

# An efficient numerical method for one-dimensional hyperbolic interface problems\*

Chartese Jones<sup>1\*\*</sup> and Xu Zhang<sup>2\*\*\*</sup>

<sup>1</sup>Department of Mathematics and Statistics, Mississippi State University, Mississippi State, MS 39762

<sup>2</sup>Department of Mathematics and Statistics, Mississippi State University, Mississippi State, MS 39762

**Abstract.** In this paper, we develop an efficient numerical scheme for solving one-dimensional hyperbolic interface problems. The immersed finite element (IFE) method is used for spatial discretization, which allows the solution mesh to be independent of the interface. Consequently, a fixed uniform mesh can be used throughout the entire simulation. The method of lines is used for temporal discretization. Numerical experiments are provided to show the features of these new methods.

## 1 Introduction

Consider the inhomogeneous one-dimensional acoustic wave equation

$$u_{tt} - (\beta u_x)_x = f(x, t) \quad \text{in } \Omega \times (0, T), \tag{1}$$

with the following initial and boundary conditions

$$u(x, 0) = u_0(x) \quad u_t(x, 0) = u_1(x). \tag{2}$$

$$u(a, t) = g_0(t) \quad u(b, t) = g_1(t). \tag{3}$$

Here,  $\Omega = (a, b)$  denotes the spatial domain, and  $\beta$  denotes the sound speed. We assume that  $\beta$  is discontinuous across an interface point  $\alpha \in \Omega$  which separates  $\Omega$  into two sub-domains  $\Omega^+ = (\alpha, b)$  and  $\Omega^- = (a, \alpha)$ . The exact solution  $u$  is assumed to satisfy the following interface jump conditions:

$$[u] |_{\alpha} = 0, \tag{4}$$

$$[\beta u_x] |_{\alpha} = 0. \tag{5}$$

The coefficient function  $\beta$  is a piece-wise continuous function defined as follows:

$$\beta(x, t) = \begin{cases} \beta^-(x, t), & x \in \Omega^-, \\ \beta^+(x, t), & x \in \Omega^+. \end{cases} \tag{6}$$

There are many numerical methods that can be used to solve this problem. For instance, the finite difference method [1], finite volume method [2], and finite element method [3].

\*The project is partially supported by National Science Foundation Grant DMS-1720425.

\*\*e-mail: [cdj192@msstate.edu](mailto:cdj192@msstate.edu)

\*\*\*e-mail: [xuzhang@math.msstate.edu](mailto:xuzhang@math.msstate.edu)

When using standard finite element methods, one needs to tailor the solution mesh to fit the interface in order to obtain the optimal convergence. This meshing procedure can be time-consuming if the interface is moving, or the interface is geometrically complicated in higher dimension domains. In this paper, we adopt the immersed finite element method (IFEM) [4] which allows the use of unfitted meshes that is independent of the location of the interface. As a result, a fixed uniform mesh can be used to solve interface problems even if the interface moves.

The fundamental idea of IFEM is to locally modify the basis functions on the interface element to better accommodate the interface jump conditions. To be more specific, piecewise polynomials instead of polynomials will be used as basis functions on interface elements. The IFEM has been used to solve elliptic interface problems [5] and parabolic interface problems [6]. Recently, it has been used for solving hyperbolic interface problems [7] with discontinuous Galerkin spatial discretization. In this paper, we will use the high order IFEM in standard Galerkin formulation to solve the model problem (1) - (6).

This article will be presented in the following order. In section 2, we will recall the linear and higher degree IFEM basis functions. In section 3, we will derive the numerical scheme. In particular, we use the methods of lines for time discretization. Finally, in section 4, we will present some numerical examples.

## 2 Immersed Finite Element Spaces

In this section, we recall some immersed finite element spaces. The linear IFE space was first introduced in [4]. Let  $\phi_{0,T}$   $\phi_{1,T}$  be two IFE basis functions on an interface element  $T = (x_i, x_{i+1})$  containing the interface point  $\alpha$ . These two linear IFE basis functions are constructed to satisfy the nodal value conditions as well as interface jump conditions as follows

$$\phi_{0,T}(x_i) = 1, \quad \phi_{0,T}(x_{i+1}) = 0, \quad [\phi_{0,T}(\alpha)] = 0, \quad [\beta\phi'_{0,T}(\alpha)] = 0. \tag{7}$$

$$\phi_{1,T}(x_i) = 0, \quad \phi_{1,T}(x_{i+1}) = 1, \quad [\phi_{1,T}(\alpha)] = 0, \quad [\beta\phi'_{1,T}(\alpha)] = 0. \tag{8}$$

