UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
DEPARTAMENTO DE INGENIERÍA MECÁNICA

APPLICATIONS OF GENERALIZED ISOGEOMETRIC ANALYSIS TO PROBLEMS OF TWO-DIMENSIONAL STEADY-STATE HEAT CONDUCTION

MEMORIA PARA OPTAR AL TÍTULO DE
INGENIERO CIVIL MECÁNICO

MAXIMILIANO ANDRÉS PÉREZ HERMOSILLA

PROFESOR GUÍA:
ELENA ATROSHCHENKO

MIEMBROS DE LA COMISIÓN:
ALEJANDRO ORTIZ BERNARDIN
ÁLVARO VALENCIA MUSALEM

SANTIAGO DE CHILE
2017

RESUMEN DE LA MEMORIA PARA OPTAR
AL TÍTULO DE INGENIERO CIVIL MECÁNICO
POR: MAXIMILIANO ANDRÉS PÉREZ HERMOSILLA
FECHA: 20/12/2017
PROF. GUÍA: ELENA ATROSHCHENKO

APPLICATIONS OF GENERALIZED ISOGEOMETRIC ANALYSIS TO PROBLEMS OF TWO-DIMENSIONAL STEADY-STATE HEAT CONDUCTION

El método de aproximación de campo de geometría independiente (GIFT) introducido recientemente en [1] como una extensión del análisis isogeométrico (IGA), permite diferentes bases para la parametrización de la geometría y la aproximación de campo. Por ejemplo, se demostró en [1] cómo se puede emparejar la geometría CAD original dada por NURBS con la solución dada por PHT-splines para problemas bidimensionales de conducción de calor regidos por la ecuación de Poisson. Esta modificación permite superar ciertas deficiencias del método IGA, como la ausencia de refinamiento local, que es crucial para los problemas donde las soluciones muestran altos gradientes, manteniendo la principal ventaja de IGA, es decir, la estrecha integración del análisis con el diseño de CAD.

Este trabajo es una extensión del trabajo realizado en [1]. Tiene dos objetivos principales: el primero es estudiar la adaptabilidad local de la solución, esto se logra implementando y comparando tres medidas de error diferentes; la segunda parte consiste en el estudio de la dependencia de la parametrización de la geometría en la solución, para este objetivo, se analizan dos parametrizaciones diferentes del mismo problema.

Para estudiar la adaptabilidad local, se resuelven dos problemas usando tres métodos diferentes, usando indicadores de error para seleccionar las celdas a refinar: el Método 1 usa un indicador de error de norma-L^2, el Método 2 usa un indicador de error basado en residuos y el Método 3 usa un algoritmo jerárquico. Para probar estos métodos se utiliza el método GIFT con su geometría definida por NURBS y un campo de solución definido por PHT-Splines, las tasas de convergencia, las mallas refinadas y las gráficas de error se comparan con la solución analítica y el refinamiento homogéneo.

Finalmente, dos de los tres métodos muestran mejoras en comparación con el refinamiento homogéneo, donde el Método 2 presenta los mejores resultados, seguido de cerca por el Método 1 y finalmente el método 3, que presenta peores resultados que el refinamiento homogéneo. Para la dependencia de la parametrización de la geometría, los resultados obtenidos muestran una diferencia entre las parametrizaciones, concluyendo que la parametrización de la geometría si afecta la solución, además, se observa que al utilizar el método de refinamiento adaptativo local se presentan mejores tasas de convergencia para ambas parametrizaciones.

Se espera que este trabajo conduzca a posteriores mejoras del método GIFT, propuesto en [1].

Abstract

Geometry Independent Field approximaTion (GIFT) is a method recently introduced in [1] as an extension of the IsoGeometric Analysis (IGA), which allows different bases for the geometry parameterization and the field approximation. For example, it was demonstrated in [1] how the original CAD geometry given by NURBS can be paired with the solution given by PHT-splines for 2D problems of heat conduction governed by Poisson's equation. This modification allows to overcome such deficiencies of IGA, as absence of local refinement, which is crucial for the problems, where solutions exhibit high gradients, while keeping the main advantage of IGA, i.e. tight integration of analysis with CAD design.

This work is an extension of the work done in [1]. It has two principal objectives: the first one is to study the local adaptivity of the solution, this will be achieved by implementing and comparing three different error measures; the second part consists in the study of the dependence of the parameterization of the geometry on the solution, for this goal, two parameterizations of the same problem are tested.

To study the local adaptivity, two problems were solved using three different methods, that used error indicators to select the cells to refine: Method 1 uses an L^2-norm error indicator, Method 2 uses a residual based error indicator and Method 3 uses a hierarchical algorithm. To test this methods, the GIFT method with a NURBS geometry and a PHT-Splines solution field was used, the convergence rates, the refined meshes and the error plots were compared to the analytical solution and the homogeneous refinement.

Finally, two of the three methods show improvements compared to the homogeneous refinement, where Method 2 presents the best results, followed closely by Method 1, and finally, the method 3, which presents worse results than the homogeneous refinement. For the geometry parameterization dependence, the results obtained show an important difference between the parameterizations, concluding that the geometry parameterization does affect on the solution, whilst the local adaptive refinement method works fine for both parameterizations.

It is expected that this work will lead to further improvement of the GIFT method, proposed in [1].

To my little girl.
Para mis hermanos, Vicente y Nicolás,
belive in yourself, be yourself, you own your future,
embrace it or just change it, be happy.

Agradecimientos

Agradezo a quienes estuvieron durante estos 6 años junto a mi, a todos aquellos que estuvieron conmigo durante cada prueba y control. A los grandes amigos que pude conocer, a Tropikalismos y los viejos amigos que aún perduran luego del colegio, los que siempre estarán ahí. A cada persona que me hizo pasar un buen rato en la universidad, a mi equipo de basquetbol por esas incontables mañanas de entrenamiento a las 7 am que alegraban mis días desde temprano. A Gonzálo Bustos, más que un compañero de sección, un gran amigo, de esas amistades que perduran en el tiempo. Agradezco en especial a Miguel Inostroza "El rata" y Sebastián Goza "El Goza" mis mejores amistades durante la carrera, con quienes pasteamos todos los ramos y estuvieron en las buenas y en las malas apoyandome. Agradezco a todos los que me ayudaron económicamente para poder estudiar y a mis padres las veces que me apoyaron.

Mención honrosa a quienes más me apoyaron en este último tramo, a Felipe Contreras por todo su apoyo y buena onda y a Elena Atroshchenko mi profesora guía quién fue pilar fundamental durante mi trabajo de título, gracias a su excelente disposición, ayuda e interés.

Finalmente agradezco a quien fue la persona más importante durante esta larga carrera, mi polola Javiera González, quien me apoyó siempre que lo necesité, desde sus llamados previo a los controles los sábados a las 9 am cuando mechón, hasta esas noches hasta las 4 am en la universidad terminando la tesis o preparando presentaciones cuando viejón. Por todos esos lindos momentos que pasamos, por las preciadas vacaciones descansando en las playas de ecuador, acampando en la patagonia y por todos esos viajes que se vendrán, agradezco tu amor incondicional.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Motivation

Over the last few decades, computer aided numerical methods have been extensively used to solve or to obtain an approximate solution to complex engineering problems (e.g. heat propagation, vibratory responses, fluid dynamics, etc.) through the resolution of partial differential equations. The development of numerical methods is motivated by the great complexity of obtaining such solutions analytically, and due to rapid developments in Computer Sciences, the time spent on solving such problems numerically, dramatically decreased. Due to the nature of numerical methods, the solutions are discrete, which implies that is always possible to obtain a more detailed and precise solution because of the real numbers density, making necessarily the use of larger amounts of computational data to improve the quality of the desired solution. This, together with the great technological advances lead the career of the numerical methods to achieve better responses to the problems. The classic numeric method for the resolution of problems has been the Finite Element Method (FEM), nevertheless, it presents certain limitations, that result in large computational times and inaccuracy in the solutions.

Nowadays, the numerical methods are being very important in different areas of science, including engineering and health. That is why engineers and mathematicians seek to obtain increasingly accurate solutions to known problems in order to be able to solve faster and more accurately problems that still have not been resolved by now.

In the pursuit for the numerical methods optimization, a new method was proposed by Hughes in [2], known as IsoGeometric Analysis (IGA), which gathers the best of Computer Aided Design methods (CAD) and Finite Element methods (FEM or FEA). While in FEM, for h-method (standard FEM), the geometry of the problem is constantly refined, to achieve this is necessary the use of lots of computational resources. Whilst in the IGA method the geometry is represented exactly as the CAD generates it, so there is no need to refine and change the geometry to refine the solution, significantly reducing the resources used. This

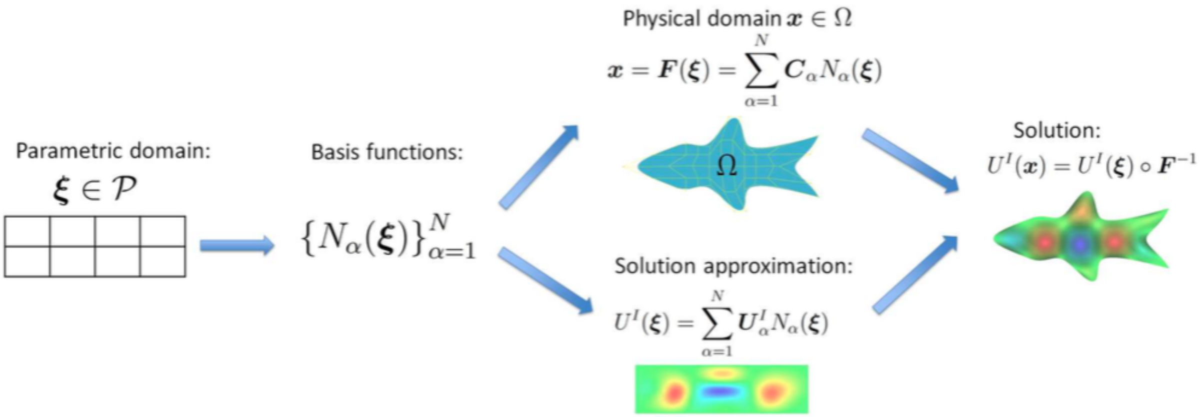
Physical domain $\boldsymbol{x} \in \Omega$
$\boldsymbol{x} = \boldsymbol{F}(\boldsymbol{\xi}) = \sum_{\alpha=1}^{N} \boldsymbol{C}_\alpha N_\alpha(\boldsymbol{\xi})$
Parametric domain:
$\boldsymbol{\xi} \in \mathcal{P}$
Basis functions:
$\{N_\alpha(\boldsymbol{\xi})\}_{\alpha=1}^{N}$
Ω
Solution:
$U^I(\boldsymbol{x}) = U^I(\boldsymbol{\xi}) \circ \boldsymbol{F}^{-1}$
Solution approximation:
$U^I(\boldsymbol{\xi}) = \sum_{\alpha=1}^{N} \boldsymbol{U}_\alpha^I N_\alpha(\boldsymbol{\xi})$

Figure 1.1: The main idea of isogeometric analysis (IGA). Source: [1].

is why Isogeometric Analysis allows to mix the best of CAD and FEM, generating an exact geometry by using basis functions called Splines (the most commonly used ones are B-Splines and NURBS [Non Uniform Rational B-Splines]), that are used for both, parameterization of the CAD geometry and approximation of the solution field. Despite the advantages of IGA, there are shortcomings in this method like the difficult to represent complex multi patch geometries and the lack of local adaptivity.

Recently, in [1] a new method was introduced, called Geometry-Independent Field approximation (GIFT), which presents further generalization of the IGA. IGA has the potential to provide closer integration between geometric design and numerical analysis since any change in the CAD model is directly inherited by the field approximation, but sometimes using the same functions for both geometry and solution field may create unwanted constraints, this is why unlike IGA, in GIFT, the solution field and geometry basis can be chosen independently. Allowing GIFT method to address the main shortcomings of IGA, by enabling:

1. Flexibility of solution representation.
2. Straightforward treatment of multi-patch configurations.
3. Independent field refinement.

In Figures 1.1 and 1.2, the main ideas of IGA and GIFT methods are simply represented.

As GIFT method allows to select separately the basis functions for the geometry and solution field, it helps to take advantage of two different Splines in the same problem. This main advantage using PHT-Splines for the solution field, allowing local adaptivity, and a simple parameterization of the geometry using NURBS, allows the method getting better convergence rates in problems with sharp peaks. In Figure 1.3 one can see the comparison of convergence rates between IGA and GIFT method made in [1], using PHT-Splines and B-Splines for the solution field and NURBS for the geometry in case of GIFT method and PHT-Splines and B-Splines for solution and geometry field for IGA.

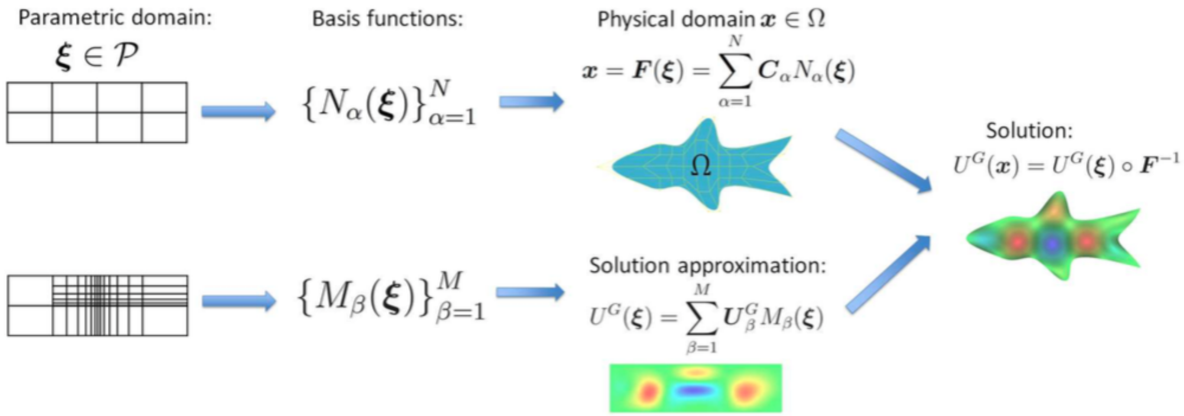
Parametric domain:
$\boldsymbol{\xi} \in \mathcal{P}$
Basis functions:
$\{N_\alpha(\boldsymbol{\xi})\}_{\alpha=1}^{N}$
Physical domain $\boldsymbol{x} \in \Omega$
$\boldsymbol{x} = \boldsymbol{F}(\boldsymbol{\xi}) = \sum_{\alpha=1}^{N} \boldsymbol{C}_\alpha N_\alpha(\boldsymbol{\xi})$
Ω
Solution:
$U^G(\boldsymbol{x}) = U^G(\boldsymbol{\xi}) \circ \boldsymbol{F}^{-1}$
$\{M_\beta(\boldsymbol{\xi})\}_{\beta=1}^{M}$
Solution approximation:
$U^G(\boldsymbol{\xi}) = \sum_{\beta=1}^{M} \boldsymbol{U}_\beta^G M_\beta(\boldsymbol{\xi})$

Figure 1.2: The main idea of geometry-independent field approximation (GIFT). Source: [1].

In this work the local adaptivity, using PHT-Splines for the solution field and NURBS for the geometry, will be studied using different methods to trigger local refinement, also the dependency of the parameterization of the geometry with the convergence of the solution will be studied.

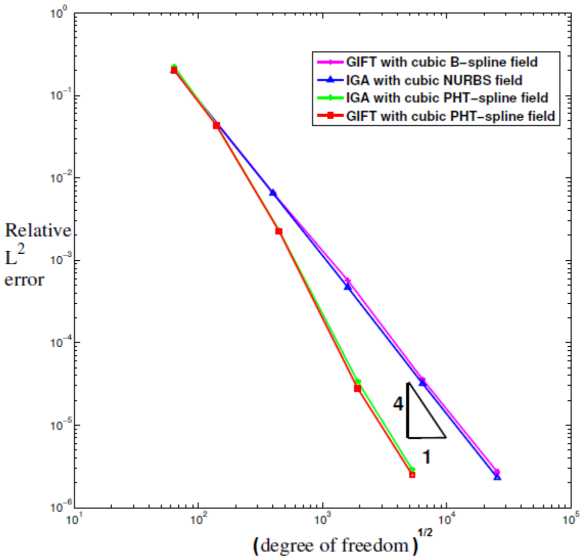

Figure 1.3: Comparison of the convergence results with the IGA method using quadratic NURBS and GIFT method using cubic B-Spline spaces during h-refinement. Source: [3]

1.2 Objectives and scope

1.2.1 General objective

The general objective of this work is to continue studying performance of the GIFT method, initiated in [1].

1.2.2 Specific objectives

To achieve the main objective, two specific objectives are established:

1. The study of the local adaptivity of the solution. This will be achieved by implementing three different error measures methods as indicators for the code to self-select the cells to be refined in two different problems:

   - Using absolute error with L^2-norm for every cell.
   - Using an error indicator that does not depend on the analytical solution to select the cells to be refined.
   - Using a hierarchical method that depends on the previous solution to select the cells to be refined.

2. The study of the dependence of the parameterization of the geometry on the solution.

   - Analyzing and comparing two problems with the same geometry and the same solution using two different parameterizations.
   - Using a uniform parameterization and an irregular parameterization.

1.2.3 Scope

This work will analyze two-dimensional (2D) problems governed by partial differential equations. Comparing analytical and numerical results obtained by an in-house code programmed in C++, which has to be modified and extended to accomplish the objectives of this work.

Chapter 2

Theoretical Background

2.1 Heat transfer

The transfer of heat is normally done from a high temperature object to a lower temperature object. Heat transfer changes the internal energy of both systems involved according to the First Law of Thermodynamics:

$$Q = Wk + \frac{\mathrm{d}U}{\mathrm{d}t}, \tag{2.1}$$

where Q is the heat transfer rate and Wk is the work transfer rate. They may be expressed in joules per second (J/s) or Watts (W). The derivative $\mathrm{d}U/\mathrm{d}t$ is the rate of change of internal thermal energy U, with time t. This works for closed systems [4].

The analysis of the heat transfer processes can generally be done without reference to any work processes, although heat transfer might subsequently be combined with work in the analysis of real systems. If $p\ \mathrm{d}V$ work is the only work that occurs, then Eq. (2.1) can be rewritten as:

$$Q = p\frac{\mathrm{d}V}{\mathrm{d}t} + \frac{\mathrm{d}U}{\mathrm{d}t}. \tag{2.2}$$

For heat transfer there are three methods:

- Conduction.
- Convection.
- Radiation.

For this work we will focus just on conduction. The heat equation for a function u is:

$$\rho c_p \frac{\mathrm{d}u}{\mathrm{d}t} - k\Delta u = q, \tag{2.3}$$

where ρ is the density, c_p is the specific heat capacity, k is a positive constant that represents the thermal conductivity, Δ denotes the laplace operator and q represents the heat source.

In a steady state Eq. (2.3) can be reduced to the form:

$$\Delta u = \frac{-q}{k}, \tag{2.4}$$

this equation is known as the Poisson's equation. In this particular case, for Heat conduction problems, q is expressed in Joules or Calories and k is expressed in $\frac{W}{Km}$. For this work we focus on the mathematical expressions, this is why the units are not shown in the posterior analysis.

2.1.1 Poisson's equation

Poisson's equation is a partial differential equation, the inhomogeneous equivalent of Laplace's equation given by Eq. (2.6). It is encountered in the modeling of a variety of problems in mechanics and physics, ranging from the study of fluid flows in porous media to the theory of gravitation [5] and it is defined by Eq. (2.5):

$$\Delta u = f, \quad \Omega \subset \mathbb{R}^n, \tag{2.5}$$

$$\Delta u = 0, \quad \Omega \subset \mathbb{R}^n, \tag{2.6}$$

where Δ is the laplace operator, a differential operator given by the divergence of the gradient of the function, u is the unknown solution and f is a function in Ω.

This equation is particularly important because is going to be used in this work. The weak form of Poisson's equation is given by:

$$\int_\Omega \nabla w \cdot \nabla u \ \mathrm{d}\Omega = \int_\Omega wf \ \mathrm{d}\Omega + \int_\Gamma w \ (\overrightarrow{n} \cdot \nabla u) \ \mathrm{d}\Gamma, \tag{2.7}$$

where w is the weight function and $\overrightarrow{n}$ is the unit outward normal of the boundary Γ of the problem domain.

To solve Laplace or Poisson's equation is necessary to impose boundary conditions.

The most common boundary conditions are:

- Dirichlet condition.

$$u(x) = g(x), \quad x \in \delta\Omega. \tag{2.8}$$

- Neumann condition.

$$\frac{\partial u(x)}{\partial n} = h(x), \quad x \in \delta\Omega. \tag{2.9}$$

- Robin condition.

$$au(x) + b\frac{\partial u(x)}{\partial n} = \beta(x), \quad x \in \delta\Omega. \tag{2.10}$$

Where u is the solution, g, h and β are functions in $\delta\Omega$, a and b are constants and $\partial u/\partial n$ denotes the normal derivative of u at the boundary.

2.2 Splines

It is commonly accepted that the first mathematical reference to splines is Schoenberg's paper [6], which is probably the first place that the word "Spline" is used in connection with smooth, piecewise polynomial approximation. According to [7], the splines were first used in the aircraft and ships crafting in the days before computer modeling. The engineers needed to draw smooth curves through a set of points, this was achieved by using thin metal or wooden beam (called spline) and bended through the control points with metal weights (called ducks). This meant that if more influence of the control point was needed, the simple solution was to add more weights to the control point. Despite this solution, a more detailed method was required, a mathematical way was needed to describe the shape of the curve. For more information about spline history refer to [8].

Cubic splines were the first ones used to recreate the spline-duck method mathematically due to its close relation with the draftsman's spline, an important aspect is that this spline is represented actually by piecewise cubic polynomial between control points, but to assure a smooth continuity they must follow certain rules, i.e. they must be continuously differentiable (C^0, C^1 and C^2 continuity).

The splines proved to be an effective tool in the elementary processes of interpolation and approximate integration. An outstanding characteristic, however, is its effectiveness in numerical differentiation. In computer graphics, the pursuit of representing smooth curves for desired designs makes splines the best functions to approximate the geometry of the graphic, that is why it is widely used for Computer Aided Design softwares. In [9] the reader may find more rigorous and complete information about the use of splines in computer graphics.

Nowadays there are multiple kinds of splines that are used to represent smooth curves, these curves are used from graphic designs to solutions of complex engineering problems. In the next sections, three important kinds of splines relevant to this work are presented: B-Splines, NURBS and PHT-Splines.

B-Splines

B-Splines are a kind of spline that represent curves like piecewise Bezièr curves, see an example in Figure 2.1 (for more information about Bezièr curves the reader is referred to [10]). A particular property of the B-Splines is local control, by which we mean that altering the position of a single control vertex causes only a part of the curve to change. This makes it possible to modify part of a curve (or surface) without affecting other portions that are already satisfactory, which is often useful in geometric design and modeling. An added benefit of local control is that it minimizes the work required to recompute a curve after a control vertex has been moved since only a small part of the curve has changed [9].

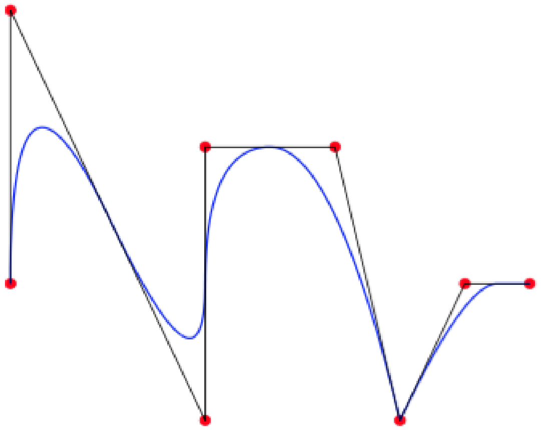

Figure 2.1: Quadratic B-Spline curve in R^2. Control point locations are denoted by red dots. Source: [2].

A *p-th-degree B-Spline curve* is defined by:

$$C(u) = \sum_{\mathrm{i}=0}^{n} N_{\mathrm{i},p}(u) P_{\mathrm{i}}, \tag{2.11}$$

where P_{i} are the control points, and $N_{\mathrm{i},p}(u)$ are the pth-degree B-Spline basis functions, which can be defined with Cox-de Boor recursion formula:

$$N_{\mathrm{i},0}(u) = \begin{cases} 1 & \mathrm{i}f\ u_{\mathrm{i}} < u < u_{\mathrm{i}+1} \\ 0 & otherwise \end{cases},$$

$$N_{\mathrm{i},p}(\xi) = \frac{\xi - \xi_{\mathrm{i}}}{\xi_{\mathrm{i}+1} - \xi_{\mathrm{i}}} N_{\mathrm{i},p-1}(\xi) + \frac{\xi_{\mathrm{i}+p+1} - \xi}{\xi_{\mathrm{i}+p+1} - \xi_{\mathrm{i}+1}} N_{\mathrm{i}+1,p-1}(\xi). \tag{2.12}$$

Important properties of B-Splines curves are:

1. They have continuous derivatives of order $p - k$, where k is a number of repetition of the knots.

2. An affine transformation of a B-spline curve is obtained by applying the transformation to the control points. We refer to this property as affine covariance.

B-Splines surfaces: For a B-Spline surface of $p-th$ degree in ξ direction and $q-th$ degree in η direction, given a control points net $P_{\mathrm{i},j}$, $\mathrm{i} = 1, 2, ..., n$, $j = 1, 2, ..., m$, and knot vectors Ξ_1=$\{\xi_1, \xi_2, ..., \xi_{n+p+1}\}$, and Ξ_2=$\{\eta_1, \eta_2, ..., \eta_{m+q+1}\}$ is defined by:

$$S(\xi, \eta) = \sum_{\mathrm{i}=1}^{n} \sum_{j=1}^{m} N_{\mathrm{i},p}(\xi) M_{j,q}(\eta) P_{\mathrm{i},j}, \tag{2.13}$$

where $N_{\mathrm{i},p}$ and $M_{j,q}$ are B-Splines basis functions of the surface.

Non Uniform Rational B-Splines (NURBS)

A pth-degree NURBS curve is defined by:

$$C(u) = \frac{\sum_{\mathrm{i}=0}^{n} N_{\mathrm{i},p}(u) w_{\mathrm{i}} P_{\mathrm{i}}}{\sum_{\mathrm{i}=0}^{n} N_{\mathrm{i},p}(u) w_{\mathrm{i}}} \qquad a \leq u \leq b, \tag{2.14}$$

where P_{i} are the control points (forming a control polygon), w_{i} are the weights and $N_{\mathrm{i},p}(u)$ are the pth-degree B-spline basis functions defined by Eq. (2.12), on the non periodic knot vector

$$U = \{\underbrace{a, \ldots, a}_{p+1}, u_{p+1}, \ldots, u_{n-p-1}, \underbrace{b, \ldots, b}_{p+1}\}.$$

Unless otherwise stated, we assume that $a = 0,\ b = 1$ and $w_{\mathrm{i}} > 0$ for all i. Setting

$$R_{\mathrm{i},p}(u) = \frac{N_{\mathrm{i},p}(u)w_{\mathrm{i}}}{\sum_{j=0}^{n} R_{j,p}(u)w_j}, \tag{2.15}$$

allows us to rewrite Eq. (2.14) in the form:

$$C(u) = \sum_{\mathrm{i}=0}^{n} R_{\mathrm{i},p}(u)P_{\mathrm{i}}, \tag{2.16}$$

where $R_{\mathrm{i},p}(u)$ are the rational basis functions of NURBS, they are piecewise rational functions on $u \in [0, 1]$.

