University of Cincinnati

Date: 2/28/2024

I, Kevin T Schmitt, hereby submit this original work as part of the requirements for the degree of Doctor of Philosophy in Mathematical Sciences.

It is entitled:

Enriched Discontinuous Galerkin Methods for Highly Oscillatory Differential Equations

Student's name: **Kevin T Schmitt**

This work and its defense approved by:

Committee chair: Benjamin Vaughan, Ph.D.

Committee member: Sookkyung Lim, Ph.D.

Committee member: Deniz Bilman, Ph.D.

47957

Enriched Discontinuous Galerkin Methods for Highly Oscillatory Differential Equations

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy

Department of Mathematical Sciences
College of Arts and Sciences
University of Cincinnati, March 2024

Author: Kevin Schmitt
Degrees: B.S. Mathematics, 2017,
University of Cincinnati

Chair: Benjamin Vaughan, Ph.D.
Committee: Deniz Bilman, Ph.D.
Sookyung Lim, Ph.D.

Abstract

This dissertation presents a comprehensive study of a novel enriched Discontinuous Galerkin (xDG) method, designed specifically for solving highly oscillatory differential equations, with a primary focus on its application to High-Intensity Focused Ultrasound (HIFU). HIFU is a medical procedure that employs ultrasound waves, ranging between 0.1 to 20 MHz, to target and ablate abnormal tissues within the body, serving as a motivating application for this research.

Central to this study is a comparative analysis between the proposed enriched Symmetric Interior Penalty Galerkin (xSIPG) method and its predecessors, highlighting the advancements in solving crucial differential equations, notably the Helmholtz and Bioheat equations. These equations are pivotal for understanding the propagation of ultrasound waves and their interaction with human tissues. A significant achievement of this research is the optimization of penalty parameters within the xDG framework, which plays an essential role in the accuracy and computational efficiency of the methods.

The results indicate xSIPG's significant improvement in modeling efficiency for HIFU simulations, potentially enhancing computational performance by up to three orders of magnitude compared to conventional FEM. Moreover, this study establishes the limitations of previous xDG methods in fully capturing the complexities of the HIFU model, a challenge addressed by the xSIPG approach. This advancement not only highlights the methodological leap facilitated by the xSIPG method but also reinforces the potential of applying xDG techniques to simulate HIFU.

Acknowledgments

First and foremost, my heartfelt appreciation goes to Professor Vaughan, whose expert guidance and unwavering support have been pivotal in my research. At a crucial point in my academic journey, during the summer before my 6th year, he saw the potential in me to complete the dissertation and generously took me on as a new student. His mentorship has profoundly shaped both my intellectual pursuits and personal growth. The insights he provided and the encouragement he offered have been invaluable. His ability to recognize my capabilities and nurture them has been a beacon of hope and a driving force behind my success. For his belief in me and for his invaluable guidance, I am deeply grateful.

I extend my sincere thanks to Professor French, with whom I've had the privilege of engaging in many discussions over coffee. Our conversations have sparked new ideas and perspectives, greatly contributing to the depth and breadth of this work. His wisdom and camaraderie were a valuable source of motivation.

To the vibrant communities within the Philosophy and Mathematics student groups, I owe a debt of gratitude. Their friendship, support, and intellectual camaraderie have formed a cornerstone of my journey. The shared experiences, challenges, and triumphs with them have not only made this intellectual endeavor more enriching but have also cultivated relationships that hold a special place in my heart.

Last but by no means least, I wish to express my love and gratitude to my dogs, Bailey and Max. Their unconditional love provided me with joy amidst the rigors of a PhD student's life.

Contents

Chapter 1

Introduction

1.1 Motivation

Finite element methods (FEM) are a class of numerical techniques used to solve partial differential equations (PDEs) over complex geometries. They have been widely used in various engineering and scientific fields such as elasto-dynamics, fluid mechanics, electromagnetics, and molecular dynamics [18] [17]. FEM offers high flexibility and accuracy in solving PDEs with arbitrary geometries and material properties. These methods discretize a domain into a finite number of subdomains, called elements, to approximate the solution within each element using a given set of local basis functions. The solutions within each element are then combined to form an approximation over the entire domain.

One domain where FEM has been applied is in the simulation of oscillatory phenomenon governed by the Helmholtz equation. This equation describes the propagation of waves in a medium and is commonly used to model oscillatory phenomena in a variety of applications such as medical imaging, molecular dynamics, ultrasound surgery, and quantum mechanics. The application we will have in mind is using high intensity focused ultrasound (HIFU) for ultrasound surgery. We will need to solve the Helmholtz equations

for high wave numbers ranging from one thousand to 20 million hertz [12]. However, when solving the Helmholtz equation with high frequency, FEM becomes computationally expensive and impractical due to the need for an ultra-fine mesh to resolve the wave behavior accurately [11]. There have been many attempts at generalizing FEM [2] to address this issue such as Galerkin least squares (GLS) and discontinuous Galerkin (DG) methods [**gls**]. We will focus our attention on the latter.

DG methods are an extension of finite element methods where the numerical solution is allowed to be discontinuous across element boundaries. Discontinuities emerge by dropping the smoothness constraint between interacting elements, resulting in an expansion of the spectrum of permissible basis functions and a genuine local scheme. By taking into account the analytic properties of the problem at hand when choosing the numerical flux between elements, DG methods provide more flexibility than classical FEM.

Enriched discontinuous Galerkin (xDG) [7] methods are a further extension of DG methods that improve the efficiency and accuracy of solving more complex problems, such as the (highly) oscillatory behavior inherent in the Helmholtz equation. xDG methods achieve this by enriching the approximation space according to a priori knowledge of the problem at hand. In the case of Helmholtz, one enriches the local approximation space with appropriate wave functions to capture the behavior of oscillatory phenomena, allowing for a more efficient numerical solution by significantly reducing the required number of elements to achieve a desired level of accuracy.

The use of xDG methods has several advantages over other generalizations of FEM. Firstly, xDG methods offer an ease of implementation, it is quite simple to modify existing FEM implementations by treating each element as a separate FEM problem and incorporating the enriched basis. Secondly, they provide a more flexible and robust framework for handling complex geometries and material properties. Finally, xDG methods are enormously more efficient simulations than traditional FEM methods, particularly for problems with high wave numbers.

Overall, xDG methods are a promising approach for modeling highly oscillatory phenomena, and ongoing research is focusing on further refining this method for more efficient and accurate simulations in a variety of practical applications.

1.2 Contributions

In this work, we propose an enriched discontinuous Galerkin (xDG) method for simulating oscillatory behavior in high frequency regimes with an application aimed at high intensity focused ultrasound (HIFU) surgery. Our method involves enriching the approximation space with appropriate wave functions to capture the a priori behavior of the solution.

We show that the XDG method significantly reduces the computational cost compared to traditional finite element methods (FEM) while maintaining high accuracy. Furthermore, we show that our xDG framework is an improvement over previous xDG attempts, opening up the genuine possibility of using the method for a general oscillatory problem. Lastly, we test an implementation on the method in a simplified one dimensional HIFU model.

Overall, our work contributes to advancing the field by offering an efficient and accurate method for solving highly oscillatory phenomena. We benchmark the method on applicable problems and showcase the method's applicability and ease of implementation. Our approach provides a flexible and robust framework for handling complex geometries and problem specific behavior, while maintaining high accuracy and reducing computational cost.

Chapter 2

Background

2.1 The Finite Element Method

The finite element method (FEM) [13] is a numerical scheme used to solve partial differential equations (PDEs) by discretizing the domain into a collection of polygons called finite elements. The method has gained significant popularity due to its ability to provide accurate solutions to complex problems with arbitrary geometries.

At its core, the FEM involves the approximation of a smooth problem on a discrete domain by constructing a system of linear equations that can be solved numerically. This is achieved by dividing the domain into a finite number of simple geometric shapes, called elements. The solution is then obtained as a combination of basis functions defined over these finite elements. The values of the solution at finitely many nodes of each element are approximated by the values of the basis functions at these nodes. The collection of all these nodal values forms the numerical solution to the discrete problem.

The FEM involves three key steps: discretization of the domain, approximation by basis functions, and numerical solution. In the discretization step, the domain is divided into finite elements, and nodes are typically defined at the vertices of these elements. The approximation step involves the selection of basis functions, which represent the

numerical solution over each element. The solution step involves solving the system of linear equations obtained from the discretized problem.

2.1.1 Notation

Let $\Omega \subset \mathbb{R}^d$ be a domain of interest in d-dimensional Euclidean space, and let $L^2(\Omega)$ be the space of all square integrable real-valued functions on Ω. L^2 is equipped with the inner product $(\cdot, \cdot)$ and norm defined as follows:

$$(v, w) = \int_\Omega v(x)w(x)dx, \qquad \|v\|_2 = \sqrt{(v, v)}$$

We will use $H^1(\Omega)$ to denote the Sobolev subspace of V, consisting of those functions that have all their first-order partial derivatives square integrable. $H^1(\Omega)$ is equipped with the energy norm given by:

$$\|v\|_{H^1} = \sqrt{\sum_{|\alpha|\leq 1} \|D^\alpha v\|_2^2}$$

where $\alpha = (\alpha_1, \alpha_2, \alpha_3)$ is a multi-index, and D^α denotes the αth weak derivative of v. Alternatively, the H^1 energy norm is the square root of the sum of the square of the L^2 norm of the original function and the square of the L^2 norm of all its first-order partial derivatives. In one dimension, the norm reduces to:

$$\|v\|_{H^1} = \sqrt{\|v\|_2^2 + \|v'\|_2^2}$$

2.1.2 Discretization

In the finite element method, the continuous domain Ω is divided into smaller subdomains (called elements), and the equations governing the behavior of the problem are approximated within these elements. The discretization step is handled by choosing an appropriate mesh for the geometry Ω. The mesh should capture the important features

of the geometry, such as boundaries and interfaces, and it should be able to accurately represent the solution within the desired level of accuracy. In one dimension, equally spaced intervals are commonly used due to their simplicity and ease of implementation, but other spacing schemes may also be appropriate depending on the problem at hand. In two dimensions, triangles or quadrilaterals are common (figure 2.1), and tetrahedrons for three dimensions. The choice of discretization and the methods used to compute them have been extensively studied, and falls under the subject of computational geometry which is outside the scope of this dissertation. We will simply use the popular Delaunay triangulization in higher dimensions and uniformly spaced nodes for one dimension in our MATLAB implementation and proceeding discussion. We employ the notation $\mathcal{K} \subset \Omega$ for our discretization of the geometry Ω.

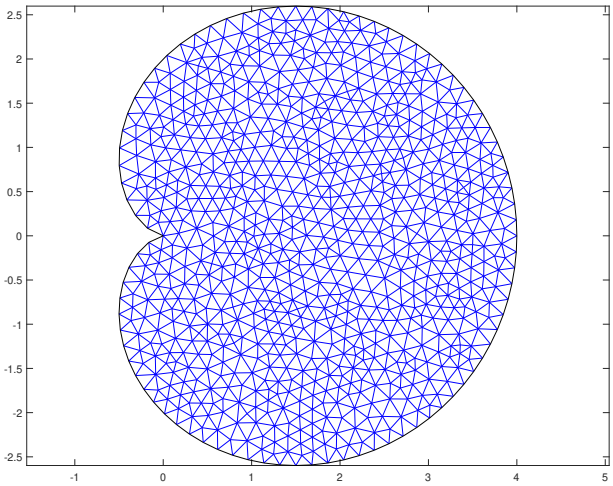


Figure 2.1 Delaunay Triangulization $\mathcal{K}$ of a Cardiod Ω

2.1.3 Formulation

After discretizing the domain Ω into a set of elements $\mathcal{K}$, the next step in the finite element method is to approximate the solution of the problem within these elements. The approximation is constructed using a set of basis functions $\{\phi_i\}_{i=1}^{N}$, where N is the number of nodes in the mesh. The basis functions are chosen such that they have local support, meaning that they are nonzero only within the elements containing the corresponding node.

The approximate solution u_h of the problem is then represented as a linear combination of these basis functions:

$$u_h(x) = \sum_{j=1}^{N} u_j \phi_j(x), \quad x \in \Omega$$

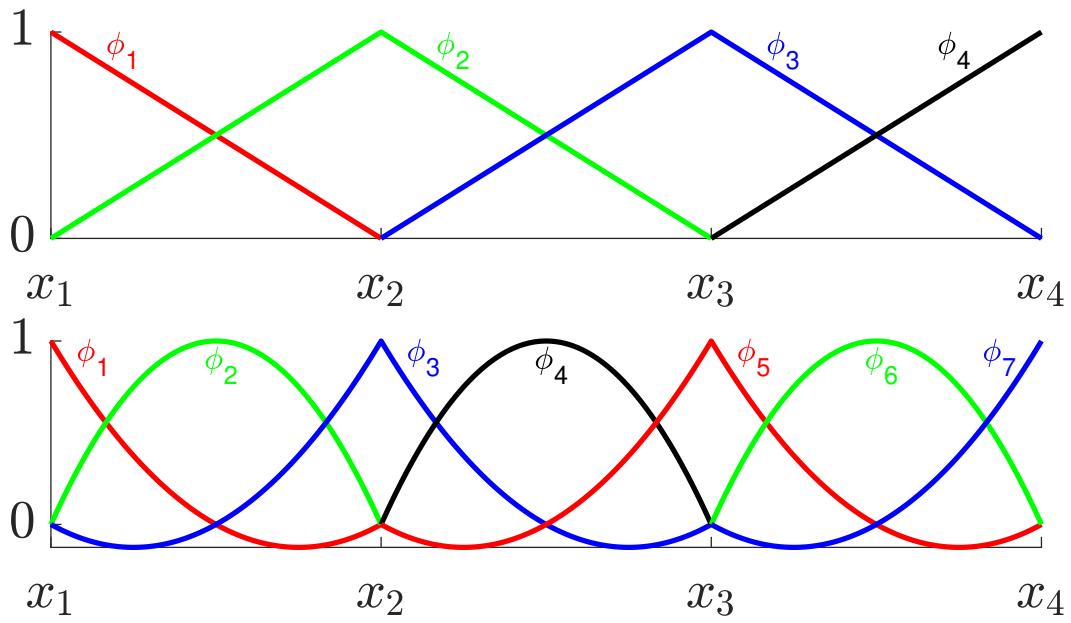


Figure 2.2 (Top) Order 1 Lagrange basis functions on a discretization of an interval using 4 nodes. (Bottom) Order 2 Lagrange basis functions on the same discretization, note the additional nodes at the midpoint of the elements

Here, u_j are unknown coefficients to be determined, and $\phi_j(x)$ are the basis functions. The choice of basis functions is crucial, as it influences the accuracy and stability of the numerical solution. Some common choices of basis functions include Lagrange polynomials,

piecewise linear functions, and Bessel functions. We will use order 1 Lagrange polynomials, which are defined at each node as being the piecewise linear defined as 1 at that specific node and 0 at neighboring nodes (Figure 2.2). One can extend this idea to higher orders; in the case of order 2, piecewise quadratics are used.

To find the coefficients u_i, the problem is formulated into a weak form by multiplying the governing equation with a test function $v \in V$ and integrating by parts over the domain Ω. The weak form ensures that the problem is well-posed and can be solved using variational methods.

The weak form is used to formulate the variational version of our initial problem and can be formulated generally as follows. Let $\mathbf{a}(\cdot, \cdot)$ be a bilinear form on V and $\mathbf{l}(\cdot)$ be a continuous linear functional. Then the abstract variational problem takes the form: find $u \in V$ such that for each $v \in V$

$$\mathbf{a}(u, v) = \mathbf{l}(v)$$

The famous Lax-Milgram lemma guarantees the uniqueness of the solution to the variational problem as long as $\mathbf{a}(\cdot, \cdot)$ is continuous and coercive.

