ISOGEOMETRIC COLLOCATION METHOD FOR ELASTICITY PROBLEMS CONTAINING SINGULARITIES

by

Puja Rattan

A dissertation proposal submitted to the faculty of
The University of North Carolina at Charlotte
in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in
Applied Mathematics

Charlotte

2016

Approved by:

_____

Dr. Hae-Soo Oh

_____

Dr. Shaozhong Deng

_____

Dr. Wei Cai

_____

Dr. Russell Keanini

ABSTRACT

PUJA RATTAN. Isogeometric collocation method for elasticity problems containing singularities. (Under the direction of DR. HAE-SOO OH)

Isogeometric analysis(IGA), introduced by Hughes, et al. [2,3], is a method which combines both engineering design obtained by CAD and Finite Element(FE)-analysis of the design. Prior to manufacturing, it is necessary to design the shape of the object and then analyze the durability of the design. To design complex structures, generally NURBS basis functions are used. Isogeometric analysis is effective in the design-analysis-manufacture loop.

Babuŝka and Oh [10] introduced mapping techniques called the Method of Auxiliary Mapping(MAM) to handle singularities that occur in PDEs. However, this method is unable to follow the framework of IGA. Thus, we are looking for another way to handle singularity in IGA using collocation method.

For this, we are modifying B-spline basis functions using partition unity functions. Then the neighborhood of singularity will be enriched by these modified basis functions so that they can capture the singular behavior of the true solution. In this dissertation, this method is tested to one-dimensional problems as well as 2-dimensional problems. Also, we claim that our method is more effective and economical than other existing methods in handling problems with singularities because the collocation method requires less computation than Galerkin method or any other existing method.

Schwarz alternating method in the framework of IGA-Collocation is also discussed

in this dissertation. In this method, domain is decomposed into two subdomains and then the problem is solved by solving subproblems for each subdomain. We start with some initial guess and iterate until we get a solution of desired accuracy. This technique has been applied in one- and two-dimensions for overlapping as well as non-overlapping subdomains. Elasticity problems with singularities were also solved using this method. Numerical results are presented and compared with the results obtained by IGA-Galerkin method.

DEDICATION

To my loving daughter Nyra

ACKNOWLEDGMENTS

I would like to express my deep appreciation and gratitude to my advisor, Dr. Hae-Soo Oh, for the patient guidance and mentor-ship he has provided to me continuously. He has always been very supportive and patient.

I would also like to thank my committee members, Drs. Shaozhong Deng and Wei Cai for their friendly guidance and thought provoking suggestions. I would like to thank Dr. Russell Keanini for agreeing to serve on my doctoral dissertation committee as an outside department committee member.

I would also like to thank Graduate School and Mathematics department for providing me financial support. Special thanks to Dr. Hyunju Kim who has helped me with computations at the beginning stage of my research. It would have been very difficult for me to compute numerical results if he had not helped me with the programming.

I would also like to thank my parents who have always been there for me, and came all the way from India to take care of my newborn daughter during the final stage of my PhD program.Without their blessings and support it would have been very difficult for me to achieve this dream.

Last but not the least I would like to thank my husband for the love, support, and constant encouragement he has given over the years. I undoubtedly could not have done this without his support and motivation. Finally, I would like to thank and dedicate this thesis to my daughter, Nyra who has been the most wonderful daughter anyone can have.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1: INTRODUCTION

Isogeometric analysis(IGA), introduced by Hughes, et al. [2,3], is a method which combines both engineering design obtained by CAD and Finite Element(FE)-analysis of the design. For any practical problem, it is necessary to design the shape of the object and then analyze the problem. To design complex curves, generally NURBS basis functions are used. Isogeometric analysis is effective in the design-analysis-manufacture loop.

In Chapter 2, we review definitions and terminologies that are needed to understand this paper. Readers are suggested to read books such as Rogers [8], Piegl and Tiller [9] for more information. In section 2.2, three types of refinement methods are explained with examples. Section 2.3 gives the definition of Sobolev space and norm along with formula of norms which are used to compute error in this dissertation.

In Chapter 3, the basic IGA-Galerkin and IGA-Collocation methods are briefly explained and compared with an example. In section 3.2, several methods for construction of partition of unity functions are discussed and used to enrich the region of singularity. Numerical results are shown in section 3.3.Also, the problem with oscillating singularity was tested in this section.

In Chapter 4, modification of basis functions is introduced so that C^0-continuous functions can be made C^1-continuous and then IGA-Collocation method can be used to solve the problem. In section 4.2, this method is extended to two dimen-

sions.Section 4.3 explains global basis numbering which is used for assembling local stiffness matrix for two dimensional problem. In section 4.4, this method is tested for one as well as two dimensional problems.

In Chapter 5, Schwarz alternating method in the framework of IGA-Collocation is introduced. In section 5.1, Schwarz alternating method is explained whereas in section 5.2, parallel Schwarz method is explained. Results of numerical tests done in one and two dimensions are shown in section 5.3.

In Chapter 6, we explained Schwarz alternating method for two nonoverlapping subdomains. Section 6.1 explains this method when Dirichlet-Neumann boundary condition is imposed at interface and Section 6.2 explains this method when Neumann-Neumann boundary condition is imposed at interface

In Chapter 7, Schwarz alternating iterative technique is explained for elasticity problem. In section 7.1, we first review definitions and terminologies used in linear elasticity. In section 7.2, Schwarz alternating method is used to solve coupled elliptic equations with IGA-Collocation approach. In section 7.3, this method is tested on non-singular as well as singular problems of elasticity.

Finally, the concluding remarks and future work is discussed in Chapter 8 of this dissertation.

CHAPTER 2: PRELIMINARIES

2.1 B-Splines and NURBS

In this section, we briefly review definitions and terminologies that are needed to understand this dissertation. Readers are suggested to read books such as Rogers [8], Piegl and Tiller [9] for details.