$R_{\mathrm{i},p}(u)$ have the following properties derived from Eq. (2.14) and the corresponding properties of $N_{\mathrm{i},p}(u)$:

1. Non negativity and partition of the unity.
2. Affine invariance: an affine transformation is applied to the curve by applying it to the control points; NURBS curves are also invariant under perspective projections, a fact which is important in computer graphics.
3. Strong convex hull property.
4. $C(u)$ is infinitely differentiable on the interior of knots spans and is $p - k$ times differentiable at a knot of multiplicity k.
5. NURBS curves contain nonrational B-Splines and rational and nonrational Bézier curves as special cases.

NURBS surfaces: Analogously to B-Splines, a NURBS surface of $p - th$ degree in ξ direction and $q - th$ degree in η direction is defined by:

$$S(\xi, \eta) = \sum_{\mathrm{i}=1}^{n} \sum_{j=1}^{m} R_{\mathrm{i},p}^{p,q}(\xi, \eta)P_{\mathrm{i},j}, \tag{2.17}$$

where $P_{\mathrm{i},j}$ is a set of $n \times m$ bidirectional control net and $R_{\mathrm{i},p}^{p,q}$ is defined by:

$$R_{\mathrm{i},p}^{p,q}(\xi, \eta) = \frac{N_{\mathrm{i},p}(\xi)M_{j,q}(\eta)}{\sum_{\hat{\mathrm{i}}}^{n} \sum_{\hat{j}}^{m} N_{\hat{\mathrm{i}},p}(\xi)M_{\hat{j},q}(\eta)w_{\hat{\mathrm{i}},\hat{j}}}, \tag{2.18}$$

where $N_{\hat{\mathrm{i}},p}(\xi)$ and $M_{\hat{j},q}(\eta)$ are B-Splines functions defined on knots vectors Ξ_1 and Ξ_2 respectively.

PHT-Splines

An important progress of locally refinable splines began with T-splines [11, 12] which overcomes a weakness of NURBS models that contains a large number of superfluous control points by allowing T-junctions in the control mesh, calling the new control mesh as T-mesh and have already shown its potential as a powerful modeling tool for advanced geometric modeling and adaptive isogeometric analysis [13, 14].

Polynomials Splines over Hierarchical T-meshes (PHT-Splines), introduced by Deng et al. in 2006 [15], are a generalization of B-Splines over hierarchical T-meshes which in addition to main properties of B-Splines it inherits the main properties of T-Splines such as adaptivity and flexibility, also exhibits several advantages over T-Splines. The conversion between NURBS and PHT-Splines is very fast unlike T-Splines. The local refinement algorithm of PHT-Splines is local and simple. However, the trade off is PHT-Splines are only C^1 continuous but this is enough for most applications in solid and structural mechanics.

PHT-Splines possess a very efficient local refinement property. This property makes PHT-Splines preferable in geometric processing, adaptive finite elements and isogeometric analysis. Owing to these properties, PHT-Splines have been widely applied in geometric modeling and isogeometric analysis. With PHT-Splines, surface models can be reconstructed and simplified efficiently [15].

Given a T-mesh $\mathbb{T}$, $\mathcal{T}$ denotes all the cells in $\mathbb{T}$ and Ω the region occupied by all the cells in $\mathcal{T}$. A spline space over the given T-mesh $\mathbb{T}$ is defined by:

$$\mathbf{T}(m, n, \alpha, \beta, \mathbb{T}) := \left\{ s(x, y) \in C^{\alpha,\beta}(\Omega) \qquad |s(x, y)|_{\phi} \in \mathbb{P}_{mn} \quad \forall \phi \in \mathcal{T} \right\}, \tag{2.19}$$

where $\mathbb{P}_{mn}$ is the space of all the polynomials of bi-degree (m, n), and $C^{\alpha,\beta}(\Omega)$ is the space consisting of all the bivariate functions which are continuous in Ω with order α along x direction and with order β along y direction.

Due to the nature of hierarchical T-meshes, the local refinement of PHT-Splines is achieved by cross insertion, i.e., dividing a cell into four subcells with a cross.

PHT-Splines basis functions have the same important properties as B-Splines, such as nonnegativity, local support and partition of unity.

PHT-Splines surfaces: Let $\mathbb{T}$ be a hierarchical T-mesh, and $b_j(u, v)$, $j = 1, ..., \mathrm{d}$ be the basis functions of PHT-Splines. Then the polynomial spline surface over $\mathbb{T}$ (called PHT-Spline surface) is defined by:

$$S(u,v) = \sum_{j=1}^{\mathrm{d}} C_j b_j(u,v) \qquad (u,v) \in [0,1] \times [0,1], \tag{2.20}$$

where C_j are control points. PHT-Splines surfaces have similar properties with B-Splines surfaces such as convex-hull property, affine invariant, local support, etc. To efficiently manipulate and evaluate a PHT-Spline surface, one should maintain the Bézier representation of the PHT-Splines surface in every cell, since the basis functions are represented in Bézier forms in every cell.

2.2.1 Finite Element Method

The Finite Element Method (FEM) is used to solve physics and engineering problems where other methods are difficult to use. Mathematical models of complex problems are solved by numerical simulations, by its simplicity to approximate solutions despite complex geometries, physics and/or boundary conditions.

In the Finite Element Method, a given domain is viewed as a collection of subdomains, and over each subdomain the governing equation is approximated by any of the traditional variational methods. This way it is easier to represent a complicated function as a collection of simple polynomial.

The FEM is characterized by three features [16]:

1. The domain of the problem is represented by a collection of simple subdomains, called finite elements. The collection of finite elements is called the finite element mesh.

2. Over each finite element, the physical process is approximated by functions of the desired type (polynomials or otherwise), and algebraic equations relating physical quantities at selective points, called nodes, of the element are developed.

3. The element equations are assembled using continuity and/or "balance" of physical quantities.

In the finite element method, we seek an approximation u_h of the solution u in the form:

$$u \approx u_h = \sum_{j=1}^{n} u_j \psi_j, \tag{2.21}$$

where u_j are the values of u_h at the element nodes and ψ_j are the interpolation functions. This representation is substituted into a weak form of the problem and a system of linear algebraic equation is obtained to determine unknown nodal values u_j.

2.2.2 IsoGeometric Analysis (IGA)

The isogeometric analysis (IGA) is a method introduced in [2] where basis functions generated from NURBS are employed to construct an exact geometric model and for analysis purposes. The basis is refined and/or its order elevated without changing the geometry (see Figure 1.1). This makes a complete integration between CAD and Finite Element Analysis (FEA), which by the unnecessary subsequent communication with the CAD description allows better computational efficiency. More detailed examples can be found in [2].

The main features of IGA are the use of same spline representation for both, geometry and the solution field and having every element iso-geometric, the union of the fields is better achieved.

In the following equations the geometry and the solution field splines representation are shown:

$$\mathbf{x}(\boldsymbol{\xi}) = \sum_{\alpha=1}^{N} \mathbf{C}_\alpha N_\alpha(\boldsymbol{\xi}), \tag{2.22}$$

$$u(\boldsymbol{\xi}) = \sum_{\alpha=1}^{N} D_\alpha N_\alpha(\boldsymbol{\xi}), \tag{2.23}$$

where $\mathbf{C}_\alpha \in \mathbb{R}^{\mathrm{d}}$ are the control points, $\mathbf{D}_\alpha$ are the control variables and $N_\alpha(\boldsymbol{\xi})$ are the NURBS basis functions.

This representation is subsequently substituted into the weak form of the problem given by Eq. (2.7) together with $w = N_\beta(\boldsymbol{\xi})$ to yield the following algebraic equation for the vector of unknown control variables U:

$$\mathbf{K}U = \mathbf{F}, \tag{2.24}$$

where $\mathbf{K}$ is the stiffness matrix and $\mathbf{F}$ is the force vector.

Poisson's equation weak form

For example, for Poisson's problem given by Eq. (2.5) with Neumann's boundary condition (Eq. (2.9)), using the weak form of the equation, given by Eq. (2.7) with the solution representation $u(\boldsymbol{\xi})$ defined in Eq. (2.23) and selecting the weight function to be the i^{th} shape function, i.e. $w = N_{\mathrm{i}}(x, y)$ we get the following i^{th} linear algebraic equation for N unknown variables:

$$\sum_{j=1}^{N}\left\{\int_{\Omega}\left(\frac{\partial N_{\mathrm{i}}}{\partial_x}\frac{\partial N_j}{\partial_x}+\frac{\partial N_{\mathrm{i}}}{\partial_y}\frac{\partial N_j}{\partial_y}\right)\,\mathrm{d}\Omega\right\}u_j=\int_{\Omega}N_{\mathrm{i}}f\ \mathrm{d}\Omega+\int_{\Gamma}N_{\mathrm{i}}h(x)\ \mathrm{d}\Gamma. \tag{2.25}$$

This equation is valid for $\mathrm{i} = 1, 2, ..., N$. All N equations can be represented by Eq. (2.24) where:

$$\mathbf{K}_{\mathrm{i}j}=\int_{\Omega}\left(\frac{\partial N_{\mathrm{i}}}{\partial_x}\frac{\partial N_j}{\partial_x}+\frac{\partial N_{\mathrm{i}}}{\partial_y}\frac{\partial N_j}{\partial_y}\right)\mathrm{d}\Omega, \tag{2.26}$$

$$\mathbf{F}_{\mathrm{i}}=\mathbf{f}_{\mathrm{i}}+\mathbf{b}_{\mathrm{i}}, \tag{2.27}$$

$$\mathbf{f}_{\mathrm{i}}=\int_{\Omega}N_{\mathrm{i}}\mathbf{f}\ \mathrm{d}\Omega, \tag{2.28}$$

$$\mathbf{b}_{\mathrm{i}}=\int_{\Gamma}N_{\mathrm{i}}h(x)\ \mathrm{d}\Gamma, \tag{2.29}$$

U is the unknown vector with N entries, $\mathbf{f}$ is the force vector and $\mathbf{b}$ is the boundary integral vector.

2.2.3 Geometry-Independent Field approximaTion (GIFT)

The Geometry-Independent Field approximaTion (GIFT) is a new method explained in [1]. It is a generalization of the isogeometric analysis (IGA) method where the spline spaces used for the geometry and the field variables can be chosen and adapted independently while preserving geometric exactness and tight CAD integration.

In GIFT, for a given computational domain with spline form, the solution field can have a different spline representation, such as B-Splines of different degrees, PHT-Splines, T-Splines and generalized B-Splines. Importantly, the geometry of the computational domain has the same spline representation as that of the given CAD boundary.

In Figure 2.2, one can see an example of the GIFT method in a complex geometry defined by NURBS and PHT-Splines for the solution field. The parametric domain mesh and the color-map error of the GIFT solution are shown.

Analogously to the IGA, the computational domain $\Omega \in \mathbb{R}^{\mathrm{d}},\ \mathrm{d} \geq 2$, defined on parametric domain $\mathcal{P}$ has the following spline representation:

$$\mathbf{x}(\boldsymbol{\xi}) = \sum_{\alpha=1}^{N} \mathbf{C}_\alpha N_\alpha(\boldsymbol{\xi}). \tag{2.30}$$

Unlike in IGA, in GIFT method, the solution field $u(\boldsymbol{\xi})$ can have a different spline representation, as shown in Figure 1.2:

$$u(\boldsymbol{\xi}) = \sum_{\beta=1}^{M} U_\beta M_\beta(\boldsymbol{\xi}), \tag{2.31}$$

where U_β, $\beta = 0, 1, \ldots, M$ are the control variables to be solved and $M_\beta(\boldsymbol{\xi})$ are the basis functions of the specified spline space defined on the parametric domain $\mathbb{P}$.

The main features of GIFT are:

1. Possibility to flexibly choose between different spline spaces with different properties to better represent the solution of the problem, e.g. the continuity of the solution field, boundary layers, singularities, whilst retaining geometrical exactness of the domain boundary.

2. For multi-patch analysis, where the domain is composed of several spline patches, the continuity condition between neighboring patches on the solution field can be automatically guaranteed without additional constraints in the variational form.

3. Refinement operations by knot insertion and degree elevation are performed directly on the spline space of the solution field, independently of the spline space of the geometry of the domain, which makes the method versatile and allows local refinement. GIFT with PHT-spline solution spaces and NURBS geometries is used to show the effectiveness of the proposed approach in [1].

Formulation of GIFT

Consider an open domain $\Omega \subset \mathbb{R}^{\mathrm{d}}$, $\mathrm{d} \geq 2$, with boundary Γ consisting of two parts Γ_D and Γ_N, such that $\Gamma = \Gamma_D \bigcup \Gamma_N,\ \Gamma_D \bigcap \Gamma_N = \varnothing$. The domain Ω is parameterized on a parametric domain $\mathcal{P}$ by mapping $\boldsymbol{F}$:

$$\boldsymbol{F} : \mathcal{P} \to \Omega, \qquad \boldsymbol{x} = \boldsymbol{F}(\boldsymbol{\xi}), \qquad \boldsymbol{x} \in \Omega,\ \boldsymbol{\xi} \in \mathcal{P}. \tag{2.32}$$

Using Eq. (2.30) one can represent the geometry where $N_\alpha(\boldsymbol{\xi})$ is the chosen basis function known as geometry basis. For change of variables we will also need the Jacobian matrix $J(\boldsymbol{\xi})$ of the mapping F, which is given by:

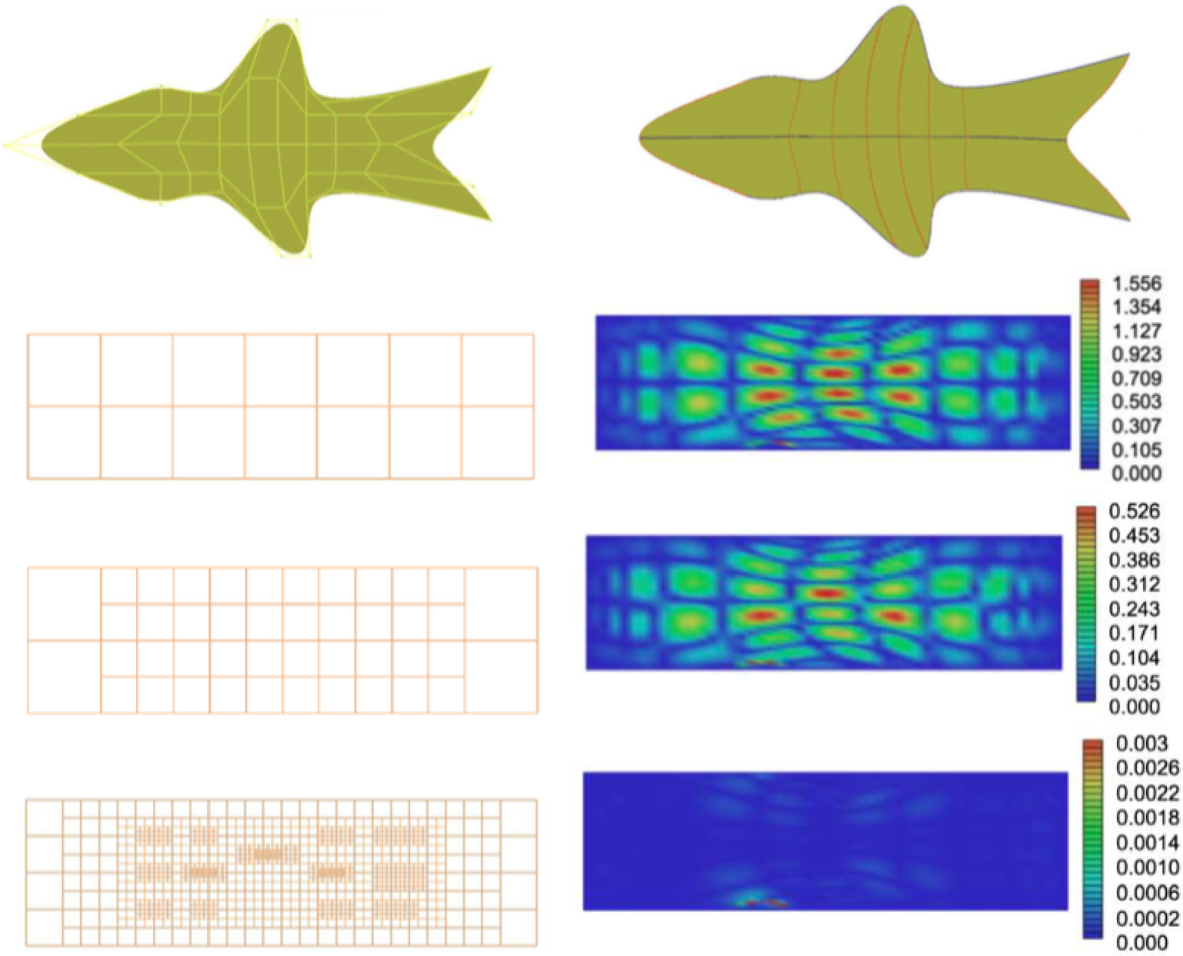

Figure 2.2: GIFT utilized with local refinement using PHT-splines. The first row shows the computational domain with the control mesh (left) and knot patches (right). From the second row to the fourth row, we show the T-mesh in the parametric domain (left) and the corresponding color-map of error measurement (right) with different scales during local refinement operations. Source: [1].

$$J_{\mathrm{i}j}(\boldsymbol{\xi}) = \frac{\partial x_{\mathrm{i}}}{\partial \xi_j}(\boldsymbol{\xi}) = \sum_{\mathrm{i}} C_{\mathrm{i}} \frac{\partial N_{\mathrm{i}}(\boldsymbol{\xi})}{\partial \xi_j}. \tag{2.33}$$

The departure from classical isogeometric analysis consists in choosing a solution basis $M_\beta(\boldsymbol{\xi})$, which is different from the geometry basis, and looking for the solution as in Eq. (2.31). In order to evaluate derivatives of the solution basis function $M_\beta(\boldsymbol{\xi})$ with respect to variables $\boldsymbol{x}$, the standard chain rule is used, which in two dimensions read:

$$\begin{pmatrix} \frac{\partial M_\beta(\xi,\eta)}{\partial x} \\ \frac{\partial M_\beta(\xi,\eta)}{\partial y} \end{pmatrix} = \begin{pmatrix} \frac{\partial \xi}{\partial x} & \frac{\partial \eta}{\partial x} \\ \frac{\partial \xi}{\partial y} & \frac{\partial \eta}{\partial y} \end{pmatrix} \begin{pmatrix} \frac{\partial M_\beta(\xi,\eta)}{\partial \xi} \\ \frac{\partial M_\beta(\xi,\eta)}{\partial \eta} \end{pmatrix} \tag{2.34}$$

Now let the weak form of the boundary value problem be given by:

$$a(u,v) = l(v), \tag{2.35}$$

the substituting Eq. (2.30) and (2.31) into (2.35) we obtain the linear system of equations given by Eq. (2.24) where the stiffness matrix $\boldsymbol{K}$ and the force vector $\boldsymbol{f}$ are given by:

$$\boldsymbol{K}_{\mathrm{i}j} = a(M_{\mathrm{i}}(\boldsymbol{x}), M_j(\boldsymbol{x})), \qquad \boldsymbol{f}_{\mathrm{i}} = l(M_{\mathrm{i}}(\boldsymbol{x})). \tag{2.36}$$

2.3 Measurement of errors

Before proceeding further it is necessary to define what we mean by error. It is considered to be the difference between the exact solution and the approximate one, given by:

$$\mathbf{e} = u_n - u_a, \tag{2.37}$$

where u_n denotes the numerical solution and u_a the exact or analytical solution.

There are 2 general types of errors and different methods to calculate them. The absolute error and the relative error are given by:

$$\mathbf{e}_{abs} = \|u_n - u_a\|, \tag{2.38}$$

$$\mathbf{e}_{rel} = \frac{\|u_n - u_a\|}{\|u_a\|}, \tag{2.39}$$

where $\| \ \|$ is the norm, which will be explained in the following section.

In this work we will use absolute errors in the L^2-norm to calculate the global error of problems, and for every cell of the field mesh also we will use H^1-norm and two different error indicators.

2.3.1 L^2-Norm

The L^2-norm is the most typical and mostly used to estimate errors. L^2-norm of a function u defined in domain Ω is represented as follows:

$$\|u\|_{L^2(\Omega)} = \left(\int_{\Omega} |u|^2 \mathrm{d}x \right)^{\frac{1}{2}}, \tag{2.40}$$

where $|u|$ denotes the modulus of u.

2.3.2 H^1-Norm

The H^1-norm is defined in the Sobolev space $W^{1,2}(\Omega)$ that could be also denoted as $H^1(\Omega)$. It is a Hilbert space, with an important subspace $H_0^1(\Omega)$ defined to be the closure in $H^1(\Omega)$ of the infinitely differentiable functions compactly supported in Ω. The $H^1(\Omega)$-norm is defined as follows:

$$\|u\|_{H^1(\Omega)} = \left(\int_{\Omega} (|u|^2 + |\nabla u|^2) \mathrm{d}x \right)^{\frac{1}{2}}. \tag{2.41}$$

2.3.3 Error indicators

Residual based

Error indicators are used to select elements that meet specific requirements to be refined. Requirements could be as simple as a constant limit, e.g. an error indicator that selects all values that are over a chosen value $\mathrm{e}_{ind} = a$, where every element to analyze of a vector that are greater than the specific value a is selected.

An important error indicator is proposed in [1] as a strategy for marking the parametric cells to be refined, using GIFT method with PHT-Splines in the field solution and NURBS for the geometry. Suppose $\mathcal{K}$ as the cell on the T-mesh $\mathbb{T}$ of the parametric domain $\mathcal{P}$ for the PHT-Splines solution u_h. The local error indicator $\mathrm{e}_{\mathcal{K}}$ on each parametric cell $\mathcal{K}$ is given by:

$$\mathrm{e}_{\mathcal{K}} = h_{\mathcal{K}} \| f(\boldsymbol{x}) + \Delta u_h(\boldsymbol{x}) \|_{L^2(\mathcal{K})}, \tag{2.42}$$

where $h_{\mathcal{K}}$ is the circumference of the subpatch in the NURBS parameterization $\mathbf{F}(\xi, \eta)$ of Ω with respect to the cell $\mathcal{K}$ in the parametric domain $\mathcal{P}$ and $\mathbf{x} = \mathbf{F}(\xi, \eta) = (x(\xi, \eta), y(\xi, \eta))$.

Hierarchical

The hierarchical error indicator compares the solution of step n and step $n + 1$. Being u_n the numerical solution of the iteration n and u_{n+1} the numerical solution of the iteration $n + 1$, the error marker to select the cells to be refined can be calculated using L^2-norm of the difference by Eq.(2.43) or H^1-norm of the difference by Eq. (2.44) in each $\mathcal{K}$ cell of the domain.

$$\mathbf{e}_{ind} = \| u_{n+1} - u_n \|_{L^2(\mathcal{K})}. \tag{2.43}$$

$$\mathbf{e}_{ind} = \| u_{n+1} - u_n \|_{H^1(\mathcal{K})}. \tag{2.44}$$

2.3.4 Refinement strategy

To select the cells to be refined in addition of the error indicators, a strategy has to be implemented. Using the error indicators, cells are arranged by their errors in descending order, so the cells with the highest errors are selected. In this work, two simple methods are used, selecting the 5% of the cells with the highest errors and the second one selecting the 20% of the cells.

Chapter 3

Methodology and Resources

3.1 Methodology

To achieve the main objectives of this research it is necessary to divide the work in three stages:

The first one is the detailed theorical background review of the principal subjects as Isogeometric Analysis (IGA), Geometry-Independent Field approximaTion (GIFT) and the necessarily functions, such as NURBS and PHT-Splines.

The second stage is the study of the local refinement of the solution. For this purpose, the code used to solve two known problems has to be adapted, so the algorithm can self-adapt to improve the solutions by locally refining the meshes where it's needed, specifically where the algorithm detects that the solutions exhibits high gradients or sharp peaks. This will be verified by the implementation and comparison of three different error measures to determine if they improve the accuracy on the solutions compared to homogeneous refinements, selecting by three different methods the cells to be refined. The first method uses the analytic solution, but the other two methods do not depend on the exact solution. Because many times there is not a known exact solution available, this two methods would be very helpful if they work accurately.

1. Method 1 selects the cells to be refined by using the L^2-norm of the absolute error of the solution (analytical and numerical solution).

2. Method 2 selects the cells to be refined by using the Residual based error indicator, mentioned in Eq. (2.42).

3. Method 3 uses a hierarchical algorithm using L^2-norm and H^1-norm as mentioned in Eq. (2.43) and (2.44).

Finally as the third stage, we study the dependence of the parameterization of the geometry on the solution. To do this we will use two different parameterizations of a quarter annulus geometry, a uniform one and a nonuniform, to compare the solution given by the GIFT method of a known problem, the Poisson's problem.

3.2 Resources

The resources involved in this work are related to the software employed. For the algorithm coding it is necessary a cross platform like Code Blocks, which is a free C++ IDE available for Windows, OS X and Linux operating systems. For the data analysis and all the post-processing, the software Matlab® will be used, which is a not free numerical computing software, available for Windows, OS X and Linux operating systems.

Chapter 4

Definition of The problems

We seek to investigate the performance of the GIFT method, and the improvements we can get exploiting it's advantages. For this work, we will research the advantages of the local refinement using PHT Splines. Analyzing two problems using different methods to select the cells to refine, aiming to decreasing the convergence rate.

For the second part we are using multiple parameterization of a geometry to test the solution in each of them.

4.1 Poisson's problem

The first problem to solve is the Poisson's equation given by Eq. (2.5) in a quarter annulus domain Ω (see Figure 4.1). The control points used to parameterize the geometry are shown in Table 4.1.

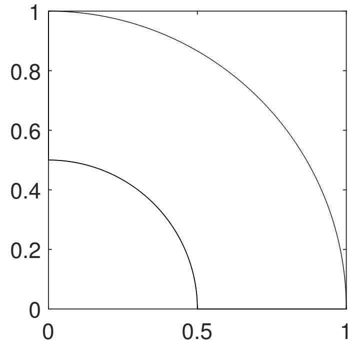

Figure 4.1: Quarter annulus geometry with external radius of 1 and internal radius of 0.5.

Table 4.1: Control points and weights for coarsest parameterization of Poisson's problem.

(i, j)	P^x_{ij}	P^y_{ij}	w_{ij}
(1,1)	0.500	0.000	1.000
(1,2)	0.500	0.500	0.7071
(1,3)	0.000	0.500	1.000
(2,1)	0.750	0.000	1.000
(2,2)	0.750	0.750	0.7071
(2,3)	0.000	0.750	1.000
(3,1)	1.000	0.000	1.000
(3,2)	1.000	1.000	0.7071
(3,3)	0.000	1.000	1.000

We impose zero Dirichlet boundary condition on the entire boundary $\delta\Omega$. We use the solution $u(r, \theta)$ defined by Eq. (4.1), where $r = \sqrt{x^2 + y^2}$ and $\theta = \arctan(y/x)$.

$$u(r, \theta) = \frac{(r-1)(r-\frac{1}{2})\theta(\theta - \frac{\pi}{2})}{\exp(100(r\cos(\theta) - 0.5))} \tag{4.1}$$

To formulate the numerical solution we use GIFT method to solve Eq. (2.24). To do this we define the geometry using NURBS as in Eq. (2.30) and the solution field using PHT-Splines as in Eq. (2.31) aiming the local refinement.

For NURBS geometry we need the control points given and two knot vectors: $\Xi_\xi = [0, 0, 0, 1, 1, 1]$ for ξ direction and $\Xi_\eta = [0, 0, 0, 1, 1, 1]$ for η direction. As one can infer, the degrees of the splines are 3 for each direction.