Each differential equation will have a slightly different bilinear form associated with it. For illustration purposes, let us consider the general second-order elliptic linear PDE:

$$Lu = f, \quad \text{in } \Omega$$

$$u = 0, \quad \text{on } \partial\Omega$$

Where L is the second-order differential operator

$$Lu = -\nabla \cdot (a\nabla u) + b \cdot \nabla u + cu$$

To obtain the weak form we multiply the equation $Lu = f$ by a test function v in the subspace V of $H^1(\Omega)$, then integrate by parts to obtain the weak form defined by

$$\begin{aligned}\mathbf{a}(u, v) &= (a\nabla u, \nabla v) + (b \cdot \nabla u, v) + (cu, v) \\ \mathbf{l}(v) &= (f, v)\end{aligned}$$

To further simplifyok , let us consider the one dimensional case where $Lu = -u'' + u$ over the domain $\Omega = [0, 1]$. To obtain the bilinear form $\mathbf{a}(\cdot, \cdot)$, we multiply Lu by a test function v and integrate, followed by integrating by parts:

$$\mathbf{a}(u, v) = -\int_0^1 u''v + \int_0^1 uv = \int_0^1 u'v' + \int_0^1 uv - u'(1)v(1) + u'(0)v(0)$$

To simply our first example, let us suppose that the problem at hand has homogeneous Neumann boundary conditions. Then the resulting variational formulation is

$$\begin{aligned}\mathbf{a}(u, v) &= \int_0^1 u'v' + \int_0^1 uv \\ \mathbf{l}(v) &= \int_0^1 fv\end{aligned}$$

It is widely known that such weak forms will have a unique solution due to Lax-Milgram. Thus, we seek to compute the solution by restricting our attention to an approximation space of V, which corresponds to our choice of basis functions and discretization of the domain into finite elements. Let V_h be one such finite dimensional subspace of V. The finite element problem becomes: find $u_h \in V_h$ such that for each $v \in V_h$

$$\mathbf{a}(u_h, v) = \mathbf{l}(v)$$

2.1.4 Approximation Space and Numerical Solution

Let $\{\phi_i\}_{i=1}^N$ be a basis for the approximation space V_h. The finite element problem can be equivalently expressed as find $u_h \in V_h$ such that for each basis function ϕ_j:

$$\mathbf{a}(u_h, \phi_j) = \mathbf{l}(\phi_j), \text{ for } j = 1, ..., N$$

Applying the fact that $u_h \in V_h$ itself, we obtain the system of linear equations given by:

$$b_j = \mathbf{l}(\phi_j) = \sum_{j=1}^{N} u_j \mathbf{a}(\phi_i, \phi_j) = \sum_{j=1}^{N} A_{ij} u_j$$

This results in the linear system $Au = b$, where A is the matrix corresponding to the bilinear form $\mathbf{a}(\cdot, \cdot)$ by $A_{ij} = \mathbf{a}(\phi_i, \phi_j)$, u is the vector of unknown coefficients u_j from which we can construct u_h, and b is the load vector corresponding to the linear functional $\mathbf{l}(\cdot)$. The matrix A and vector b are determined by evaluating the integrals in the weak form using the chosen basis functions.

Reference Element

In order to better handle the many basis functions we will inevitably be working with (in the thousands or more), we parametrize each basis function with respect to the variable s ranging in the interval $[-1, 1]$. This allows us to work over a canonical space for computing integrals and values for the basis functions in the weak form. The interval $\hat{E} = [-1, 1]$ is called the *reference* element. In the case of two dimensions, the reference element will be a suitably chosen triangle or square, and for three dimensions one would have a reference tetrahedron.

For illustrative purposes, let us come back to the specific problem $Lu = -u'' + u$ and $\Omega = [0, 1]$. We will discretize the domain into 3 elements E_1, E_2, and E_3 using 4 nodes x_1, x_2, x_3, and x_4 just as in Figure 2.2. We refer to the initial elements $E_j = [x_j, x_{j+1}]$

as the *physical* elements. The parametrization of the physical element E_k with respect to the reference element is given by the affine mapping $\mathbb{A}_j : \hat{E} \to E_j$, sending $s \to x$, is given below

$$\mathbb{A}_j(s) = x_j\xi_1(s) + x_{j+1}\xi_2(s), \text{ where } \xi_1(s) = \frac{1-s}{2} \text{ and } \xi_2(s) = \frac{1+s}{2}$$
$$\mathbb{A}_j(s) = x = \frac{(x_{j+1} - x_j)s + x_{j+1} + x_j}{2} = \frac{h_j s + x_{j+1} + x_j}{2}$$

We have set h_j to be the size of the element E_j. In our proceeding work we assume a uniformly spaced discretization, so we simplify h_j to just h. Note that ξ_1 is the basis function on $\hat{E}$ which is 1 on the left and 0 on the right; and ξ_2 vice versa. We will call ξ_i the *reference* basis functions. Additionally, since FEM requires continuity at interior nodes, the physical basis elements at these nodes, ϕ_i, will gain contributions from both the left (ξ_1) and right (ξ_2) interfacing elements when we compute the weak form using reference elements.

We note here that the inversion of $\mathbb{A}_j$ is given by the mapping $x \to s$:

$$s = \frac{2x - x_{j+1} - x_j}{h}$$

The Shape of the Linear System

With our physical space set up as Figure 2.2, let's compute the resulting $4x4$ linear system $Au = b$ and take note of its general shape. The shape of the matrix A is given by $A_{ij} = a(\phi_i, \phi_j)$, resulting in

$$A = \begin{bmatrix} \mathbf{a}(\phi_1, \phi_1) & \mathbf{a}(\phi_1, \phi_2) & \mathbf{a}(\phi_1, \phi_3) & \mathbf{a}(\phi_1, \phi_4) \\ \mathbf{a}(\phi_2, \phi_1) & \mathbf{a}(\phi_2, \phi_2) & \mathbf{a}(\phi_2, \phi_3) & \mathbf{a}(\phi_2, \phi_4) \\ \mathbf{a}(\phi_3, \phi_1) & \mathbf{a}(\phi_3, \phi_2) & \mathbf{a}(\phi_3, \phi_3) & \mathbf{a}(\phi_3, \phi_4) \\ \mathbf{a}(\phi_4, \phi_1) & \mathbf{a}(\phi_4, \phi_2) & \mathbf{a}(\phi_4, \phi_3) & \mathbf{a}(\phi_4, \phi_4) \end{bmatrix}$$

But referring to Figure 2.2, and taking note that

$$\mathbf{a}(\phi_i, \phi_j) = \int_0^1 \phi_i' \phi_j' + \int_0^1 \phi_i \phi_j$$

We see that ϕ_1 makes zero contribution with ϕ_3 and ϕ_4, and vice versa, and similarly ϕ_2 makes zero contributions with ϕ_4 and vice versa. This results in the general shape of A taking that of a banded matrix with nonzero entries just above and below the main diagonal:

$$A = \begin{bmatrix} \mathbf{a}(\phi_1, \phi_1) & \mathbf{a}(\phi_1, \phi_2) & 0 & 0 \\ \mathbf{a}(\phi_2, \phi_1) & \mathbf{a}(\phi_2, \phi_2) & \mathbf{a}(\phi_2, \phi_3) & 0 \\ 0 & \mathbf{a}(\phi_3, \phi_2) & \mathbf{a}(\phi_3, \phi_3) & \mathbf{a}(\phi_3, \phi_4) \\ 0 & 0 & \mathbf{a}(\phi_4, \phi_3) & \mathbf{a}(\phi_4, \phi_4) \end{bmatrix}$$

Let's compute the local contributions from the element E_j using the reference element $\hat{E}$. After the change of variables $x \to s$ and taking into account that the determinant of the Jacobian is $\frac{2}{h}$, this amounts to computing the matrix

$$A_j = \frac{2}{h} \begin{bmatrix} \mathbf{a}(\xi_1(s), \xi_1(s)) & \mathbf{a}(\xi_1(s), \xi_2(s)) \\ \mathbf{a}(\xi_2(s), \xi_1(s)) & \mathbf{a}(\xi_2(s), \xi_2(s)) \end{bmatrix} = \frac{2}{h} \begin{bmatrix} \int_{-1}^1 [(\xi_1')^2 + \xi_1^2] & \int_{-1}^1 [\xi_1' \xi_2' + \xi_1 \xi_2] \\ \int_{-1}^1 [\xi_1' \xi_2' + \xi_1 \xi_2] & \int_{-1}^1 [(\xi_2')^2 + \xi_2^2] \end{bmatrix}$$

The global matrix A is then constructed with the following shape

$$A = \begin{bmatrix} A_1(1,1) & A_1(1,2) & 0 & 0 \\ A_1(2,1) & A_1(2,2) + A_2(1,1) & A_2(1,2) & 0 \\ 0 & A_2(2,1) & A_2(2,2) + A_3(1,1) & A_3(1,2) \\ 0 & 0 & A_3(2,1) & A_3(2,2) \end{bmatrix}$$

Now let's compute the local contributions b_j arising from element E_j to the target vector b. After the change of variables $x \to s$, the local target vector is given by

$$b_j = \frac{2}{h}\begin{bmatrix} \mathbf{l}(\xi_1) \\ \mathbf{l}(\xi_2) \end{bmatrix} = \frac{2}{h}\begin{bmatrix} \int_0^1 f\xi_1 \\ \int_0^1 f\xi_2 \end{bmatrix}$$

Keeping in mind that FEM requires continuity of the solution vector, the global target vector is then constructed by piecing together the local contributions in the following manner:

$$b = \begin{bmatrix} b_1(1) \\ b_1(2) + b_2(1) \\ b_2(2) + b_3(1) \\ b_3(2) \end{bmatrix}$$

Once the coefficients u_j are determined by solving the linear system $Au = b$, the approximate solution u_h can be reconstructed using the basis functions. The solution can then be evaluated at any point within the domain by evaluating the linear combination of basis functions at that point.

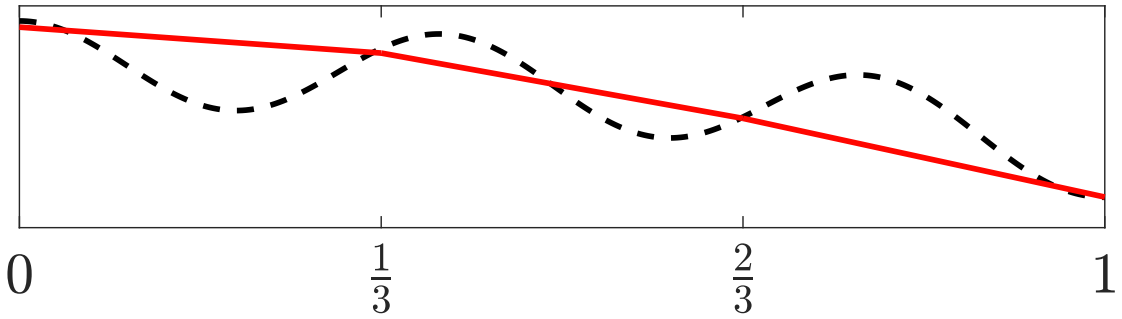


Figure 2.3 FEM solution u_h (red curve) to the problem $u'' - u = \cos(16\pi x)$ with homogeneous Neumann boundary conditions. The black dashed curve shows the exact solution. 3 linear elements were used.

2.2 The Discontinuous Galerkin Method

The discontinuous Galerkin (DG) method [15][8] is an extension of the finite element method, which allows for discontinuities in the approximate solution across element boundaries. This feature leads to increased flexibility and robustness for certain types of problems, especially those with highly varying or discontinuous solutions. In this section, we will discuss the formulation of the DG method, some popular penalization techniques, and the advantages and applications of the method.

2.2.1 Formulation

The primary difference between the DG method and the finite element method lies in the choice of the finite-dimensional subspace V_h and the formulation of the weak form. In the DG method, each element has its own set of local basis functions, and there is no requirement for continuity across element boundaries. Consequently, there will be two degrees of freedom at each interior node rather than the one in FEM, leading to increased flexibility in the representation of the solution.

Since we will be working with the boundaries of elements in the interior of the domain Ω frequently, let us introduce the notation Γ_h for the set of boundaries of interior elements. In one dimension, Γ_h contains exactly the interior nodes of the interval Ω (Ω need not be an interval but it helps illimunate the situation at hand). In higher dimensions, it includes the interior edges or faces and excludes $\partial\Omega$. Lastly, with each $e \in \Gamma_h$, we associate a unit normal vector $\mathbf{n}_e$.

The finite dimensional subspace V_h is chosen as the set of functions whose restriction to each element E of $\mathcal{K}$ is contained in $H^1(E)$. To derive the weak form of the problem, we first multiply the governing equation with a test function v over the broken Sobolev subspace $V_h \subset H^1(\Omega)$ and integrate over each element $E \in \mathcal{K}$ (it is important to note here that the integration is taking place over E rather than Ω, as is done in FEM). For

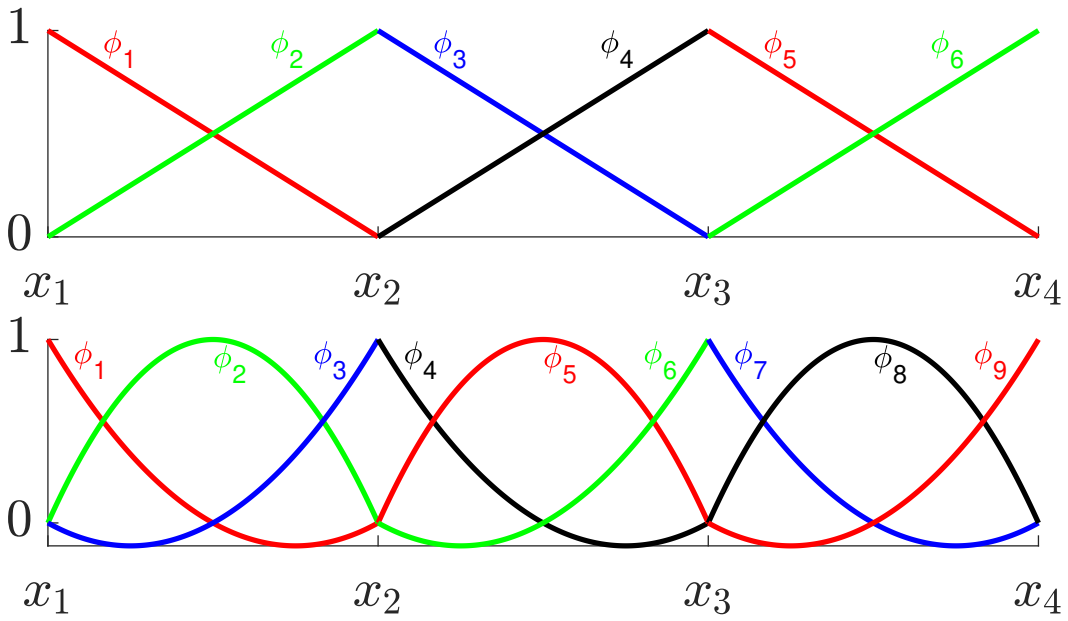


Figure 2.4 (Top) Order 1 DG Lagrange basis functions on a discretization of an interval using 4 nodes. (Bottom) Order 2 DG Lagrange basis functions on the same discretization. Note that the basis functions split at the interior node points giving rise to an extra degree of freedom there.

simplification, we take the second order differential operator $Lu = -\nabla \cdot a\nabla u + cu$.

$$\int_E -\nabla \cdot (a\nabla u)v + \int_E cuv = \int_E fv$$

Next, we integrate by parts, as in the finite element method, to obtain a local weak form. However, since the solution is allowed to be discontinuous, additional terms, known as jump terms, arise from the boundary integrals of the integration by parts process.

$$\int_E a\nabla u \cdot \nabla v - \int_{\partial E} \nabla u \cdot \mathbf{n}_E v + \int_E cuv = \int_E fv$$

In contrast to the finite element method, where the bilinear form $a(u_h, v)$ consists of only full volume integrals, the discontinuities in the DG method introduce boundary terms in the solution. In addition to the naturally arising boundary terms, there have been numerous proposals to add additional terms to ensure stability and accuracy of the

solution. The local weak form is then summed over all elements to construct the global weak form.

Let's go through the same illustrating example as we did in FEM and observe the shape of the resulting linear system corresponding to the variational formulation of the problem according to DG. The problem of interest is in solving $-u'' + u = f$ with homogeneous Nuemann boundary conditions. We discretize the domain $\Omega = [0, 1]$ into three equally spaced elements E_1, E_2, and E_3. Since DG is locally formulated the use of the reference element $\hat{E} = [-1, 1]$ in computations is straightforward. However, compared to FEM, now we have a total of six basis function ϕ_i to handle.

To compute the resulting linear matrix A, we break the problem up into computing only the contributions to A from each element $\{E_j\}_{j=1}^n$ and store them in the matrix A_j. Due to the local formulation of DG, this approach is more reasonable since each basis function is only nonzero within exactly one element. In our current situation, the variational formulation simplifies to

$$\int_{E_j} u'v' + \int_{E_j} uv = \int_{E_j} fv$$

Thus the local contributions appear as

$$A_j = \begin{bmatrix} \int_{E_j}[(\phi_1')^2 + \phi_1^2] & \int_{E_j}[\phi_1'\phi_2' + \phi_1\phi_2] \\ \int_{E_j}[\phi_1'\phi_2' + \phi_1\phi_2] & \int_{E_j}[(\phi_2')^2 + \phi_2^2] \end{bmatrix} = \frac{2}{h} \begin{bmatrix} \int_{-1}^{1}[(\xi_1')^2 + \xi_1^2] & \int_{-1}^{1}[\xi_1'\xi_2' + \xi_1\xi_2] \\ \int_{-1}^{1}[\xi_1'\xi_2' + \xi_1\xi_2] & \int_{-1}^{1}[(\xi_2')^2 + \xi_2^2] \end{bmatrix}$$

The global matrix A is constructed as the $6x6$ block diagonal matrix with diagonals A_k due to the relaxing of continuity criterion at the interface of elements.

$$A = \begin{bmatrix} A_1 & 0 & 0 \\ 0 & A_2 & 0 \\ 0 & 0 & A_3 \end{bmatrix}$$

The target vector b is straightforward to compute

$$b = \begin{bmatrix} \int_{E_j} f\phi_1 \\ \int_{E_j} f\phi_2 \\ \int_{E_j} f\phi_3 \\ \int_{E_j} f\phi_4 \\ \int_{E_j} f\phi_5 \\ \int_{E_j} f\phi_6 \end{bmatrix} = \frac{2}{h} \begin{bmatrix} \int_{-1}^{1} f\xi_1 \\ \int_{-1}^{1} f\xi_2 \\ \int_{-1}^{1} f\xi_1 \\ \int_{-1}^{1} f\xi_2 \\ \int_{-1}^{1} f\xi_1 \\ \int_{-1}^{1} f\xi_2 \end{bmatrix}$$

2.2.2 Penalizing The Numerical Solution

Observe that there are no continuity conditions in place in our current formulation, manifesting itself in the linear system A having the shape of a block diagonal matrix; the basis functions are free to differ from their neighbor where they meet at a node point, resulting in the possibility of *jumps* at the interface between elements. To stabilize and ensure accuracy of the numerical solution, penalization techniques are used in the DG method to handle the discontinuities in the approximation. Two popular approaches we focus on are the Babuška-Zlámal (BZ) [1][3] and the Symmetric Interior Penalty Galerkin (SIPG) [5] methods.