By direct calculation, we have

$$\phi_{0,T}(x) = \begin{cases} \frac{y_1 - 1}{\alpha - x_i}(x - x_i) + 1, & x \in [x_i, \alpha] \\ \frac{-y_1}{x_{i+1} - \alpha}(x - x_{i+1}), & x \in [\alpha, x_{i+1}] \end{cases} \tag{9}$$

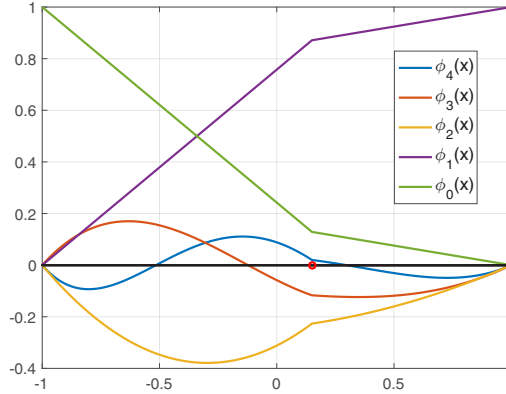
where  $y_1 = \frac{\rho(x_{i+1}-\alpha)}{-x_i+\alpha+\rho(x_{i+1}-\alpha)}$  and  $\rho = \frac{\beta^-}{\beta^+}$ . Similarly, we have

$$\phi_{1,T}(x) = \begin{cases} \frac{y_2}{\alpha - x_i}(x - x_i), & x \in [x_i, \alpha] \\ \frac{1 - y_2}{x_{i+1} - \alpha}(x - x_{i+1}) + 1, & x \in [\alpha, x_{i+1}] \end{cases} \tag{10}$$

where  $y_2 = \frac{x_i-\alpha}{x_i+\alpha(\rho-1)-x_{i+1}\rho}$ .

For higher order approximation, the  $k$ -th degree IFE basis functions can be constructed in a similar manner by enforcing the nodal value conditions, interface jump conditions (4) - (5), and the extended jump conditions [8]:

$$[\beta u^{(j)}(\alpha)] = 0, \quad j = 2, 3, \dots, k. \tag{11}$$



**Figure 1.** IFE basis functions on a reference element  $[-1, 1]$ .

Alternatively, the  $k$ -th degree IFE basis functions can also be constructed in the framework of orthogonal polynomials with discontinuous weight functions. For more details we refer readers to [9, 10]. We plot a few orthogonal high order IFE basis functions in Figure 1.

The local  $k$ -th degree IFE space is  $S_h(T) = span\{\phi_{0,T}, \phi_{1,T}, \dots, \phi_{k,T}\}$ . The global IFE space is defined by

$$\mathcal{U} = \{v \in H^1(\Omega) \mid v \in P_k(T) \text{ if } T \text{ is a regular element}; v \in S_h(T) \text{ if } T \text{ is an interface element}\}. \quad (12)$$

Here  $P_k(T)$  is the standard polynomial space with degree no more than  $k$ .

### 3 Numerical Algorithm

In this section, we use the Galerkin IFEM for spatial discretization and the method of lines for temporal discretization. The method of lines is an efficient numerical method for time-dependent partial differential equations that proceeds by first discretizing the spatial derivatives and leaving the time variable continuous, then further discretize in temporal discretization.

The weak formulation of the hyperbolic initial boundary value problem (1)-(6) is to find  $u \in H_g^1 = \{v \in H^1(\Omega) : v(a, t) = g_0(t), v(b, t) = g_1(t)\}$  for each time  $t$  and such that

$$\int_{\Omega} u_{tt} v dx + \int_{\Omega} \beta u_x v_x dx = \int_{\Omega} f v dx, \quad \forall v \in H_0^1(\Omega). \quad (13)$$