For the stiffness matrix K, using the weak form of Poisson's equation given by Eq. (2.7) with $u(x, y)$ defined by Eq. (2.31) and the weight function, we can find the stiffness matrix as in Eq. (2.26)

The force vector is defined by:

$$f(r, \theta) = \frac{-(4 + A + B)}{C}, \tag{4.2}$$

where A, B and C are defined as follows:

$$A = r(-12 + 8r + \theta(2\theta - \pi)(-3 + 16r(613 + 25r(-72 + r(97 + 50r(-3 + 2r)))))),$$

$$B = 200r(\theta r(-2\theta + \pi)((3 + 4r(49 + 50r(-3 + 2r)))\cos(\theta) - r(3 + 4r(24 + 25r(-3 + 2r)))\cos(2\theta)) - (-4\theta + \pi)(-1 + r)(-1 + 2r)(-1 + 2r\cos(\theta))\sin(\theta),$$

$$C = 4exp(25(1 - 2r\cos(\theta))^2)r^2,$$

where $r = \sqrt{x^2 + y^2}$ and $\theta = \arctan(y/x)$.

In Figure 4.2, the analytical solution of the problem given by Eq. (4.1) is plotted. One can see the gradient in the middle of the domain, where local refinement is expected.

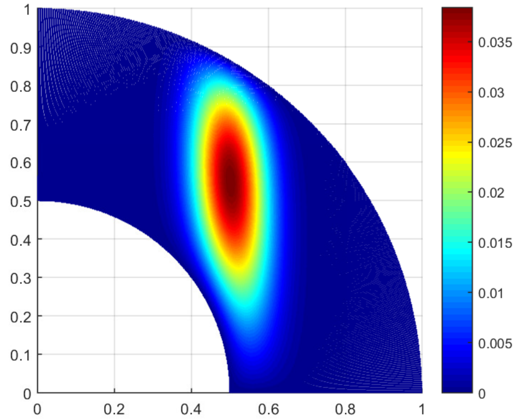

Figure 4.2: Analytical solution for Poisson's equation in a quarter annulus geometry domain.

4.2 Two Heat source problem

The second problem to solve has 2 peaks in its domain. We use this problem to test the adaptability of the solution of the in-house code in presence of multiple focuses of interests.

The domain Ω used for this problem is a square which control points used to parameterize the geometry are shown in Table 4.2.

Table 4.2: Control points and weights for coarsest parameterization of Two Heat source problem.

(i, j)	$P^x_{\mathrm{i}j}$	$P^y_{\mathrm{i}j}$	$w_{\mathrm{i}j}$
(1,1)	0.000	0.000	1.000
(1,2)	0.000	1.000	1.000
(2,1)	1.000	0.000	1.000
(2,2)	1.000	1.000	1.000

As for the first problem we impose zero Dirichlet boundary condition for entire boundary $\delta\Omega$ and the exact solution is given by Eq. (4.3):

$$u(x, y) = (x^2 - x)(y^2 - y) \left(\mathrm{e}^{-100|(x,y)-(0.8,0.05)|^2} + \mathrm{e}^{-100|(x,y)-(0.8,0.95)|^2}\right), \tag{4.3}$$

To formulate the numerical solution we use GIFT method to solve Eq. (2.24). To do this we define the geometry using NURBS as in Eq. (2.30) and the solution field using PHT-Splines as in Eq. (2.31) aiming the local refinement.

For geometry and solution field we need the control points given and two knot vectors: $\Xi_\xi = [0, 0, 1, 1]$ for ξ direction and $\Xi_\eta = [0, 0, 1, 1]$ for η direction. As one can infer, the degrees of the splines are 3 for each direction.

The force vector of this problem is obtained by solving the following equation:

$$f(x, y) = -\left(\frac{\partial^2 u}{\partial x} + \frac{\partial^2 u}{\partial y}\right), \tag{4.4}$$

where u is the exact solution of the problem given by Eq. (4.3).

In Figure 4.2, the analytical solution, given by Eq. (4.1), is plotted. One can see the two gradients next to the boundaries of the domain.

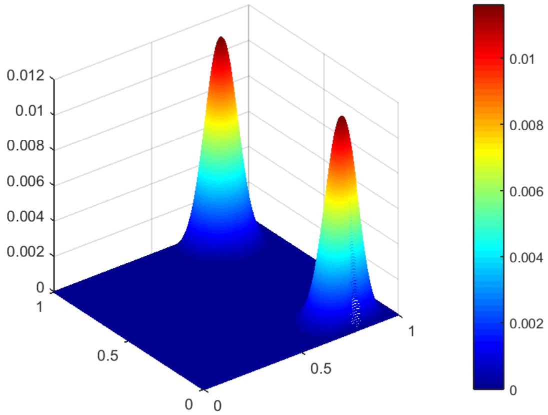

Figure 4.3: Analytical solution for Two Heat source problem, given by Eq. (4.3), in a square geometry domain.

4.3 Geometry parameterization

Using the Poisson's problem given by Eq. (2.5), two different parameterizations of the same geometry are tested, to study the dependence of the solution on the parameterization. The geometry is a quarter annulus, the parameterizations used are shown in the Figure 4.4, where the control points are listed in the Tables 4.3 and 4.4.

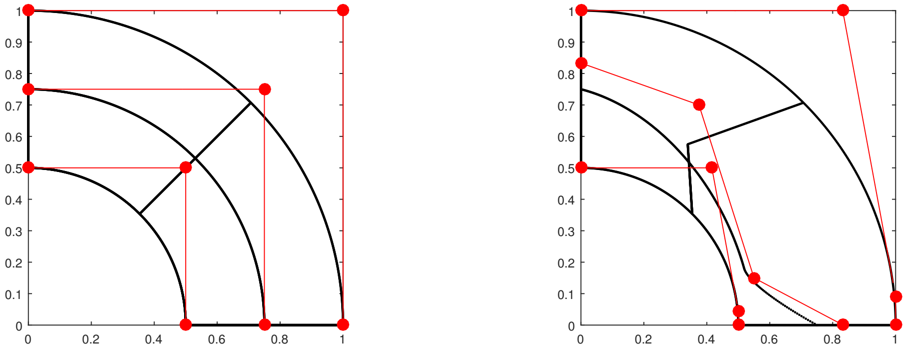

Figure 4.4: In the left, the uniform parameterization of a quarter annulus geometry domain given by control points listed in Table 4.3. In the right, the non-regular parameterization of a quarter annulus geometry domain given by control points listed in Table 4.4.

Table 4.3: Control points and weights for uniform parameterization of a quarter annulus geometry domain.

(i, j)	$P^x_{\mathrm{i}j}$	$P^y_{\mathrm{i}j}$	$w_{\mathrm{i}j}$
(1,1)	0.500	0.000	1.000
(1,2)	0.500	0.500	0.7071
(1,3)	0.000	0.500	1.000
(2,1)	0.750	0.000	1.000
(2,2)	0.750	0.750	0.7071
(2,3)	0.000	0.750	1.000
(3,1)	1.000	0.000	1.000
(3,2)	1.000	1.000	0.7071
(3,3)	0.000	1.000	1.000

Table 4.4: Control points and weights for a non-uniform parameterization of a quarter annulus geometry domain.

(i, j)	$P^x_{\mathrm{i}j}$	$P^y_{\mathrm{i}j}$	$w_{\mathrm{i}j}$
(1,1)	0.500	0.000	1.000
(1,2)	0.500	0.04585	0.9634
(1,3)	0.416	0.500	0.7437
(1,4)	0.000	0.500	1.000
(2,1)	0.83335	0.000	1.000
(2,2)	0.550	0.150	0.9634
(2,3)	0.375	0.700	0.7437
(2,4)	0.000	0.83335	1.000
(3,1)	1.000	0.000	1.000
(3,2)	1.000	0.09175	0.9634
(3,3)	0.83195	1.000	0.7437
(3,4)	0.000	1.000	1.000

Chapter 5

Results

This chapter is divided in two sections, First the results of the Study of the local refinement will be shown for Poisson's and Two Heat source problem using each one of the different error measures as seen in the methodology, using convergence curves and error plots to analyze the results. The second section is dedicated for the results of the multiple patch test.

5.1 Poisson's problem

5.1.1 Method 1: L^2-norm error indicator

This method is divided in two parts: The first results shown are obtained using L^2-norm of the absolute error of the cells with refinement strategy of 20% and 5%. For the second part, the results are obtained using a defined error limit for each cell using as well L^2-norm . The refinement strategy is selecting all cells that overcomes this limit, making a final result where all cells error is below the limit. The used values are 10^{-6}, 10^{-7} and 10^{-8}.

L^2-norm error indicator: From the results, in Figures 5.1 and 5.3 we can see the convergence curve of the adaptive refinement presents a sinusoidal behavior for lower degrees of freedom. In Figure 5.1 after 10^3 degrees of freedom and in Figure 5.3 after 3×10^2 degrees of freedom, the convergence curve shows a similar steady decreased slope compared to the homogeneous refinement convergence curve, this represents that the greater the degree of freedom, the better accuracy shows this method against the homogeneous refinement.

In Figure 5.2 we can see that the refined cells are grouped in the middle of the geometric domain with a tendency on the upper edge. Which is very close where the peak of the solution is located, where we expect to see the local refinement. The sinusoidal behavior of the convergence curve of both refinement strategies can be explained because at lower degrees of freedom the mesh has to be refined almost in all the domain, this is why the convergence curve of the adaptive refinement is close to the homogeneous refinement convergence curve until it shows a more local adaptive process where the refinement occurs most of all in the middle of the domain where the exact solution presents greater slopes and so we expect greater errors in those cells.

The refined mesh in Figures 5.2 and 5.4 for the two refinement strategy looks very similar, both follows the same pattern where the refined cells are close to the analytic solution peak but there is a little difference that makes one of the refinement strategies better than the other. In Figure 5.2 in the three cases shown even though most of the refined cells are in the center there are a few ones that break this pattern. In Figure 5.4 for 5% refinement strategy there are even fewer cells outside this pattern. This is why the 5% refinement strategy shows a better convergence curve than 20%. When selecting the cells to be refined using 20% too much cells are selected, inducing some cells that do not have a big error to be selected causing a greater global error. When using 5% refinement strategy we can see in Figure 5.4 that the refined cells are locally adaptive to the solution peak in a better way than using 20%.

L^2-norm limit error: The results obtained using a limit value for each cell is presents similar behavior to the homogeneous convergence curve. We can see in Figure 5.5 the adaptive convergence curves follow the same path until a few degrees of freedom before reaching the limit where every cell error is below the defined limit. This is caused by the strategy used, the defined limits are low enough to select almost every cell in the domain to be refined until a few degrees of freedom before reaching the limit as said before. In Figure 5.6 we can see the refinement process and error plot when using the error value limit of 10^{-8}, here we can observe that in each iteration almost all the domain is refined imitating the homogeneous refinement until less cells are selected to be refined then we can see an improvement in the convergence curve when the refinement starts being more local.

L^2-norm error indicator 20%

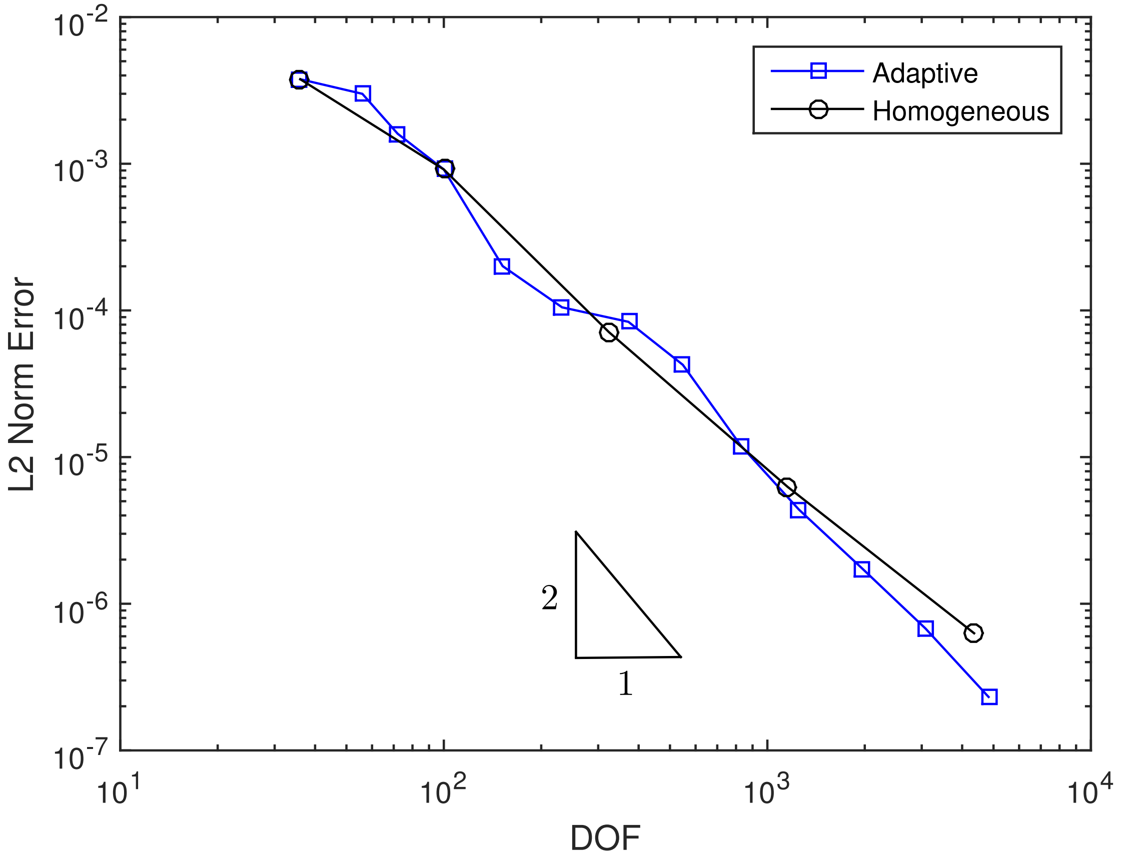

Figure 5.1: Comparison of Homogeneous refinement and Adaptive local refinement for the Poisson's solution by Method 1, L^2-*norm error indicator*, using 20% refinement strategy.

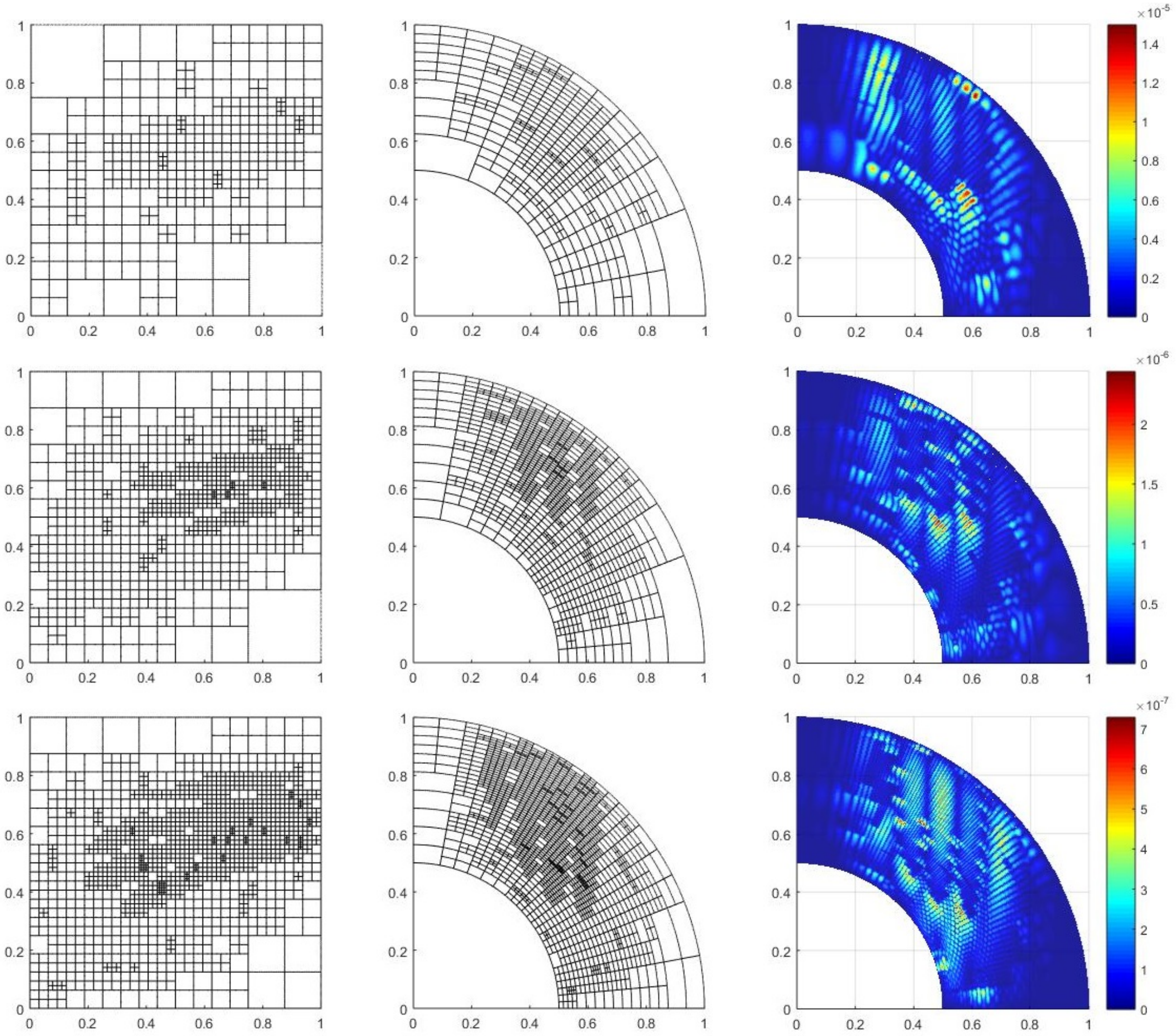

Figure 5.2: In 3 rows, from left to right: Parametric mesh, Geometric mesh and error plot of different degrees of freedom for self-adaptive local refinement for the Poisson's solution by Method 1, L^2-*norm error indicator*, using 20% refinement strategy.

L^2-norm error indicator 5%

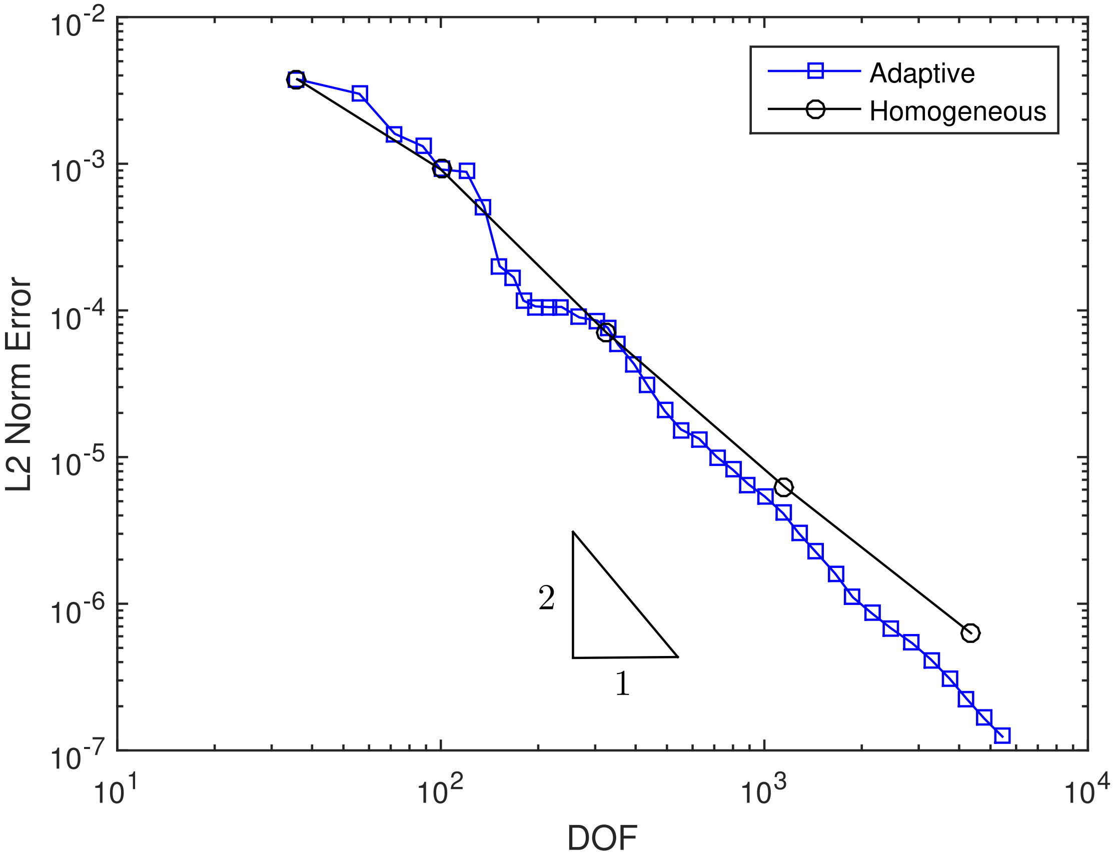

Figure 5.3: Comparison of Homogeneous refinement and Adaptive local refinement for the Poisson's solution by Method 1, L^2-*norm error indicator*, using 5% refinement strategy.

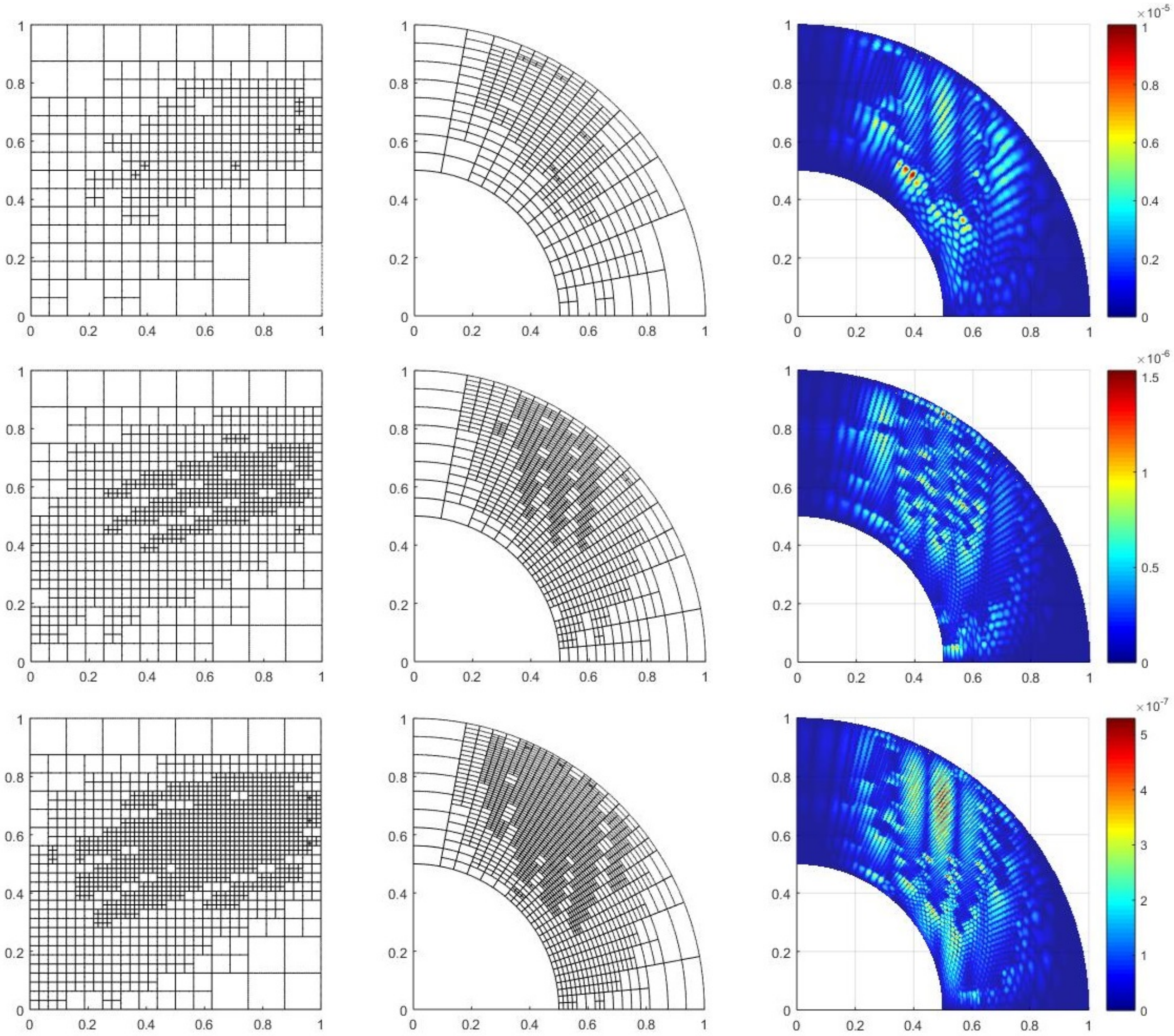

Figure 5.4: In 3 rows, from left to right: Parametric mesh, Geometric mesh and error plot of different degrees of freedom for self-adaptive local refinement for the Poisson's solution by Method 1, L^2-*norm error indicator*, using 5% refinement strategy.

L^2-norm error limit

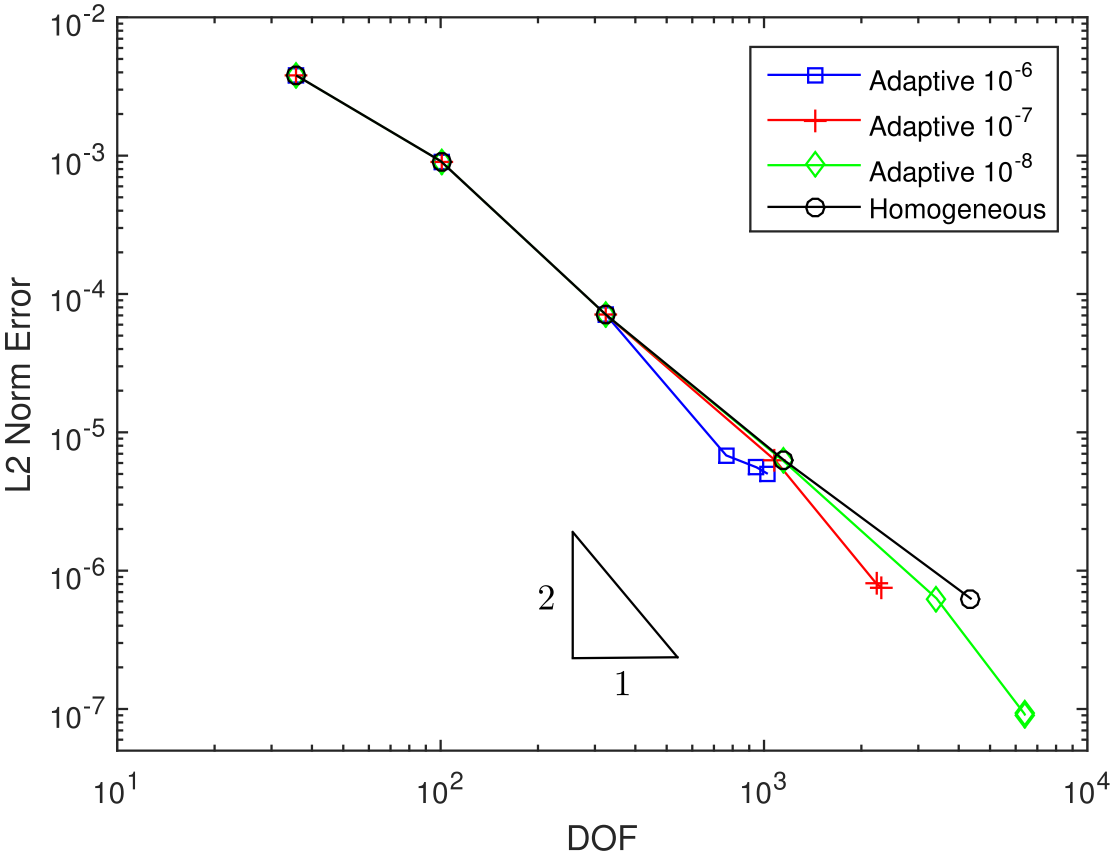

Figure 5.5: Comparison of Homogeneous refinement and Adaptive local refinement for the Poisson's solution by Method 1, L^2-*norm error limit*, using limits of 10^{-6}, 10^{-7} and 10^{-8}.