2.1.1 B-Splines

A **knot vector** $U = \{u_1, u_2, u_3, ..., u_m\}$ is a non-decreasing sequence of real numbers in the parameter space $[0, 1]$, and the components u_i for $i = 1, 2, ..., m$ are called **knots**. An **open knot vector** of order $p + 1$ is a knot vector in which the first and the last knots are repeated $p + 1$ times. The interior knots can be repeated at most p times.

$$u_1 = ... = u_{p+1} < u_{p+2} \leq ... \leq u_{m-p-1} < u_{m-p} = ... = u_m$$

There are many ways to define B-spline basis functions. Here it is defined by the recurrence formula due to Cox-de Boor. The ith **B-spline basis functions** of order $k = p + 1$ corresponding to the knot vector U are piecewise polynomials of degree p which are denoted by $N_{i,t}(u)$, is defined as

$$N_{i,1}(u) = \begin{cases} 1 & \text{if} \quad u_i \leq u < u_{i+1} \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

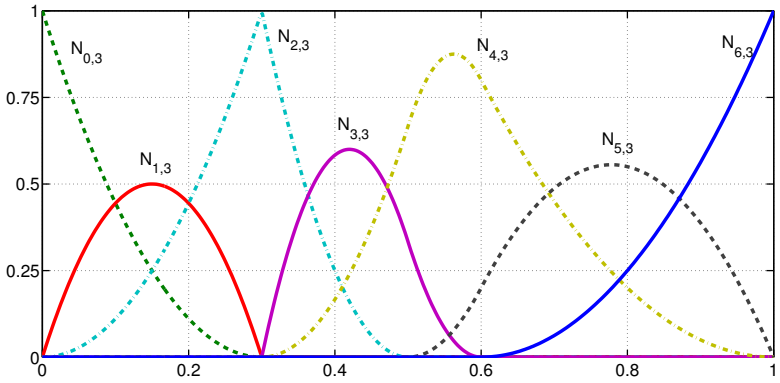


Figure 1: B-Spline functions $N_{i,3}(u), i = 1, 2, ..., 7$ of order $k = 3$ for knot vector $U = \{0, 0, 0, 0.3, 0.3, 0.5, 0.6, 1, 1, 1\}$.

$$N_{i,t}(u) = \frac{u - u_i}{u_{i+t-1} - u_i} N_{i,t-1}(u) + \frac{u_{i+t} - u}{u_{i+t} - u_{i+1}} N_{i+1,t-1}(u) \tag{2}$$

where $1 \leq i \leq m - 1$ and $2 \leq t \leq k$

B-Spline function possesses the following important properties:

1. $N_{i,k}(u)$ is non-negative for all i, k and u.

2. Each polynomial $N_{i,k}(u)$ has local support on $[u_i, u_{i+k})$.

3. On any span $[u_i, u_{i+1})$, at most $p+1$ basis functions of degree p are non-zero,i.e, $N_{i-p,k}(u), N_{i-p+1,k}(u), N_{i-p+2,k}(u), ...,$ and $N_{i,k}(u)$

4. The sum of all non-zero degree p basis functions on span $[u_i, u_{i+1})$ is 1.

5. B-Spline functions are linearly independent.

6. $N_{1,k}(0) = N_{m-1,k}(1) = 1$.

7. If the number of knots is m, the degree of the basis functions is p, then the number of basis functions is $n = m - (p + 1)$.

8. Basis function $N_{i,k}(u)$ is a composite curve of degree p polynomials with joining points at knots in $[u_i, u_{i+p+1})$.

9. At a knot of multiplicity k, basis function $N_{i,k}(u)$ is C^{p-k} continuous.

A **B-spline curve** is defined as follows:

$$C(u) = \sum_{i=1}^{m-k} N_{i,k}(u) B_i, \tag{3}$$

where B_i are control points that make B-spline functions draw a desired curve.B-spline functions corresponding to the open knot vector of order $k = n + 1$.

$$U = \{\underbrace{0, 0, 0, ..., 0}_{n+1}, \underbrace{1, 1, 1, ..., 1}_{n+1}\}$$

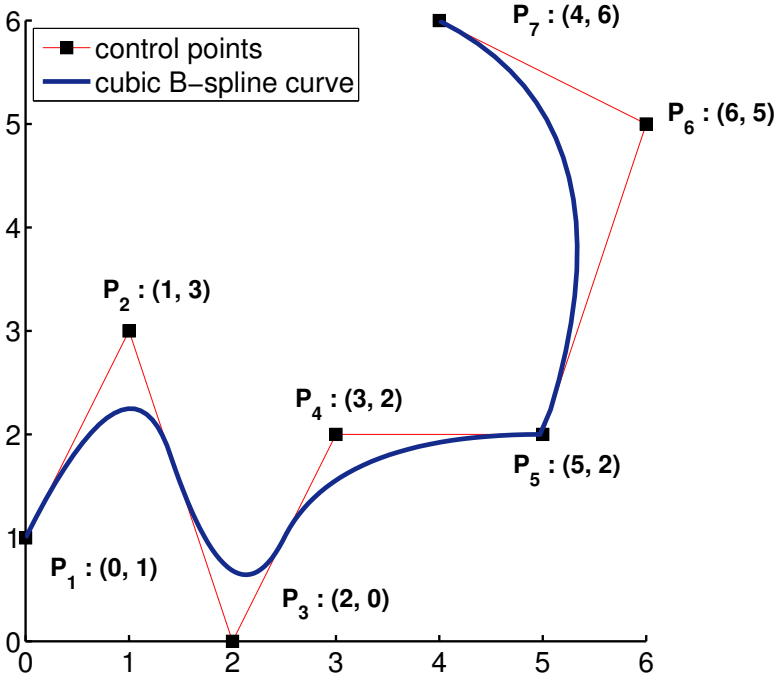
are global polynomials, called Bézier-Bernstein polynomials.The B-spline curve obtained by Bézier polynomials is called Bézier Curve. B-Spline curve possesses the following important properties:

1. A B-spline curve $C(u)$ is a piecewise curve where each piece is a curve of degree p.

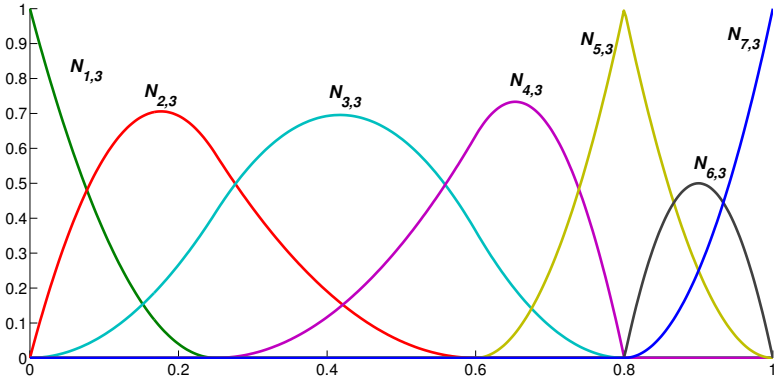
2. A B-spline curve $C(u)$ satisfies convex hull property, which means that the curve is contained in the convex hull of its control polyline. If u is in knot span $[u_i, u_{i+1})$, then $C(u)$ is in the convex hull of control points $B_{i-p}, B_{i-p+1}, ..., B_i$.