The discontinuities allowed in the formulation lead to the fact that the solution has two traces along the boundary of any two interfacing elements E^l and E^r, so we can add or subtract those values. In order to be mathematically precise about the behaviors of these discontinuities we introduce some notation; let $\{\cdot\}$ and $[\cdot]$ be the average and jump operators respectively defined by

$$\{v\} = \frac{v|_{E^l} + v|_{E^r}}{2} \text{ and } [v] = v|_{E^l} - v|_{E^r}, \quad \forall e \in E^l \cap E^r$$

In one-dimension, the average is simply the average at the node from the left and right elements, while the jump is the difference between the two. Let x be the node

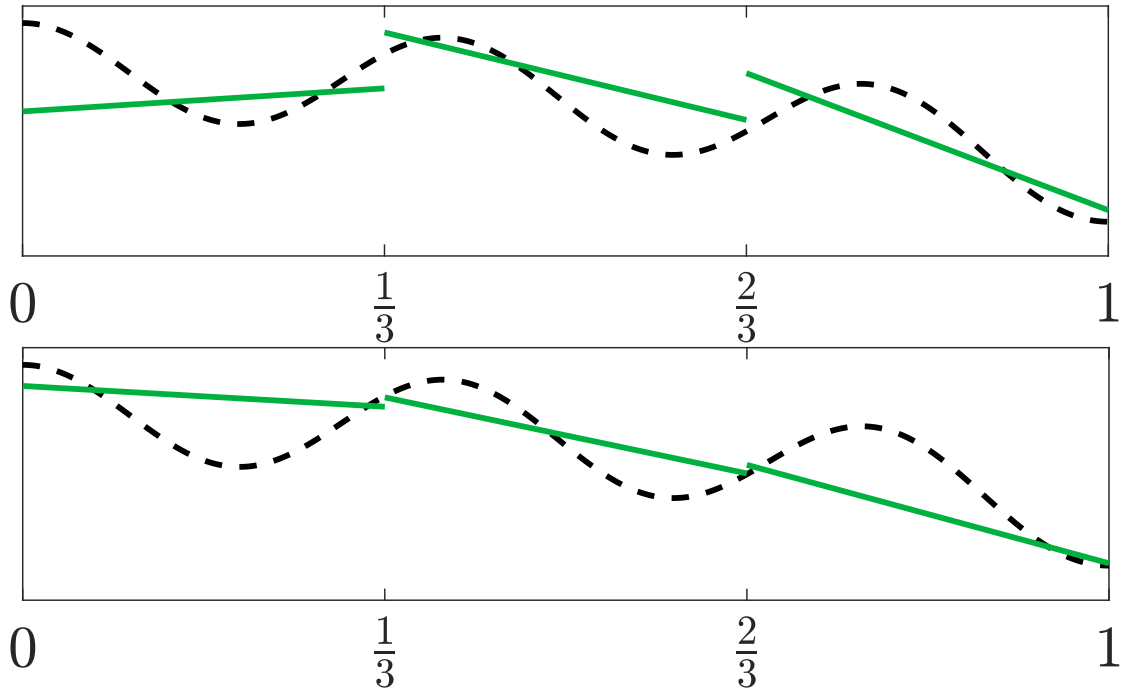


Figure 2.5 DG solutions u_h (green curve) to the problem $u'' - u = \cos(16\pi x)$ with homogeneous Neumann boundary conditions. Top figure is with an insubstantial penalty and the bottom figure is with a more substantial penalty. The black dashed curve shows the exact solution. 3 linear elements were used.

point between the two elements and denote by $v(x^+) = \lim_{\epsilon \to 0} v(x + \epsilon)$ and $v(x^-) = \lim_{\epsilon \to 0} v(x - \epsilon)$, then the jump and average terms are

$$\{v(x)\} = \frac{v(x^+) + v(x^-)}{2} \text{ and } [v(x)] = v(x^-) - v(x^+)$$

The several different ways of choosing the penalization method can be concisely discussed by introducing the jump bilinear form, $J_{\sigma,\alpha}$, defined as follows:

$$J_{\sigma,\alpha}(u, v) = \sum_{e \in \Gamma_h \cup \partial\Omega} \int_e \frac{\sigma}{|e|^\alpha} [u][v]$$

The values σ and α are penalty parameters and will be specified later. There are a variety of different methods for penalizing the jumps in the numerical solution. We will focus attention on three discontinuous Galerkin methods as a foundation; Babuška-Zlámal

(BZ), Incomplete Interior Penalty Galerkin (IIPG), and the Symmetric Interior Penalty Galerkin (SIPG) methods. In the one dimensional case, the jump operator simplifies to

$$J_{\sigma,\alpha}(u,v) = \sum_{j=1}^{n+1} \frac{\sigma}{h}[u(x_j)][v(x_j)]$$

Babuška-Zlámal Method (BZ)

The BZ method introduces a penalty term to the weak form of the problem in perhaps the most simple, yet naive, way. In the BZ method, the naturally arising boundary terms are ignored and the jumps in the solution are penalized solely using the jump penalty form with $\alpha = 3$, $J_{\sigma,3}$ [6]. Summing over all elements in the mesh and ignoring the naturally arising boundary terms, the BZ weak formulation of the problem is to compute the unique solution $u_h \in V_h$ which satisfies for each $v \in V_h$,

$$\sum_{E\in\mathcal{K}} \int_E a\nabla u_h \cdot \nabla v + \sum_{E\in\mathcal{K}} \int_E c u_h v + J_{\sigma,3}(u_h, v) = \sum_{E\in\mathcal{K}} \int_E fv$$

For our running example $u'' + u = f$; the edges in the jump term are simply the boundary points of E_j and the formulation requires us to compute u_h which satisfies

$$\sum_{j=1}^{n}\Big[\int_{E_j} u_h' v' + \int_{E_j} u_h v\Big] + \sum_{j=1}^{n+1} \frac{[u_h](x_j)[v](x_j)}{h^3} = \sum_{j=1}^{n} \int_{E_j} fv$$

The additional terms are only the jump terms involving $[\cdot]$. Let's see how this changes the shape of the linear system. We collect all the terms into what we call the penalty matrix P. Let's compute the local penalty matrix P_j corresponding to the penalty contributions

from the interior nodes x_j for $j = 2, 3$:

$$P_j = \frac{\sigma}{h^3}\begin{bmatrix} [\phi_{2j}(x_j)][\phi_{2j}(x_j)] & [\phi_{2j}(x_j)][\phi_{2j+1}(x_j)] \\ [\phi_{2j}(x_j)][\phi_{2j+1}(x_j)] & [\phi_{2j+1}(x_j)][\phi_{2j+1}(x_j)] \end{bmatrix}$$
$$= \frac{\sigma}{h^3}\begin{bmatrix} \phi_{2j}(x_j^-)\phi_{2j}(x_j^-) & -\phi_{2j}(x_j^-)\phi_{2j+1}(x_j^+) \\ -\phi_{2j}(x_j^-)\phi_{2j+1}(x_j^+) & \phi_{2j+1}(x_j^+)\phi_{2j+1}(x_j^+) \end{bmatrix} = \frac{\sigma}{h^3}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$$

Note that P_j is offset from A_j, entering into the global linear system in the rows and columns corresponding to $2, 3$ for $j = 2$ and $4, 5$ for $j = 3$. This will result in a global linear system with a banded matrix that is no longer block diagonal due to the interaction of basis functions between interfacing elements. Thus the global penalty matrix takes the shape

$$P = \frac{\sigma}{h^3}\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Finally, the BZ formulation is put together in the global linear system $Su = b$, where $S = A + P$. We denote by a to indicate a contribution from the matrix A and a p to indicate a contribution to the shape from the penalty matrix P. With this notation, S has the following shape

$$S = \begin{bmatrix} a & a & 0 & 0 & 0 & 0 \\ a & a & p & 0 & 0 & 0 \\ 0 & p & a & a & 0 & 0 \\ 0 & 0 & a & a & p & 0 \\ 0 & 0 & 0 & p & a & a \\ 0 & 0 & 0 & 0 & a & a \end{bmatrix}$$

Incomplete Interior Penalty Galerkin (IIPG)

In contrast with BZ, the IIPG method [14] retains the naturally arising boundary terms in addition to penalizing by the jump form $J_{\sigma,1} := J_\sigma$. Summing the boundary integrals over all elements $E \in \mathcal{K}$ and employing jump notation, we obtain the equation

$$\sum_{E\in\mathcal{K}} \int_{\partial E} a\nabla u \cdot \mathbf{n}_e v = \sum_{e\in\Gamma_h\cup\partial\Omega} \int_e [a\nabla u \cdot \mathbf{n}_e v]$$

Since the exact solution u is sufficiently smooth, we know that $\{a\nabla u \cdot \mathbf{n}_e\} = a\nabla u \cdot \mathbf{n}_e$. Thus, we obtain the resulting variational form for IIPG

$$\mathbf{a}(u,v) = \sum_{E\in\mathcal{K}} \int_E a\nabla u \cdot \nabla v + \sum_{E\in\mathcal{K}} \int_E cuv - \sum_{e\in\Gamma_h\cup\partial\Omega} \int_e \{a\nabla u \cdot \mathbf{n}_e\}[v] - J_\sigma(u,v)$$

$$\mathbf{l}(v) = \sum_{E\in\mathcal{K}} \int_E fv$$

For the one dimensional case and our running example, the blinear form $\mathbf{a}$ simplifies to

$$\mathbf{a}(u,v) = \sum_{j=1}^{n} \int_{E_j} \Big[u'v' + uv\Big] - \sum_{j=1}^{n+1} \{u'\}(x_j)[v](x_j) + \frac{\sigma}{h} \sum_{j=1}^{n+1} [u](x_j)[v](x_j)$$

Therefore, to discern the shape of the resulting linear system we only need to calculate the contributions arising from $\{u'\}[v]$. We collect all the terms into a matrix I. The local contributions from these terms arising from each interior node x_r will add a nonzero term $\{\phi_i'\}[\phi_j]$ when $i = 2r-3, 2r-2, 2r-1, 2r$ and $j = 2r-2, 2r-1$; these are the basis functions which are neighbors to the node x_r, it may be seen that $[\phi_j](x_r) = 0$

when $j = 2r-3, 2r$. Hence the local matrix is

$$I_j = -\begin{bmatrix} \{\phi'_{2j-3}\}[\phi_{2j-2}] & \{\phi'_{2j-3}\}[\phi_{2j-1}] \\ \{\phi'_{2j-2}\}[\phi_{2j-2}] & \{\phi'_{2j-2}\}[\phi_{2j-1}] \\ \{\phi'_{2j-1}\}[\phi_{2j-2}] & \{\phi'_{2j-1}\}[\phi_{2j-1}] \\ \{\phi'_{2j}\}[\phi_{2j-2}] & \{\phi'_{2j}\}[\phi_{2j-1}] \end{bmatrix} = \frac{1}{h}\begin{bmatrix} 1 & -1 \\ -1 & 1 \\ 1 & -1 \\ -1 & 1 \end{bmatrix}$$

The global IIPG terms are combined together into the following matrix in our running example with $n = 3$,

$$I = \frac{1}{h}\begin{bmatrix} 0 & 1 & -1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 \\ 0 & 1 & -1 & 1 & -1 & 0 \\ 0 & -1 & 1 & -1 & 1 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 \end{bmatrix}$$

When we add I to the global system S, denoting its contributions by i, we obtain the following shape for IIPG

$$S = \begin{bmatrix} a & a & i & 0 & 0 & 0 \\ a & a & p & 0 & 0 & 0 \\ 0 & p & a & a & i & 0 \\ 0 & i & a & a & p & 0 \\ 0 & 0 & 0 & p & a & a \\ 0 & 0 & 0 & i & a & a \end{bmatrix}$$

Symmetric Interior Penalty Galerkin (SIPG)

The SIPG method can be viewed as an extension of the IIPG method, as it incorporates additional boundary terms to the weak form to help stablize the solution. Since the exact

solution u is continuous, it follows that $[u] = 0$, and hence,

$$\sum_{e \in \Gamma_h \cup \partial\Omega} \int_e \{a\nabla v \cdot \mathbf{n}_e\}[u] = 0$$

However, the approximate solution will have jump discontinuites so it turns out that adding these terms to the weak form is not trivial; rather it helps to stabilize numerical properties in the hope for a more robust method. We will see later how this small tweak in the formulation will lead to a wider potential application of problems. The SIPG weak form is then to find the unique solution $u \in V_h$ such that for each $v \in V_h$, $\mathbf{a}(u, v) = \mathbf{l}(v)$, where

$$\begin{aligned}\mathbf{a}(u,v) = &\sum_{E\in\mathcal{K}} \int_E a\nabla u \cdot \nabla v + \sum_{E\in\mathcal{K}} \int_E cuv - \sum_{e\in\Gamma_h\cup\partial\Omega} \int_e \{a\nabla v \cdot \mathbf{n}_e\}[u] \\ &- \sum_{e\in\Gamma_h\cup\partial\Omega} \int_e \{a\nabla u \cdot \mathbf{n}_e\}[v] - J_0^{\sigma_0,\alpha_0}(u,v)\end{aligned}$$

$$\mathbf{l}(v) = \sum_{E\in\mathcal{K}} \int_E fv$$

For our one dimensional example, this reduces to

$$\mathbf{a}(u,v) = \sum_{j=1}^{n} \int_{E_k} \Big[u'v' + uv\Big] - \sum_{j=1}^{n+1} \Big[\{v'\}(x_j)[u](x_j) + \{u'\}(x_j)[v](x_j)\Big] + \frac{\sigma}{h} \sum_{j=1}^{n+1} [u](x_j)[v](x_j)$$

To determine the shape of the resulting linear system we just need to finish by computing the contributions from $\{v'\}[u]$. Again, we collect the terms in the matrix X. The local contributions from these terms arising from each interior node x_r will add a nonzero term $\{\phi_j\}[\phi_i]$ when $i = 2r-2, 2r-1$ and $j = 2r-3, 2r-2, 2r-1, 2r$. Hence the local matrix is given by

$$X_j = (I_j)^T$$

Therefore when X is added to the global system S and denoting its contributions by x, we obtain the following (symmetric) shape for SIPG

$$S = \begin{bmatrix} a & a & i & 0 & 0 & 0 \\ a & a & p & x & 0 & 0 \\ x & p & a & a & i & 0 \\ 0 & i & a & a & p & x \\ 0 & 0 & x & p & a & a \\ 0 & 0 & 0 & i & a & a \end{bmatrix}$$

Chapter 3

Enriched Discontinuous Galerkin Method

3.1 Enriching the Approximation Space

The enriched discontinuous Galerkin (xDG) method is an extension of the discontinuous Galerkin (DG) method, which integrates enrichment functions like trigonometric, Heaviside, harmonic, or other special functions to enhance the accuracy and efficiency of numerical approximations. By combining the adaptability, local formulation, and flexibility of the DG method with the enriched Finite Element Method (xFEM), the xDG method offers a powerful tool for various computational applications. In this chapter, we will provide an overview of the xDG method's formulation, establish our first model problems, and then compare different xDG methods.

3.1.1 Formulation

The primary distinction between the xDG and DG methods lie in the enrichment of the approximation space by incorporating additional functions that provide known or a priori information about the application problem at hand. Like the DG method, in the xDG

approach, each element has its own set of local basis functions, and continuity across element boundaries is not required. The basis functions are enhanced by including extra functions, tailored to the specific problem, to better approximate the solution. These enrichment functions can include trigonometric functions like cosine and sine, or special functions like Heaviside functions.

In the xDG method, the weak form of the problem is derived using the enriched basis functions, similar to the DG method. Penalization techniques from the DG method, such as SIPG or BZ, can be utilized to stabilize the numerical solution. The resulting system of linear equations is then solved to obtain the approximate solution.

Enrichment functions play a critical role in the xDG method, as they provide additional degrees of freedom for the solution representation, enabling better capture of complex or rapidly varying solution features. Popular enrichment functions include trigonometric functions for periodic or oscillatory behavior, Heaviside functions for discontinuities, and other special functions (such as Bessel functions) depending on the specific application.