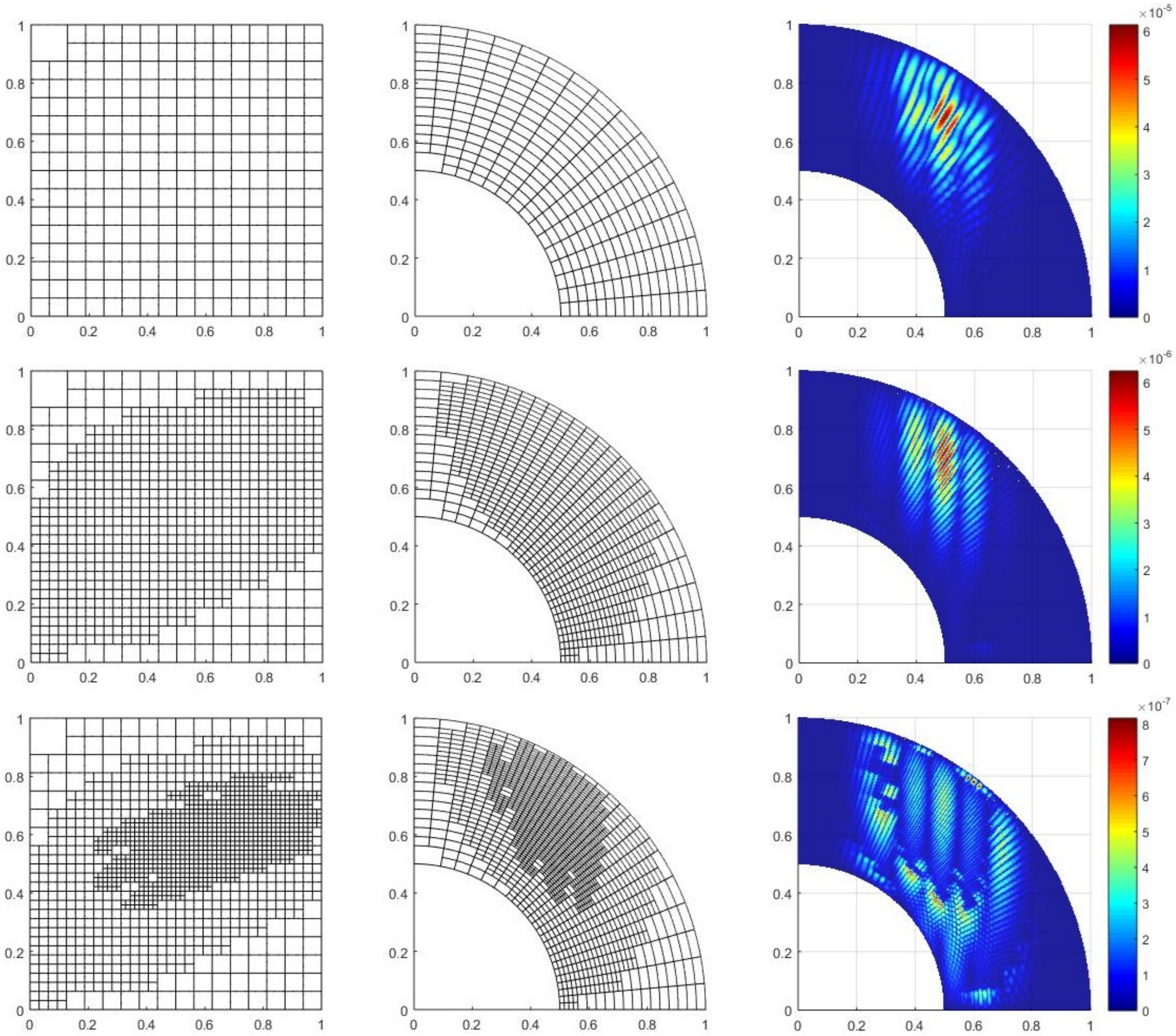

Figure 5.6: In 3 rows, from left to right: Parametric mesh, Geometric mesh and error plot of different degrees of freedom for self-adaptive local refinement for the Poisson's solution by Method 1, L^2-*norm error limit*, using limit of 10^{-8}.

5.1.2 Method 2: Residual based error indicator

The following results are obtained using the residual based error indicator, given in Eq. (2.37), over the Poisson's solution and using the refinement strategy of 20% and 5%.

Residual based error indicator: From the results, comparing convergence curves of 20% and 5% refinement strategy, in Figures (5.7) and (5.9), we can see both behave similar with a slightly difference right after 10^2 degrees of freedom, where both get close to the homogeneous convergence curve For 20% refinement strategy the error decreases compared to the homogeneous curve but for 5% refinement strategy the error has a little increase before decreasing and behaving like 20% refinement strategy. This could be because when using 5% refinement strategy, less cells are selected to be refined on each iterations, making degrees of freedom increase slowly, this shows more errors for different degrees of freedom that are not shown in Figure 5.7.

Like in Method 1, after a threshold close to 3×10^2 degrees of freedom, the slope of adaptive local refinement convergence curve increases passing below the homogeneous curve, showing better results with lower errors for higher degrees of freedom.

Figures 5.8 and 5.10 show the refinement process for several degrees of freedom, we can see that for both strategies the solution behaves very similar. In both cases the refinement is centered in the middle of the geometry domain very similar to Method 1, with the difference that for this Method the refined cells centered in the middle also show an oval geometry more similar to the analytical problem.

A great difference of this method is that the analytical solution of the problem is no needed. Making this method a great option to treat problems with unknown analytic solution.

Residual based error indicator 20%

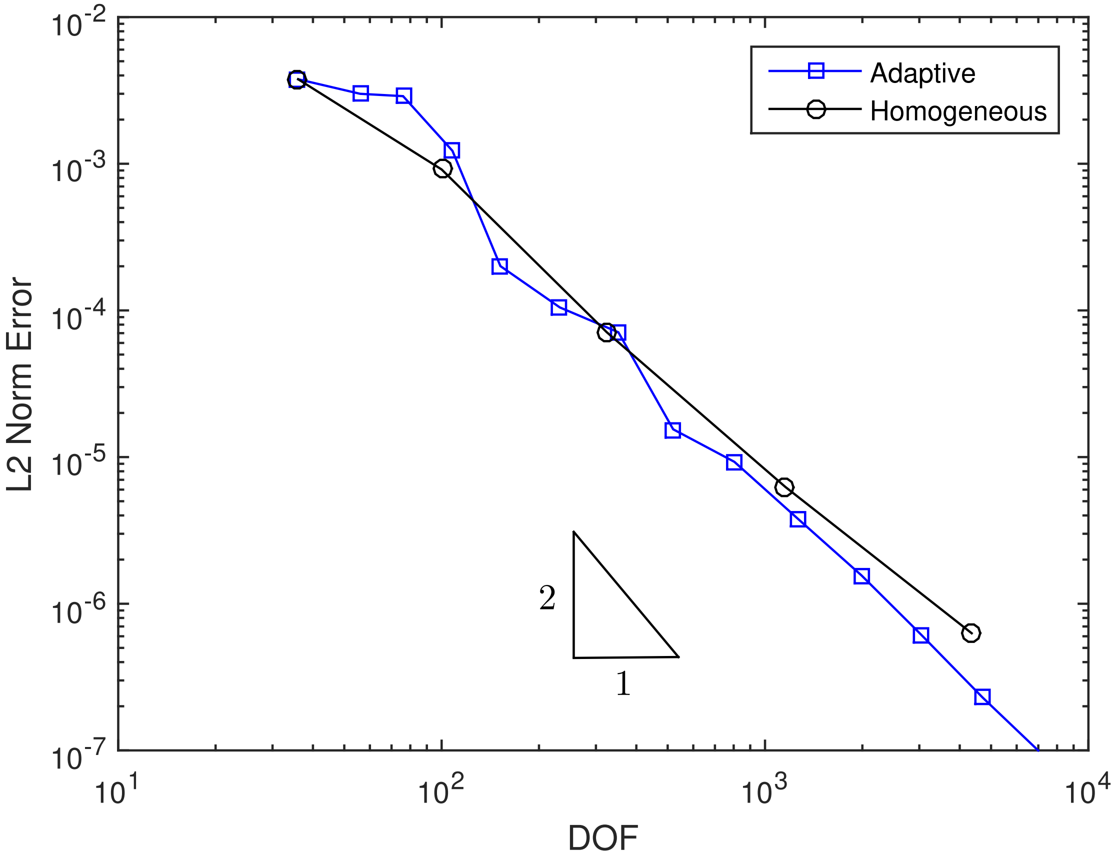

Figure 5.7: Comparison of Homogeneous refinement and Adaptive local refinement for the Poisson's solution by Method 2, *Residual based error indicator*, using 20% refinement strategy.

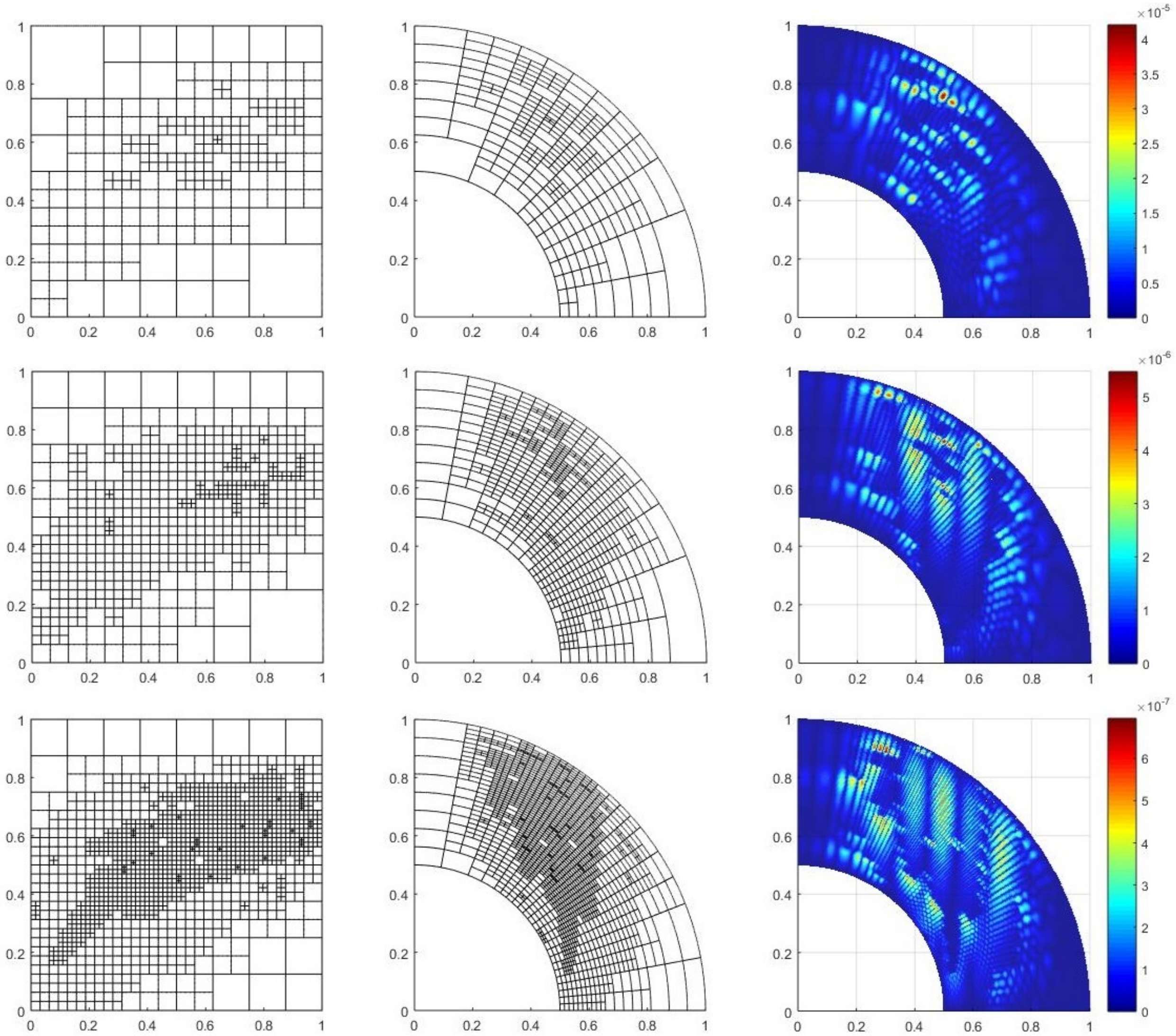

Figure 5.8: In 3 rows, from left to right: Parametric mesh, Geometric mesh and error plot of different degrees of freedom for self-adaptive local refinement for the Poisson's solution by Method 2, *Residual based error indicator*, using 20% refinement strategy.

Residual based error indicator 5%

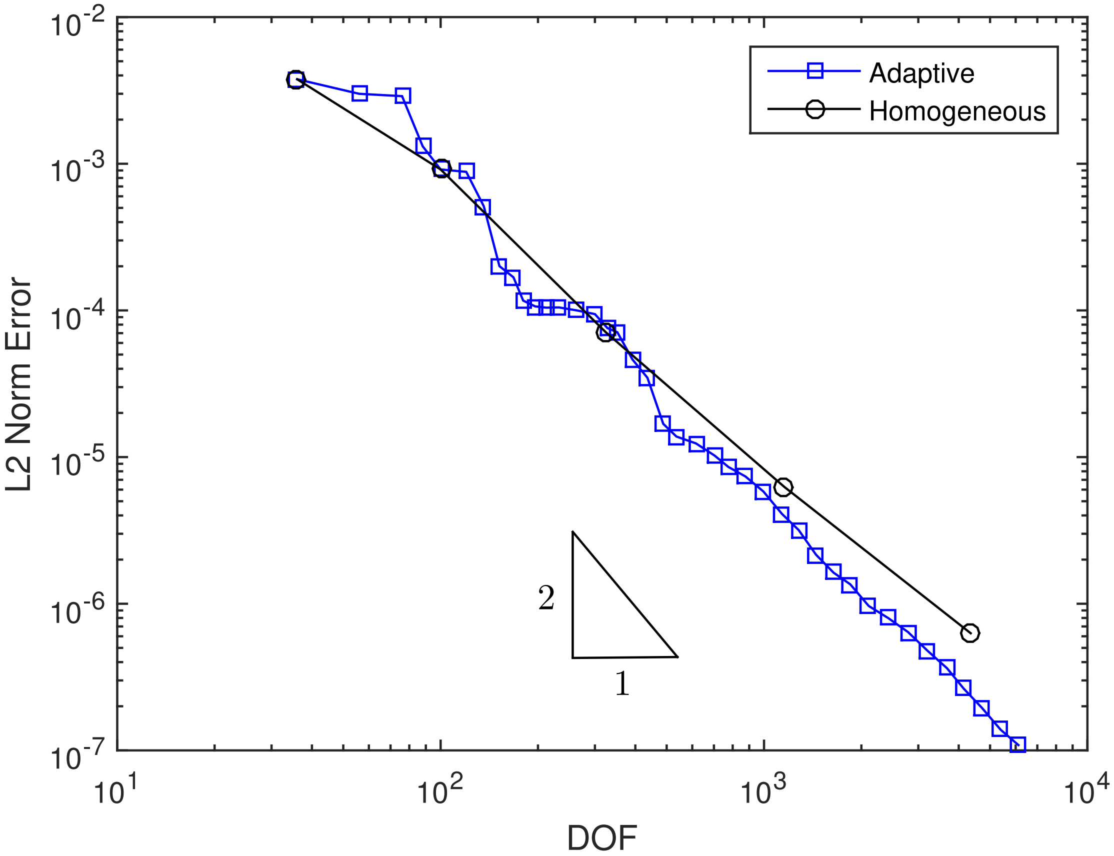

Figure 5.9: Comparison of Homogeneous refinement and Adaptive local refinement for the Poisson's solution by Method 2, *Residual based error indicator*, using 5% refinement strategy.

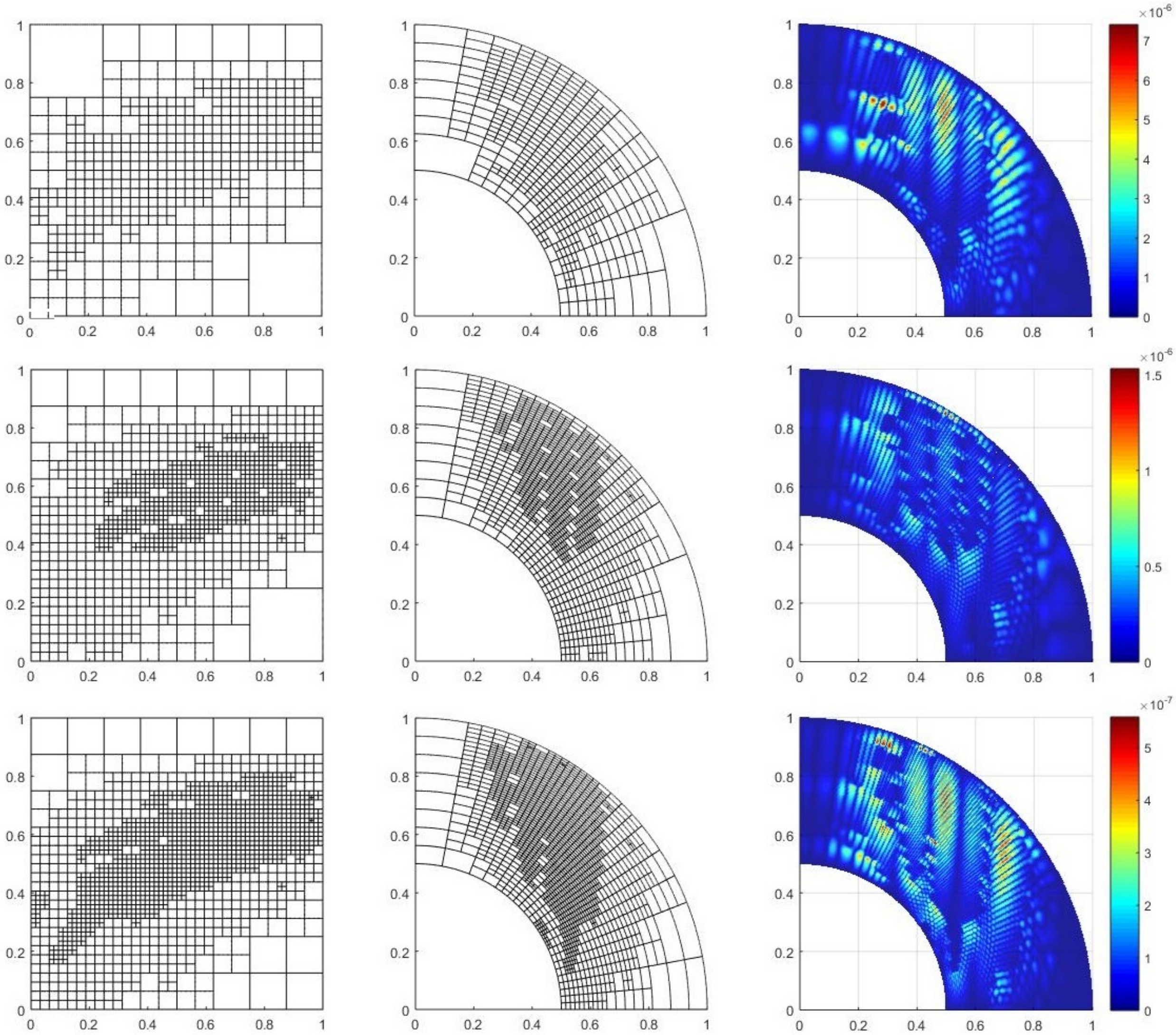

Figure 5.10: In 3 rows, from left to right: Parametric mesh, Geometric mesh and error plot of different degrees of freedom for self-adaptive local refinement for the Poisson's solution by Method 2, *Residual based error indicator*, using 5% refinement strategy.

5.1.3 Method 3: Hierarchical Adaptivity error indicator

This method is divided in two parts. The first results are obtained using the hierarchical error indicator with L^2-norm given by Eq. (2.43) using refinement strategy of 20% and 5%. The second part, the hierarchical error indicator is used with H^1-norm given by Eq. (2.44) and the refinement strategy of 20% and 5%.

Hierarchical Adaptivity L^2-norm: For both refinement strategies of 20% and 5% the convergence rates are worse than the homogeneous refinement. In Figure 5.11 we can see a behavior similar to a stair with three notorious steps. First the solution stays in a constant error while refining some cells and increasing the degrees of freedom until there is a big drop in the error measured with a very high slope and then this is repeated. This may be because the hierarchical method after refining a cell when it compares the previous solution to the new one there is a big difference, a big new error, even though the solution could get better, more accurate, as the method just compares with itself when it refine the cells, it can not see the difference. This causes the algorithm to enter in a 'loop', refining the same place over and over until the error is so small that this loop relocates in other place. This explains the constant error in the convergence curve and the behavior of the refinement process seen in Figures 5.12 and 5.14 where a scattered refinement is shown. For Figure 5.3 when using the 5% refinement strategy makes the same stair behavior, and the scattered behavior is even more evident. This happens because with the 5% strategy refinement when less cells are selected, the refinement process focuses in one place, as previously seen, and it takes more time to lower the error to move on to other place, this is why the convergence curve in Figure 5.13 shows more results with a constant value than Figure 5.11.

For both strategy refinement in Figures 5.12 and 5.14 in the right column where the error plot is shown, we can see that the algorithm does not follow the analytical solution, we see in both cases that the refinement process do not appear where the highest errors shown by the error plot are. This also makes the overall error of the solution to improve very slowly as shown in the convergence curves or even not improve.

Hierarchical Adaptivity H^1-norm: H^1-norm, unlike L^2-norm, involves the derivatives of the function (in this case the solution), this is the only difference between this two methods. Just like with L^2-norm, the convergence curves for both refinement strategies in Figures 5.11 and 5.13 show a behavior similar to a stair. And Figures 5.12 and 5.14 also show an even more scattered pattern in the refinement process.

The results obtained using H^1-norm are very similar to the ones obtained using L^2-norm, using the hierarchical algorithm for both cases makes the solution very similar, but using H^1-norm gives slightly better results, more like the homogeneous refinement. This happens because it takes in consideration the derivatives of the function which are the gradients of the solution and helps to select the cells where the peaks of the problems are located. Even though is not a good indicator when the exact solution is not known.

Hierarchical Adaptivity L^2-norm error indicator 20%

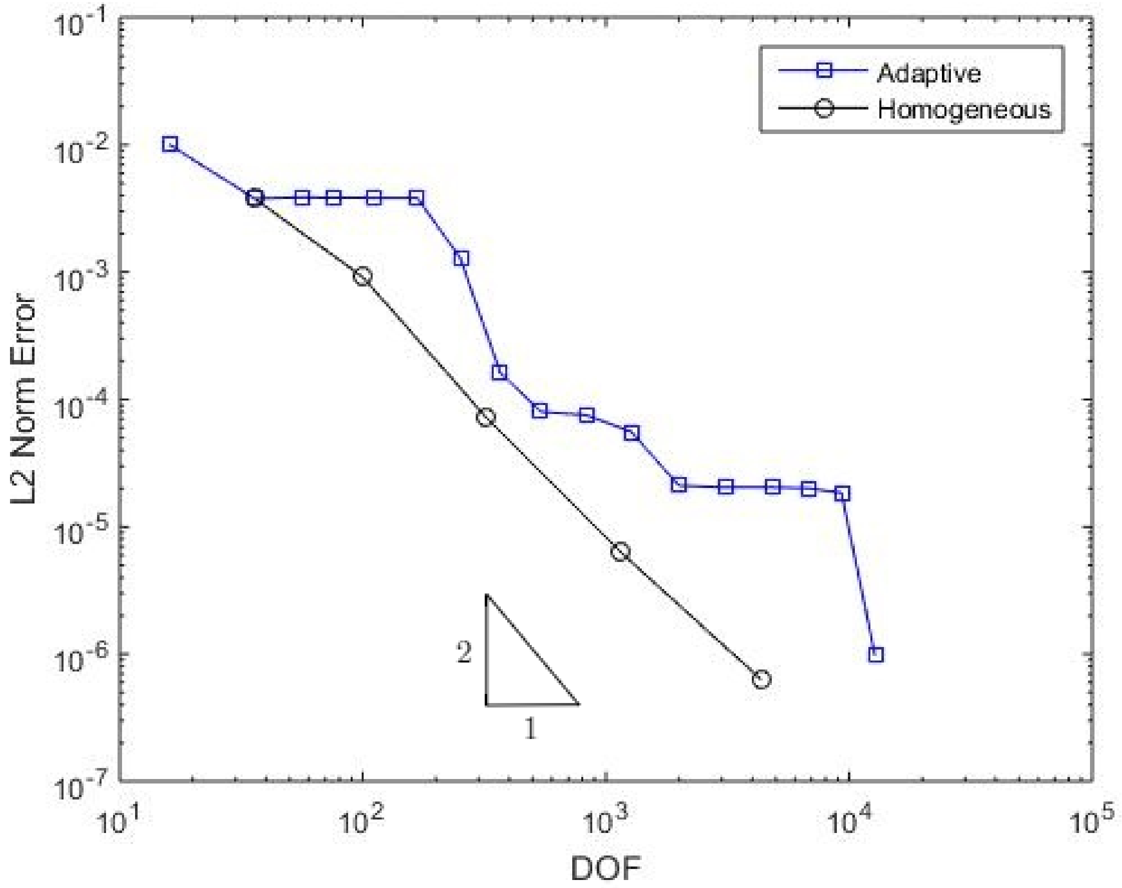

Figure 5.11: Comparison of Homogeneous refinement and Adaptive local refinement for the Poisson's solution by Method 3, *Hierarchical adaptivity L^2-norm error indicator*, using 20% refinement strategy.

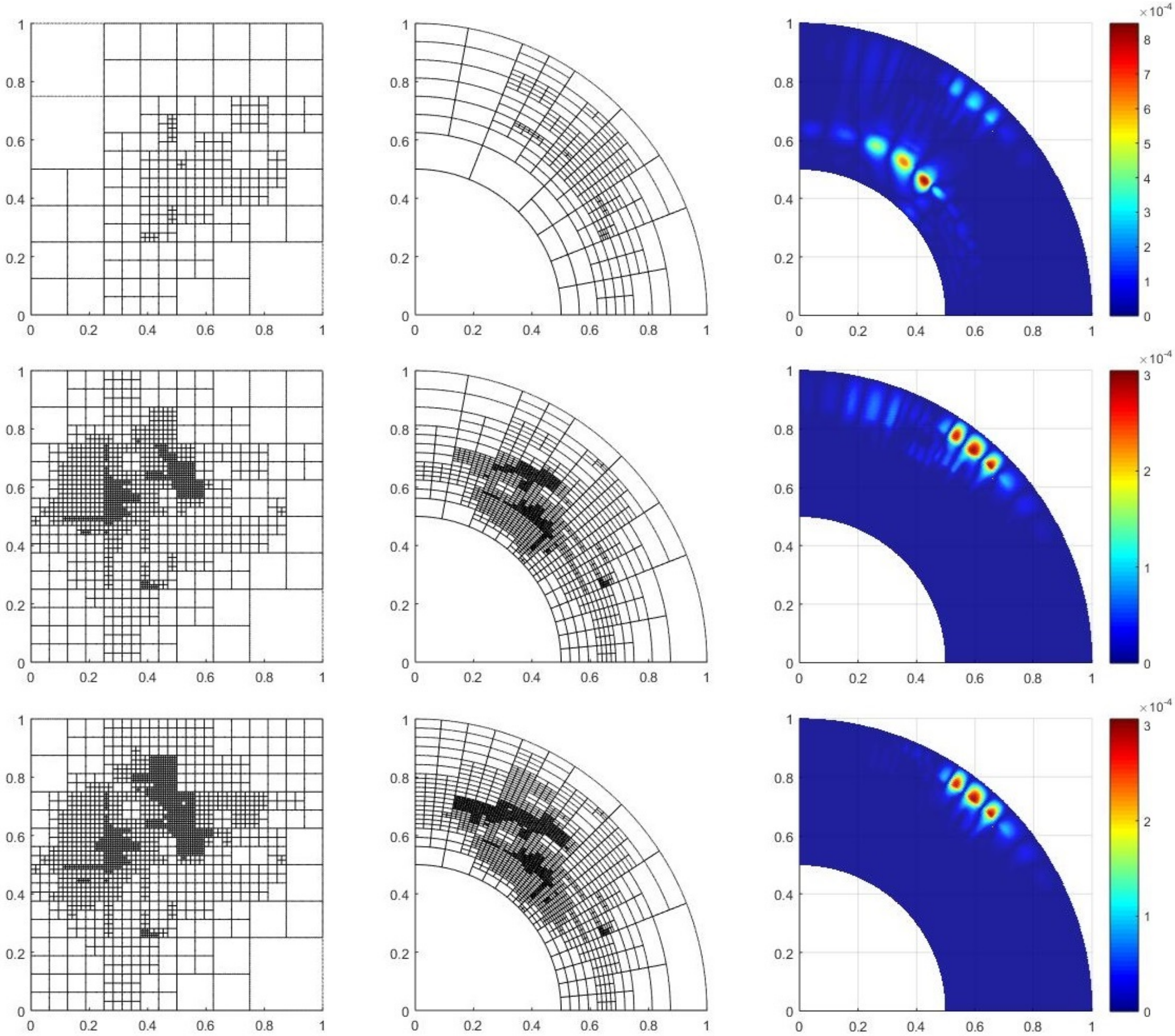

Figure 5.12: In 3 rows, from left to right: Parametric mesh, Geometric mesh and error plot of different degrees of freedom for self-adaptive local refinement for the Poisson's solution by Method 3, *Hierarchical adaptivity L^2-norm error indicator*, using 20% refinement strategy.

