	2.2245
159	8.8146×10^{-5}	3.4083	3.1101×10^{-3}	1.9612	1.1298×10^{-3}	2.9887
319	1.9315×10^{-5}	4.5636	1.6142×10^{-3}	1.9267	3.6684×10^{-4}	3.0798

Table 3. Interface results for example 1.

Table 4 gives the error in the normal derivative on the interface. This data is quite important when the speed of an evolving interface depends on the gradient of the solution at the interface, eg when the speed is derived from a potential. With the X-FEM using ramp enrichments and IIM, the normal derivative is not accurately captured with $O(1)$ errors, which is expected since the IIM is only an $O(h)$ method on the interface and taking the derivative costs the method an order of accuracy. On

the other hand, using X-FEM with step enrichments captures the normal derivative with first order accuracy.

n	Step Enrichment		Ramp Enrichment		IIM	
	$\|T_n\|_\infty$	ratio	$\|T_n\|_\infty$	ratio	$\|T_n\|_\infty$	ratio
19	4.1828×10^{-1}		1.8292×10^{-0}		4.6176	
39	1.6067×10^{-1}	2.6033	1.6479×10^{-0}	1.1100	4.4095	1.0472
79	9.3826×10^{-2}	1.7124	1.3096×10^{-0}	1.2583	4.2222	1.0444
159	4.5301×10^{-2}	2.0712	1.4733×10^{-0}	0.8889	4.1219	1.0243
319	2.2290×10^{-2}	2.0323	1.3818×10^{-0}	1.0662	4.0640	1.0142

Table 4. Interface derivative results for example 1.

Since using step enrichments with the X-FEM yields much better accuracy while only slightly increasing the system size, the remaining examples will only use step enrichments with the X-FEM.

4.2. ***Example 2.*** The second example has discontinuous coefficients along with a singular source term. The equation is

$$\nabla \cdot (\beta \nabla u) = f + C \int_{\Gamma_I} \delta \left(\boldsymbol{x} - \boldsymbol{X}\left(s\right)\right) \partial \Gamma_I \tag{13}$$

where

$$f\left(x, y\right) = 8\left(x^2 + y^2\right) + 4$$

and

$$\beta\left(x, y\right) = \begin{cases} r^2 + 1 & r \leq \frac{1}{2} \\ b & r > \frac{1}{2} \end{cases}$$

with the following jump conditions

$$[\![u]\!] = 0$$

$$\left[\!\left[\beta \frac{\partial u}{\partial n}\right]\!\right] = 0$$

The exact solution to (13) is:

$$u\left(x, y\right) = \begin{cases} r^2 & r \leq \frac{1}{2} \\ \frac{1}{4}\left(1 - \frac{1}{8b} - \frac{1}{b}\right) + \frac{1}{b}\left(\frac{r^4}{2} + r^2\right) + C \log\left(2r\right) & r > \frac{1}{2} \end{cases}$$

with $b = 10$ and $C = 0.1$.

Table 5 shows the results for the IIM and the X-FEM. Both methods handle the discontinuous variable coefficient with the IIM still being first order while the X-FEM achieves second order accuracy. The results are similar for interpolation on

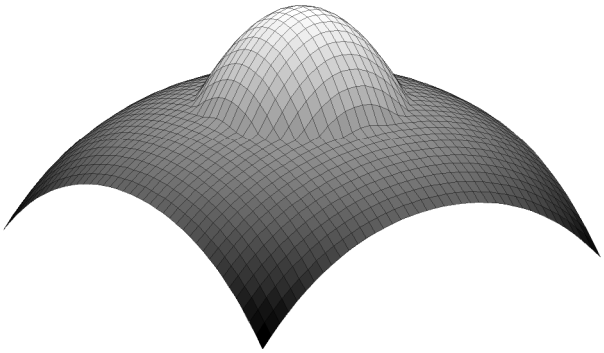


Figure 9. Solution for example 2.

n	X-FEM with steps		IIM	
	$\|T_n\|_\infty$	ratio	$\|T_n\|_\infty$	ratio
19	1.7613×10^{-3}		2.5520×10^{-2}	
39	4.1771×10^{-4}	4.2166	8.4159×10^{-3}	3.0324
79	1.0289×10^{-4}	4.0598	3.5290×10^{-3}	2.3848
159	3.0164×10^{-5}	3.4110	2.1227×10^{-3}	1.6625
319	6.7960×10^{-6}	4.4385	9.8789×10^{-4}	2.1487

Table 5. Numerical results for example 2.

the interface as show in Table 6. In addition, Table 7 shows the same convergence results as the previous example problem for evaluating the normal derivative on the interface with no convergence for the IIM and first order convergence for the X-FEM.