We look to approximate  $u(x, t)$  by  $u_h(x, t) \in \mathcal{U}_g \triangleq \mathcal{U} \cap H_0^1(\Omega)$  in the form of

$$u_h(x, t) = \sum_{j=1}^N c_j(t) \phi_j(x),$$

where  $\phi_j$  are global IFE basis functions. Then the discretized weak formulation can be written as

$$\sum_{j=1}^N c_j''(t) \int_{\Omega} \phi_j(x) \phi_i(x) dx + \sum_{j=1}^N c_j(t) \int_{\Omega} \beta \phi_j'(x) \phi_i'(x) dx = \int_{\Omega} f(x, t) \phi_i(x) dx. \quad (14)$$

In matrix form, we have

$$M\ddot{\vec{c}}(t) + S\dot{\vec{c}}(t) = \vec{F}(t), \tag{15}$$

where  $\vec{c}(t) = [c_1(t), \dots, c_N(t)]^T$  is the unknown vector.  $M$  and  $S$  are mass and stiffness matrices, respectively.  $\vec{F}(t)$  is the source vector. We note that the time dependent PDE becomes a system of ordinary differential equations.

Next we proceed with full discretization. Let the partition of the time domain be

$$0 = t_0 < t_1 < \dots < t_{N-1} < t_N = T$$

where  $\Delta t_n = t_n - t_{n-1}$  is the time step size. In our scheme, we consider the uniform time step size, that is  $\Delta t_n = \Delta t = T/N_t$  for all  $n$ . For time discretization we use the central difference scheme:

$$\ddot{\vec{c}}(t_n) \approx \frac{\vec{c}(t_{n+1}) - 2\vec{c}(t_n) + \vec{c}(t_{n-1}))}{(\Delta t)^2}, \quad \dot{\vec{c}}(t_n) \approx \frac{\vec{c}(t_{n+1}) + 4\vec{c}(t_n) + \vec{c}(t_{n-1}))}{6}. \tag{16}$$

We note that both formulas in (16) yield a second order approximation in time. Evaluating equation (15) at  $t_n$ , and using (16) to substitute  $\ddot{\vec{c}}(t_n)$  and  $\dot{\vec{c}}(t_n)$  we have

$$\left(M + \frac{1}{6}(\Delta t)^2 S\right) \vec{c}^{n+1} = \left(2M - \frac{2}{3}(\Delta t)^2 S\right) \vec{c}^n + \left(-M - \frac{1}{6}(\Delta t)^2 S\right) \vec{c}^{n-1} + (\Delta t)^2 \vec{F}^n. \tag{17}$$

Here we use the simplified notations  $\vec{c}^n$  to replace  $\vec{c}(t_n)$ , and  $\vec{F}^n$  to replace  $\vec{F}(t_n)$ . The initial vector  $\vec{c}^0$  is chosen as the interpolation of the initial function  $u_0$  in equation (2):

$$\vec{c}^0 = [u_0(x_1), u_0(x_2), \dots, u_0(x_N)]^T, \tag{18}$$

and  $\vec{c}^1$  is calculated by

$$\vec{c}^1 = \vec{c}^0 + \Delta t \begin{bmatrix} u_1(x_1) \\ u_1(x_2) \\ \vdots \\ u_1(x_N) \end{bmatrix} + \frac{(\Delta t)^2}{2} \begin{bmatrix} f(x_1, 0) + (\beta(x_1, 0)u'_0(x_1))' \\ f(x_2, 0) + (\beta(x_2, 0)u'_0(x_2))' \\ \vdots \\ f(x_N, 0) + (\beta(x_N, 0)u'_0(x_N))' \end{bmatrix}. \tag{19}$$

Note that equation (19) is to ensure the second order accuracy of the initial data. This can be easily verified by Taylor expansion.

## 4 Numerical Examples

In this section, we present a few numerical examples to test the performance of our full discretization numerical algorithm (17). Since our spatial meshes are independent of the interface, we use a sequence of uniform meshes  $\{\mathcal{T}_h\}$  with size  $h = \frac{b-a}{N}$ . When using the linear IFEM for spatial discretization, due to the CFL condition the time step size  $\Delta t$  must be chosen such that  $\sqrt{\beta_{max}}\Delta t \leq h$  where  $\beta_{max} = \max_{x \in \Omega, t \in (0, T)} \beta(x, t)$ . For quadratic IFEM, we choose  $\Delta t$  such that  $\beta_{max}\Delta t \leq h$ .