Hierarchical Adaptivity L^2-norm error indicator 5%

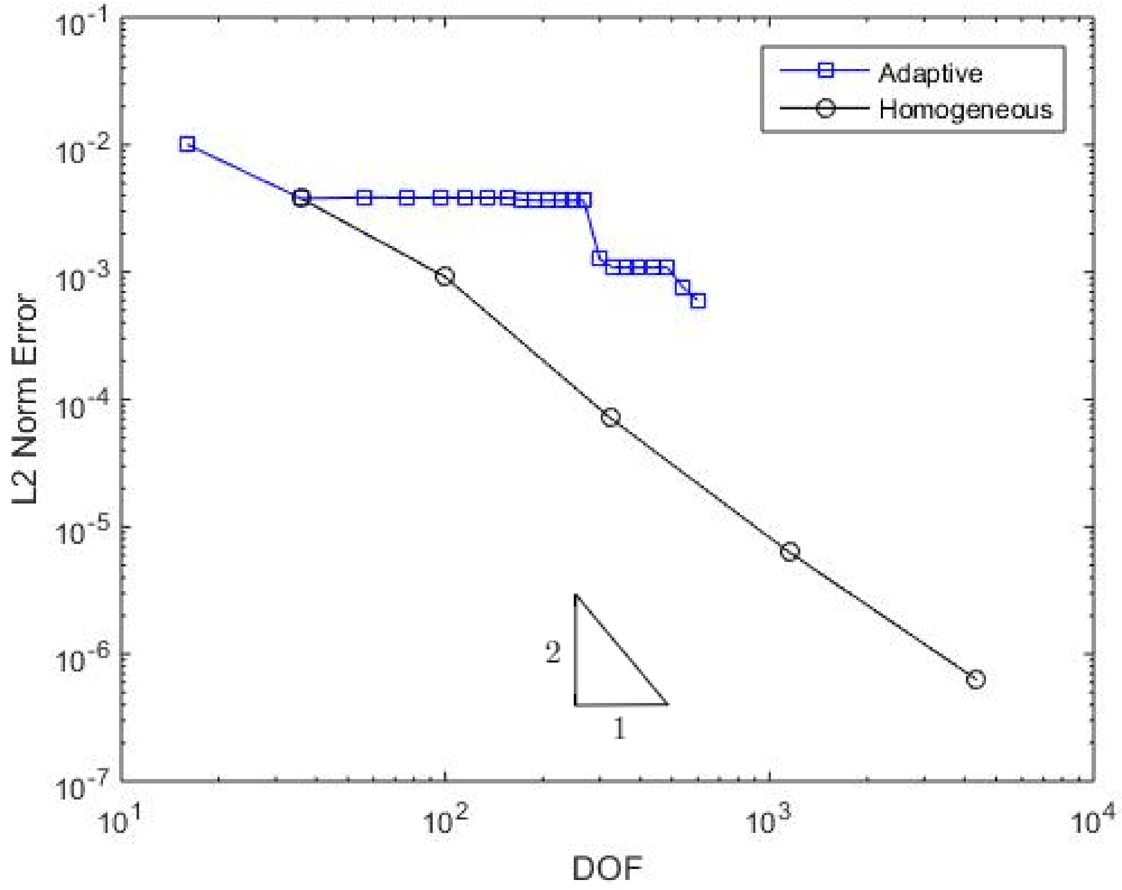

Figure 5.13: Comparison of Homogeneous refinement and Adaptive local refinement for the Poisson's solution by Method 3, *Hierarchical adaptivity L^2-norm error indicator*, using 5% refinement strategy.

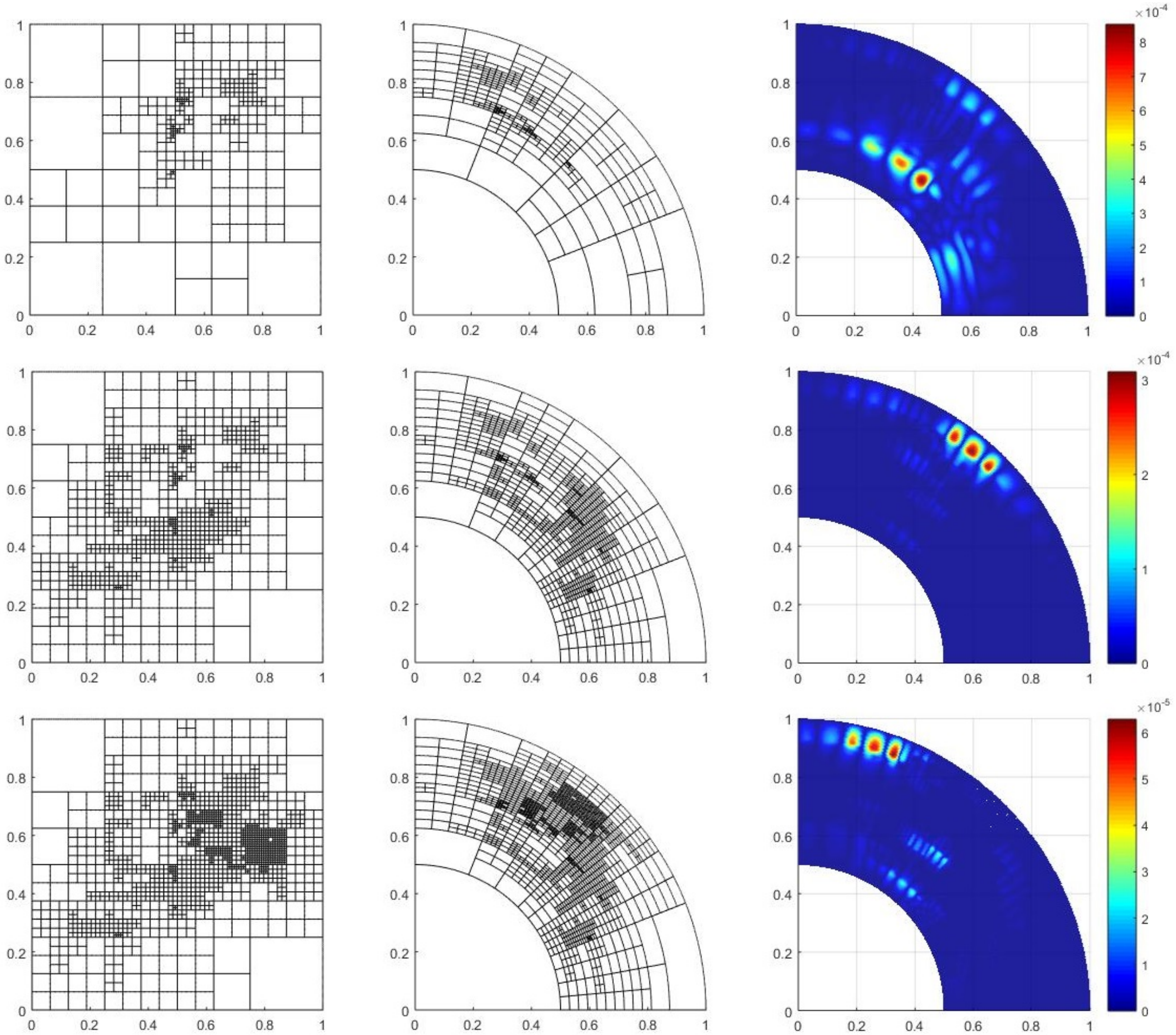

Figure 5.14: In 3 rows, from left to right: Parametric mesh, Geometric mesh and error plot of different degrees of freedom for self-adaptive local refinement for the Poisson's solution by Method 3, *Hierarchical adaptivityL^2-norm error indicator*, using 5% refinement strategy.

Hierarchical Adaptivity H^1-norm error indicator 20%

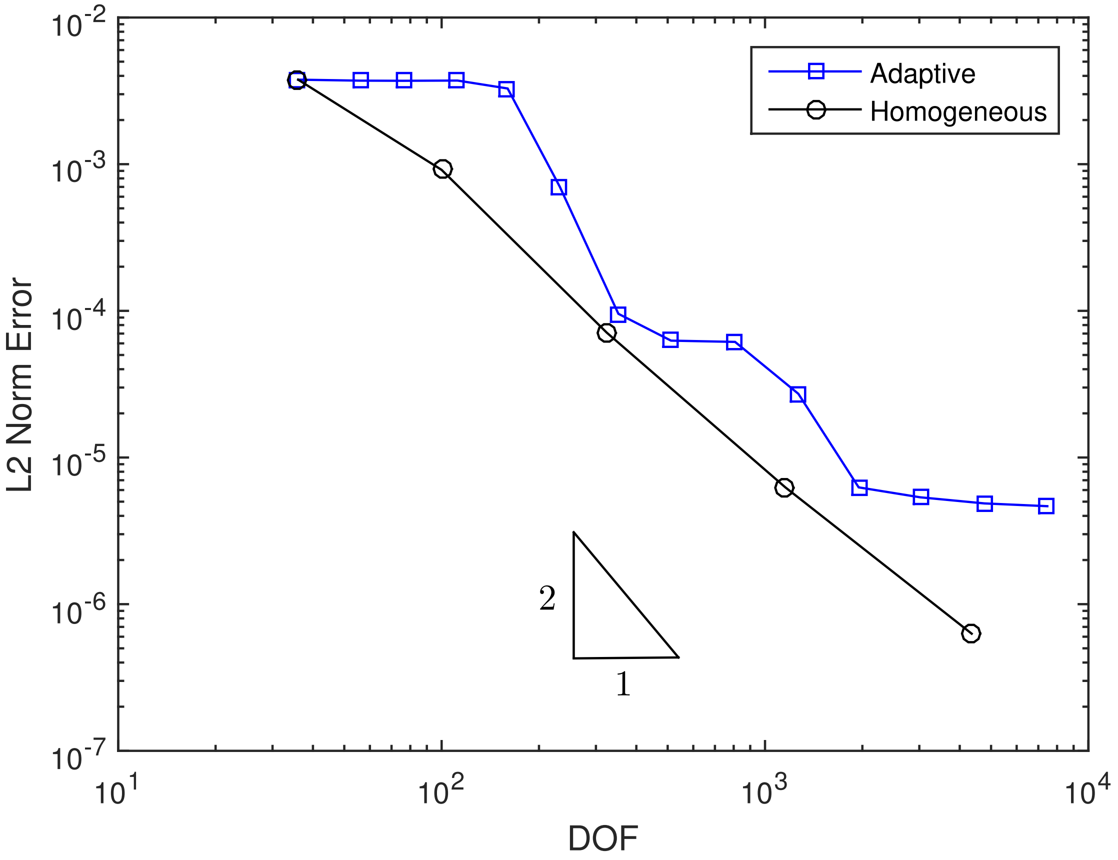

Figure 5.15: Comparison of Homogeneous refinement and Adaptive local refinement for the Poisson's solution by Method 3, *Hierarchical adaptivity H^1-norm error indicator*, using 20% refinement strategy.

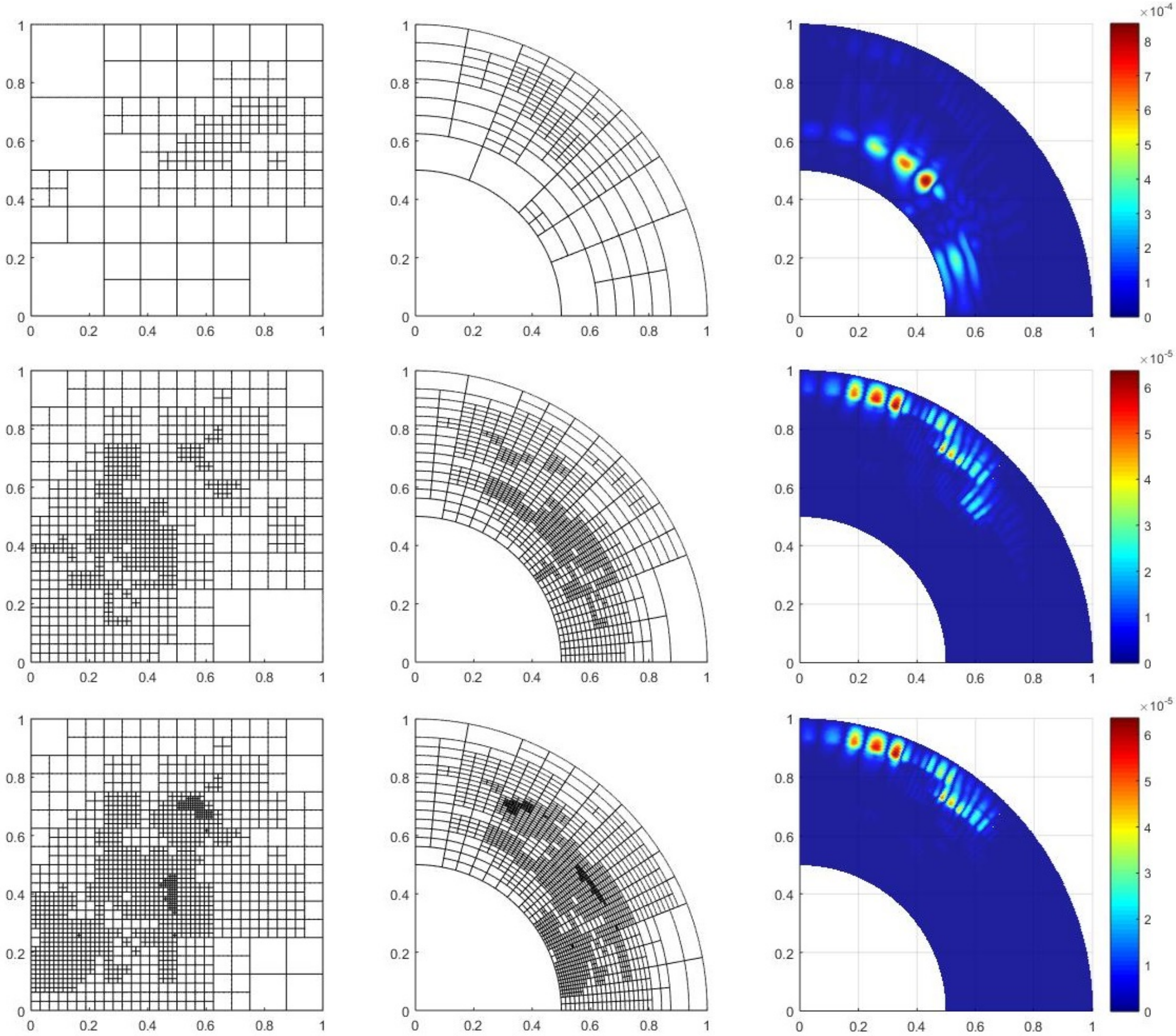

Figure 5.16: In 3 rows, from left to right: Parametric mesh, Geometric mesh and error plot of different degrees of freedom for self-adaptive local refinement for the Poisson's solution by Method 3, *Hierarchical adaptivity H^1-norm error indicator*, using 20% refinement strategy.

Hierarchical Adaptivity H^1-norm error indicator 5%

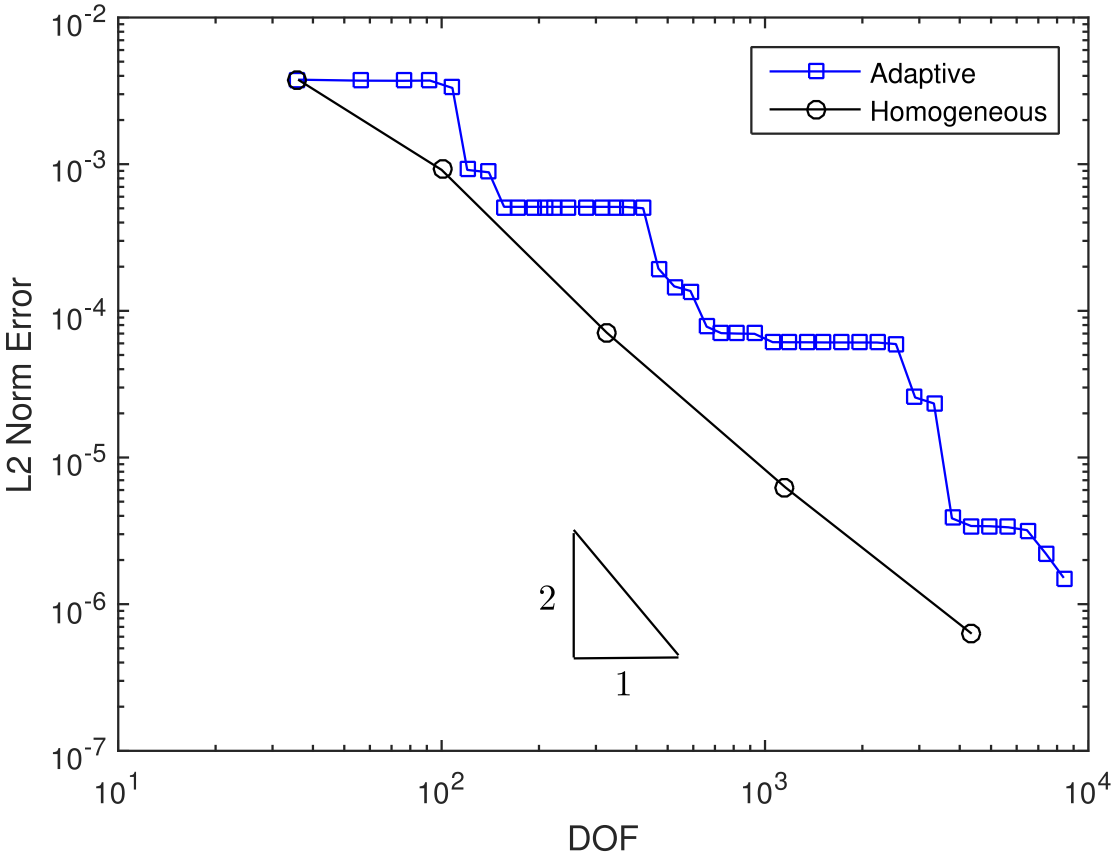

Figure 5.17: Comparison of Homogeneous refinement and Adaptive local refinement for the Poisson's solution by Method 3, *Hierarchical adaptivity H^1-norm error indicator*, using 5% refinement strategy.

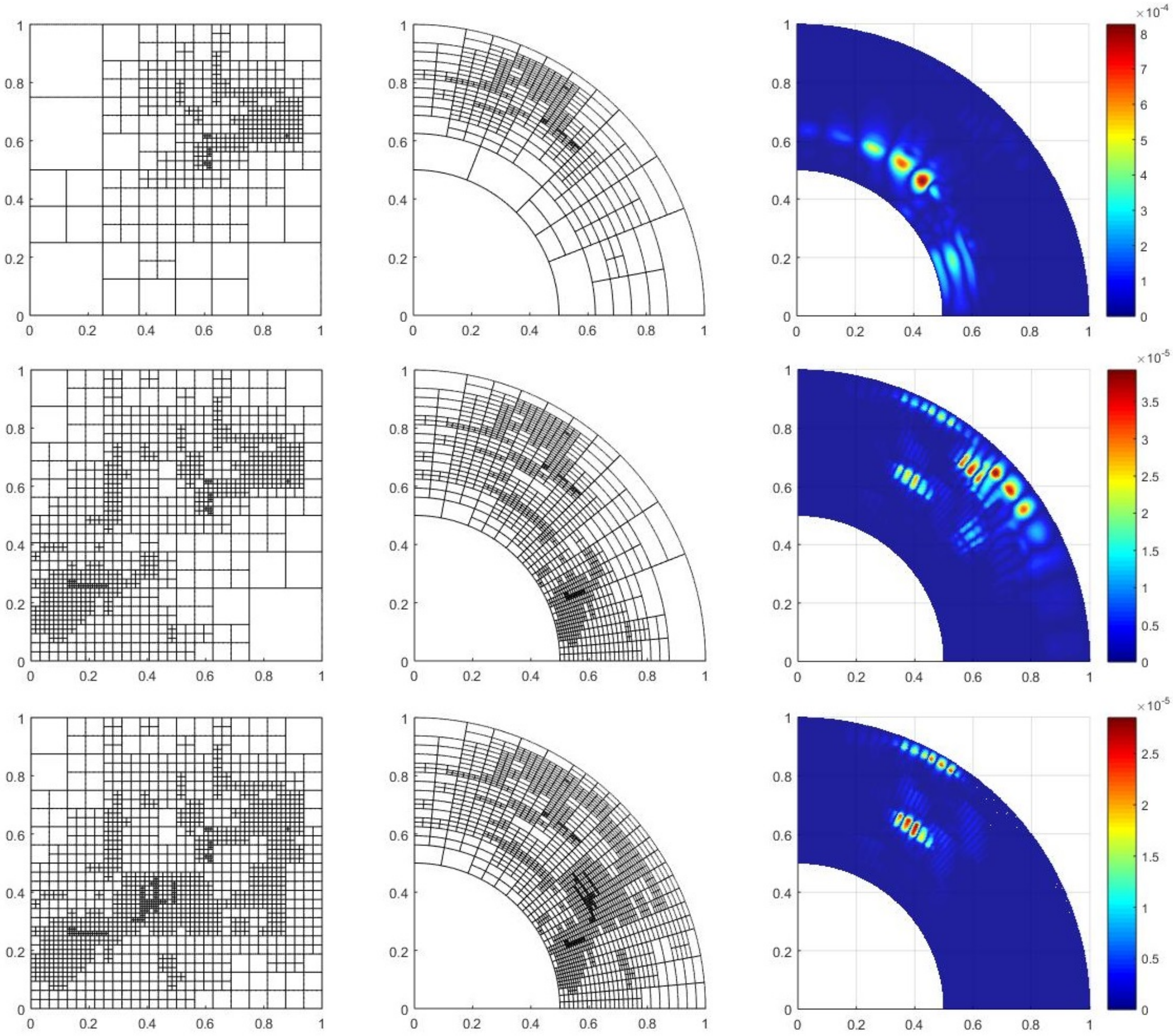

Figure 5.18: In 3 rows, from left to right: Parametric mesh, Geometric mesh and error plot of different degrees of freedom for self-adaptive local refinement for the Poisson's solution by Method 3, *Hierarchical adaptivity H^1-norm error indicator*, using 5% refinement strategy.

5.2 Two Heat source problem

5.2.1 Method 1: L^2-norm error indicator

This method is divided in two parts: The first results shown are obtained using L^2-norm of the absolute error of the cells with refinement strategy of 20% and 5%. For the second part, the results are obtained using a defined error limit for each cell using as well L^2-norm . The refinement strategy is selecting all cells that overcomes this limit, making a final result where all cells error is below the limit. The used values are 10^-6, 10^-7 and 10^-8.

L^2-norm error indicator: In Figures 5.19 and 5.21 the convergence curves of the adaptive local refinement are very similar, no big difference can be seen. Both have better convergence rates than the homogeneous refinement, with an even higher slope which makes big difference at high degrees of freedom with considerably smaller errors.

Although both convergence curves are very similar, the errors using refinement strategy of 5% are slightly minor. In Figures 5.20 and 5.22, comparing the refinement process one can see for 5% refinement strategy the mesh at higher degrees of freedom is denser where the solution peaks are, whilst for 20% refinement strategy the mesh is a little more scattered around the solution peaks, which explains the small difference in the errors obtained.

L^2-norm limit error: The Results obtained using an error threshold for each cell presents good convergence rates for higher degrees of freedom. Figure 5.23 shows a big difference between the slopes of the homogeneous curve and the adaptive curve. For lower degrees of freedom the adaptive refinement sticks to the homogeneous process because all the errors in the cells are higher than the limit, causing the refinement of the entire domain until some cells reaches this limit and the refinement process focuses where the solution peaks are, as it can be seen in Figure 5.24, so that the solution improves considerably compared to the homogeneous refinement process.

L^2-norm error indicator 20%

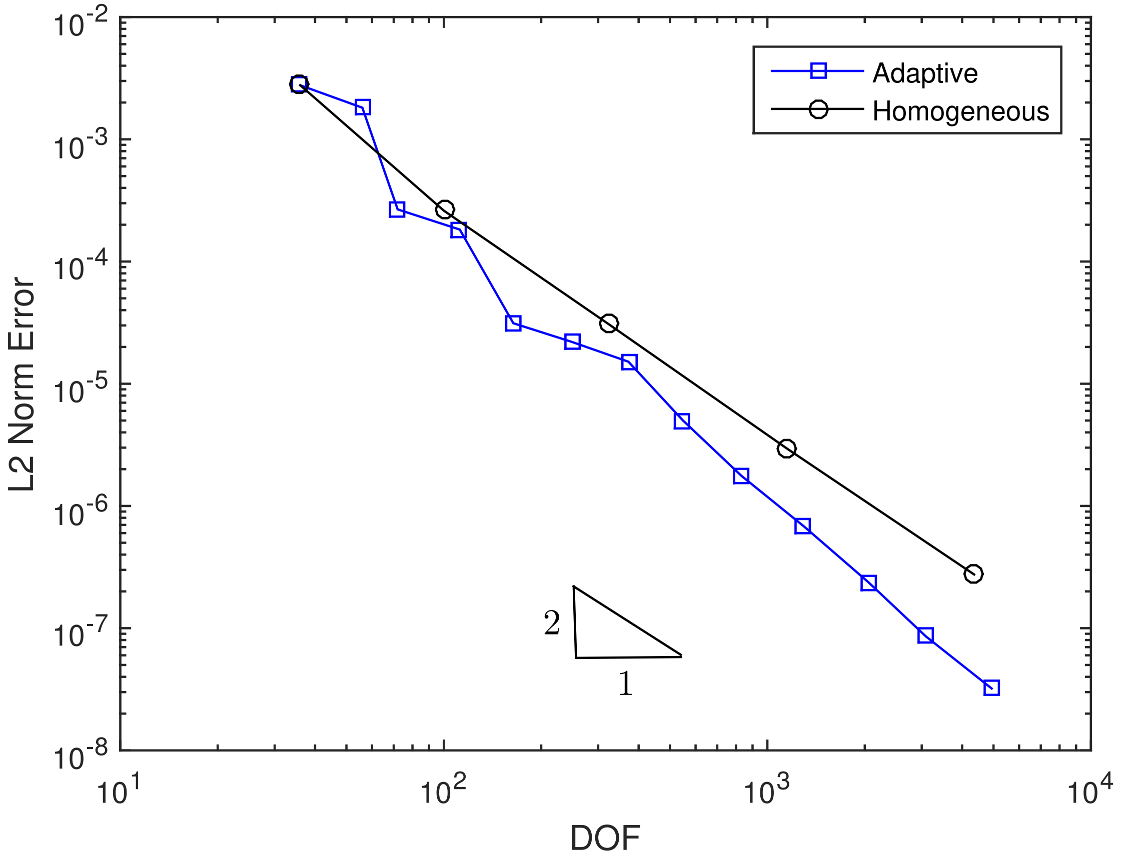

Figure 5.19: Comparison of Homogeneous refinement and Adaptive local refinement for the Two heat source solution by Method 1, L^2-*norm error indicator*, using 20% refinement strategy.