3. Changing the position of control point B_i only affects the curve $C(u)$ on interval $[u_i, u_{i+p+1})$.

4. A B-spline curve $C(u)$ is C^{p-k} continuous at a knot of multiplicity k.



(a) B-Spline curve and control points



(b) B-Spline basis functions

Figure 2: (a) B-Spline curve and control points for open knot vector $U = \{0, 0, 0, 0.25, 0.6, 0.8, 0.8, 1, 1, 1\}$. (b) B-Spline functions corresponding to the B-Spline curve shown in (a)

5. If the curve is in a plane (or space) then no straight line (or plane) can intersect a B-spline curve more than it intersects the curve's control polyline. This property is called *variation diminishing property* for B-spline curves.

6. B-spline curves also holds affine invariance property which means if an affine transformation is applied to a B-spline curve, the result can be constructed from the affine images of its control points.

If there are knot vectors $U = \{u_1, u_2, u_3, ..., u_m\}$ and $V = \{v_1, v_2, v_3, ..., v_n\}$ in u and v-direction respectively then a **B-Spline surface** is defined by

$$S(u, v) = \sum_{i=1}^{m-k} \sum_{j=1}^{n-k'} N_{i,k}(u) M_{j,k'}(v) B_{i,j}, \tag{4}$$

where $N_{i,k}(u)$ and $M_{j,k'}(v)$ are B-Spline functions of degree p and q respectively.

2.1.2 NURBS

A **Non-Uniform Rational Basis(NURBS)** function for the sets of weights $\{w_i : i = 1, ..., m - k\}$ is defined by

$$R_{i,k}(u) = \frac{N_{i,k}(u) w_i}{W(u)}, \quad W(u) = \sum_{s=1}^{m-k} N_{s,k}(u) w_s > 0. \tag{5}$$

The NURBS basis functions are piecewise rational functions which possess these properties.

1. $R_{i,k}(u)$ is non-negative for all i, k and u.

2. Each polynomial $R_{i,k}(u)$ has local support on $[u_i, u_{i+k})$.

3. On any span $[u_i, u_{i+1})$, at most $p + 1$ basis functions of degree p are non-zero

if the weights are non-negative i.e, $R_{i-p,k}(u), R_{i-p+1,k}(u), R_{i-p+2,k}(u), ...,$ and $R_{i,k}(u)$

4. The sum of all non-zero degree p basis functions on span $[u_i, u_{i+1})$ is 1.

5. NURBS basis functions are linearly independent.

6. If the number of knots is m, the degree of the basis functions is p, and the number of degree p basis functions is n, then $m = n + p + 1$.

7. Rational basis function $R_{i,k}(u)$ is a composite curve of degree p polynomials with joining points at knots in $[u_i, u_{i+p+1})$.

8. At a knot of multiplicity k, basis function $R_{i,k}(u)$ is C^{p-k} continuous.

9. If $w_i = c$ for all i, where c is a non-zero constant, $R_{i,k}(u) = N_{i,k}(u)$.

A **NURBS curve** for weights w_i and control points B_i is :

$$C(u) = \sum_{i=1}^{m-k} R_{i,k}(u) B_i. \tag{6}$$

NURBS curve possesses the following important properties:

1. A NURBS curve $C(u)$ is a piecewise curve where each piece is a rational curve of degree p.

2. A NURBS curve $C(u)$ satisfies convex hull property.

3. Changing the position of control point B_i only affects the NURBS curve $C(u)$ on interval $[u_i, u_{i+p+1})$.

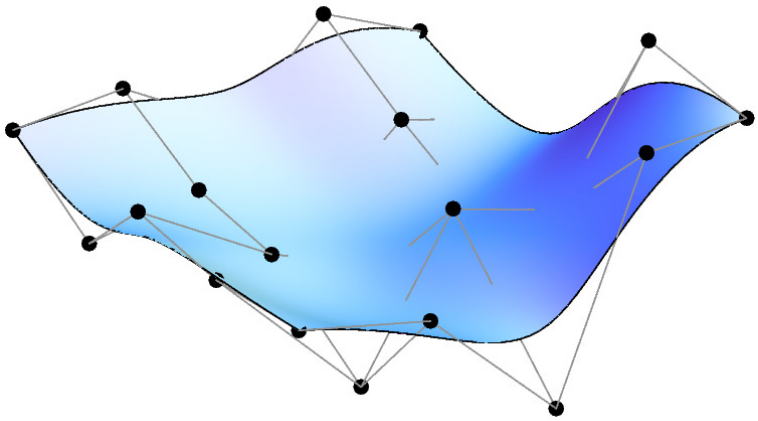


Figure 3: B-Spline surface and control net

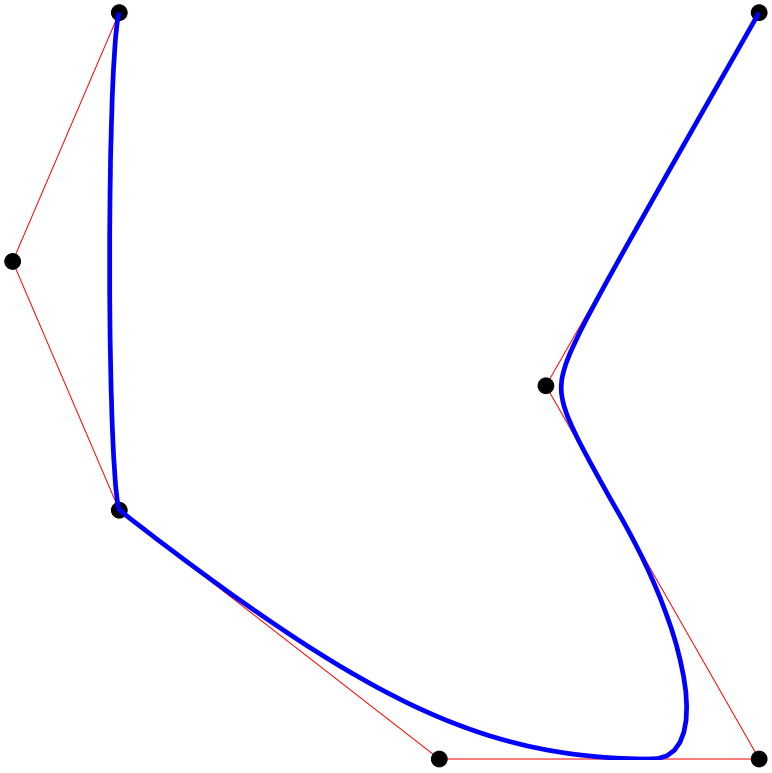


Figure 4: NURBS curve and control points

4. A NURBS curve $C(u)$ is C^{p-k} continuous at a knot of multiplicity k.

5. NURBS curves also hold variation diminishing property.

6. NURBS curve do not hold affine invariance property but they do hold projective invariance property. If the projective transformation is applied to a NURBS curve, the result can be constructed from the projective images of its control points.

If there are knot vectors $U = \{u_1, u_2, u_3, ..., u_m\}$ and $V = \{v_1, v_2, v_3, ..., v_n\}$ in u and v-direction respectively then a **NURBS surface** is defined for the given sets of weights $\{w_{i,j} : i = 1, ..., m-k, j = 1, ..., n-k'\}$

$$S(u,v) = \sum_{i=1}^{m-k} \sum_{j=1}^{n-k'} \frac{N_{i,k}(u) M_{j,k'}(v) w_{i,j} B_{i,j}}{W(u,v)}, \tag{7}$$

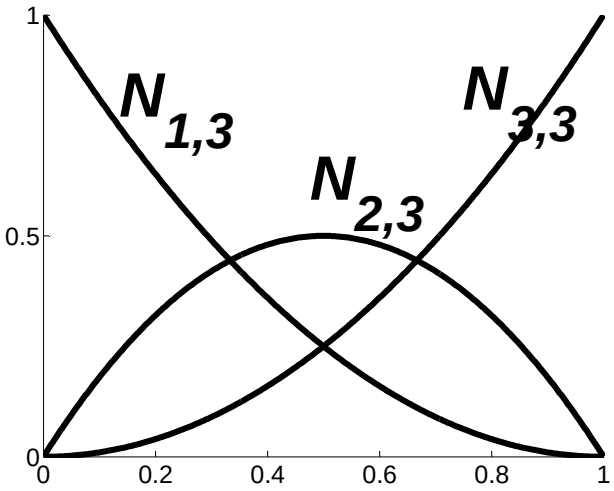
where $N_{i,k}(u)$ and $M_{j,k'}(v)$ are NURBS basis functions of degree p and q respectively.

2.2 Refinement

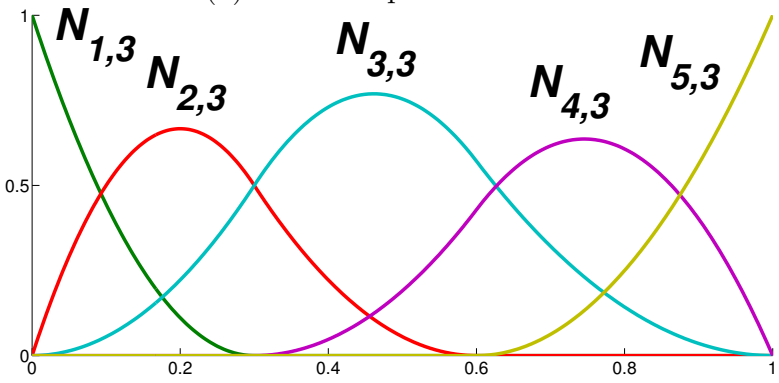
The B-spline basis can be enriched by three types of refinements. Knot insertion, degree elevation or degree and continuity elevation. Knot insertion is equivalent to h-refinement in classical FEM and degree elevation is equivalent to p-refinement in classical FEM. Degree and continuity elevation does not exist in classical FEM.

2.2.1 Knot Insertion(h-refinement)

Knot insertion meaning, adding a new knot into the existing knot vector without changing the shape of the curve. This new knot may or may not be equal to an



(a)Initial B-Spline function



(b)B-Spline function after knot insertion

Figure 5: (a)Initial B-Spline basis function with knot vector $U = \{0, 0, 0, 1, 1, 1\}$.(b)B-Spline basis function after knot insertion with knot vector $U = \{0, 0, 0, \mathbf{0.3}, \mathbf{0.6}, 1, 1, 1\}$.

existing knot. If it's equal to an existing knot then the multiplicity of that knot is increased by one.

If we have a knot vector $U = \{u_1, u_2, u_3, ..., u_m\}$ with m knots and $\{P_1, P_2, ..., P_n\}$ n control points. We want to insert a new knot t into the knot vector without changing the shape of the B-spline curve $C(u)$ then the new knot vector will be $\overline{U} = \{\overline{u_1} = u_1, \overline{u_2} = u_2, ..., \overline{u_s} = t, ..., \overline{u_{m+1}} = u_m\}$. Suppose the new knot t lies in the knot span $[u_s, u_{s+1})$, the new control points Q_i will be given by

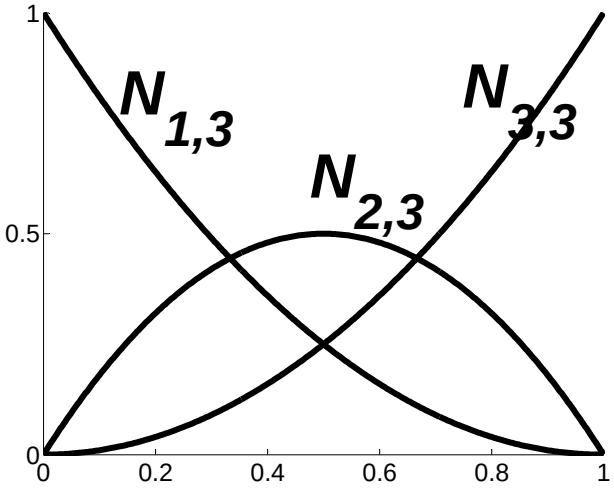
$$Q_i = (1 - a_i)P_{i-1} + a_i P_i, \tag{8}$$