### 4.1 Example 1: Small Jump

In this example, we choose the interface point  $\alpha = \frac{\pi}{6}$ . The coefficient  $\beta(x, t)$  is a piecewise constant function such that

$$\beta(x, t) = \begin{cases} 5, & x \in \Omega^-, \\ 1, & x \in \Omega^+. \end{cases}$$

The exact solution of this example is

$$u(x, t) = \begin{cases} \frac{1}{\beta^-} \sin(2\pi x)e^t, & x \in \Omega^-, \\ \frac{1}{\beta^+} \sin(2\pi x)e^t + \left(\frac{1}{\beta^-} - \frac{1}{\beta^+}\right) \sin(2\pi\alpha)e^t, & x \in \Omega^+. \end{cases} \quad (20)$$

We test with both linear ( $k = 1$ ) and quadratic ( $k = 2$ ) IFEM. The errors for the numerical solutions at the final time ( $t = 1$ ) are recorded. Errors for linear and quadratics IFEM solutions are listed in Table 1 and Table 2, respectively.

**Table 1.** Errors for Linear IFEM at final time for  $\beta^+ = 1$  and  $\beta^- = 5$ ,  $\Delta t = h/3$

h	L2 Error	L2 Rate	H1 Error	H1 Rate
$\frac{1}{5}$	1.9003e-01	—	3.0725e-00	—
$\frac{1}{10}$	4.8190e-02	1.9043	1.5593e-00	0.9785
$\frac{1}{20}$	1.2208e-02	1.9809	7.8727e-01	0.9859
$\frac{1}{40}$	3.0610e-03	1.9958	3.9464e-01	0.9963
$\frac{1}{80}$	7.6552e-04	1.9995	1.9737e-02	0.9996

**Table 2.** Errors for Quadratic IFEM at final time for  $\beta^+ = 1$  and  $\beta^- = 5$ ,  $\Delta t = h/5$

h	L2 Error	L2 Rate	H1 Error	H1 Rate
$\frac{1}{5}$	1.3218e-02	—	4.5149e-01	—
$\frac{1}{10}$	1.7475e-03	2.9191	1.1270e-01	2.0022
$\frac{1}{20}$	2.3269e-04	2.9088	2.9947e-02	1.9121
$\frac{1}{40}$	3.0227e-05	2.9445	7.6560e-03	1.9677
$\frac{1}{80}$	3.9649e-06	2.9305	1.9111e-03	2.0022

We observe that the numerical solutions in  $H^1$  norm converge in first order for linear IFEM, and second order for quadratic IFEM. This is consistent with the convergence rate for the standard FEM approximation. Errors in  $L^2$  norm decay in second order for linear IFEM and third order for quadratics IFEM. Note that, the time step size  $\Delta t$  is much smaller than  $h$  in this case, so the convergence rate will be dominated by the spatial discretization.

### 4.2 Example 2: Large Jump

In this example, we test the same problem as in Example 1, but with a much large coefficient jump, i.e.,

$$\beta(x, t) = \begin{cases} 100 & x \in \Omega^-, \\ 1 & x \in \Omega^+. \end{cases}$$

Errors for numerical solutions using linear and quadratic IFEM are reported in Table 3 and Table 4, respectively. The orders of convergence in  $L^2$  and  $H^1$  norms of the larger jump coefficient are similar to the small jump case in the Example 1.

### 4.3 Example 3: Variable Coefficient

In this example, we consider the non-constant discontinuous coefficient function

$$\beta(x) = \begin{cases} x^2 + 1 & x \in \Omega^-, \\ (x + 1)^2 & x \in \Omega^+. \end{cases}$$

**Table 3.** Errors for Linear IFEM at final time for  $\beta^+ = 1$  and  $\beta^- = 100$ ,  $\Delta t = h/10$

h	L2 Error	L2 Rate	H1 Error	H1 Rate
$\frac{1}{5}$	1.8729e-01	—	3.0122e-00	—
$\frac{1}{10}$	4.7458e-02	1.9806	1.5280e-00	0.9792
$\frac{1}{20}$	1.2051e-02	1.9775	7.7183e-01	0.9853
$\frac{1}{40}$	3.0232e-03	1.9950	3.8697e-01	0.9961
$\frac{1}{80}$	7.5601e-04	1.9996	1.9353e-01	0.9996