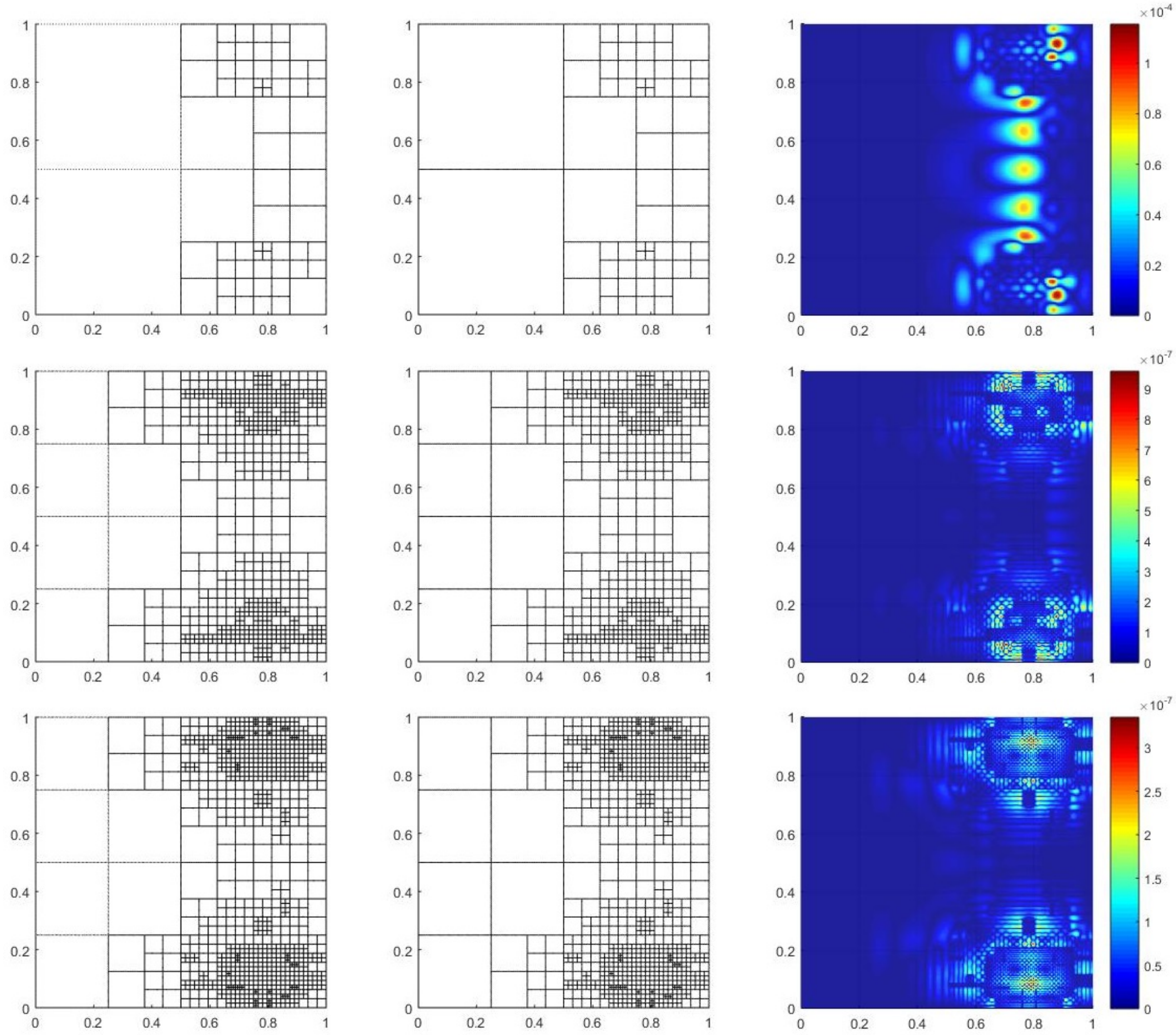

Figure 5.20: In 3 rows, from left to right: Parametric mesh, Geometric mesh and error plot of different degrees of freedom for self-adaptive local refinement for the Two heat source solution by Method 1, L^2-*norm error indicator*, using 20% refinement strategy.

L^2-norm error indicator 5%

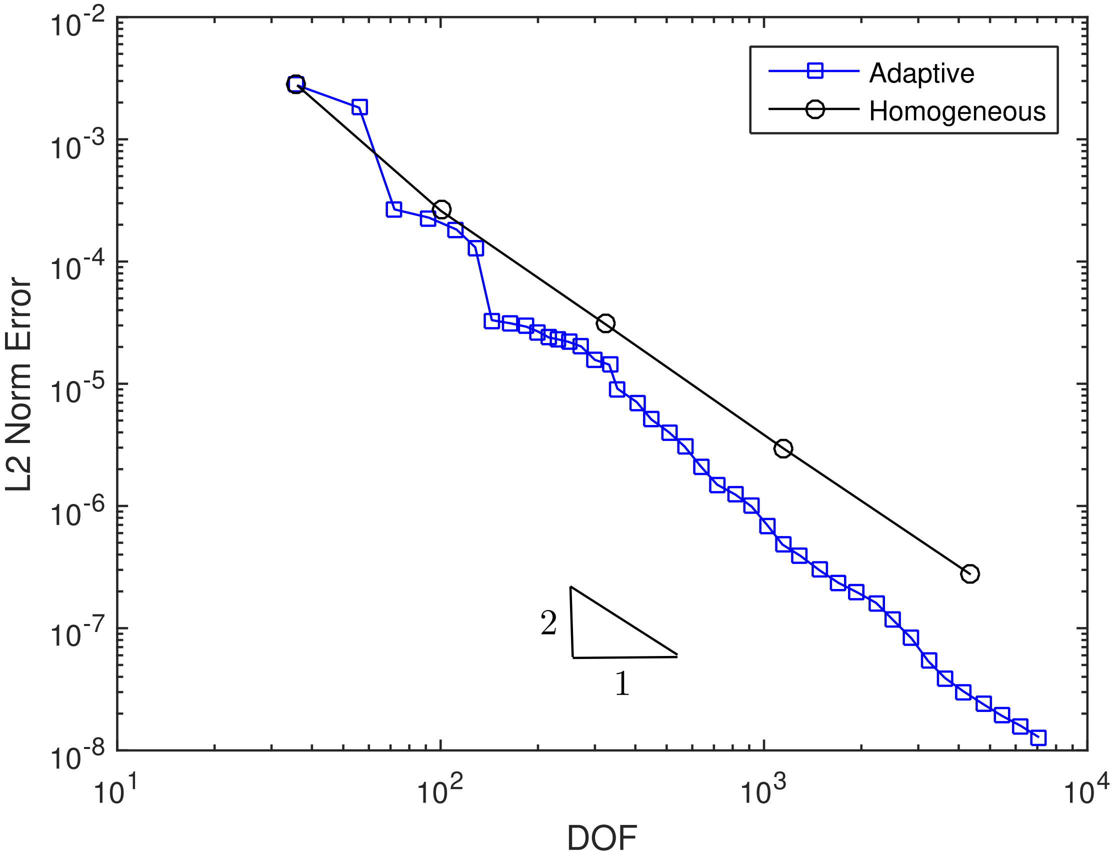

Figure 5.21: Comparison of Homogeneous refinement and Adaptive local refinement for the Two heat source solution by Method 1, L^2-*norm error indicator*, using 5% refinement strategy.

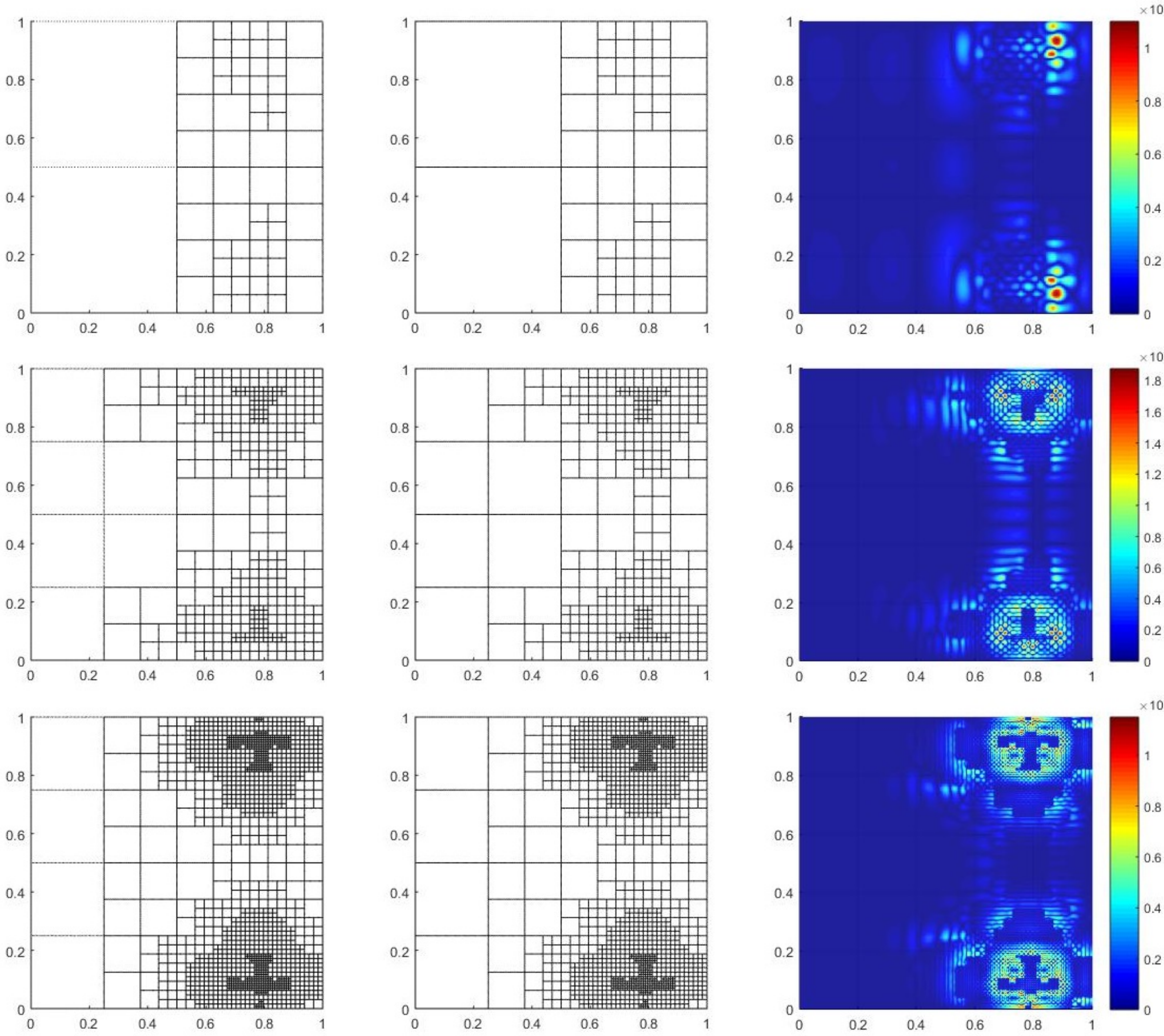

Figure 5.22: In 3 rows, from left to right: Parametric mesh, Geometric mesh and error plot of different degrees of freedom for self-adaptive local refinement for the Two heat source solution by Method 1, L^2-*norm error indicator*, using 5% refinement strategy.

L^2-norm limit error

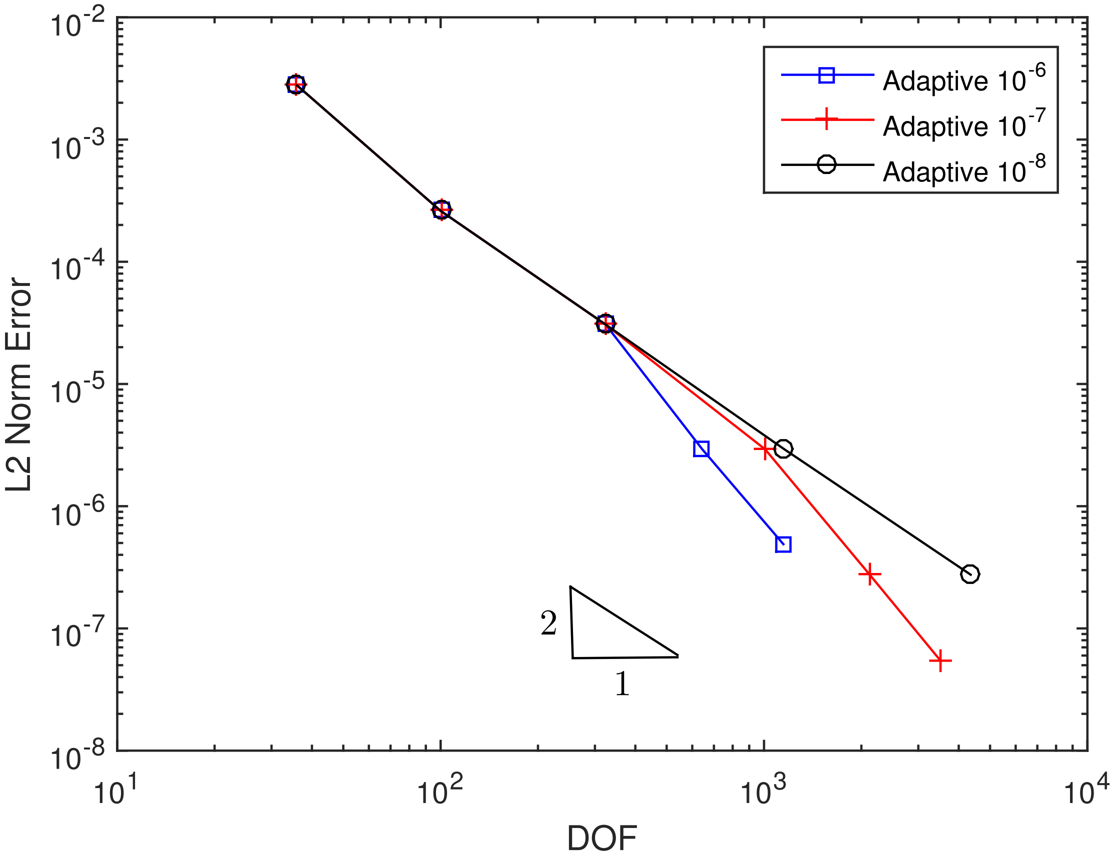

Figure 5.23: Comparison of Homogeneous refinement and Adaptive local refinement for the Two heat source solution by Method 1, L^2-*norm error indicator*, using 5% refinement strategy.

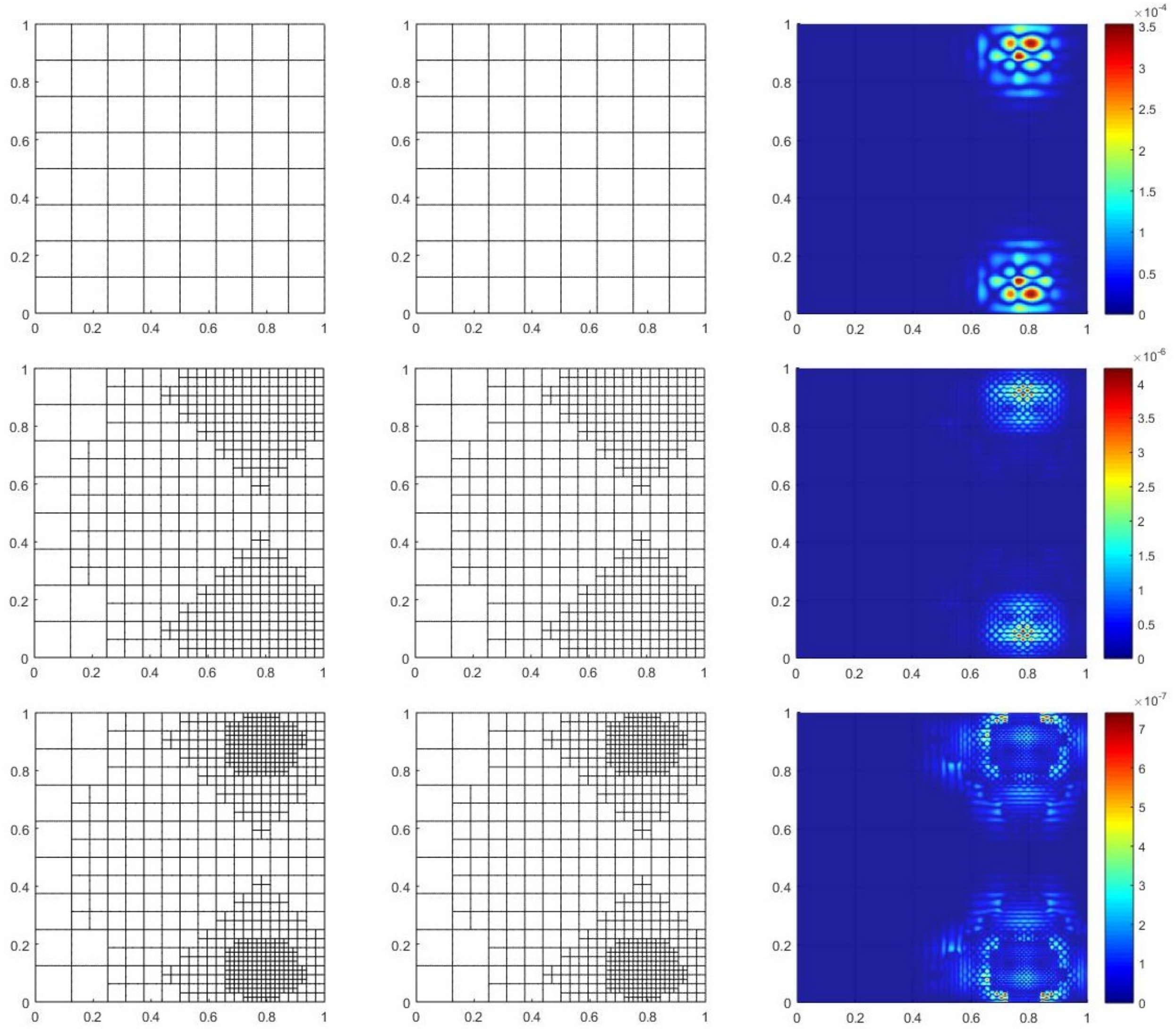

Figure 5.24: In 3 rows, from left to right: Parametric mesh, Geometric mesh and error plot of different degrees of freedom for self-adaptive local refinement for the Two heat source solution by Method 1, L^2-*norm error indicator*, using 5% refinement strategy.

5.2.2 Method 2: Residual based error indicator

The following results are obtained using the residual based error indicator (Eq. (2.37)) over the Two heat source solution and using the refinement strategy with 20% and 5%.

Residual based error indicator: In Figures 5.25 and 5.27 we can see the convergence curves for both refinement strategies, that has an almost identical behavior, where the local refinement convergence rates are better than the homogeneous refinement process, after 10^2 degrees of freedom we see both convergence curves going below the homogeneous curve, meaning lower errors for same degrees of freedom, what implies a better solution, being more accurate than using the homogeneous refinement. When comparing to the Poisson's problem with same method (Figures 5.7 and 5.9), one can see for this problem, where there are two peaks, that the local refinement makes better improvements for this solution.

Looking at Figures 5.26 and 5.28 we can see what was expected from the local adaptive refinement, the location where the solution peaks are, given by Eq. (4.3), is where the refinement process is centered, leaving the left side of the domain without refining until higher degrees of freedom. This convergence rates demonstrate the effectiveness of the method.

Using method 2 in this problem, one can see there is barely no difference between strategy refinement of 20% and 5%, both convergence rates are very similar.

Residual based error indicator 20%

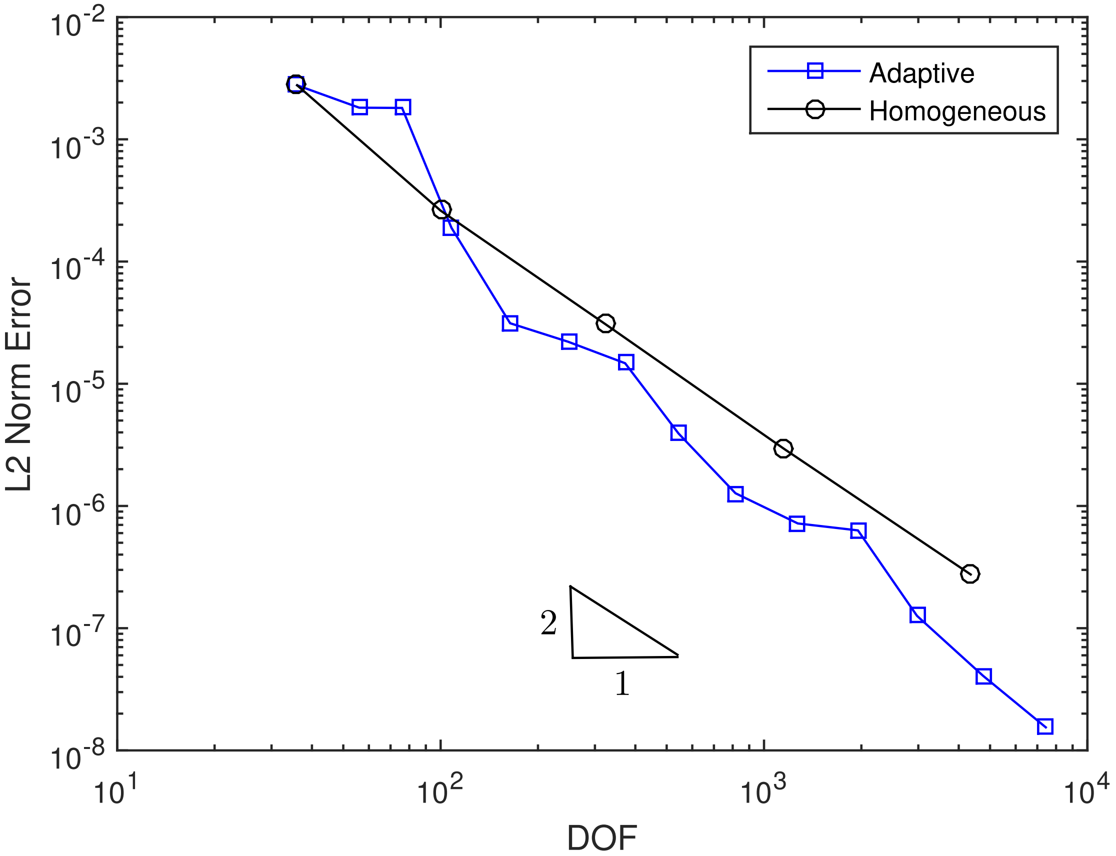

Figure 5.25: Comparison of Homogeneous refinement and Adaptive local refinement for the Two heat source solution by Method 2, *Residual based error indicator*, using 20% refinement strategy.

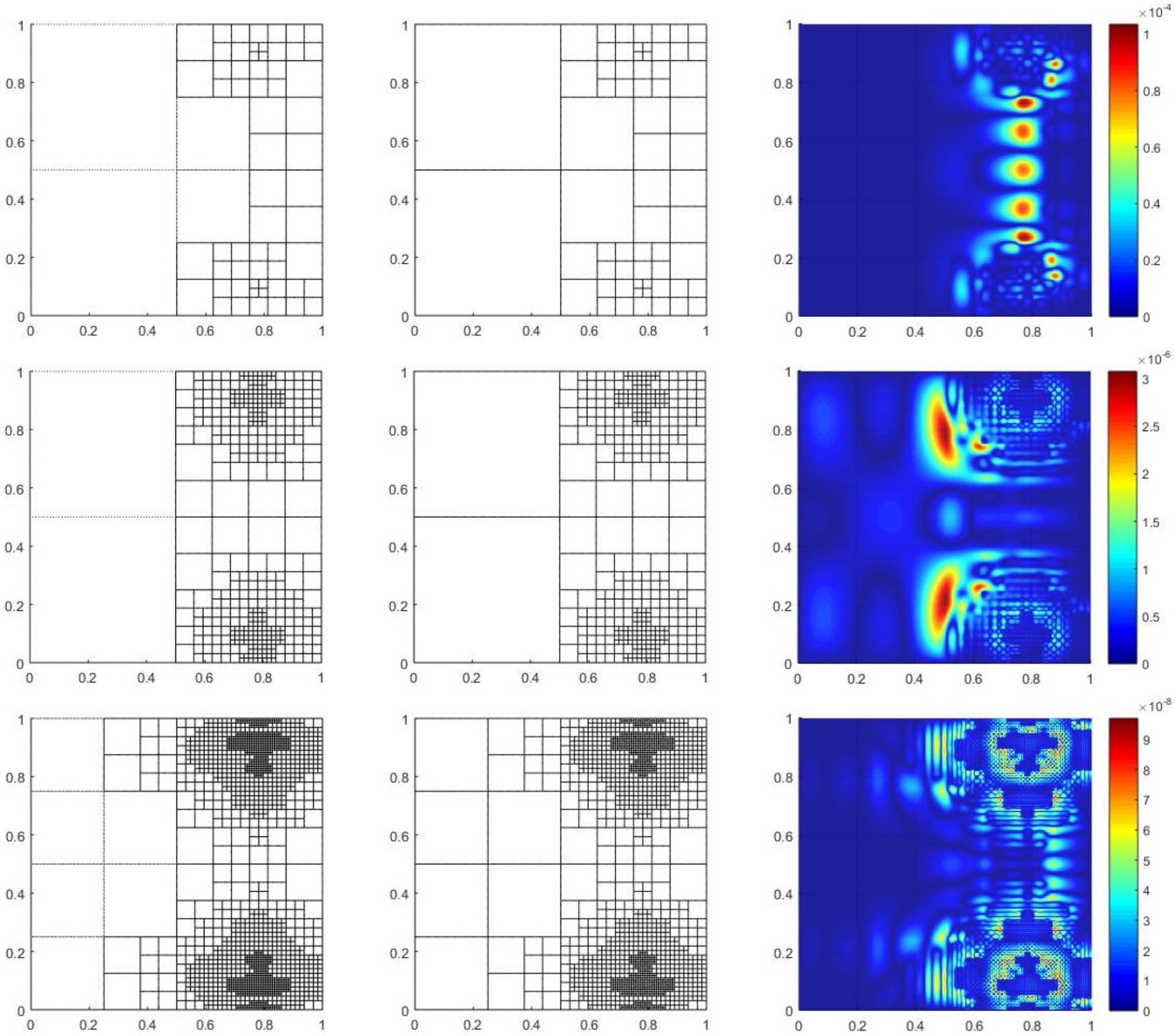

Figure 5.26: In 3 rows, from left to right: Parametric mesh, Geometric mesh and error plot of different degrees of freedom for self-adaptive local refinement for the Two heat source solution by Method 2, *Residual based error indicator*, using 20% refinement strategy.