where a_i can be calculated from

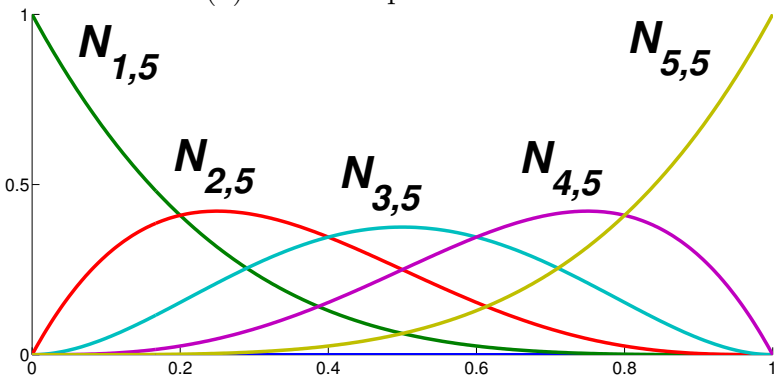
$$a_i = \begin{cases} 1 & \text{if} \quad i \leq s - p \\ \frac{t-u_i}{u_{i+p}-u_i} & \text{for} \quad s - p + 1 \leq i \leq s \\ 0 & \text{if} \quad i \geq s + 1 \end{cases} \tag{9}$$

2.2.2 Degree Elevation(p-refinement)

In this refinement we increases the degree of a curve without changing the shape of the curve. To keep the geometry and the parametrization same we also increase the multiplicity of each knot by 1 if the degree was elevated by 1. This way we can preserve the discontinuities in the various derivatives already existing in the original curve. For a surface, one can elevate degree in either the u- or the v- direction or both. If we have a knot vector $\{u_1 = ... = u_{p+1} < u_{p+2} \leq ... \leq u_{m-p-1} < u_{m-p} = ... = u_m\}$ and n control points $\{P_1, P_2, ..., P_n\}$. We want to increase the degree of the curve by 1 without changing the shape of the B-spline curve $C(u)$ then the new knot vector will



(a)Initial B-Spline function



(b)B-Spline basis function after degree elevation

Figure 6: (a)Initial B-Spline function with knot vector $U = \{0, 0, 0, 1, 1, 1\}$.(b)B-Spline basis function after degree elevation with knot vector $U = \{0, 0, 0, 0, 0, 1, 1, 1, 1, 1\}$.

be $\{u_1 = ... = u_{p+1} = u_{p+2} < u_{p+3} = u_{p+4} \leq ... \leq u_{2m-3p-4} = u_{2m-3p-3} < u_{2m-3p-2} = ... = u_{2m-2p-2}\}$ and the new control points Q_i will be given by

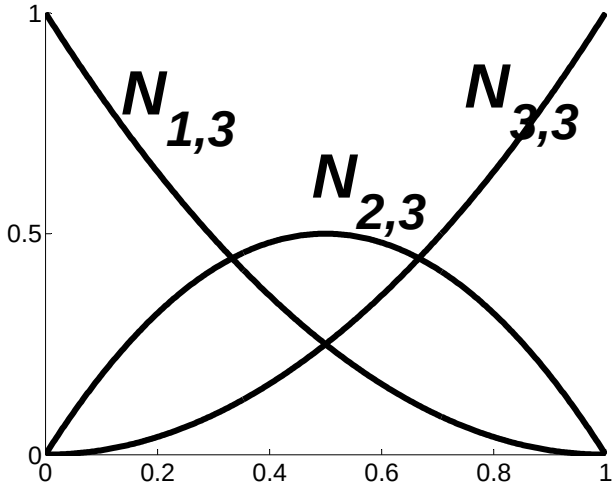
$$Q_i = \begin{cases} P_1 & \text{if} \quad i = 1 \\ \frac{(p+1-i)P_i + (i)P_{i-1}}{p+1} & \text{for} \quad 2 \leq i \leq p+1 \\ P_{p+1} & \text{if} \quad i = p+2 \end{cases} \tag{10}$$

2.2.3 k-refinement

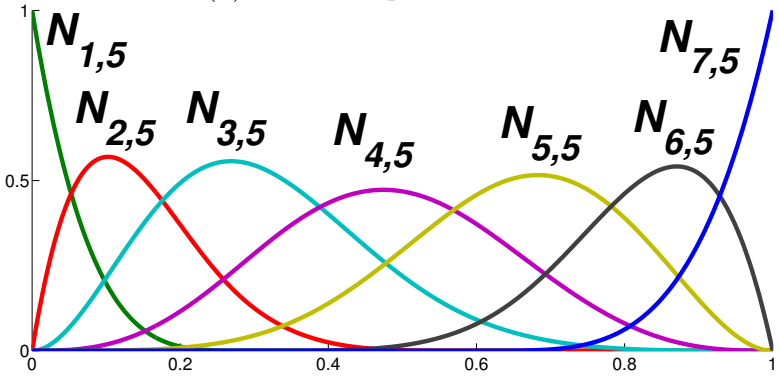
In this refinement we elevate the degree as well as we insert a new knot without changing the shape of the curve. This has no equivalent refinement in FEA. First we increase the degree of the curve and also increase the multiplicity of all intermediate knot values so the continuity of the curve does not change at these specific knots. Then we insert a new knot. Significant amount of degrees of freedom can be saved by using this refinement. These processes (Degree elevation and inserting knots) are not commutative and therefore the order in which these refinements are applied will change the final basis.

Suppose initial knot vector is $U = \{\underbrace{0, ..., 0}_{3}, 0.5, \underbrace{1, ..., 1}_{3}\}$ and we want to increase the degree from $p = 2$ to $p = 4$ and also want to insert knot $t = 0.6$. If first degree elevation occurs, we get $U = \{\underbrace{0, ..., 0}_{5}, \underbrace{0.5, ..., 0.5}_{3}, \underbrace{1, ..., 1}_{5}\}$. Initially, the regularity of the curve was C^{2-1} at knot 0.5 which remained same(C^{4-3}) after degree elevation. Now inserting new knot $U = \{\underbrace{0, ..., 0}_{5}, \underbrace{0.5, ..., 0.5}_{3}, 0.6, \underbrace{1, ..., 1}_{5}\}$ gives us total 9 basis functions.