For instance, consider an original set of basis functions $\{\phi_i\}_{i=1}^{N_e}$, where N_e is the number of such functions per element. Let $\{\psi_j\}_{j=1}^{N_x}$ be a set of enrichment functions defined over the element E. The enriched approximation space over element E consists of functions $u_h(x)$ that can be expressed as:

$$u_h(x) = \sum_{i=1}^{N_e} u_{i0}\phi_i(x) + \sum_{i=1}^{N_e}\sum_{j=1}^{N_x} u_{ij}\phi_i(x)\psi_j(x)$$

For any node point at an interface between elements, there will be $2(N_e + N_eN_x)$ degrees of freedom. For highly oscillatory problems, trigonometric enrichments prove valuable, and in the following text we will restrict our attention to them. In particular, we enrich each element E with both sin and cos at a frequency of k so that as the element is

traversed, their domain is $[0, h]$:

$$\psi_1(s) = \sin\left(kh\frac{1-s}{2}\right), \quad \psi_2 = \cos\left(kh\frac{1-s}{2}\right)$$

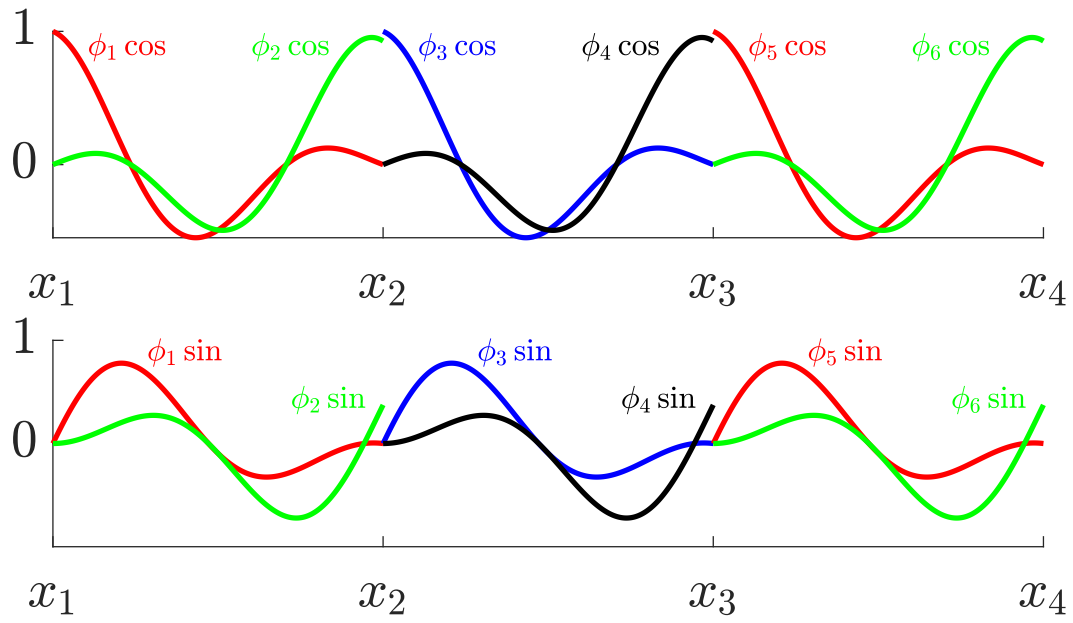


Figure 3.1 Linear DG basis functions ϕ_i enriched by (top) cos and (bottom) sin on 3 elements with enrichment frequency $k_e = 10$.

The approximate solution on element E_j takes the form

$$u_h(x) = \sum_{i=1}^{N_e} u_i\phi_i(x) + \sum_{i=1}^{N_e} v_i\phi_i(x)\sin\left(k(x-x_j)\right) + \sum_{i=1}^{N_e} w_i\phi_i(x)\cos\left(k(x-x_j)\right)$$

The overall approximation space V_h for xDG methods is the collection of functions whose restriction on each element can be expressed in the above form. The penalization methods and weak form are established in a manner similar to DG, with the key difference being the increased degrees of freedom at each node, which correspond to the additional basis functions in our formulation.

3.2 Model Problems

In this section, we present preliminary model problems for further study before moving on to a simplified one dimensional HIFU model. The chosen problems are a 2nd order elliptic and Helmholtz problem.

3.2.1 Elliptic

The 2nd order elliptic equation is a widely used partial differential equation (PDE) in various engineering and scientific disciplines, such as heat transfer, fluid dynamics, and solid mechanics, among others. It is given by:

$$-\Delta u + cu = f,$$

where c is a constant, u is the unknown solution, and f is a given source term. The natural boundary condition to apply when using FEM, SIPG, or xSIPG are of the Neumann type due to ease of implementation. Therefore, to establish benchmarks, we will focus our attention on the elliptic equation with Neumann boundary conditions over the interval $[0, 1]$:

$$u'(0) = a_l, \qquad u'(1) = a_r$$

Furthermore, our focus will be on oscillatory problems with given frequency k and in one dimension; the model simplifies to finding a solution u to the problem

$$-u'' + k^2 u = f, \qquad u'(0) = a_l \text{ and } u'(1) = a_r$$

Our benchmark problem is constructed to have the solution $u(x) = e^x \sin(kx)$ so there is a smooth component being multiplied by a highly oscillatory component. We study this more involved problem because our enriched method will be able to solve a problem

exactly with only one element if the exact solution is a linear combination of the pure harmonics sin and cos, and the HIFU ultrasound pressure exhibits similar behavior.

The final form of what we will call the *elliptic* problem is formulated as

$$-u''+k^2u = e^x((2k^2-1)\sin(kx) - 2k\cos(kx))$$
$$u'(0) = k \quad \text{and} \quad u'(1) = e(\sin(k) + k\cos(k))$$

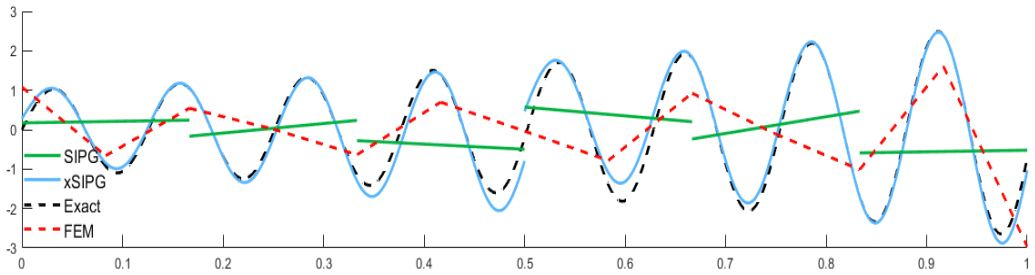


Figure 3.2 FEM, SIPG, and xSIPG solutions to the elliptic problem with $k = 50$. Only 2 elements were used for xSIPG, 6 elements for SIPG, and 12 elements for FEM. These number of elements were chosen for each method so that the resulting linear system would have about the same size (13 for FEM). The penalty parameter used for both SIPG and xSIPG was set to be $\sigma = 1$. The enrichment frequency for xDG was $k_e = 50$.

3.2.2 Helmholtz

The Helmholtz equation arises in various applications such as acoustics, electromagnetics, and wave propagation problems. The equation is given by:

$$\Delta u + k^2 u = f,$$

where Δ is the Laplace operator, k is the wavenumber, u is the unknown solution, and f is a given source term. We focus our attention on the case of Neumann type boundary conditions:

$$u'(0) = a_l, \qquad u'(1) = a_r$$

In one dimension our problem simplifies to

$$u'' + k^2 u = f, \qquad u'(0) = a_l \quad \text{and} \quad u'(1) = a_r$$

Just as was the case for the elliptic problem, we construct our model problem to have an exact solution of $u(x) = e^x \sin(kx)$. Resulting in the final formulation

$$u''+k^2 u = e^x(\sin(kx) + 2k\cos(kx))$$
$$u'(0) = k \quad \text{and} \quad u'(1) = e(\sin(k) + \cos(k))$$

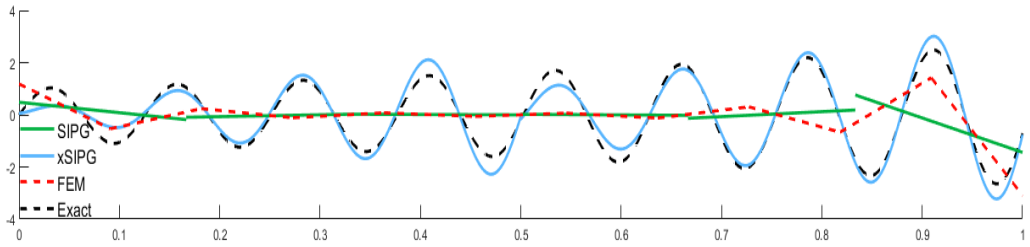


Figure 3.3 FEM, SIPG, and xSIPG solutions to the helmholtz problem with $k = 50$. Only 2 elements were used for xSIPG, 6 elements for SIPG, and 12 elements for FEM. These number of elements were chosen for each method so that the resulting linear system would have about the same size. The penalty parameter used for both SIPG and xSIPG was set to be $\sigma = 1$. The enrichment frequency for xDG was $k_e = 50$.

3.3 Supremacy of xSIPG over xBZ

This section compares the convergence and conditioning of xSIPG, xBZ, and FEM in the elliptic and Helmholtz problems stated above with particularly high frequency k. We have chosen $\sigma = 1, \alpha = 3$ for xBZ [6] and $\sigma = 10^8, \alpha = 1$ for xSIPG to match with our numerical results from the next chapter where we determine σ.

3.3.1 Elliptic

We show below the H^1 energy error, L^2 error, and the condition numbers for the elliptic problem with frequencies $k = 1000, 10000, 20000$. One important observation is the increased efficiency and stability of the xSIPG method as the frequency grows larger. There appears to be a "knee" for xBZ, corresponding to the number of elements needed for xBZ to begin converging, while for xSIPG we are observing no "knee."

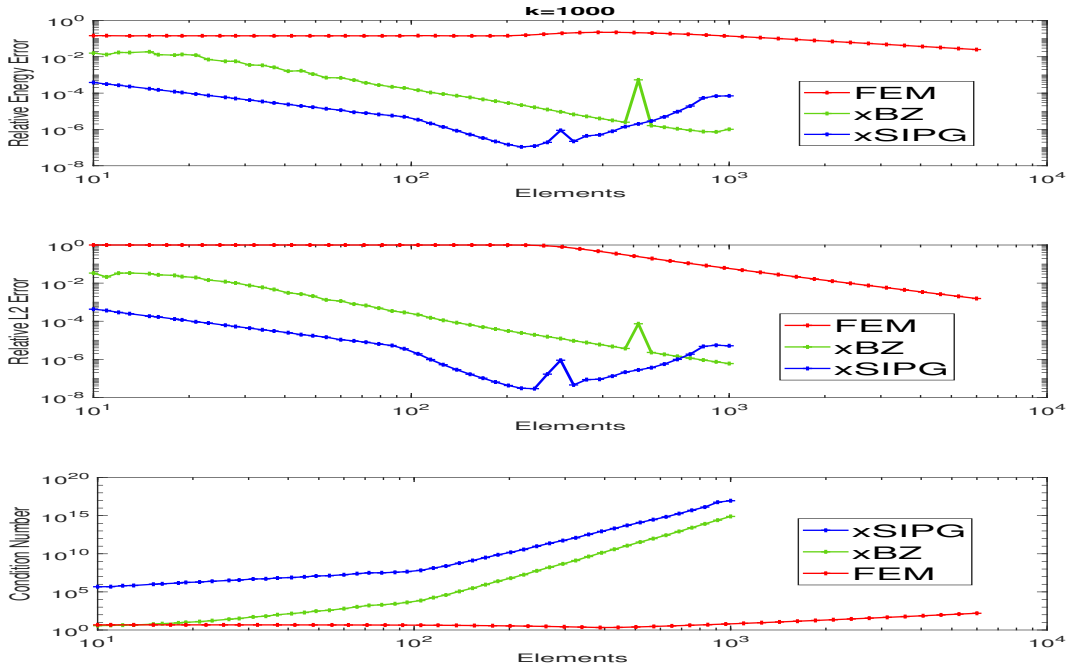


Figure 3.4 H^1, L^2, and conditioning plots for the elliptic problem with a frequency of $k = 1000$. For the enriched methods we carried out the computations to 1000 elements, while for FEM we stopped at 6000 elements.

We need to address here why it appears our method begins to increase at error at some point. This is due to the default MATLAB precision for round-off error of $2.2 * 10^{16}$. For example, when the condition number reaches 10^{10} we can expect no more than 6 digits of accuracy, and when the condition number reaches 10^{12} we can expect only 4 digits of accuracy. This is why we see the error increase once the condition number reaches a sufficiently large number. It is important to note this behavior would also be seen for FEM, if one were to use a sufficient number of elements to reach an error of 10^{-6}; and that if a higher precision were to be used, convergence would be expected to prolong correspondingly.

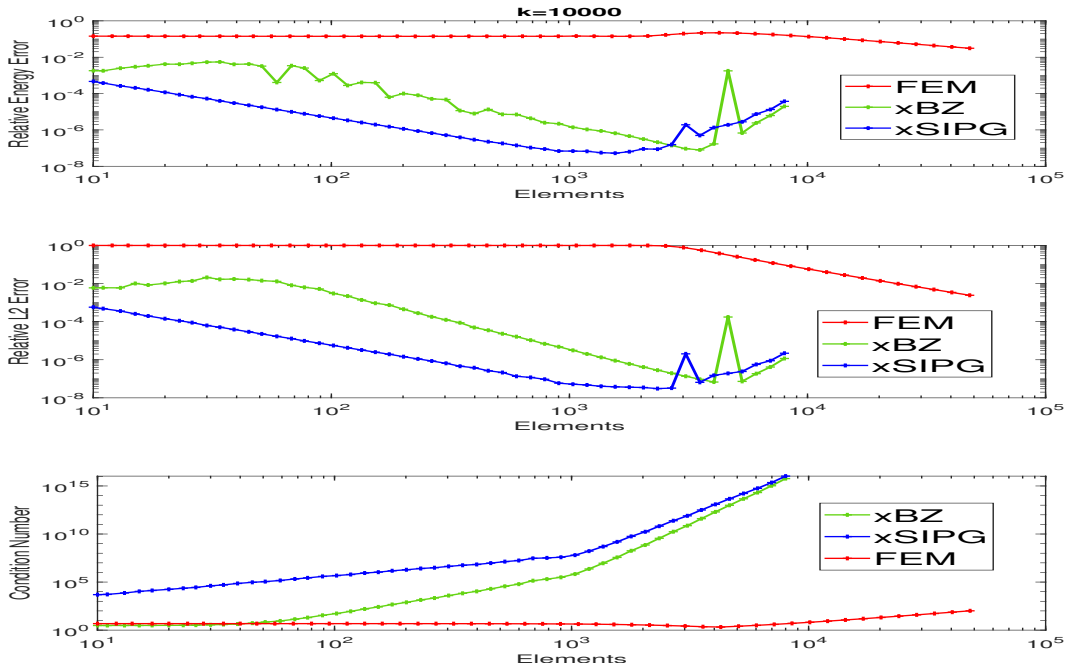


Figure 3.5 H^1, L2, and conditioning plots for the elliptic problem with a frequency of $k = 10000$. For the enriched methods we carried out the computations to 8000 elemenets, while for FEM we stopped at 48000 elements.

These numerical results show that xSIPG provides a substantial improvement over xBZ for solving highly oscillatory elliptic problems, which will be crucial for solving the full HIFU model.

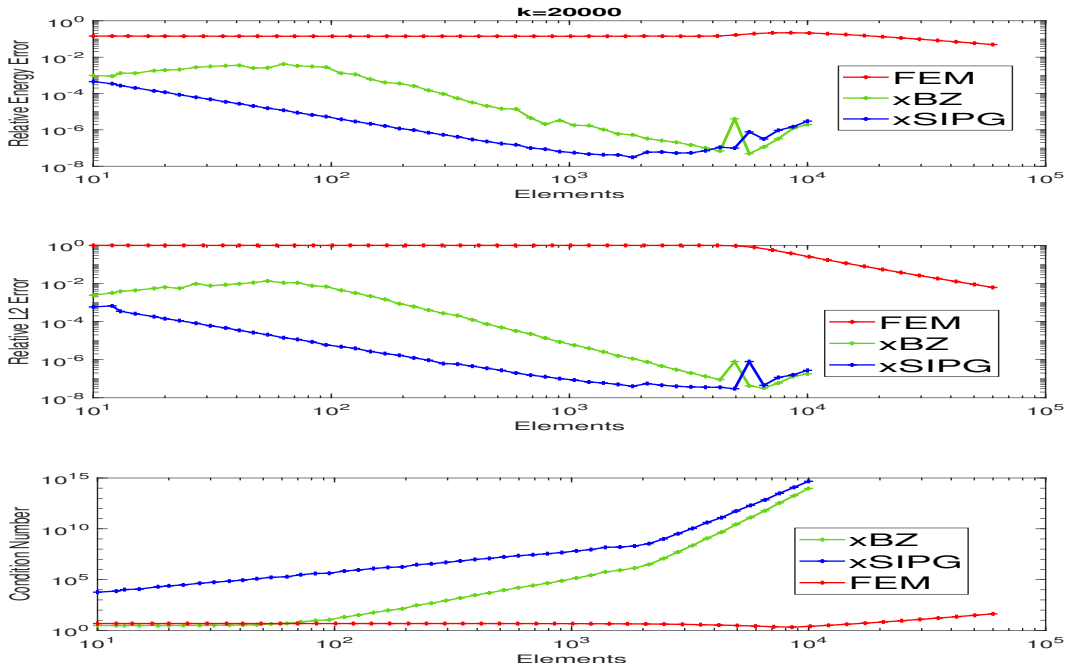


Figure 3.6 H^1, L2, and conditioning plots for the elliptic problem with a frequency of $k = 20000$. For the enriched methods we carried out the computations to 10000 elemenets, while for FEM we stopped at 60000 elements.