4.3. ***Example 3.*** For the third example, jumps in the function u are imposed on the interface Γ_I. The differential equation is

$$\nabla^2 u = 0 \tag{14}$$

with the jump conditions

$$[\![u]\!] = e^x \cos y$$

$$\left[\!\left[\frac{\partial u}{\partial n}\right]\!\right] = 2e^x \left(x \cos y - y \sin y\right)$$

The exact solution of (14) is:

n	X-FEM with steps		IIM	
	$\|T_n\|_\infty$	ratio	$\|T_n\|_\infty$	ratio
19	1.6517×10^{-3}		2.5988×10^{-2}	
39	3.3824×10^{-4}	4.8832	8.8692×10^{-3}	2.9301
79	8.2238×10^{-5}	4.1129	3.6100×10^{-3}	2.4568
159	3.1568×10^{-5}	2.6051	2.1768×10^{-3}	1.6584
319	7.4612×10^{-6}	4.2310	1.0004×10^{-4}	2.1759

Table 6. Interface results for example 2.

n	X-FEM with steps		IIM	
	$\|T_n\|_\infty$	ratio	$\|T_n\|_\infty$	ratio
19	2.7307×10^{-1}		4.6176×10^{-0}	
39	1.2776×10^{-1}	2.1374	4.4095×10^{-0}	1.0472
79	6.1203×10^{-2}	2.0875	4.2222×10^{-0}	1.0444
159	4.8216×10^{-2}	1.2694	4.1219×10^{-0}	1.0243
319	2.4790×10^{-2}	1.9450	4.0640×10^{-0}	1.0142

Table 7. Interface derivative results for example 2.

$$u(x, y) = \begin{cases} e^x \cos y & r \leq \frac{1}{2} \\ 0 & r > \frac{1}{2} \end{cases} \tag{15}$$

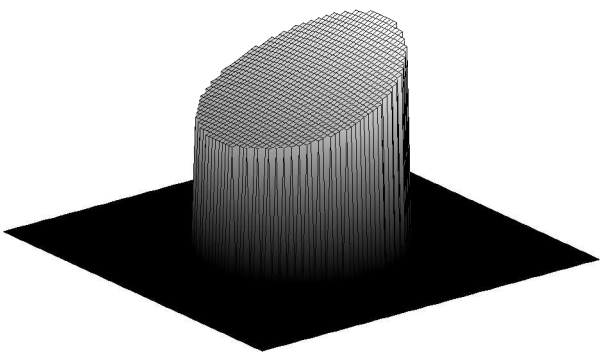


Figure 10. Solution for example 3.

Since (14) does not have a line source term explicitly, the equation is modified for the X-FEM to include one that yields the correct jump in the normal derivative. The new equation is

$$\nabla^2 u = \int_{\Gamma_I} 2e^x \left(x \cos y - y \sin y\right) \delta \left(\boldsymbol{x} - \boldsymbol{X}\left(x\right)\right) \partial\Gamma_{\boldsymbol{I}} \tag{16}$$

n	X-FEM with steps		IIM	
	$\|T_n\|_\infty$	ratio	$\|T_n\|_\infty$	ratio
19	1.7648×10^{-4}		3.6253×10^{-3}	
39	6.0109×10^{-5}	2.9360	4.6278×10^{-4}	7.8337
79	1.7769×10^{-5}	3.3828	3.0920×10^{-4}	1.4967
159	4.8626×10^{-6}	3.6542	1.1963×10^{-4}	2.5846
319	1.2362×10^{-6}	3.9335	4.5535×10^{-5}	2.6272

Table 8. Numerical results for example 3.

n	X-FEM with steps		IIM	
	$\|T_n\|_\infty$	ratio	$\|T_n\|_\infty$	ratio
19	4.7842×10^{-4}		4.0230×10^{-3}	
39	1.0659×10^{-4}	4.4884	5.7563×10^{-4}	6.9889
79	2.8361×10^{-5}	3.7583	3.1617×10^{-4}	1.8206
159	7.3603×10^{-6}	3.8532	1.2004×10^{-4}	2.6339
319	2.0634×10^{-6}	3.5671	4.5526×10^{-5}	2.6367

Table 9. Interface results for example 3.

n	X-FEM with steps		IIM	
	$\|T_n\|_\infty$	ratio	$\|T_n\|_\infty$	ratio
19	5.6520×10^{-2}		$3.0009 \times 10^{+1}$	
39	2.4190×10^{-2}	2.3365	$5.5185 \times 10^{+1}$	0.5438
79	9.4512×10^{-3}	2.5595	$1.2034 \times 10^{+2}$	0.5392
159	7.1671×10^{-3}	1.3187	$2.6466 \times 10^{+2}$	0.4547
319	2.6865×10^{-3}	2.6678	$5.2870 \times 10^{+2}$	0.5006

Table 10. Interface derivative results for example 3.

Table 8 gives the results for the X-FEM and the IIM with the X-FEM is second order while the IIM is first order with the X-FEM giving about an order of magnitude better improvement at the nodes. The conclusions are similar for errors taken on the interface as given in Table 9. Table 10 contains the errors in the normal derivative on the interface for the X-FEM and the IIM. Once again, the X-FEM is first order when computing the normal derivative and the IIM is unable give an accurate evaluation of the normal derivative at the interface.

5. Conclusion

In this paper, the Extended Finite Element Method and the Immersed Interface Method were compared. Both methods use a regular cartesian mesh, which does not conform to an internal interface.