In case we reverse the order of degree elevation and knot insertion, we get $U =$



(a)Initial B-Spline function



(b)B-Spline basis function after degree elevation

Figure 7: (a)Initial B-Spline function with knot vector $U = \{0, 0, 0, 1, 1, 1\}$.(b)B-Spline basis function after k-refinement with knot vector $U = \{0, 0, 0, 0, 0, 0.3, 0.6, 1, 1, 1, 1, 1\}$.

$\{\underbrace{0,...,0}_{3},0.5,0.6,\underbrace{1,...,1}_{3}\}$ first by knot insertion and then by increasing the degree from 2 to 4 we have $U=\{\underbrace{0,...,0}_{5},\underbrace{0.5,...,0.5}_{3},\underbrace{0.6,...,0.6}_{3},\underbrace{1,...,1}_{5}\}$ which gives total 11 basis functions. This basis is completely different than what we had in the first case. First case will have less degrees of freedom than the second case.

2.3 Sobolev Space and Norm

In PDEs we look for solutions in Sobolev space. A **Sobolev space** is a vector space of functions equipped with a norm that is a combination of L^p-norms of the function itself and its derivatives up to a given order. The Sobolev space denoted by $W^{k,p}(\Omega)$, is the collection of u defined in Ω such that for every multi-index $\alpha=(\alpha_1,...,\alpha_d)$ with $\mid\alpha\mid=\mid\alpha_1,...,\alpha_d\mid\leq k$, the weak derivative $D^\alpha u$ exists and belongs to $L^p(\Omega)$. On $W^{k,p}(\Omega)$ we shall use the norm

$$\parallel u\parallel_{W^{k,p}(\Omega)}=(\sum_{|\alpha|\leq k}\int_\Omega|D^\alpha u|^p dx)^{\frac{1}{p}}\quad\text{if}\quad 1\leq p<\infty \tag{11}$$

For a real number $p\geq 1$, the p-norm or L^p-norm of x is defined by

$$\|x\|_p=\left(|x_1|^p+|x_2|^p+\cdots+|x_n|^p\right)^{\frac{1}{p}}. \tag{12}$$

The L^∞-norm (or maximum norm) is the limit of the L^p-norms for $p\longrightarrow\infty$. It turns out that this limit is equivalent to the following definition:

$$\|x\|_\infty=\max\left\{|x_1|,|x_2|,\ldots,|x_n|\right\} \tag{13}$$

The L^2-norm (or euclidean norm) is given by

$$\|\mathbf{x}\|_2:=\left(\sum_{i=1}^{n}|x_i|^2\right)^{1/2}. \tag{14}$$

2.3.1 Weak solution in Sobolev space

Let an integer $k \geq 0$, $\Omega \subset R^d$ and $\alpha = (\alpha_1, ..., \alpha_d)$ for $u \in H^k(\Omega)$, the norm and the semi-norm, respectively, are defined by

$$\begin{aligned} &\| u \|_{k,(\Omega)} = (\textstyle\sum_{|\alpha| \leq k} \int_\Omega |D^\alpha u|^2 dx)^{\frac{1}{2}}, \\ &\| u \|_{k,\infty,(\Omega)} = \max_{|\alpha| \leq k} \{\text{ess.sup} \mid D^\alpha u(x) \mid : x \in \Omega\}; \\ &| u |_{k,(\Omega)} = (\textstyle\sum_{|\alpha| = k} \int_\Omega |D^\alpha u|^2 dx)^{\frac{1}{2}}, \\ &| u |_{k,\infty,(\Omega)} = \max_{|\alpha| = k} \{\text{ess.sup} \mid D^\alpha u(x) \mid : x \in \Omega\}. \end{aligned} \tag{15}$$

Suppose we are concerned with an elliptic boundary value problem on a domain Ω with Dirichlet boundary condition $g(x, y)$ along the boundary $\partial\Omega$. Let

$$W = \{w \in H^1(\Omega) : w|_{\partial\Omega} = g\} \quad \text{and} \quad V = \{w \in H^1(\Omega) : w|_{\partial\Omega} = 0\} \tag{16}$$

The variational formulation of the Dirichlet boundary value problem can be written as follows: Find $u \in W$ such that

$$B(u, v) = L(v), \quad \text{for all} \quad v \in V. \tag{17}$$

where B is a continuous bilinear form that is V-elliptic ([17]) and L is a linear functional on $L_2(\Omega)$. The solution to (**??**) is called a **weak solution** which is equivalent to the strong (classical) solution corresponding elliptic PDE whenever u is smooth enough.The energy norm of the trial function u is defined by

$$\|u\|_{eng} = \left[\frac{1}{2} B(u, u)\right]^{\frac{1}{2}}. \tag{18}$$

Relative error in energy norm(%) has been calculated for few problems in this disser-

tation wherever IGA-Galerkin method has been used to solve the problem.

Since the NURBS basis functions do not satisfy the Kronecker delta property, in this paper we approximate the non-homogeneous Dirichlet boundary condition by the least squares method as follows: $g^h \in W^h$ such that

$$\int_{\partial\Omega} |g - g^h|^2 d\gamma = \text{minimum}. \tag{19}$$

Throughout this dissertation, we measure the percentage relative error $\mid \frac{u - u^h}{u} \mid$ in the maximum norm(L^∞) as well as in the L^2 norm defined by:

$$\|u - u^h\|_{\infty,rel(\%)} = \frac{\|u - u^h\|_\infty}{\|u\|_\infty} \times 100, \tag{20}$$

and

$$\|u - u^h\|_{L^2,rel(\%)} = \frac{\|u - u^h\|_{L^2}}{\|u\|_{L^2}} \times 100. \tag{21}$$

CHAPTER 3: ENRICHED IGA-COLLOCATION

3.1 Isogeometric Analysis(IGA)

Throughout this dissertation, we use Collocation method and Galerkin method in the framework of IGA for numerical solutions of PDEs. Thus we present basic Galerkin and Collocation approximation methods.

3.1.1 IGA-Galerkin Method

Consider the following two-dimensional model problem

$$\begin{cases} -\Delta u = f & \text{in} \quad \Omega, \\ u = 0 & \text{on} \quad \partial\Omega. \end{cases} \tag{22}$$

where $f \in L^2(\Omega)$. Ω is a bounded connected open subset of R^2 whose boundary $\partial\Omega$ is Lipschitz continuous.