3.3.2 Helmholtz

The biggest contribution from moving to xSIPG is the ability to solve Helmholtz at higher frequencies [11] [10]. The numerical results show the convergence plots for frequencies $k = 1000, 10000$. We see that xBZ is unable to handle the additional complexities involved in solving a Helmholtz problem, most likely due to ignoring the critical boundary terms in the variational form which result from integrating by parts on each element.

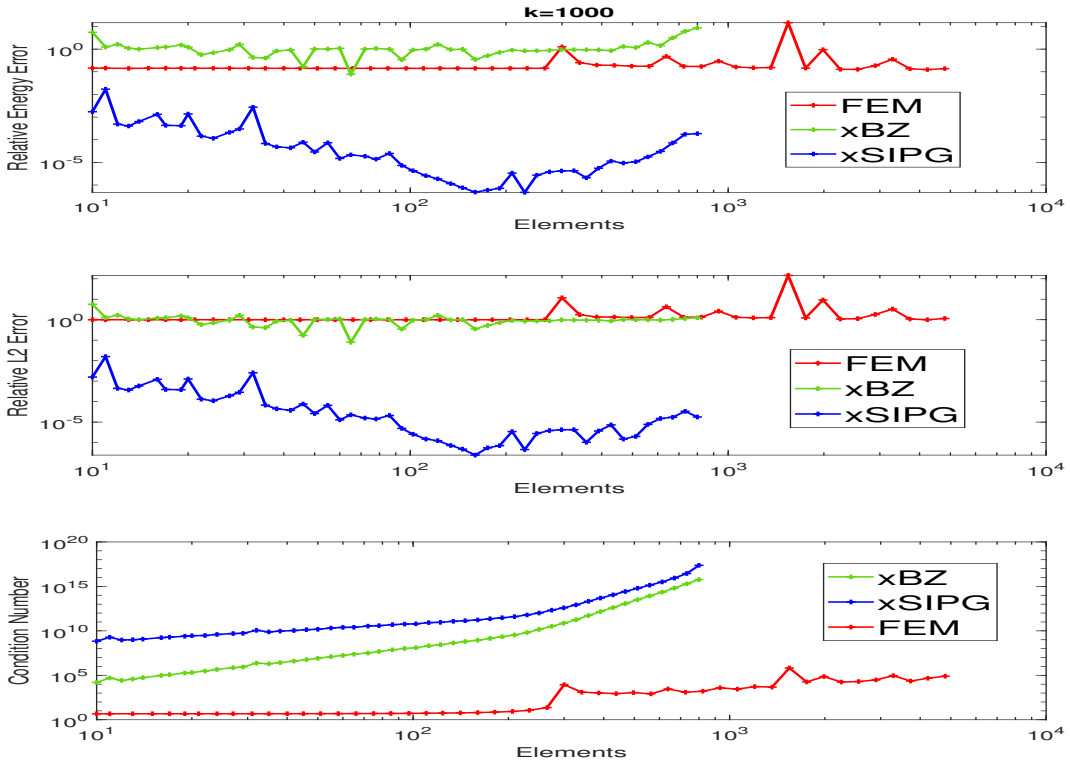


Figure 3.7 H^1, L2, and conditioning plots for the Helmholtz problem with a frequency of $k = 1000$. For the enriched methods we carried out the computations to 800 elemenets, while for FEM we stopped at 4800 elements.

This provides evidence that xSIPG has the potential to be applied to a wider range of problems for future research and endeavors, such as the Airy and Schrödinger equations. Note the non-perfect convergence of xSIPG (before conditioning issues), however this can be alleviated as one can establish element ranges for a given frequency in which the error will be guaranteed to be below a certain amount

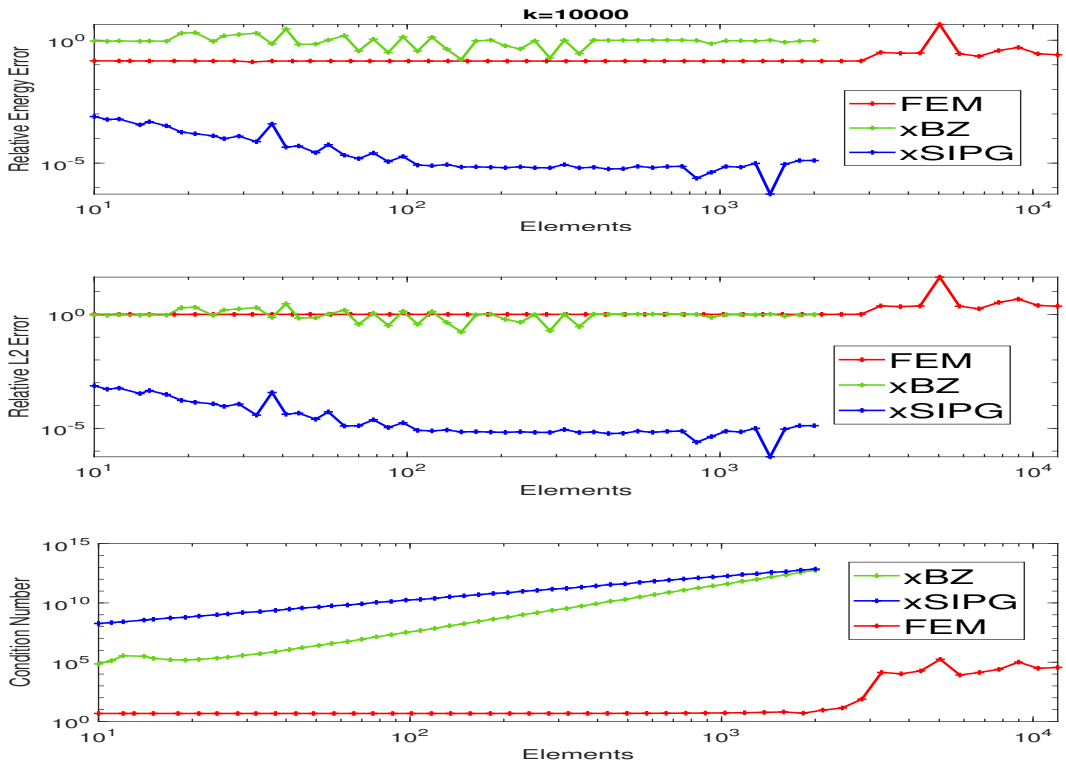


Figure 3.8 H^1, L2, and conditioning plots for the Helmholtz problem with a frequency of $k = 10000$. For the enriched methods we carried out the computations to 2000 elemenets, while for FEM we stopped at 12000 elements.

Chapter 4

Determining the Optimal Penalization Method

In this chapter, we focus on the pivotal process of identifying optimal penalty parameters for enriched Discontinuous Galerkin (xDG) methods. The selection of penalty parameters and enrichment functions is paramount to ensuring the stability, accuracy, and convergence of any xDG method [5]. We use Symmetric Interior Penalty Galerkin (SIPG) as the backbone for our xDG method, which we have refered to as the xSIPG method.

Our exploration spans the investigation of prototype elliptic boundary value problems and the benchmark Helmholtz problem. These help us illustrate the critical role of determining the optimal penalty parameters and methods. In this context, we delve into the merits of penalizing the derivative for elliptic cases, while abstaining from doing so for the Helmholtz cases. We further elucidate the impact of penalty proportionality to the wavenumber or mesh size, contributing to a deeper understanding of the xSIPG method's behavior.

Our insights are substantiated by extensive numerical experiments conducted under varying frequency regimes to adapt to the specific problem at hand.

4.1 Advocating for a Substantial Penalty

In Discontinuous Galerkin (DG) methods like SIPG, a lower bound is established for the magnitude of the penalty. For one-dimensional problems, a plausible choice for this lower limit is $\sigma = 16$. Nonetheless, to attain superior accuracy, a heftier penalty is requisite. Moreover, upon integrating enrichment functions into the method, the penalty parameter must be reassessed. For an enriched DG method, such as xSIPG, the optimal selection of the penalty parameter should be set to a value around one million or higher, as we elaborate below.

The Symmetric Interior Penalty Galerkin method (SIPG) exemplifies a discontinuous Galerkin (DG) finite element method. Ascertaining the penalty parameters is instrumental in scrutinizing the behavior of a DG method. In [5], the penalty σ was established to have a lower bound of 16 for one-dimensional elliptic problems. We demonstrate that when the SIPG method is enriched with supplementary enrichment functions [16][6], the penalty parameter must be on the order of one million, at least. We direct our focus to the wavenumber $k = 16$ and both the Elliptic and Helmholtz problems

Our numerical results (see figure 4.1 below) reveal that for penalties within the range of 10 to 1000, a relative energy accuracy of no more than roughly 1% can be achieved. However, upon escalating to a higher penalty magnitude, specifically in the range of 10^5 to 10^7, the accuracy improves to $.001 - .0001\%$. This underscores the assertion that the optimal penalty parameter should indeed be above 10^6.

Consequently, when identifying the optimal penalty parameter for xSIPG at higher wavenumbers, we initiate our search at 10^6. We also exemplify in figure 4.2 that this phenomenon is not confined to enriched methods; the original SIPG method also exhibits comparable behavior.

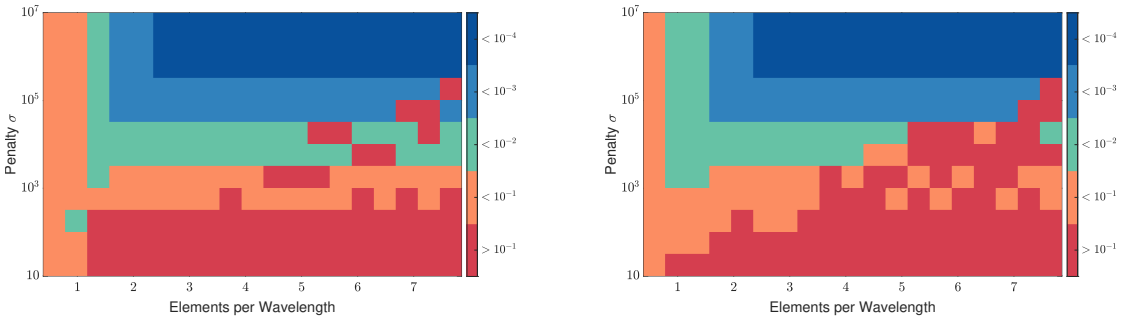


Figure 4.1 Figure comparison showing the xSIPG method in solving the Elliptic problem (left) and the Helmholtz problem (right) with wavenumber set to $k = 16$. The x-axis represents the elements per wavelength ranging from 0 to 8, while the y-axis shows the penalty parameter varying from 10 to 10^7. Colored regions denote the solutions falling below specific error thresholds: Red corresponds to an error greater than .1; Orange to an error less than .01; Green to an error less than .001; Light Blue to an error less than .0001; and Blue to an error less than .00001.

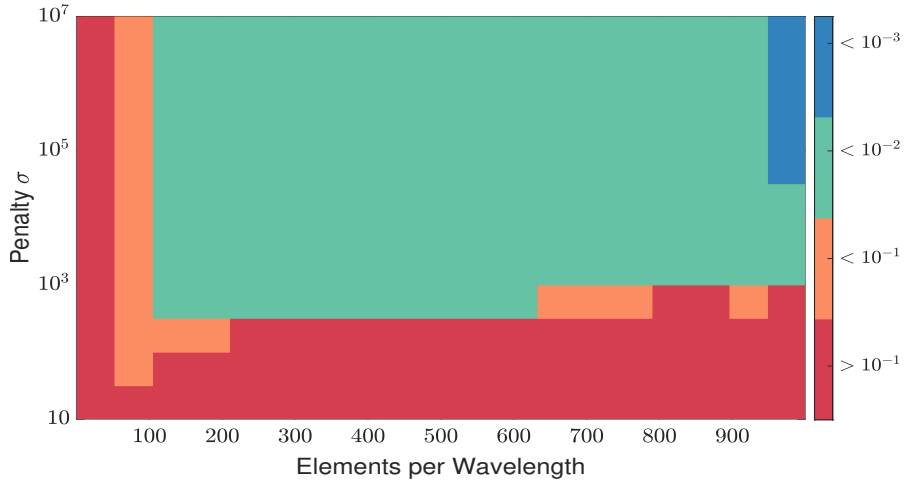


Figure 4.2 Using the SIPG method to solve the Elliptic problem with wavenumber $k = 16$. The penalty range and colors remain the same as in figure 1, while the x-axis displays elements per wavelength ranging from 0 to 1000.

4.2 Elliptic

In the elliptic case, we executed a sequence of numerical experiments to evaluate the xSIPG method's performance under various penalty parameter values. We also tested innovative penalization techniques. Our analysis of the method's stability, accuracy, and

convergence properties across these dimensions allowed us to identify the optimal penalty parameters and technique.

In the subsequent sections, we present our numerical results for different wavenumbers and penalization techniques in the elliptic case. We showcase the relative energy error, H^1, as the number of elements per wavelength, Ω, grows. We normalize the elements by wavelength to ensure that we are working with a standardized unit.

4.2.1 Penalizing in Proportion to the Mesh Size

Our penalty analysis begins with a direct adaptation of the SIPG penalization technique to xDG. With ϕ_i and ϕ_j denoting two basis functions and h representing the size of our mesh, the penalty term at the interface of elements E_i and E_j manifests as:

$$J_\sigma(\phi_i, \phi_j) = \sigma \frac{[\phi_i][\phi_j]}{h}$$

Our observations (Figure 4.3 below) indicate that the optimal choice of the penalty parameter for $k \in [10^3, 10^6]$ must lie within the range $10^8 \leq \sigma \leq 10^{10}$. However, the optimal penalty parameter seems to covary with the magnitude of the wavenumber. The best penalty for $k = 1$kHz could arguably be 10^8, while for $k = 80$kHz, the optimal penalty appears to lie within the range $10^9 \leq \sigma \leq 10^{10}$.

Another noteworthy observation is that the achievable maximum accuracy increases as the wavenumber rises. This appears counterintuitive, given that higher frequency problems are more challenging. At $k = 1$kHz, only an error threshold of 10^{-6} can be achieved within a relatively small region. However, as k increases, the 10^{-6} threshold region expands, and 10^{-7} threshold regions begin to emerge. At 40 and 80 kHz, an error of 10^{-8} is attainable. This observation is attributable to the fact that when k is smaller the regular DG basis functions are closer to approximating the enriched functions, which causes ill-conditioning faster.

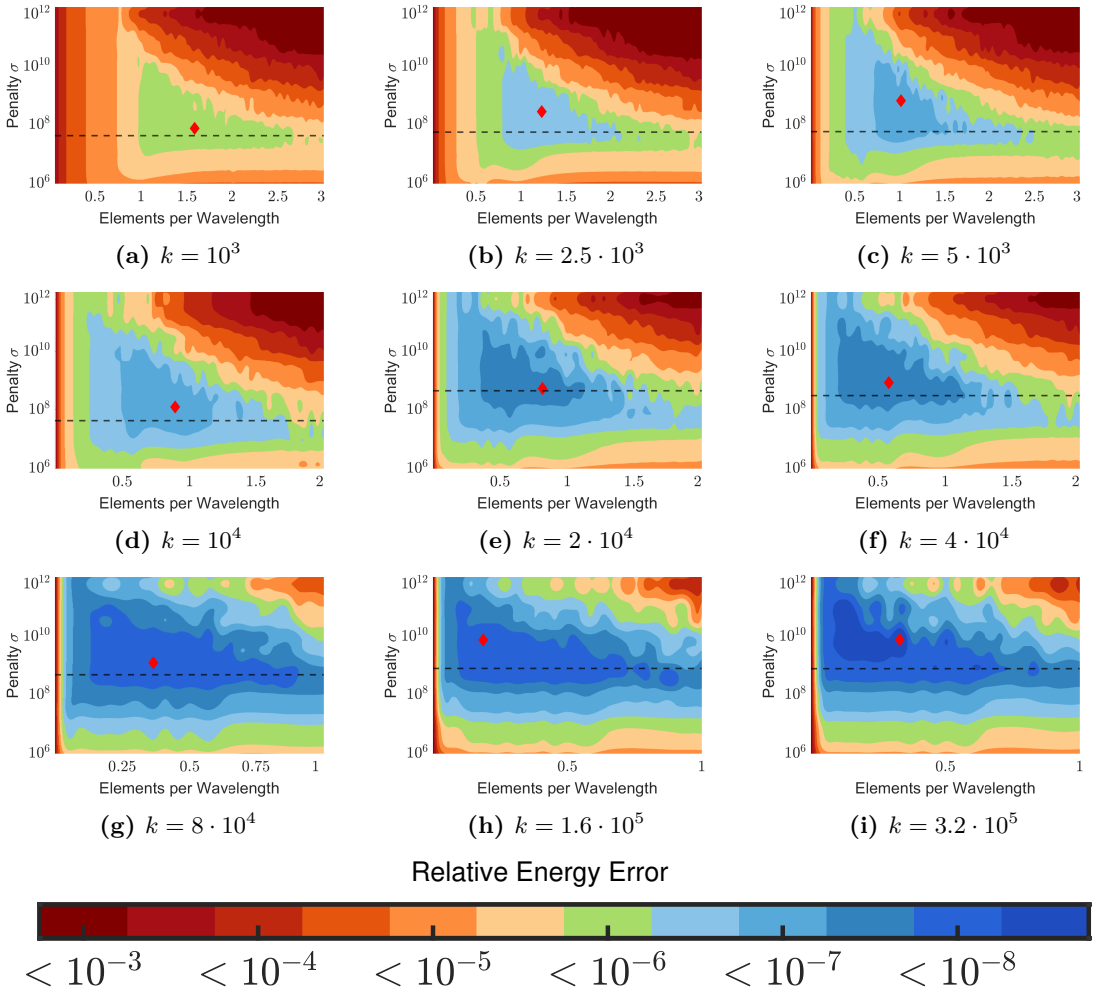


Figure 4.3 Pseudocolor plots illustrating the variation of H^1 error for different wavenumbers ω and penalty parameters σ when using J_σ. Note the shift for the range of the x-axis as you descend levels. Three key observations are to be made as ω increases. First is the striking observation that the maximum accuracy increases. Secondly, the optimal σ begins around 10^8 and shifts upwards to 10^{10}. Lastly, the region where the method achieves the minimum error shifts to the left corresponding to a decrease in elements per wavelength (note that the total number of elements used would still increase).