Residual based error indicator 5%

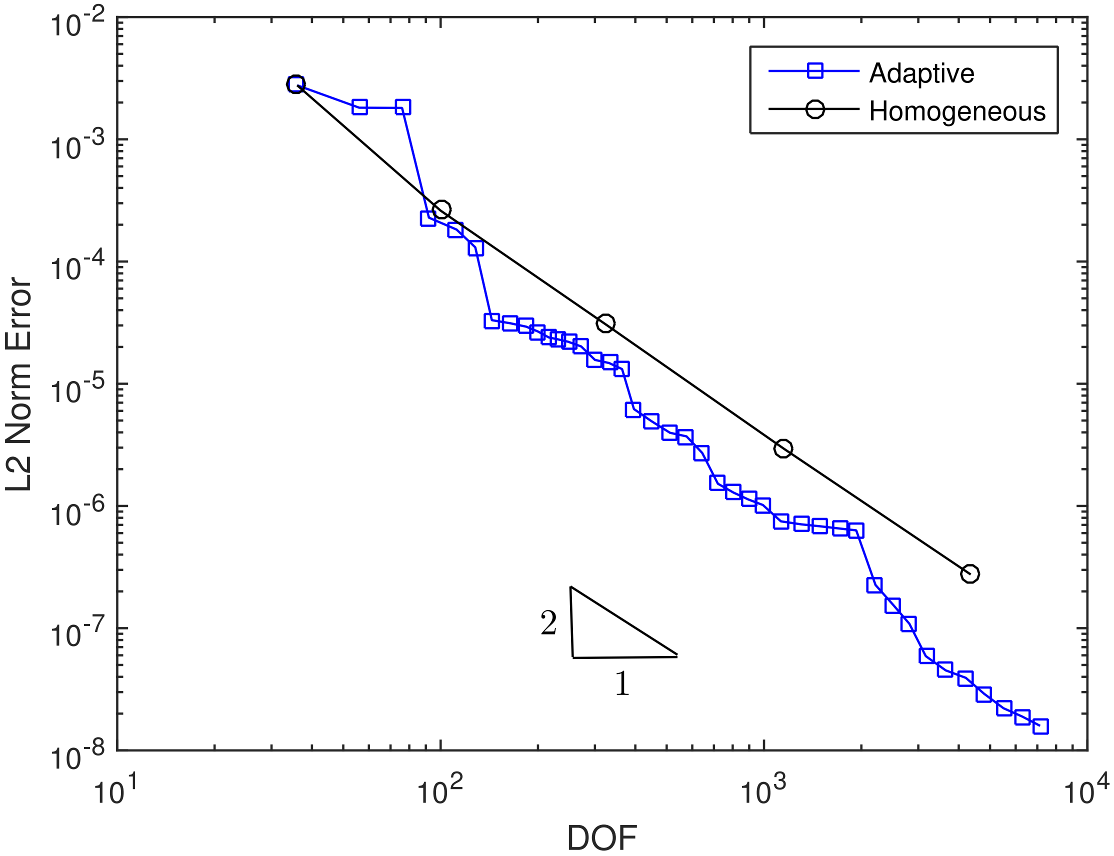

Figure 5.27: Comparison of Homogeneous refinement and Adaptive local refinement for the Two heat source solution by Method 2, *Residual based error indicator*, using 5% refinement strategy.

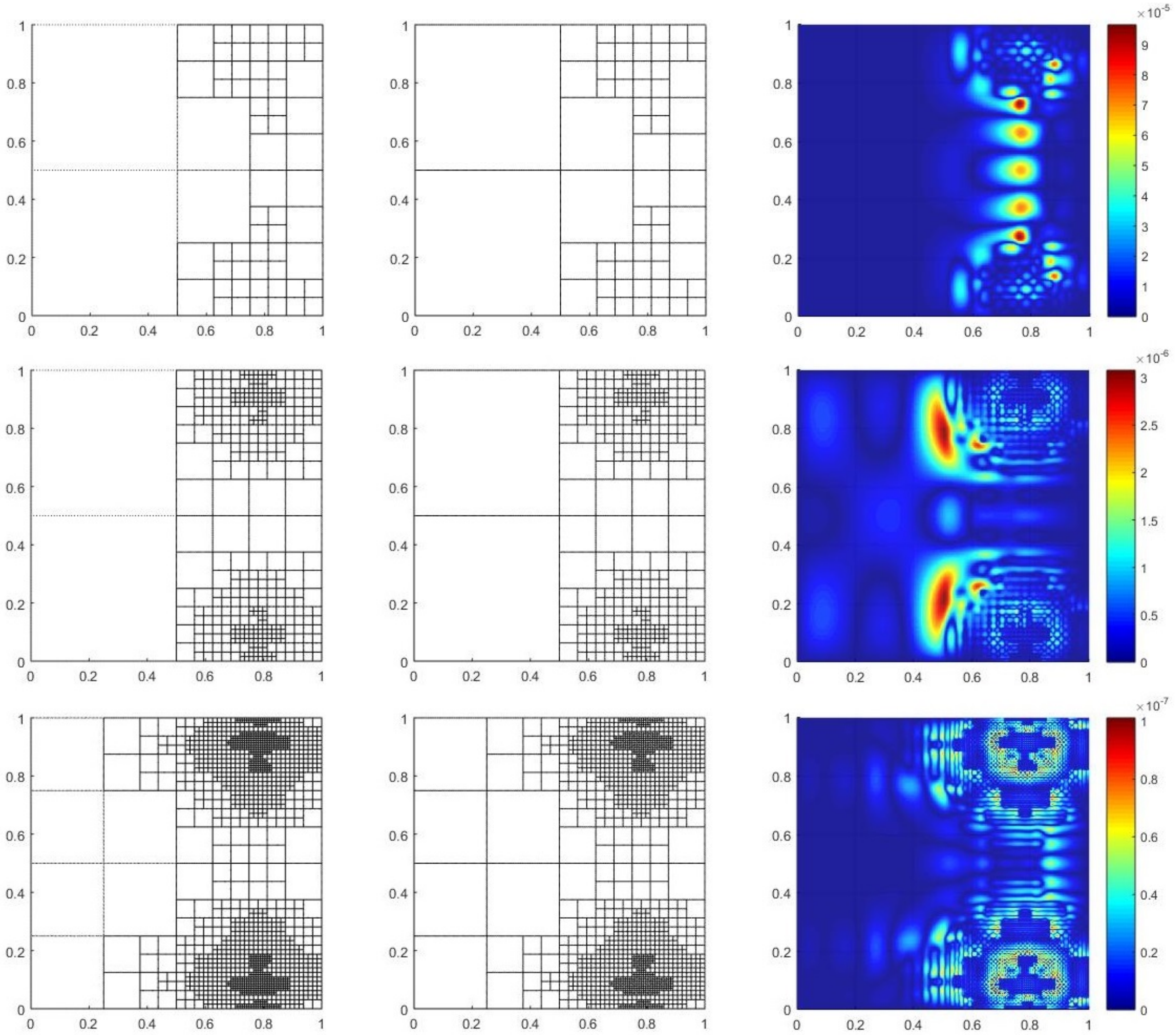

Figure 5.28: In 3 rows, from left to right: Parametric mesh, Geometric mesh and error plot of different degrees of freedom for self-adaptive local refinement for the Two heat source solution by Method 2, *Residual based error indicator*, using 5% refinement strategy.

5.2.3 Method 3: Hierarchical Adaptivity error indicator

This method is divided in two parts. The first results are obtained using the hierarchical error indicator with L^2-norm given by Eq. (2.43) using refinement strategy of 20% and 5%. The second part, the hierarchical error indicator is used with H^1-norm given by Eq. (2.44) and the refinement strategy of 20% and 5%.

Hierarchical Adaptivity L^2-norm: The convergence curve obtained with refinement strategy of 20% is very different from all the convergence curves analyzed because it maintains a steady error until really high degrees of freedom where it decreases drastically. This behavior occurs due to the 'loop' effect as seen in the Poisson's problem using Method 3 with hierarchical adaptivity, where the refinement process take place in some place and it starts refining over and over, this can be seen in Figure 5.30 where for lower degrees of freedom it refines near one peak of the solution over and over, this makes the decrease of the error negligible, looking almost constant, because even it refines over one of the solution peak, in the place where the other peak is located the error is high enough to counteract the error of the refined cells. For high degrees of freedom where the convergence rate of the adaptive refinement increases sharply, one can see in Figure 5.30 the refinement process began to take place all over the domain, making the error of the cells drastically decrease explaining the behavior of the convergence curve of the adaptive refinement.

For the refinement strategy of 5% in Figure 5.31, we can see a behavior more alike the Poisson's problem, similar to a stair, explained like the previous one because of the 'loop' effect. This refinement strategy detects better the two solution peaks unlike with 20% refinement strategy that focuses just in one solution peak. In Figure 5.32, the irregular refinement process can be observed, refining both solution peaks areas but in a very dispersed way. This makes the converge rate worse than the homogeneous refinement process.

Hierarchical Adaptivity H^1-norm: The convergence curves of refinement strategy of 20% and 5% are very similar, both follows the same pattern. In figure 5.35 for refinement strategy of 5% the behavior of the curve is similar to a stair whilst in Figure 5.33 the curve is more soft but following the same pattern. Like the other convergence rates using Method 3, the convergence rate of both 5% and 20% refinement strategies using H^1-norm are worse than the homogeneous refinement.

Figures 5.34 and 5.36 show the different refinement processes for 20% and 5%. Both are very irregular, although neither of them refines the left side of the domain, because the solution peaks are located at the right side, do not follow a regular pattern of localized refinement. Figure 5.34 shows a denser refinement mesh around the solution peaks whilst Figure 5.36 shows a very scattered refinement. However, both convergence rates are very similar.

Hierarchical Adaptivity L^2-norm error indicator 20%

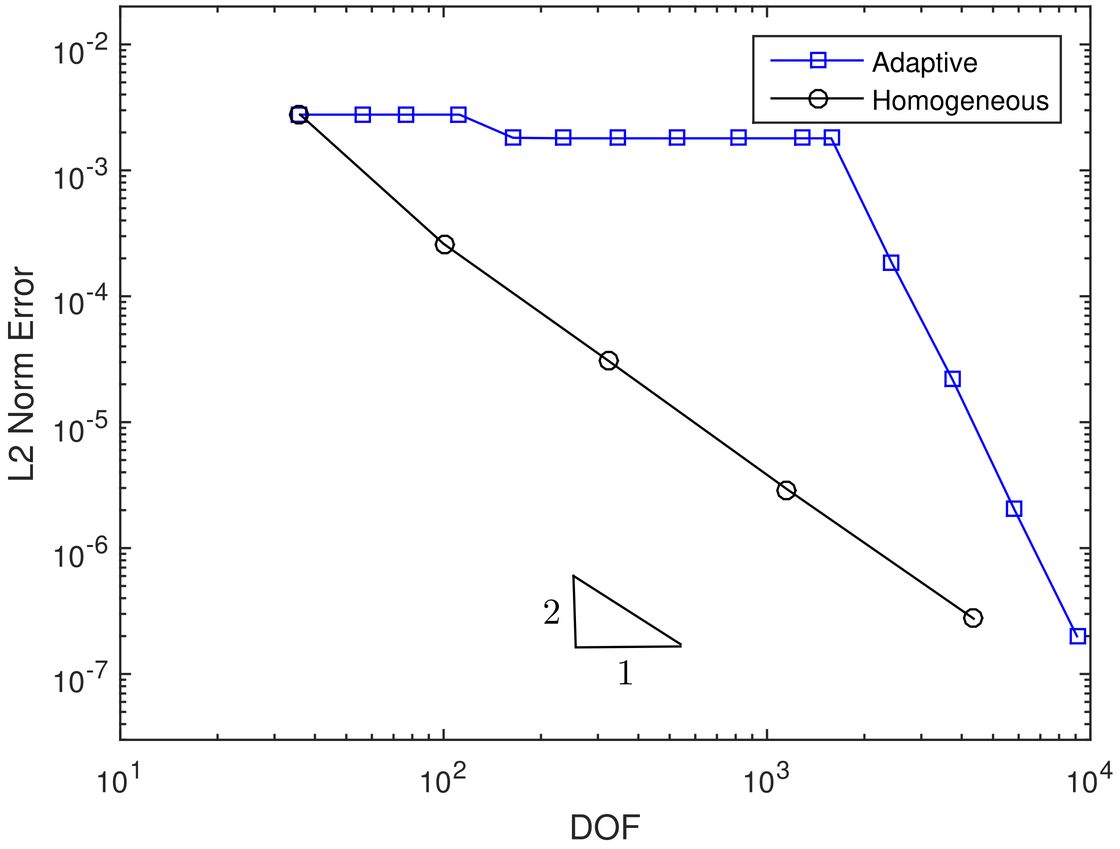

Figure 5.29: Comparison of Homogeneous refinement and Adaptive local refinement for the Poisson's solution by Method 3, *Hierarchical adaptivity L^2-norm error indicator*, using 20% refinement strategy.

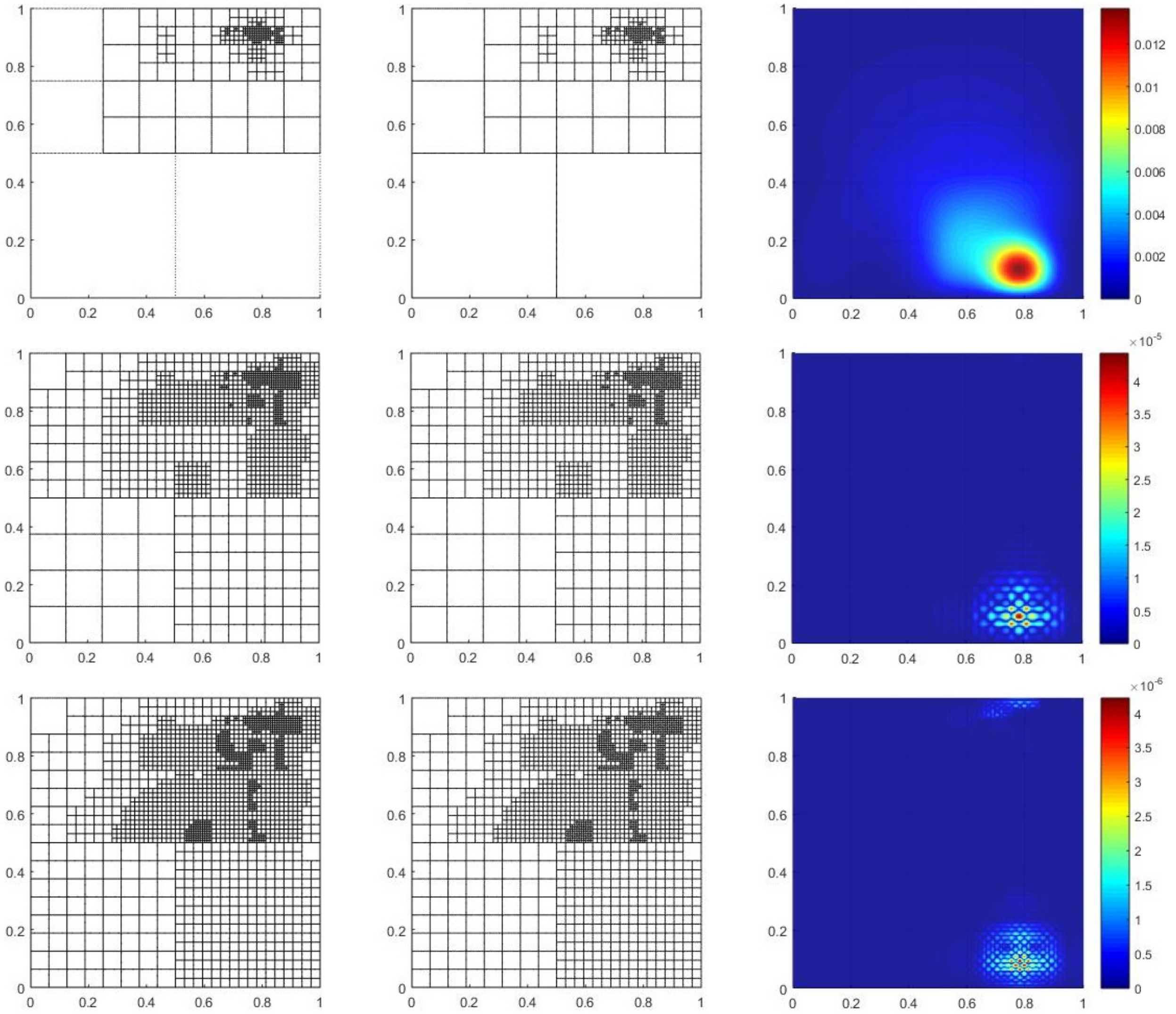

Figure 5.30: In 3 rows, from left to right: Parametric mesh, Geometric mesh and error plot of different degrees of freedom for self-adaptive local refinement for the Poisson's solution by Method 3, *Hierarchical adaptivity L^2-norm error indicator*, using 20% refinement strategy.

Hierarchical Adaptivity L^2-norm error indicator 5%

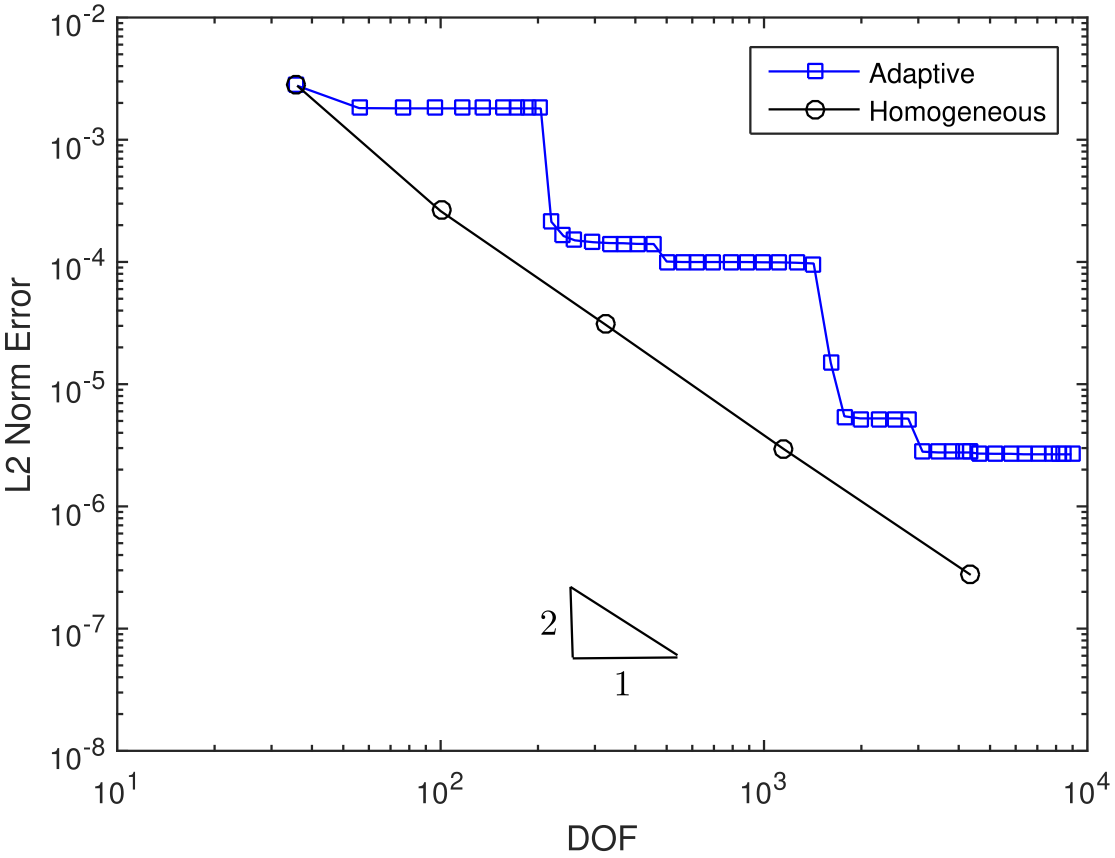

Figure 5.31: Comparison of Homogeneous refinement and Adaptive local refinement for the Poisson's solution by Method 3, *Hierarchical adaptivity L^2-norm error indicator*, using 5% refinement strategy.

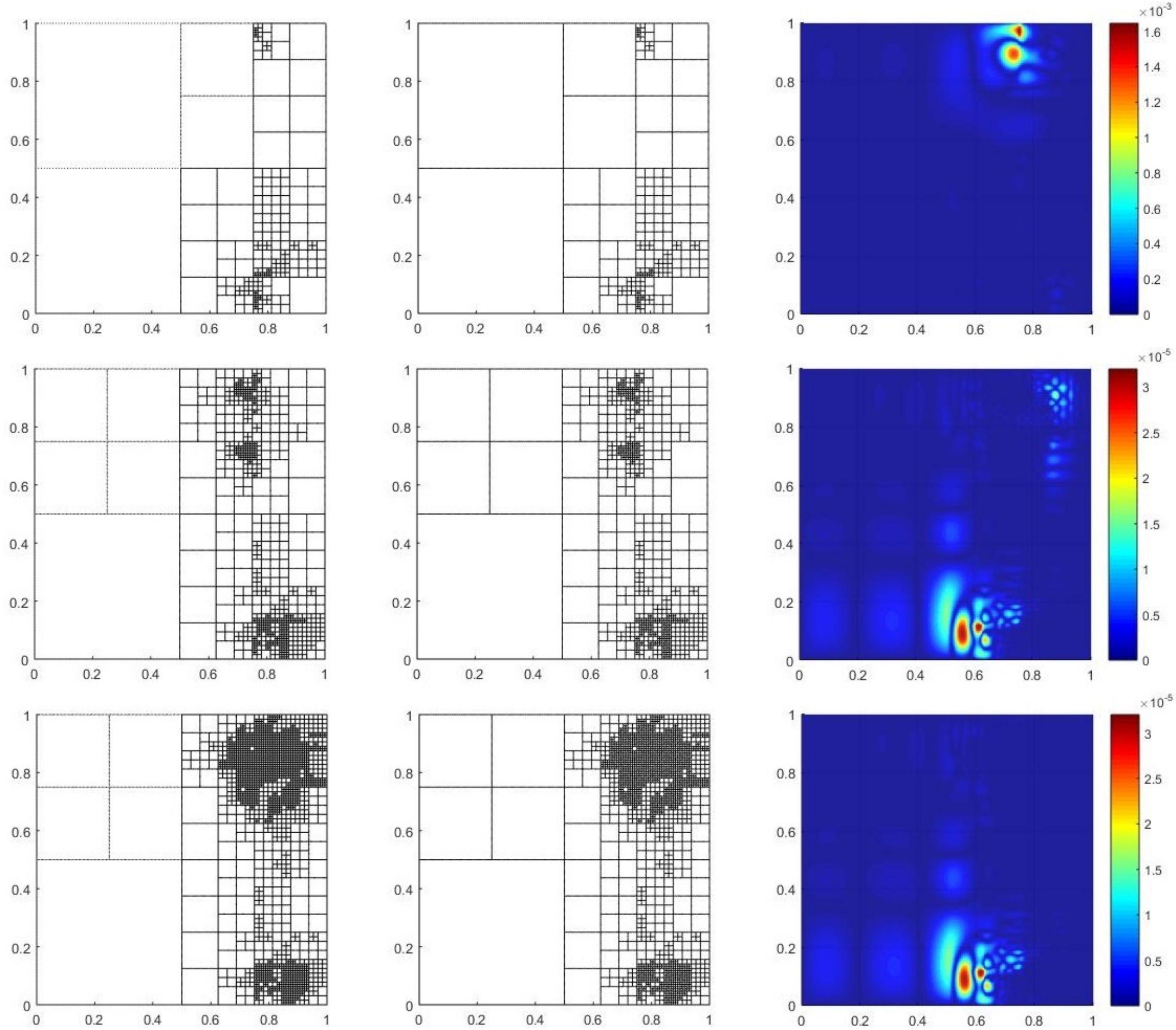

Figure 5.32: In 3 rows, from left to right: Parametric mesh, Geometric mesh and error plot of different degrees of freedom for self-adaptive local refinement for the Poisson's solution by Method 3, *Hierarchical adaptivityL^2-norm error indicator*, using 5% refinement strategy.

Hierarchical Adaptivity H^1-norm error indicator 20%

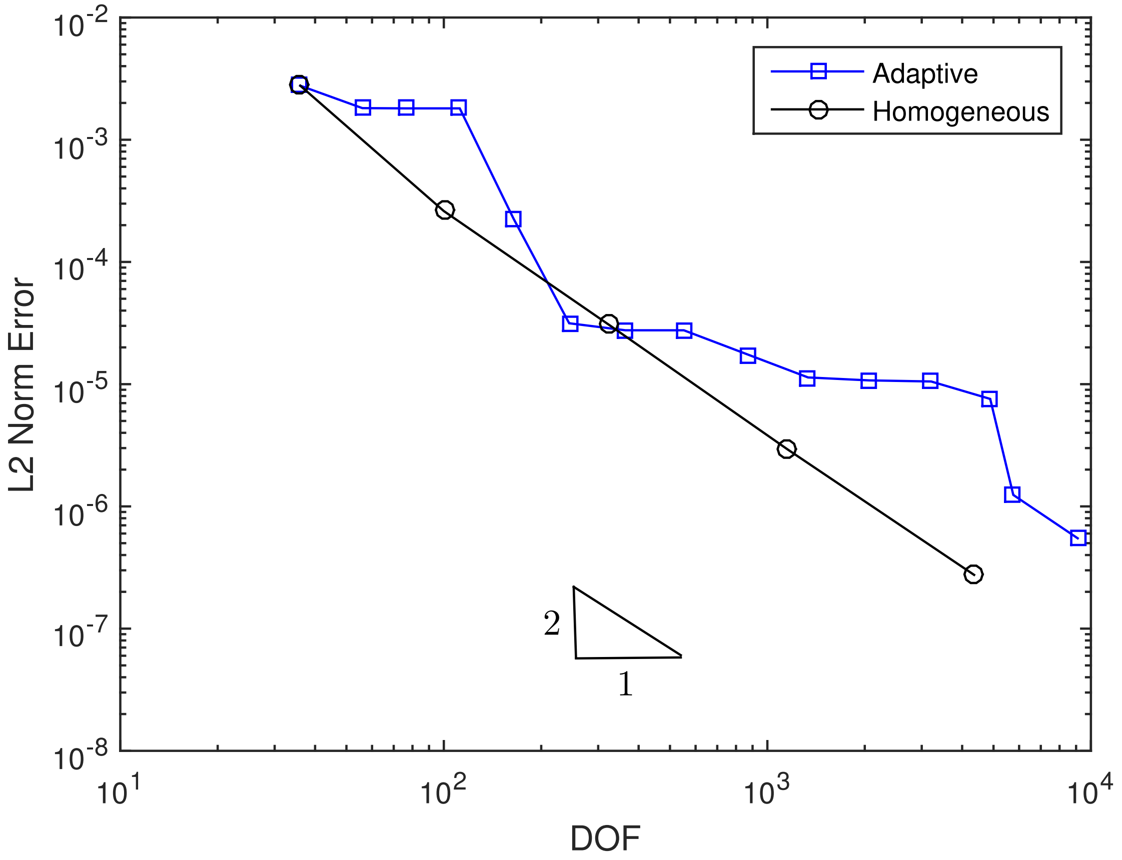

Figure 5.33: Comparison of Homogeneous refinement and Adaptive local refinement for the Poisson's solution by Method 3, *Hierarchical adaptivity H^1-norm error indicator*, using 20% refinement strategy.