Using Green's theorem we obtain the variational form of model problem (**??**) as follows

$$\iint_{\Omega} (\nabla u)^T \nabla v d\Omega = \iint_{\Omega} f v d\Omega, \quad \text{for all} \quad v \in H_0^1(\Omega). \tag{23}$$

Suppose V_h is a finite dimensional subspace of $H_0^1(\Omega)$. Then the Galerkin approximation of (**??**) is to find $u_h \in V_h$ such that

$$\iint_{\Omega} (\nabla u_h)^T \nabla v d\Omega = \iint_{\Omega} f v d\Omega, \quad \text{for all} \quad v \in V_h. \tag{24}$$

Suppose the collection $\{\phi_1, \phi_2, ..., \phi_N\}$ is a basis for V_h then since $u_h \in V_h$ we have

$$u(x, y) \approx u_h(x, y) = \sum_{i=1}^{N} c_i \phi_i(x, y), \tag{25}$$

for some constants $\{c_1, c_2, ..., c_N\}$. Substituting (**??**) into (**??**) we have the following linear system for the unknown $\{c_1, c_2, ..., c_N\}$:

$$\sum_{j=1}^{N} c_j \iint_{\Omega} (\nabla \phi_i)^T \nabla \phi_j d\Omega = \iint_{\Omega} f \phi_i d\Omega, \quad \text{for} \quad i = 1, 2, ..., N. \tag{26}$$

Let,

$$\begin{cases} \iint_{\Omega} (\nabla \phi_i)^T \nabla \phi_j = a_{ij} \\ \iint_{\Omega} f \phi_i = b_i \end{cases} \tag{27}$$

then the corresponding matrix equation for the unknown $\{c_1, c_2, ..., c_N\}$ is

$$\begin{pmatrix} \iint_{\Omega} (\nabla \phi_1)^T \nabla \phi_1 & \iint_{\Omega} (\nabla \phi_1)^T \nabla \phi_2 & \dots & \iint_{\Omega} (\nabla \phi_1)^T \nabla \phi_N \\ \iint_{\Omega} (\nabla \phi_2)^T \nabla \phi_1 & \iint_{\Omega} (\nabla \phi_2)^T \nabla \phi_2 & \dots & \iint_{\Omega} (\nabla \phi_2)^T \nabla \phi_N \\ \vdots & \vdots & \dots & \vdots \\ \iint_{\Omega} (\nabla \phi_N)^T \nabla \phi_1 & \iint_{\Omega} (\nabla \phi_N)^T \nabla \phi_2 & \dots & \iint_{\Omega} (\nabla \phi_N)^T \nabla \phi_N \end{pmatrix} \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_N \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_N \end{pmatrix} \tag{28}$$

By solving (**??**) we obtain galerkin approximate solution of (**??**) given by (**??**). When NURBS basis functions are used for galerkin approximation, it is called *IGA-Galerkin method.*

3.1.2 IGA-Collocation Method

Suppose the right hand function $f(x, y)$ in (**??**) is continuous and $p_i = (x_i, y_i)$ is a point in $\Omega \subset \mathbf{R^2}$. For brevity, we write $x = (x, y), x_i = (x_i, y_i)$.If we use Dirac δ

function $\delta(x - x_i)$ as a test function

$$\iint_\Omega (-\Delta u)\delta(x - x_i) = \iint_\Omega f\delta(x - x_i). \tag{29}$$

Then from the **shifting property** of delta function we have,

$$(-\Delta u)(x_i) = f(x_i). \tag{30}$$

Suppose the basis functions $\{\phi_1, \phi_2, ..., \phi_N\}$ are $\mathcal{C}^1$-continuous then by properly choosing N distinct points $\{p_1, p_2, ..., p_N\}$ in Ω, we obtain a system of linear equation:

$$-\Delta(\sum_{i=1}^{N} c_i\phi_i)(p_i) = f(p_i) \quad \text{for} \quad i = 1, 2, ..., N. \tag{31}$$

or,

$$-\sum_{i=1}^{N} c_i(\Delta\phi_i)(p_i) = f(p_i) \quad \text{for} \quad i = 1, 2, ..., N. \tag{32}$$

By solving the system in (**??**), one can determine the unknown coefficients $c_i, i = 1, 2, ..., N$. This method is called the **collocation approximation method**.

The collocation method using $\mathcal{C}^1$-continuous NURBS basis functions will be called *IGA-Collocation* . Even though the collocation method has many advantages over Galerkin method, the method have not widely been employed because of the complexity of constructions of $\mathcal{C}^1$-basis functions.

However, since highly smooth basis functions are used in IGA of numerical solutions of PDEs, the collocation method start to draw attentions. The success of the collocation method depends on not only constructing $\mathcal{C}^1$-continuous basis function, but the proper choice of collocation points. Commonly used collocation points are

Greville abscissae of the knot vectors and *Gaussian quadrature* points.

Greville abscissae- The Greville abscissae $\bar{u}_i$ for the knot vector $U = \{u_1, u_2, u_3, ..., u_m\}$ can be found by the formula

$$\bar{u}_i = \frac{u_{i+1} + u_{i+2} + ... + u_{i+p}}{p}, 1 \leq i \leq m - k. \tag{33}$$

For example, the Greville abscissae of a knot vector $U = \{0, 0, 0, \frac{1}{2}, 1, 1, 1\}$ are

$$\bar{u}_1 = 0, \quad \bar{u}_2 = \frac{1}{4}, \quad \bar{u}_3 = \frac{3}{4}, \quad \bar{u}_4 = 1.$$

Gaussian quadrature points-In numerical analysis, the quadrature rule is an approximation of the definite integral of a function, usually stated as a weighted sum of function values at specified points within the domain of integration. For the domain $[-1, 1]$ the rule is stated as

$$\int_{-1}^{1} f(x)dx \approx \sum_{i=1}^{n} w_i f(x_i) \tag{34}$$

where w_i's,

$$w_i = \frac{2}{(1 - x_i)^2[P_n'(x_i)]^2} \tag{35}$$

are the weights for Gauss-Legendre quadrature and x_i is the i-th root of Legendre polynomial $P_n(x)$, where by Rodrigue formula, $P_n(x)$ is defined by

$$P_n(x) = \frac{1}{2^n n!} \frac{d^n}{dx^n}[(x^2 - 1)^n]. \tag{36}$$

These quadrature points are another choice for collocation points.