**Table 4.** Errors for Quadratic IFEM at final time for  $\beta^+ = 1$  and  $\beta^- = 100$ ,  $\Delta t = h/100$

h	L2 Error	L2 Rate	H1 Error	H1 Rate
$\frac{1}{5}$	1.1603e-02	—	3.9214e-01	—
$\frac{1}{10}$	1.5876e-03	2.8696	1.0674e-01	1.8772
$\frac{1}{20}$	2.1815e-04	2.8634	2.8323e-02	1.9141
$\frac{1}{40}$	2.8739e-05	2.9243	7.4509e-03	1.9265
$\frac{1}{80}$	3.5935e-06	2.9995	1.8637e-03	1.9993

The exact solution to this problem is

$$u(x, t) = \begin{cases} \frac{1}{\alpha^2 + 1} x^4 e^t, & x \in [a, \alpha], \\ \frac{1}{(\alpha + 1)^2} x^4 e^t + \left( \frac{1}{\alpha^2 + 1} - \frac{1}{(\alpha + 1)^2} \right) \alpha^4 e^t, & x \in [\alpha, b], \end{cases}$$

where the interface point is  $\alpha = \pi/6$ . One can verify that the jump conditions (4) and (5) hold by straightforward calculation. The errors for linear and quadratic IFEM solutions at the final time level are reported in Table 5 and Table 6, respectively. The errors decay in the order of  $O(h^{k+1} + \Delta t^2)$  in  $L^2$  norm, and with an order of  $O(h^k + \Delta t^2)$  in  $H^1$  norm, where  $k$  is the polynomial degree of the IFE spaces. This is consistent with the standard FEM approximation.

**Table 5.** Errors for Linear IFEM at final time for  $\beta^+(x) = (x + 1)^2$  and  $\beta^-(x) = x^2 + 1$ ,  $\Delta t = h/2$

h	L2 Error	L2 Rate	H1 Error	H1 Rate
$\frac{1}{5}$	5.0155e-03	—	7.6951e-02	—
$\frac{1}{10}$	1.2443e-03	2.0111	3.7981e-02	1.0187
$\frac{1}{20}$	3.1415e-04	1.9858	1.9193e-02	0.9847
$\frac{1}{40}$	7.9064e-05	1.9904	9.6670e-03	0.9894
$\frac{1}{80}$	1.9763e-05	2.0002	4.8315e-03	1.0006

**Table 6.** Errors for Quadratic IFEM at final time for  $\beta^+(x) = (x + 1)^2$  and  $\beta^-(x) = x^2 + 1$ ,  $\Delta t = h/4$

h	L2 Error	L2 Rate	H1 Error	H1 Rate
$\frac{1}{5}$	1.2690e-04	—	1.5851e-03	—
$\frac{1}{10}$	2.8299e-05	2.1648	3.9646e-04	1.9993
$\frac{1}{20}$	6.8356e-06	2.0496	9.9823e-05	1.9897
$\frac{1}{40}$	1.6911e-06	2.0151	2.4995e-05	1.9977
$\frac{1}{80}$	4.2144e-07	2.0045	6.2454e-06	2.0008

## References

- [1] J.C. Strikwerda, *Finite difference schemes and partial differential equations*, Vol. 88 (SIAM, 2004)
- [2] R.J. LeVeque, *Finite volume methods for hyperbolic problems*, Cambridge Texts in Applied Mathematics (Cambridge University Press, Cambridge, 2002)
- [3] C. Johnson, *Numerical solution of partial differential equations by the finite element method* (Cambridge University Press, Cambridge, 1987)
- [4] Z. Li, *Appl. Numer. Math.* **27**, 253 (1998)
- [5] T. Lin, Y. Lin, X. Zhang, *SIAM J. Numer. Anal.* **53**, 1121 (2015)
- [6] X. He, T. Lin, Y. Lin, X. Zhang, *Numer. Methods Partial Differential Equations* **29**, 619 (2013)
- [7] S. Adjerid, K. Moon, in *Advances in applied mathematics* (Springer, Cham, 2014), Vol. 87 of *Springer Proc. Math. Stat.*, pp. 57–69
- [8] S. Adjerid, T. Lin, *Appl. Numer. Math.* **59**, 1303 (2009)
- [9] W. Cao, X. Zhang, Z. Zhang, *Adv. Comput. Math.* **43**, 795 (2017)
- [10] W. Cao, X. Zhang, Z. Zhang, Q. Zou, *J. Sci. Comput.* **73**, 543 (2017)