In our current context, it suffices to note that when penalizing in proportion to the mesh size, σ appears to covary significantly with k. One could retain this penalization method, bearing in mind that one's choice of σ must be carefully considered as k varies. Our aim, however, is to identify a penalization technique that could yield a σ value

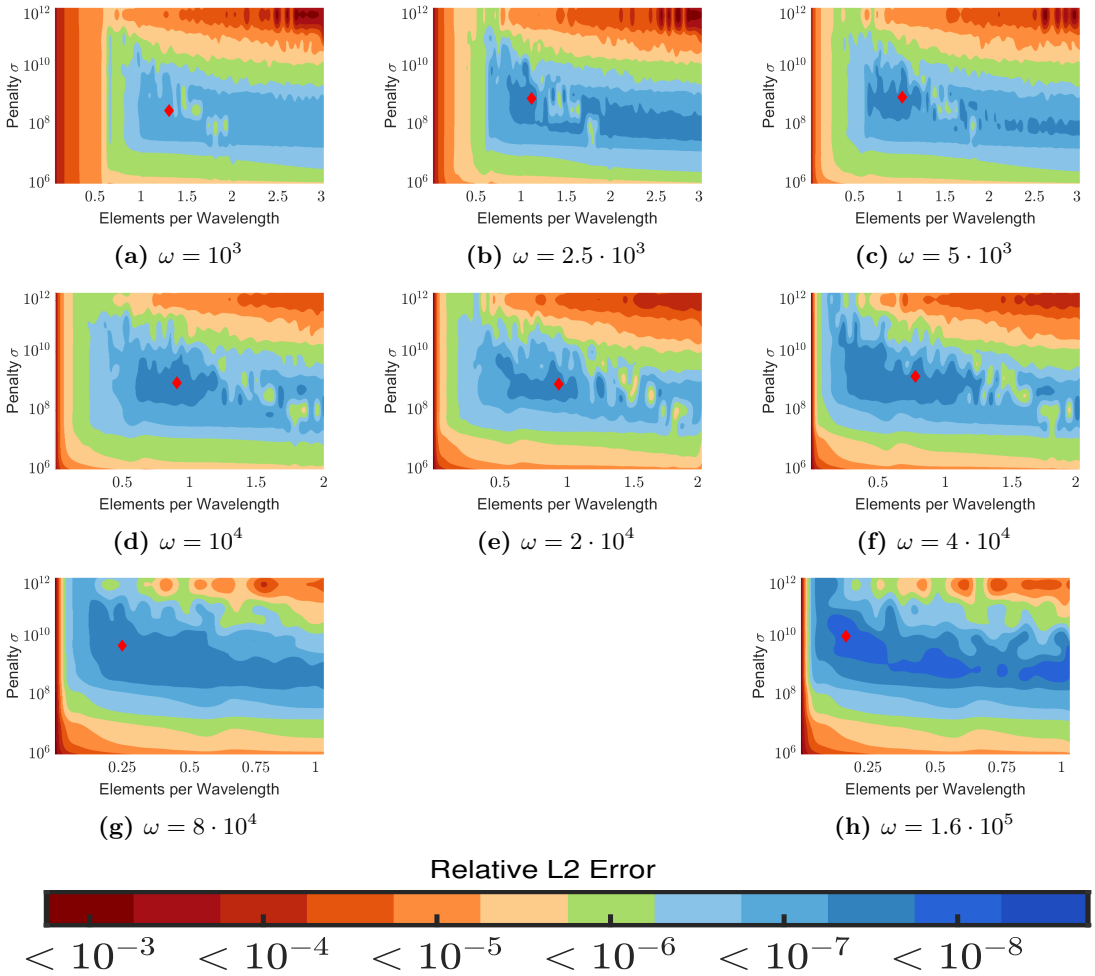


Figure 4.4 Pseudocolor plots illustrating the variation of L^2 error for different wavenumbers ω and penalty parameters σ when using J_σ. Note that similar observations can be made when using H^1 to measure error. However, in contrast, the H^1 error exhibits better behavior. This is a significant reason why we chose to analyze H^1, as it provides a better route to insight on the problem.

applicable to a wide range of frequencies. Consequently, we now turn our focus towards an innovative, more robust penalization method.

4.2.2 Penalizing in Proportion to the wavenumber

In this section, we present our findings on the performance of penalizing by the wavenumber compared to penalizing by the mesh size. Our numerical results demonstrate that

penalizing by the wavenumber yields significantly better results and is more robust than penalizing by the mesh size.

This finding is particularly noteworthy, as the idea of penalizing in proportion to the wavenumber is original to our work. The superior performance of this approach demonstrates its potential for further enhancing the effectiveness of xDG methods for oscillatory problems in various applications. More generally, the idea of incorporating known information into the penalization method itself is new to this research and represents a step forward for xDG in solving even more complex problems.

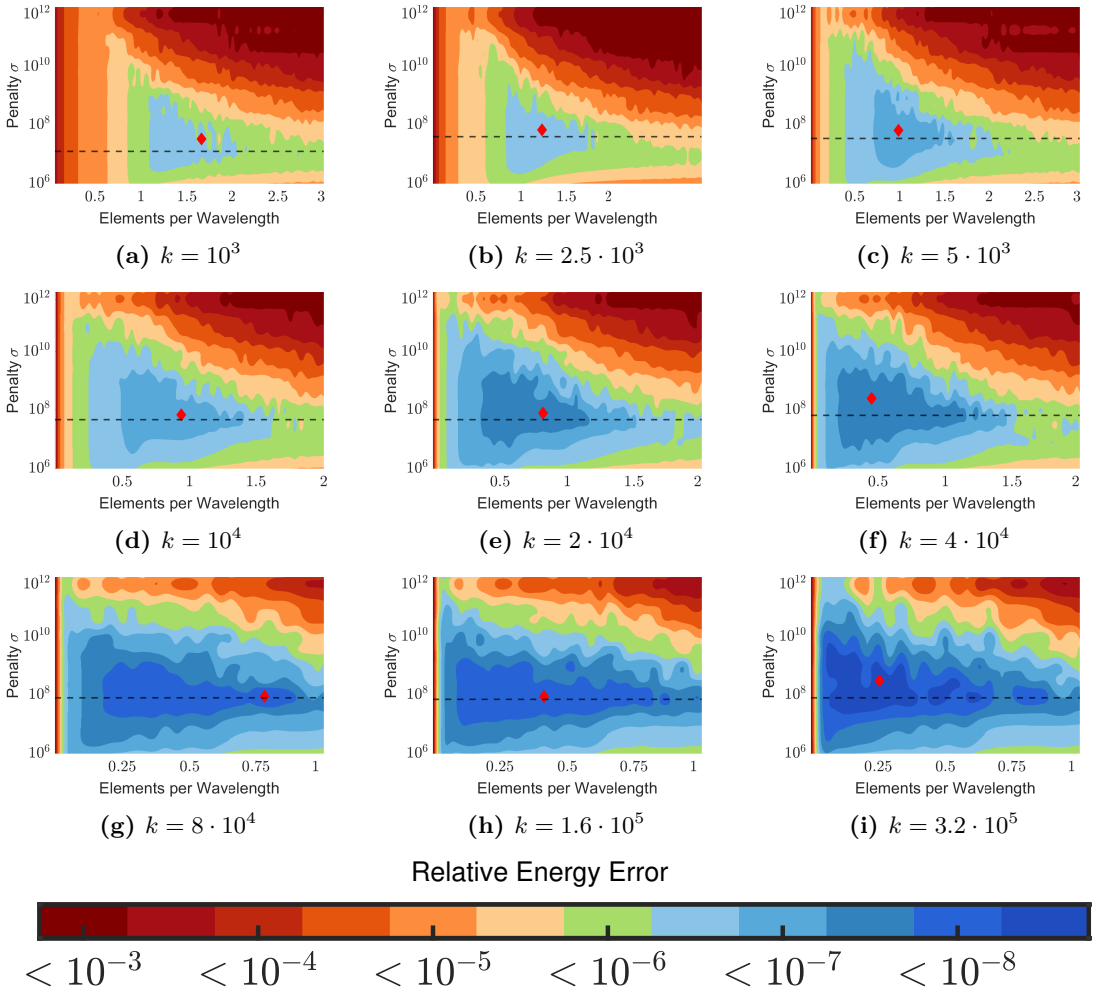
Let us be more specific about the alteration we are making. The term in the weak form we are concerned with when penalizing in proportion to the mesh size is the jump term J_σ:

$$J_\sigma(u,v) = \sum_{e\in\Gamma_h\cup\partial\Omega}\int_e \frac{\sigma}{|e|}[u][v]$$

Our change is to penalize by $k\sigma$ rather than $\frac{\sigma}{|e|}$. Thus, leaving all other terms fixed, our updated xSIPG penalization becomes J^k_σ:

$$J^k_\sigma(u,v) = \sum_{e\in\Gamma_h\cup\partial\Omega}\int_e k\sigma[u][v]$$

Upon implementing the changes in our penalization method, we carried out another series of numerical experiments. The results, as shown in Figure 4.5 on the next page, underscore the robustness and efficiency of penalizing in proportion to the wavenumber.



(a) $k = 10^3$ **(b)** $k = 2.5 \cdot 10^3$ **(c)** $k = 5 \cdot 10^3$

(d) $k = 10^4$ **(e)** $k = 2 \cdot 10^4$ **(f)** $k = 4 \cdot 10^4$

(g) $k = 8 \cdot 10^4$ **(h)** $k = 1.6 \cdot 10^5$ **(i)** $k = 3.2 \cdot 10^5$

Figure 4.5 Pseudocolor plots illustrating the variation of H^1 error thresholds for different wavenumbers k and penalty parameters σ when using J_σ^k. The most noteworthy observation is the stability of the optimal penalty parameter around $\sigma = 10^8$. This is an important advantage for J_σ^k over J_σ that will become more prevalent when we transition to the Helmholtz problem.

Unlike the mesh size penalization method, the choice of penalty does not seem to covary with the wavenumber. For all k in the interval 1 to 320 kHz, the best penalty σ is centered around 10^8. Hence, the penalty parameter need not be continually adjusted for different frequencies. This implies that the method has a greater level of stability and is not sensitive to the exact choice of σ. Therefore, a value of $\sigma = 10^8$ should generally suffice.

Remarkably, even in the challenging case of $k = 80$ kHz, we managed to achieve an error less than 10^{-8} in a rather long strip along Ω. We can see this general improvement in each wavenumber, where the state of maximum error is held for a wider range of elements compared to Figure 4.3. This is a significant improvement over the mesh size penalization method.

The introduction of this novel penalization strategy enhances our understanding of how to better handle oscillatory problems. More importantly, it enables us to solve a broad range of problems with varying frequencies without having to readjust our penalty parameter. Our results substantiate the theoretical merits of this penalization method and suggest that it may be the optimal strategy for the xSIPG method in the elliptic case.

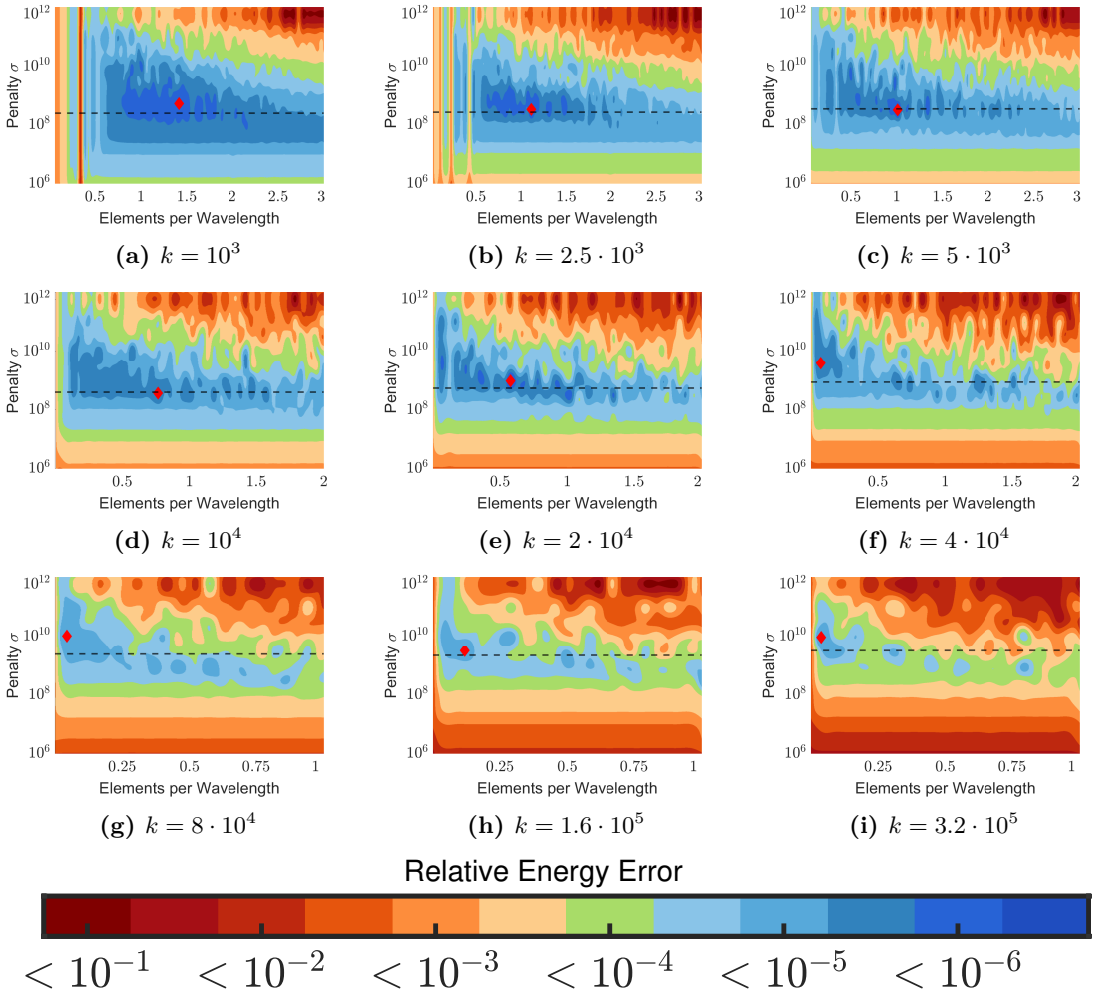
We believe that these findings have considerable potential for further development. Future research should continue to probe the possibilities and limits of penalizing in proportion to some apriori information about the problem at hand in the broader field of enriched discontinuous Galerkin methods. Not only does this approach improve the performance and robustness of the xSIPG method in oscillatory problems, but it also provides insights that could be beneficial to other methods and applications.

4.3 Helmholtz

An essential aspect of determining the optimal penalty parameter is understanding the impact of the penalty proportion on the numerical solution. In this section, we present our findings on the performance of xSIPG in solving Helmholtz; penalizing by the wavenumber compared to penalizing by the mesh size. Our numerical results demonstrate that penalizing by the wavenumber yields significantly better results than the elliptic improvement above and is more robust than penalizing by the mesh size for Helmholtz. The superior performance of this approach demonstrates its potential for further enhancing the effectiveness of the XDG method in various applications.

4.3.1 Penalizing in Proportion to the Mesh Size

Similarly to the elliptic case when penalizing in proportion to the mesh size, we see that the optimal penalty covaries with the frequency once again. It is worth nothing that the results (Figure 3.6) show that the penalty varies more in the Helmholtz case as opposed to the Elliptic case. This highlights the utility of finding a more robust penalization method to cover Helmholtz and more complex problems in the future.