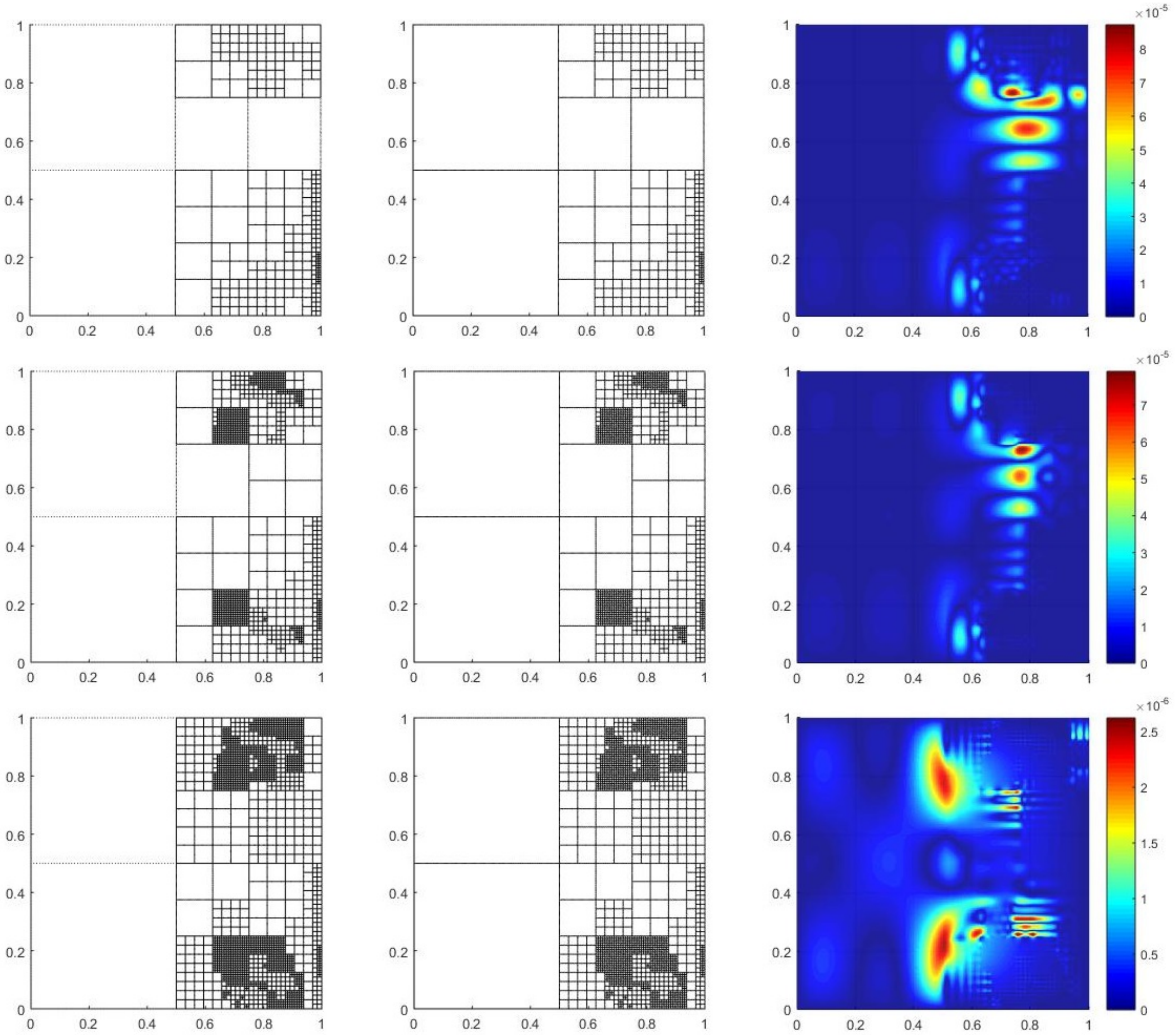

Figure 5.34: In 3 rows, from left to right: Parametric mesh, Geometric mesh and error plot of different degrees of freedom for self-adaptive local refinement for the Poisson's solution by Method 3, *Hierarchical adaptivity H^1-norm error indicator*, using 20% refinement strategy.

Hierarchical Adaptivity H^1-norm error indicator 5%

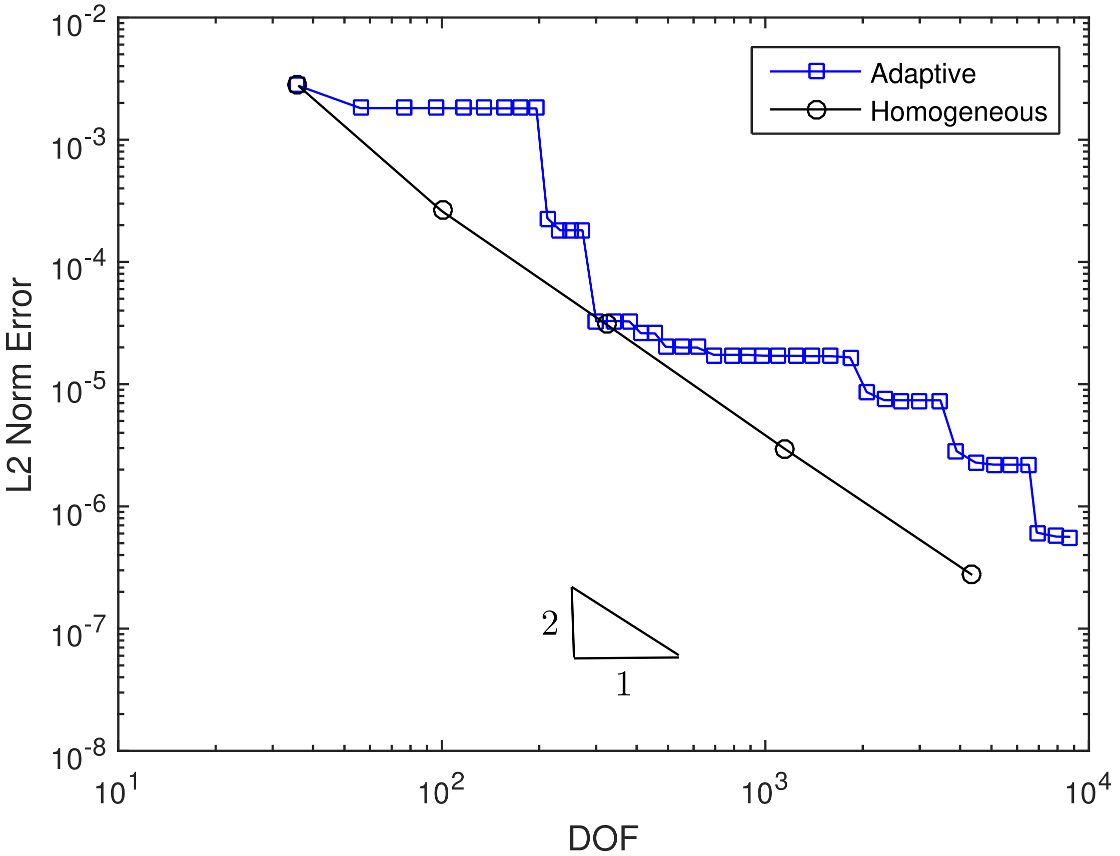

Figure 5.35: Comparison of Homogeneous refinement and Adaptive local refinement for the Poisson's solution by Method 3, *Hierarchical adaptivity H^1-norm error indicator*, using 5% refinement strategy.

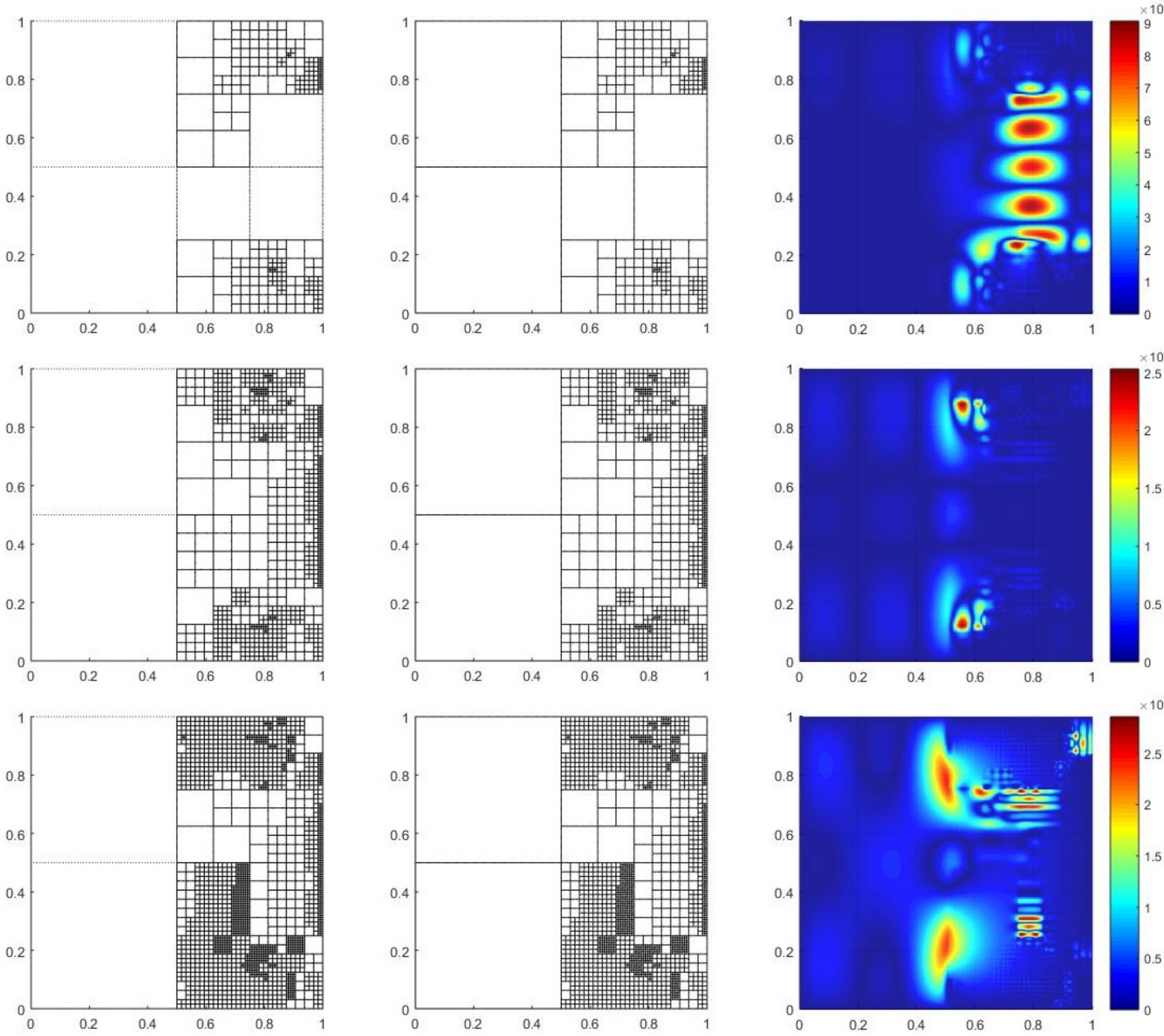

Figure 5.36: In 3 rows, from left to right: Parametric mesh, Geometric mesh and error plot of different degrees of freedom for self-adaptive local refinement for the Poisson's solution by Method 3, *Hierarchical adaptivity H^1-norm error indicator*, using 5% refinement strategy.

5.3 Dependence of the solution on the parameterization of the geometry

Poisson's problem, given by Eq. (2.5), is solved using two geometry parameterizations (see Figure 4.4), to test the dependence on the solution. A regular parameterization and an irregular parameterization were used, both have been tested using Method 2: *Residual based error indicator*, using L^2-norm. Figure 5.37 shows the numerical results in both parameterizations, at a glance one can see the difference between the solutions obtained, where using a regular parameterization, best results are achieved.

Figures 5.38 and 5.39, show the convergence curves of the regular parameterization and the irregular parameterization respectively. For the irregular parameterization one can see the homogeneous convergence rate is worse than for the regular parameterization, but using Adaptive local refinement gives better convergence rates compared to the homogeneous refinement than using a regular parameterization where the convergence rate of the adaptive refinement is slightly better than the homogeneous refinement, still, the irregular parameterization convergence curve for homogeneous and adaptive refinements are worse than using a regular parameterization. In Figure 5.40, one can see the mesh of the adaptive refinement using the irregular parameterization, where the highest errors are located near the boundaries of the irregular mesh, therefore, the refinement focuses on the edges, causing the convergence rates being worse than using a regular parameterization.

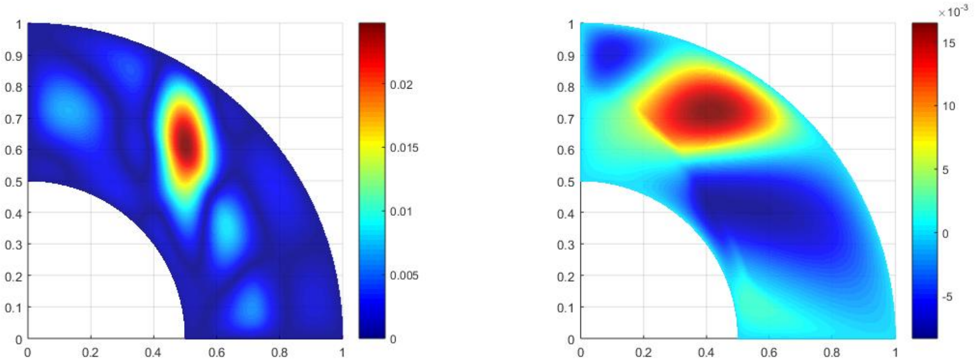

Figure 5.37: In the left is the numerical solution using GIFT method with an uniform parameterization. In the right is the numerical solution using GIFT method with a non-uniform parameterization (see Figure 4.4).

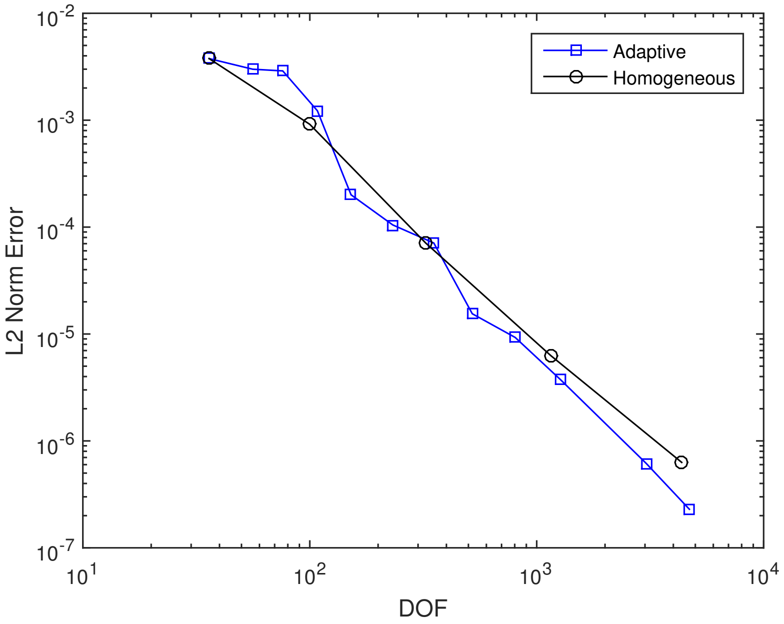

Figure 5.38: Comparison of Homogeneous refinement and Adaptive local refinement over a regular parameterization for the Poisson's solution by Method 2, *Residual based error indicator*, using 20% refinement strategy.

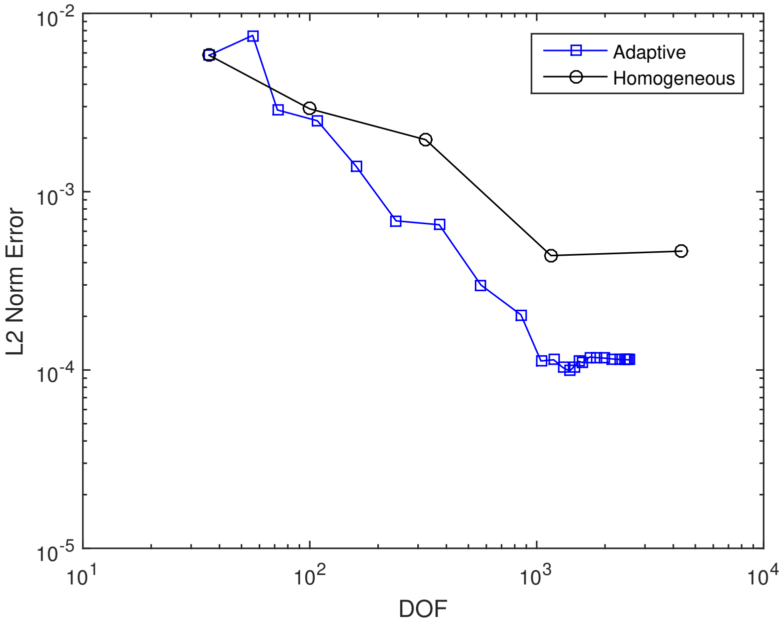

Figure 5.39: Comparison of Homogeneous refinement and Adaptive local refinement over an irregular parameterization for the Poisson's solution by Method 2, *Residual based error indicator*, using 20% refinement strategy.

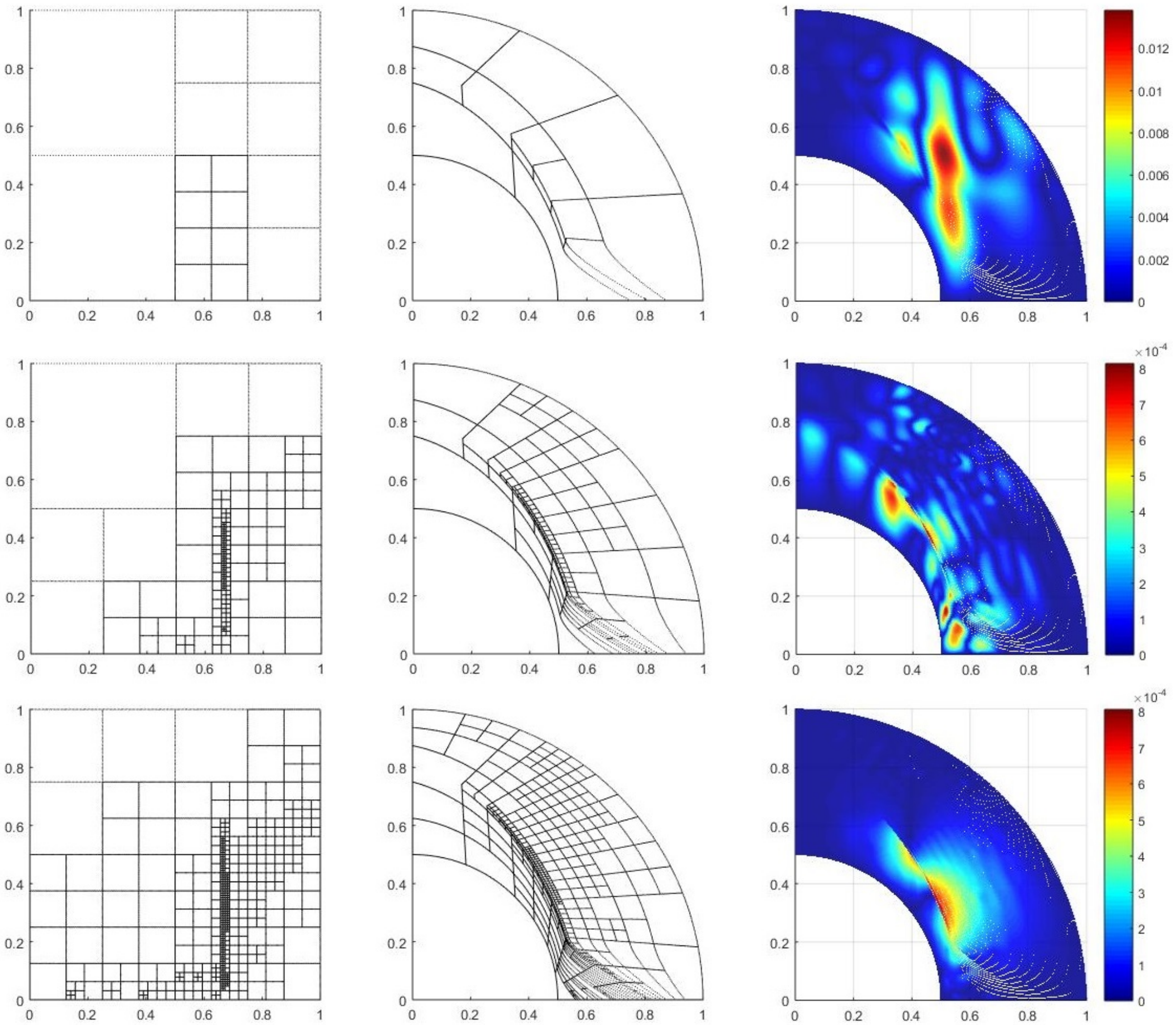

Figure 5.40: In 3 rows, from left to right: Parametric mesh, Geometric mesh and error plot of different degrees of freedom for self-adaptive local refinement for the Poisson's solution over an irregular parameterization by Method 2, *Residual based error indicator*, using 20% refinement strategy.

Chapter 6

Discussion

The GIFT method shows several improvements compared to IGA method, mainly because of the advantage of using separate fields for the geometry and the solution, this was reviewed in [1] where GIFT method was introduced. In this work the GIFT method performance was studied, specifically it's performance using a local adaptive refinement, thanks to the PHT-Splines properties used for the field solution and using NURBS for the geometry; with different error indicators that make the process self-select the cells that will be refined. Also the dependence of the geometry parameterization was reviewed.

For Poisson's problem using the Method 1, the three different refinement strategies results shown slightly improvement in the convergence rates for higher degrees of freedom and the best results were achieved using refinement strategy of 5% where the refined mesh shows a regular pattern of localized refinement. Results using Method 2 are quite similar to the results obtained using Method 1. For both methods the convergence rates are better for higher degrees of freedom where the refinement strategy of 5% show the best results. Finally, Method 3 shows different results, where the homogeneous refinement is a better option. Using H^1-norm with the hierarchical algorithm shows better convergence curves than L^2-norm and for both the refinement strategy of 20% unlike the other methods is the one that best suits the solution.

Comparing the three methods on the Poisson's problem, Method 3 is the only one that has worse convergence rates than the homogeneous refinement, because the hierarchical algorithm does not show a good selection in the cells to be refined. Method 1 and Method 2 shows fine results. Method 1 uses the analytical solution, so it is expected to present a good solution, and consequently a better rate of convergence than the homogeneous refinement. Method 2 show the best results, due to the fact that it does not uses the analytical solution to refine the numerical solution and offers similar convergence rates, it also presents a regular localized adaptive mesh around the peaks of the solution as expected.

For Two Heat source problem, Method 1 shows a very localized refined mesh, focusing where the peaks of the solution are, this causes better convergence rates than the homogeneous refinement, using 5% refinement strategy a more regular refined mesh is obtained causing small improvements on the convergence rates compared to the other refinement strategies. For Method 2, the results are similar to Method 1, like for Poisson's problem, with a slightly difference on the convergence rate being a little unsteady. Results of Method 3 are not similar as the results of the other methods, using L^2-norm the convergence rate behaves very odd, due to the inability to detect both peaks of the solution, unlike with H^1-norm that focuses the refinement near the peaks of the solution with such an irregular pattern that makes the convergence rate worse than the homogeneous one.

As with the Poisson's problem, the best results as with the best convergence rates are the results of both Method 1 and Method 2. Both methods show accurate refinement near the sharp peaks of the solution. Method 3 on the other hand presents very inaccurate results because of the inability to refine the cells near the peaks of the solution, causing worse convergence rates than the homogeneous rate.

Testing the two different geometry parameterizations, one can conclude that the geometry parameterization does affect the solution, varying the convergence rate. Due to the deformed mesh that the irregular parameterization creates, for further analysis a new error indicator that involves the deformations of the cells could be tested, to counteract the effect of the deformation for each cell.

Conclusion

An in-depth study of GIFT was conducted for three benchmark problems for Poisson's equation, demonstrating that pairing NURBS geometry with the PHT-Spline field is possible in general. However, for some choices of the geometry parameterization, the optimal convergence rate was achieved, while for others - not. Three error indicators and two different refinement strategies were tested and it was shown, that the residual-based error indicator (Method 2) yields the most improvement in the convergence rate and it's capable of capturing sharp solution peaks. An L2-norm error indicator (Method 1) based on the comparison of the numerical solution with the analytical data, also showed significant improvement in the convergence rate. The adaptive refinement, based on the hierarchical error indicator (Method 3), showed worse results in comparison with the uniform or homogeneous refinement. And finally, the dependence of the solution on the geometry parameterization was tested and it was observed, that the irregular geometry parameterizations should be avoided, as the error in numerical solution accumulates along the boundaries of the geometry elements, leading to the bad overall performance of the method.

Bibliography

[1] E. Atroshchenko, G. Xu, S. Tomar, and S. Bordas, "Weakening the tight coupling between geometry and simulation in isogeometric analysis: from sub-and super-geometric analysis to geometry independent field approximation (gift)," *arXiv preprint arXiv:1706.06371*, 2017.

[2] T. J. Hughes, J. A. Cottrell, and Y. Bazilevs, "Isogeometric analysis: Cad, finite elements, nurbs, exact geometry and mesh refinement," *Computer methods in applied mechanics and engineering*, vol. 194, no. 39, pp. 4135–4195, 2005.

[3] G. Xu, E. Atroshchenko, and S. Bordas, "Geometry-independent field approximation for spline-based finite element methods," in *Proceedings of the 11th World Congress in Computational Mechanics*, 2014.

[4] J. H. Lienhard, *A heat transfer textbook.* Courier Corporation, 2013.

[5] A. P. Selvadurai, *Partial Differential Equations in Mechanics 2: the Biharmonic Equation, Poisson's Equation*, vol. 2. Springer Science & Business Media, 2013.

[6] I. J. Schoenberg, "Contributions to the problem of approximation of equidistant data by analytic functions. part b. on the problem of osculatory interpolation. a second class of analytic approximation formulae," *Quarterly of Applied Mathematics*, vol. 4, no. 2, pp. 112–141, 1946.

[7] J. H. Ahlberg, E. N. Nilson, and J. L. Walsh, *The Theory of Splines and Their Applications: Mathematics in Science and Engineering: A Series of Monographs and Textbooks*, vol. 38. Elsevier, 2016.

[8] J. Epperson, "History of splines," *NA Digest*, vol. 98, no. 26, 1998.

[9] R. Bartels, J. C. Beatty, and B. A. Barsky, "An introduction to the use of splines in computer graphics, to be published by morgan kaufmann publishers," *Inc., Los Altos, California*, 1987.

[10] L. Piegl and W. Tiller, *The NURBS book.* Springer Science & Business Media, 2012.

[11] T. W. Sederberg, J. Zheng, A. Bakenov, and A. Nasri, "T-splines and t-nurccs," in *ACM transactions on graphics (TOG)*, vol. 22, pp. 477–484, ACM, 2003.

[12] T. W. Sederberg, D. L. Cardon, G. T. Finnigan, N. S. North, J. Zheng, and T. Lyche, “T-spline simplification and local refinement,” in *ACM transactions on graphics (TOG)*, vol. 23, pp. 276–283, ACM, 2004.

[13] Y. Bazilevs, V. M. Calo, J. A. Cottrell, J. A. Evans, T. Hughes, S. Lipton, M. A. Scott, and T. W. Sederberg, “Isogeometric analysis using t-splines,” *Computer Methods in Applied Mechanics and Engineering*, vol. 199, no. 5, pp. 229–263, 2010.

[14] M. R. Dörfel, B. Jüttler, and B. Simeon, “Adaptive isogeometric analysis by local h-refinement with t-splines,” *Computer methods in applied mechanics and engineering*, vol. 199, no. 5, pp. 264–275, 2010.

[15] J. Deng, F. Chen, X. Li, C. Hu, W. Tong, Z. Yang, and Y. Feng, “Polynomial splines over hierarchical t-meshes,” *Graphical models*, vol. 70, no. 4, pp. 76–86, 2008.

[16] J. N. Reddy, *An introduction to the finite element method*, vol. 2. McGraw-Hill New York, 1993.