The Immersed Interface Method is a finite difference method that handles interfaces by using a six point stencil where needed, along with correction terms on the right hand side, to handle the jump conditions. It is second order accurate at the grid points away from the interface and first order accurate at the grid points near the interface.

The Extended Finite Element Method is a finite element method where extra "enriched" basis functions are added to the standard finite element approximation. These enrichment functions add discontinuities that approximate the behavior near the interface. These enrichments coupled with the enforcement of the interface conditions yields accurate results both near and away from the interface. In addition, the X-FEM is not restricted to enforcing only jump conditions on the interface in its formulation. The lack of this restriction allows explicit boundary conditions to be applied, which is a subject of current research.

Overall, the X-FEM performed well compared to the IIM. It provides second order accuracy at the nodes and on the interface while more accurately capturing the gradient on the interface for each of the problems. Against other methods like the IIM, which are constructed as second order methods away from the interface but only have a local $O(h)$ truncation error near the interface, the X-FEM maintains an advantage due it being second order on all the nodes including the ones near the interface. This is an advantage because an accurate approximation of the gradient at the interface is important for moving interface problems where the velocity is often derived from a gradient of the velocity potential. This makes the X-FEM a more attractive choice for coupling with moving interface methods such as the Level Set Method.

References

[1] T. Belytschko and T. Black, *Elastic crack growth in finite element with minimal remeshing*, International Journal of Numerical Methods in Engineering **45** (1999), 601–620.

[2] J. E. Dolbow, N. Moës, and T. Belytschko, *Discontinuous enrichment in finite elements with a partition of unity method*, Finite Elements in Analysis and Design **36** (2000), 235–260.

[3] ———, *An extended finite element method for modeling crack growth with frictional contact*, Computational Methods in Applied Mechanics and Engineering **190** (2001), 6825–6846.

[4] J. C. et. al., *The extended finite element method (xfem) for solidification problems.*, International Journal of Numerical Methods in Engineering **53** (2002), 1959–1977.

[5] H. Ji, D. Chopp, and J. E. Dolbow, *A hybrid extended finite element/level set method for modeling phase transformation*, International Journal of Numerical Methods in Engineering **54** (2002), 1209–1233.

[6] H. Ji and J. E. Dolbow, *On strategies for enforcing interfacial constraints and evaulating jump conditions with the extended finite element method*, International Journal of Numerical Methods in Engineering **61** (2004), 2508–2535.

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

[8] R. J. LeVeque and Z. Li, *Immersed interface methods for stokes flow with elastic boundaries or surface tension*, SIAM Journal Scientific Computing **18** (1997), no. 3, 709–735.

[9] Z. Li, *An overview of the immersed interface method and its applications*, Taiwanese Journal of Mathematics **7** (2003), 1–49.

[10] J. M. Melenk and I. Babuška, *The partition of unity finite element method: Basic theory and applications*, Computational Methods in Applied Mechanics and Engineering **139** (1996), 289ï¿½314.

[11] N. Moës, J. Dolbow, and T. Belytschko, *A finite element method for crack growth without remeshing*, International Journal of Numerical Methods in Engineering **46** (1999), 131–150.

[12] S. Osher and J. A. Sethian, *Fronts propagating with curvature-dependent speed: algorithms base on hamilton-jacobi formulations*, Journal of Computational Physics **79** (1988), 12–49.

[13] C. S. Peskin, *Numerical analysis of blood flow in the heart*, Journal of Computational Physics **25** (1977), 220–252.

[14] H. T. Rathod, K. V. N. B. Venkatesudu, and N. L. Ramesh, *Gauss legendre quadrature over a triangle*, Journal Indian Institute of Science **84** (2004), 183–188.

[15] M. Stolarska, D. L. Chopp, N. Moës, and T. Belytschko, *Modelling crack growth by level sets in the extended finite element method*, International Journal of Numerical Methods in Engineering **51** (2001), 943–960.

[16] N. Sukumar, D. Chopp, N. Moës, and T. Belytschko, *Modeling holes and inclusions by level sets in the extended finite element method*, International Journal of Numerical Methods in Engineering **48** (2000), 1549–1570.

[17] N. Sukumar, D. L. Chopp, and B. Moran, *Extended finite element method and fast marching method for three-dimensional fatigue crack propagation*, Engineering Fracture Mechanics **70** (2003), 29–48.

[18] G. J. Wagner, N. Moës, W. K. Liu, and T. Belytschko, *The extended finite element method for rigid particles in stokes flow*, International Journal of Numerical Methods in Engineering **51** (2001), 293–313.

Received January 4, 2006.

BENJAMIN LEROY VAUGHAN, JR.: b-vaughan@northwestern.edu
Engineering Sciences and Applied Mathematics Dept., Northwestern University, 2145 Sheridan Road, Evanston, Illinois 60208, United States

BRYAN GERARD SMITH: b-smith7@northwestern.edu
Engineering Sciences and Applied Mathematics Dept., Northwestern University, 2145 Sheridan Road, Evanston, Illinois 60208, United States

DAVID L. CHOPP: chopp@northwestern.edu
Engineering Sciences and Applied Mathematics Dept., Northwestern University, 2145 Sheridan Road, Evanston, Illinois 60208, United States