(a) $k = 10^3$ (b) $k = 2.5 \cdot 10^3$ (c) $k = 5 \cdot 10^3$

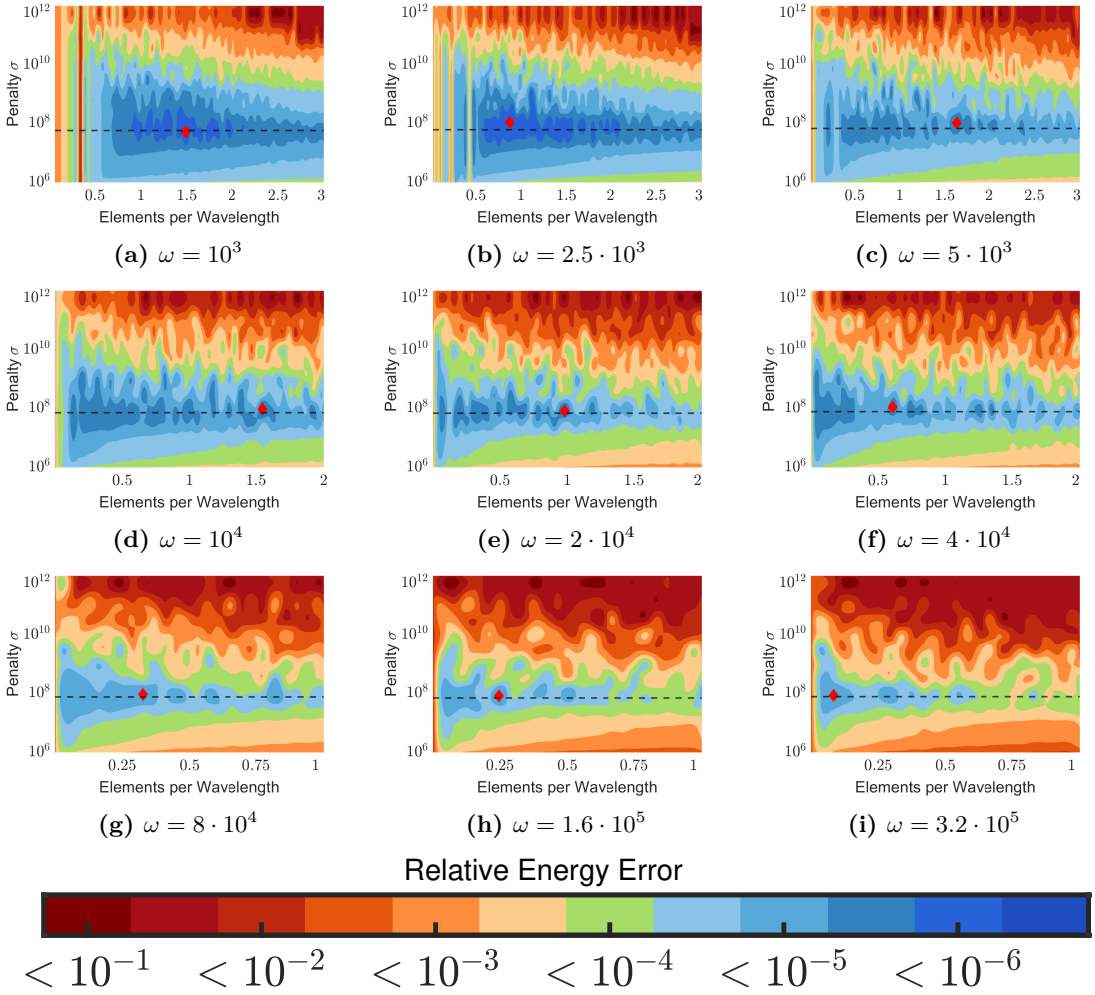
(d) $k = 10^4$ (e) $k = 2 \cdot 10^4$ (f) $k = 4 \cdot 10^4$

(g) $k = 8 \cdot 10^4$ (h) $k = 1.6 \cdot 10^5$ (i) $k = 3.2 \cdot 10^5$

Figure 4.6 Pseudocolor plots illustrating the variation of H^1 error in the Helmholtz problem for different wavenumbers k and penalty parameters σ when using J_σ. Note the optimal σ begins around 10^8 and shifts upwards to 10^{10}. The region where the method achieves the minimum error has been exceptionally reduced both vertically and horizontally.

4.3.2 Penalizing in Proportion to the Frequency

Similar to the elliptic case, we want to choose a more robust penalization method and parameter so that it can be used in a larger frequency range. The results for the new penalization method are promising for Helmholtz, as they show the robust penalty selection of $\sigma = 10^8$ continues



(a) $\omega = 10^3$ **(b)** $\omega = 2.5 \cdot 10^3$ **(c)** $\omega = 5 \cdot 10^3$

(d) $\omega = 10^4$ **(e)** $\omega = 2 \cdot 10^4$ **(f)** $\omega = 4 \cdot 10^4$

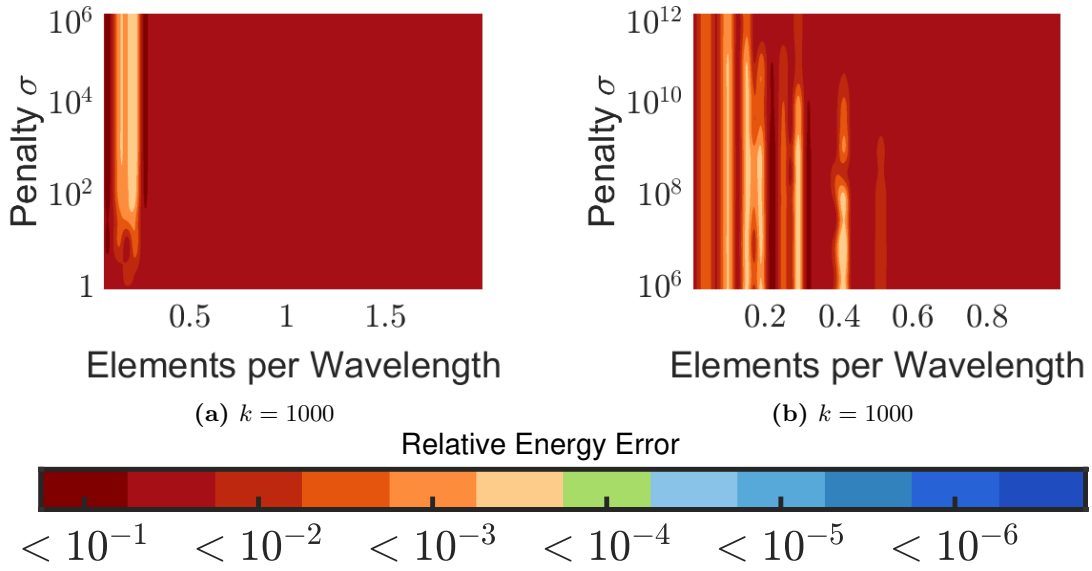
(g) $\omega = 8 \cdot 10^4$ **(h)** $\omega = 1.6 \cdot 10^5$ **(i)** $\omega = 3.2 \cdot 10^5$

Figure 4.7 Pseudocolor plots illustrating the variation of H^1 error in the Helmholtz problem for different wavenumbers k and penalty parameters σ when using J_σ^ω. Three key observations in contrast to J_σ are to be made as k increases. First is the observation that the optimal σ is fixed at 10^8. Second, is the fact that the minimum error region is slightly enlarged.

In conclusion, our analysis of the impact of penalty proportion on the numerical solution demonstrates that penalizing by the wavenumber is significantly more effective than penalizing by the mesh size for the Helmholtz cases. Furthermore, we have computationally established lower bounds on the size of the penalty parameter σ and established the number of elements to use when numerically solving.

4.4 xBZ is Intractable for Solving Helmholtz

Previously, xBZ was shown to converge for the elliptic problem, however it is clearly seen that xBZ can not be used as a viable method to solve Helmholtz as Figure 3.8 shows below. We tested penalty paremeters σ in the range $[1, 10^{12}]$ and we see that no selection produces satisfactory results. Therefore we conclude that xBZ is not a viable method for solving Helmholtz and potentially other oscillatory problems in the future.



(a) $k = 1000$ **(b)** $k = 1000$

Figure 4.8 Pseudocolor plots demonstarting the H^1 energy error in the Helmholtz problem for different wavenumbers k and penalty parameters σ using $J_{\sigma,3}$. The best error achieved is 10^{-3} in very narrow pockets of elements, showing that xBZ can not compete with xSIPG in solving Helmholtz. The left and right images show different penalty magnitudes.

4.5 Conclusion: New Frontier in xDG Methodology

We have provided a comprehensive examination of the determination of penalty parameters and method in xDG for the Elliptic and Helmholtz oscillatory problems. By presenting numerical results for both elliptic and Helmholtz cases, we have showcased the optimal penalty values, the range of elements to use, and the potential benefits of penalizing in proportion to the wave number rather than the mesh size. This original idea has the potential to greatly enhance the performance of the xDG method, and its consistent success across these two problems encourages further exploration and development in future research.

Our analysis has demonstrated the importance of carefully selecting penalty parameters and enrichment functions in the xDG method, as they have a significant influence on the method's stability, accuracy, and convergence properties. The insights gained from this investigation provide valuable guidance for researchers and practitioners working with the xDG method, contributing to improved performance in various applications. In future work, we aim to extend our analysis to other problems with highly-oscillatory solutions and higher dimensions, further substantiating the efficacy of the xDG approach in addressing a wide range of challenging problems.

As we draw this chapter to a close, we stand on the cusp of a new frontier in xDG methodology. Our investigations have not only illuminated the critical role of penalty parameters but also charted a course towards more effective and robust approaches by incorporating a priori information into the penalization method itself. The journey ahead is ripe with potential, beckoning further exploration and innovation in the pursuit of excellence in computational methods.

Chapter 5

Practical Application and Implementation

5.1 HIFU and General Implementation

Using the numerical results from the determination of penalty parameters, we may deduce an upper and lower bound on the amount of elements one should choose when using xSIPG for a frequency residing in the range of HIFU surgery applications. HIFU is a medical procedure where ultrasound transducers are set up around an arc and focused to ablate abnormal tissue such as cancerous tumors. The frequencies used in HIFU applications range anywhere from .1MHz to 20Mhz on each individual transducer [12] [4], necessitating a versatile and robust computational approach.

5.2 High Intensity Focused Ultrasound Model Problem

Central to the modeling of thermal dynamics in biological tissues for medical applications such as HIFU surgery is the bioheat equation. This PDE plays a pivotal role in simulating heat transfer, thus serving as a foundation for understanding and optimizing therapeutic

effects. The equation we study is a 1D simplification of the bioheat equation to model the effect of focused ultrasound transducers on the human body. We have coupled the bioheat equation with a solution of an appropriately oscillatory Helmholtz problem in accordance with the full HIFU model. We split the domain Ω into two parts Ω_1 and Ω_2, corresponding to healthy and cancerous tissue respectively; $\Omega_1 = [0, 0.5]$ and $\Omega_2 = (0.5, 1]$. The heat equation component takes the form:

$$\begin{aligned} &\dot{T} - T'' + \beta T = Q \\ &T(0,t) = T(1,t) = 0 \\ &T(x,0) = 0 \end{aligned}$$

where T is temperature, $\dot{T}$ is the derivative of temperature with respect to time, and Q may be thought of as the 1D heat generation due to an external source, such as an ultrasound transducer. β is the perfusion coefficient, set to be 100 on Ω_1 and 1 and Ω_2. Q is obtained as the square of the magnitude of the ultrasound pressure, p, which is given as the solution to the complex-valued Helmholtz problem below, multiplied by an absorption coefficient i.e. $Q = \alpha|p|^2$:

$$\begin{aligned} &p'' + k^2 p = 0 \\ &p'(0) = i\omega, \quad p'(1) = ikp(1) \end{aligned}$$

Here, k is the wavenumber given by $k = (\omega + i\alpha)$, where ω is the frequency of the wave field and α is the absorption coefficient for Ω, α is 1 on Ω_1 and 8 and Ω_2. This Helmholtz problem was chosen because it matches the expected behavior of the pressure induced by a single *focused* ultrasound transducer placed at the left boundary $x = 0$. The left

boundary condition corresponds to a source of amplitude ω and the right boundary is an appropriate absorbing condition. [9]

The solution to the above Helmholtz problem can be analytically obtained by solving two subproblems on $\partial\Omega_1$ and $\partial\Omega_2$ and requiring that the two solutions agree on their intersection.

Let $k_1 = \omega + i\alpha_1$ and $k_2 = \omega + i\alpha_2$ be the wave numbers on Ω_1 and Ω_2 respectively. The solution $p(x)$ is given piecewise by

$$p(x) = \begin{cases} Ae^{ik_1 x} + Be^{-ik_1 x} & \text{if } x \in \Omega_1 \\ Ce^{ik_2 x} & \text{if } x \in \Omega_2 \end{cases}$$

where the constants A, B, C are

$$\begin{aligned} A &= B - \frac{\omega}{k_1} \\ C &= \frac{B(e^{ik_1/2} + e^{-ik_1/2}) - \frac{\omega}{k_1} e^{ik_1/2}}{e^{ik_2/2}} \\ B &= \frac{ik_2\omega e^{ik_1/2}/k_1 - i\omega e^{ik_1/2}}{ik_2(e^{ik_1/2} + e^{-ik_1/2}) - ik_1(e^{ik_1/2} - e^{-ik_1/2})} \end{aligned}$$

Hence the source term $Q(x) = \alpha|p(x)|^2$ for the bioheat equation is also piecewise, with expression

$$Q(x) = \begin{cases} \alpha_1(|A|^2 e^{-2\alpha_1 x} + |B|^2 e^{2\alpha_1 x} + 2\Re(A\overline{B}e^{i2\omega x})) & \text{if } x \in \Omega_1 \\ \alpha_2|C|^2 e^{-2\alpha_2 x} & \text{if } x \in \Omega_2 \end{cases}$$

5.3 Numerical Results

The resulting numerical simulations demonstrate the behavior of the system at both low and high frequencies, which are relevant to HIFU applications. At lower frequencies, an unexpected result was observed: the performance of xBZ closely aligned with that of xSIPG. This convergence in outcomes initially complicates the differentiation between the two methods under these conditions. For the analysis presented, the Finite Element Method (FEM) was employed to resolve the temporal aspect of the Bioheat equation across all methodologies, while FEM, xBZ, and xSIPG were applied to spatially solve the Helmholtz problem. The findings suggest that, in one-dimensional settings, the enriched methods do not confer a distinct advantage for the time-component of the Bioheat equation as the source term lacks oscillatory characteristics. However, the potential benefits of these enriched methods may become more pronounced in higher-dimensional applications where oscillatory behavior in the source term is anticipated.

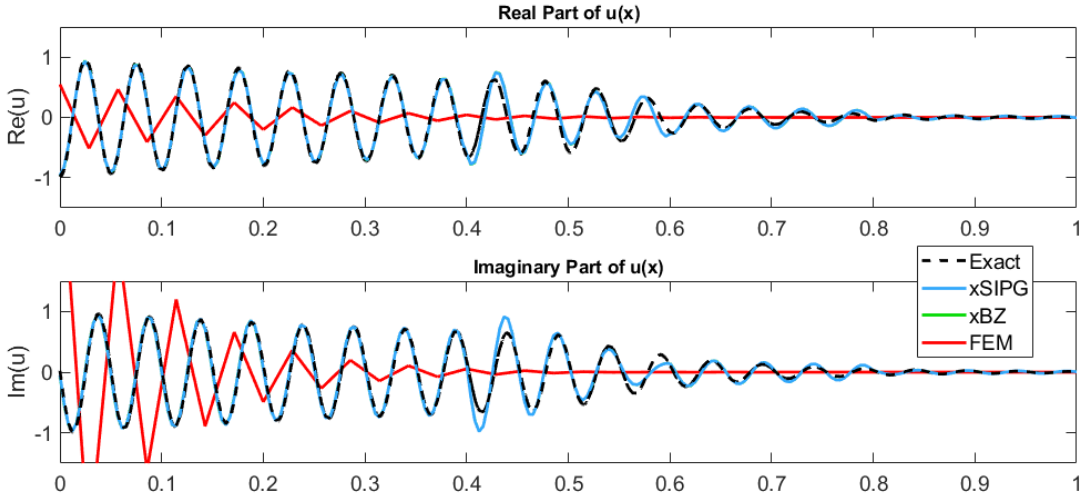


Figure 5.1 Solution p to the coupled Helmholtz problem with $\omega = 125$ and $\alpha = 1, 8$. 30 elements were used for FEM and 5 were used for xSIPG. xBZ and xSIPG matching at low frequency.

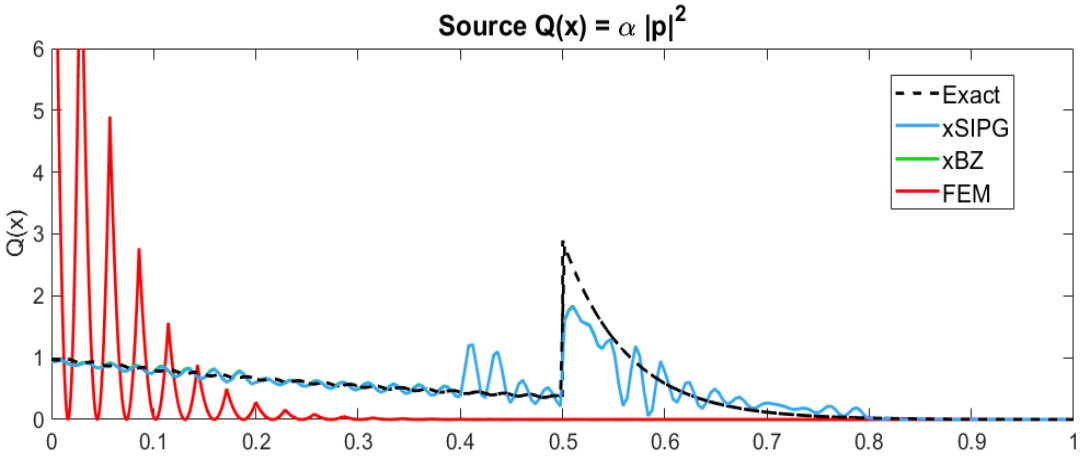


Figure 5.2 Source term Q for the bioheat equation with $\omega = 125$ and $\alpha = 1, 8$. 30 elements were used for FEM and 5 were used for xDG. xBZ and xSIPG matching at low frequency.

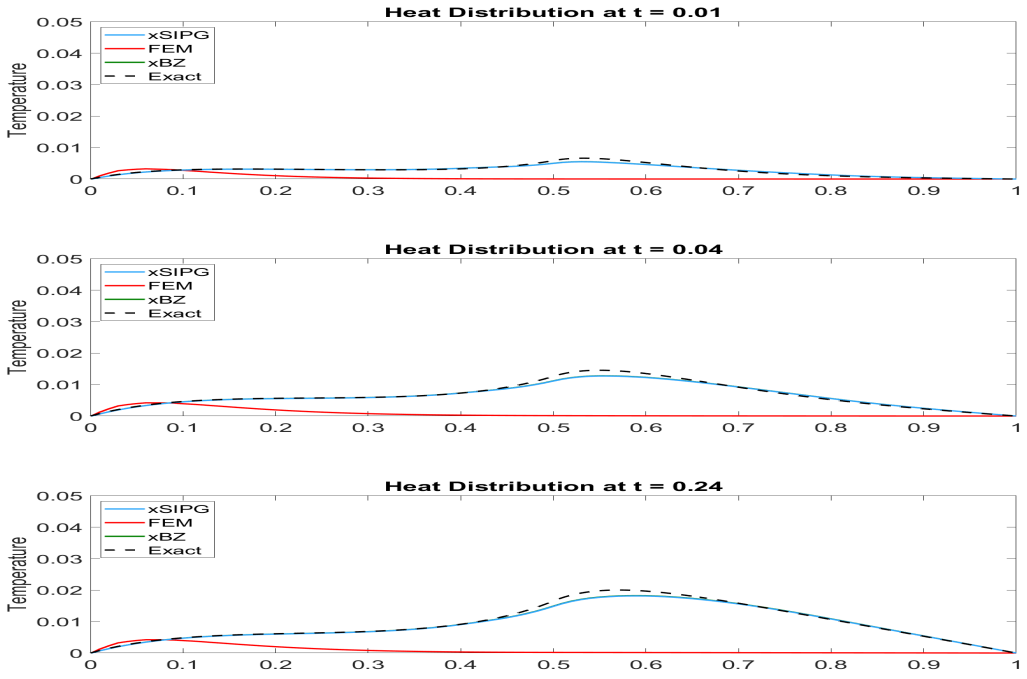


Figure 5.3 Solving the Bioheat equation with $\beta = 100, 1$ on Ω_1 and Ω_2 respectively. Backward Euler in time and FEM in space were used to solve the heat equation after the source Q was computed using xSIPG (Blue) or FEM (Red) as shown above. 100 timesteps and 100 elements were used in the Bioheat equation.

A particularly striking insight emerges when examining the efficacy of xDG methods with merely five elements: the xDG method approaches the true solution while the FEM shows minimal convergence toward an accurate heat distribution. This efficiency gain underlines the significant advantages of employing xDG methods, especially when considering the real-world frequencies utilized in HIFU procedures, which can reach up to a magnitude of one million.

In scenarios where the frequency is increased to $\omega = 1000$, the disparity between xSIPG and xBZ becomes pronounced. Where xBZ exhibits errors comparable to FEM at a frequency of $\omega = 125$, xSIPG notably achieves a solution that is nearly exact, despite using a mere 50 elements. This stark contrast underscores the superior capabilities of xSIPG, particularly in high-frequency applications, thereby highlighting its potential for broader applications in the field.

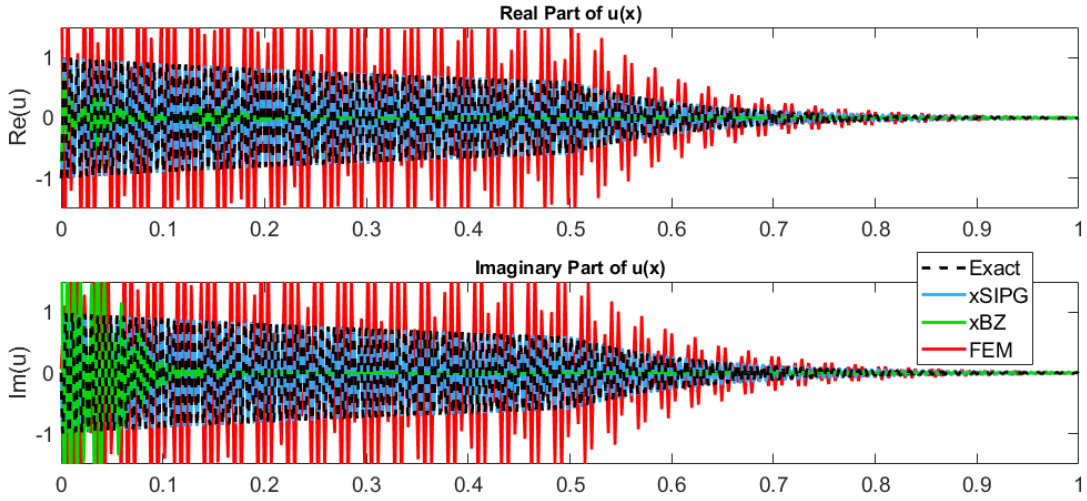


Figure 5.4 Solution p to the coupled Helmholtz problem with $\omega = 1000$ and $\alpha = 1, 8$. 300 elements were used for FEM and 50 were used for xDG. Note the significiant difference between xSIPG and xBZ.

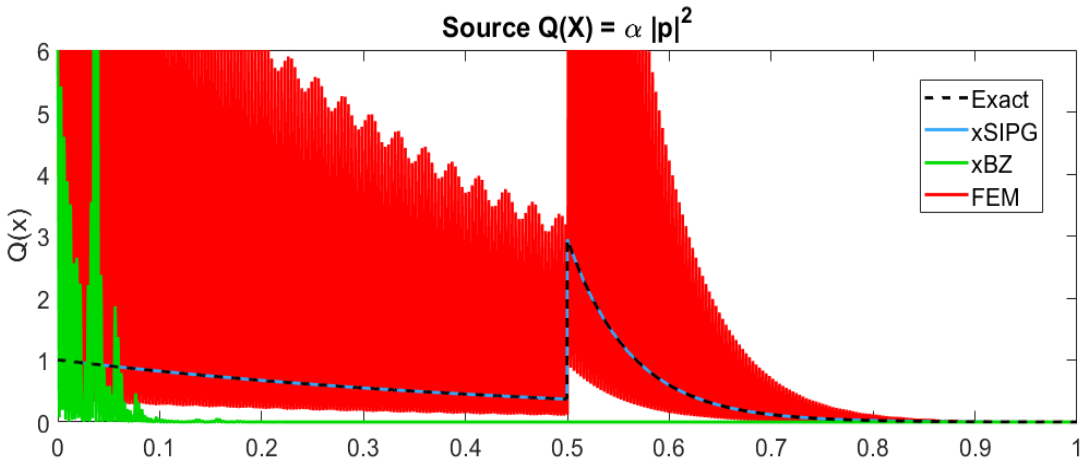


Figure 5.5 Source term Q for the bioheat equation with $\omega = 1000$ and $\alpha = 1, 8$. 300 elements were used for FEM and 50 were used for xDG. Note the significiant difference between xSIPG and xBZ.

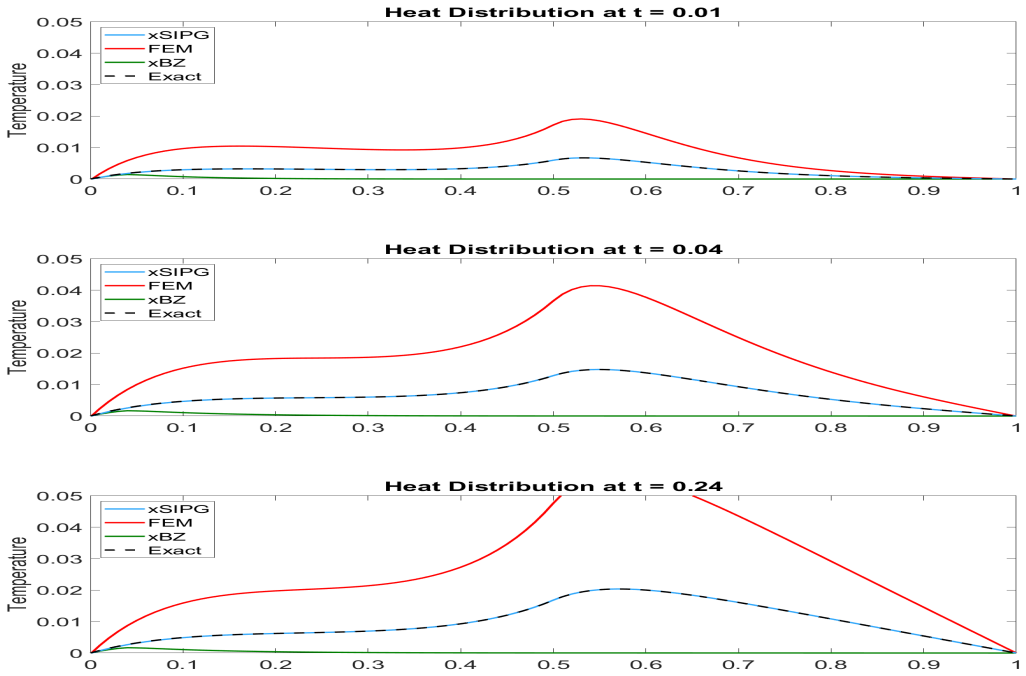


Figure 5.6 Solving the Bioheat equation with $\beta = 100, 1$ on Ω_1 and Ω_2 respectively. Backward Euler in time and FEM in space were used to solve the heat equation after the source Q was computed using xSIPG (Blue) or FEM (Red) as shown above. 100 timesteps and 100 elements were used in the Bioheat equation.

These results elucidate the improvement in xSIPG over xBZ when it comes to solving the Helmholtz problem, and displays the potential for xSIPG to be used for other applications. Furthermore, the computational experiments conducted offer a compelling argument for the adoption of xSIPG in the modeling of HIFU applications. With its ability to yield accurate results at both low and high frequencies, xSIPG stands out as a highly effective method for simulating the thermal effects of focused ultrasound in medical applications. This chapter has laid out the practical advantages of xSIPG, paving the way for its implementation in advanced medical simulations and beyond.

Chapter 6

Conclusions and Future Work

In this work, we have developed and scrutinized a generalized approach for applying the Enriched Symmnetric Interior Penalty Galerkin (xSIPG) method to tackle oscillatory problems characterized by elliptic and Helmholtz equations. Our investigations have led to the identification of lower bounds for penalty parameters and a new penalization method which has the promise of offering a more robust computational approach for oscillatory problems and beyond.

Our analysis reveals that the current xBZ method falls short of providing satisfactory solutions for the Helmholtz equation, a critical component in the accurate simulation of High-Intensity Focused Ultrasound (HIFU) surgery—a limitation that underscores the necessity for more robust and refined computational strategies.

The promise shown by the xSIPG method suggests its applicability may extend to a broader spectrum of complex problems, offering a potentially transformative tool for computational simulations. We have successfully implemented xDG methods within a one-dimensional HIFU model, laying the groundwork for further exploration.

Looking to the horizon, we aim to substantiate the convergence properties of our method through rigorous mathematical proofs, thereby solidifying its theoretical foundations. Equally important is the expansion of our numerical studies into two and three dimensional

HIFU models, which would significantly advance the scope and impact of our research. The prospect of applying xDG methods to novel classes of problems, including those governed by Airy and Schrödinger equations, presents an exciting avenue for future inquiry.

Our aspiration is that these endeavors will not only advance the field of scientific computation but also catalyze significant advancements in a range of applications. This includes enhancing the precision and effectiveness of non-invasive medical procedures such as HIFU and potentially transforming approaches to solving complex, highly oscillatory problems across various scientific disciplines.

Bibliography

[1] Ivo Babuška. "The finite element method with penalty". In: *Mathematics of computation* 27.122 (1973), pp. 221–228.

[2] Ivo Babuška and Jens M Melenk. "The partition of unity method". In: *International Journal for Numerical Methods in Engineering* 40.4 (1997), pp. 727–758.

[3] Ivo Babuška and Miloš Zlámal. "Nonconforming elements in the finite element method with penalty". In: *SIAM Journal on Numerical Analysis* 10.5 (1973), pp. 863–875.

[4] Torsten Bove, Tomasz Zawada, Jørgen Serup, Alexander Jessen, and Mattia Poli. "High-frequency (20-MHz) high-intensity focused ultrasound (HIFU) system for dermal intervention: preclinical evaluation in skin equivalents". In: *Skin Research and Technology* 25.2 (2019), pp. 217–228.

[5] Yekaterina Epshteyn and Béatrice Rivière. "Estimation of penalty parameters for symmetric interior penalty Galerkin methods". In: *Journal of Computational and Applied Mathematics* 206.2 (2007), pp. 843–872.

[6] Donald French, Benjamin Vaughan, and Şuayip Toprakseven. "Error analysis of an extended discontinuous galerkin method for highly-oscillatory problems". In: (2021).

[7] Robert Gracie, Hongwu Wang, and Ted Belytschko. "Blending in the extended finite element method by discontinuous Galerkin and assumed strain methods". In: *International Journal for Numerical Methods in Engineering* 74.11 (2008), pp. 1645–1669.

[8] Jan S. Hesthaven and Tim Warburton. *Nodal Discontinuous Galerkin Methods: Algorithms, Analysis, and Applications*. Springer Science & Business Media, 2007.

[9] Janne MJ Huttunen et al. "Determination of heterogeneous thermal parameters using ultrasound induced heating and MR thermal mapping". In: *Physics in Medicine & Biology* 51.4 (2006), p. 1011.

[10] Frank Ihlenburg and Ivo Babuska. "Finite element solution of the Helmholtz equation with high wave number part II: the hp version of the FEM". In: *SIAM Journal on Numerical Analysis* 34.1 (1997), pp. 315–358.

[11] Frank Ihlenburg and Ivo Babuška. “Finite element solution of the Helmholtz equation with high wave number Part I: The h-version of the FEM”. In: *Computers & Mathematics with Applications* 30.9 (1995), pp. 9–37.

[12] Zahra Izadifar, Zohreh Izadifar, Dean Chapman, and Paul Babyn. “An introduction to high intensity focused ultrasound: systematic review on principles, devices, and clinical applications”. In: *Journal of clinical medicine* 9.2 (2020), p. 460.

[13] Mats G. Larson and Fredrik Bengzon. *The Finite Element Method: Theory, Implementation, and Applications*. Vol. 10. Springer Science & Business Media, 2013.

[14] R Liu, MF Wheeler, CN Dawson, and R Dean. “Modeling of convection-dominated thermoporomechanics problems using incomplete interior penalty Galerkin method”. In: *Computer methods in applied mechanics and engineering* 198.9-12 (2009), pp. 912–919.

[15] Béatrice Rivière. *Discontinuous Galerkin Methods for Solving Elliptic and Parabolic Equations: Theory and Implementation*. Society for Industrial and Applied Mathematics, 2008.

[16] Shuyu Sun and Jiangguo Liu. “A locally conservative finite element method based on piecewise constant enrichment of the continuous Galerkin method”. In: *SIAM Journal on Scientific Computing* 31.4 (2009), pp. 2528–2548.

[17] Lonny L Thompson. “A review of finite-element methods for time-harmonic acoustics”. In: *The Journal of the Acoustical Society of America* 119.3 (2006), pp. 1315–1330.

[18] Ye Yang, Shardool Chirputkar, David N Alpert, Thomas Eason, Stephen Spottswood, and Dong Qian. “Enriched space–time finite element method: a new paradigm for multiscaling from elastodynamics to molecular dynamics”. In: *International journal for numerical methods in engineering* 92.2 (2012), pp. 115–140.