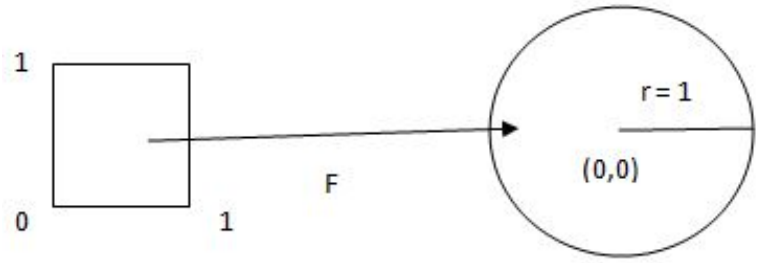


Figure 8: Mapping F maps $\widehat{\Omega} = [0,1] \times [0,1]$ to physical space is $\Omega = [(r,\theta) : r < 1, 0 < \theta < 2\pi]$ with crack along the positive x-axis

3.1.3 Mapping Techniques in IGA-Collocation and IGA-Galerkin Methods

In order to show advantage of using IGA-Collocation over IGA-Galerkin , both methods are applied to the same elliptic boundary value problem with singularity of type

$$r^{\lambda}\psi(\theta), \quad \text{where} \quad 0 < \lambda < 1, \text{and} \quad \psi \text{ is a smooth function.} \tag{37}$$

The problem discussed here has a singularity of type $r^{\frac{1}{2}}$ on a cracked circular domain of radius 1 and centered at origin.

$$\begin{cases} -\Delta u = f \quad \text{in} \quad \Omega = [(r,\theta) : r < 1, 0 < \theta < 2\pi] \\ u = 0 \quad \text{on} \quad \partial\Omega \end{cases} \tag{38}$$

which has the exact solution:

$$u(r,\theta) = \sqrt{r}(1-r)\left[\sin(\frac{\theta}{2}) + \sin(\frac{3\theta}{2})\right]. \tag{39}$$

Let F be a smooth mapping from the parameter space $\widehat{\Omega} = [0,1] \times [0,1]$ onto the

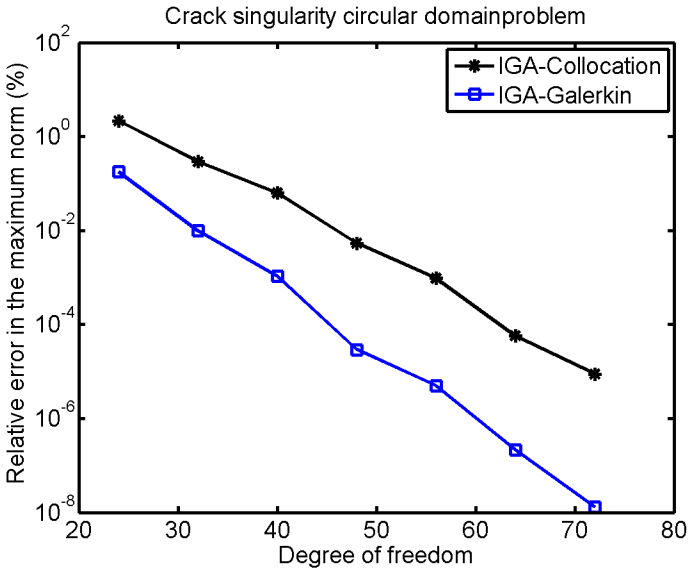


Figure 9: Relative error in the max-norm in percentage for cracked domain problem using IGA-Collocation and IGA-Galerkin methods

physical space $\Omega = [(r, \theta) : r < 1, 0 < \theta < 2\pi]$ with crack along the positive x-axis, which is defined as follows:

$$F : \hat{\Omega} \to \Omega \quad \text{and} \quad F(u, v) = (x(u, v), y(u, v)),$$

where

$$F(u, v) = \begin{cases} x(u, v) = v^2 \cos(2\pi(1 - u)) \\ y(u, v) = v^2 \sin(2\pi(1 - u)). \end{cases} \tag{40}$$

This construction of mapping F generates singular functions. For IGA-Galerkin, the basis functions should be at least $\mathcal{C}^0$-continuous but for IGA-Collocation they have to be at least $\mathcal{C}^1$-continuous basis function. Therefore we started with B-spline basis

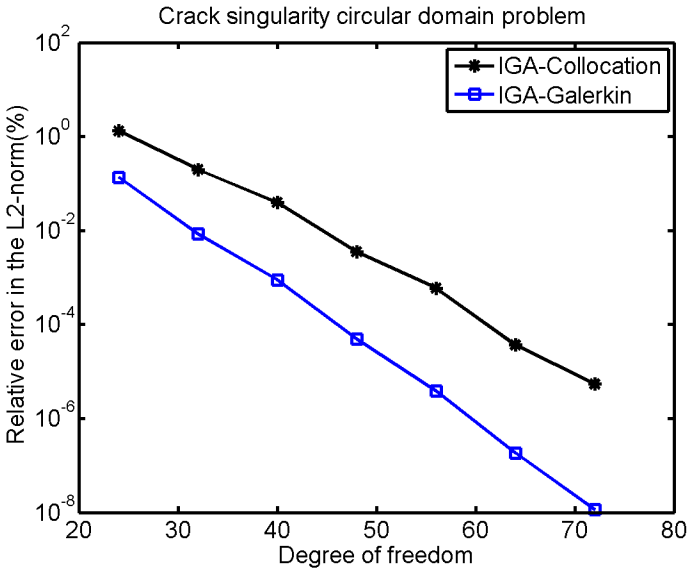


Figure 10: L^2-Norm in percentage for cracked domain problem using IGA-Collocation and IGA-Galerkin methods

Table 1: Comparison of relative error in the maximum norm(%) for IGA-Galerkin and IGA-Collocation methods for crack singularity problem. (p_u, p_v) are degrees of B-spline functions.

(p_u, p_v)	dof	IGA-Colloaction	IGA-Galerkin
$(4,3)$	24	2.132E+00	1.756E-01
$(5,3)$	32	2.916E-01	9.824E-03
$(6,3)$	40	6.304E-02	1.062E-03
$(7,3)$	48	5.360E-03	2.896E-05
$(8,3)$	56	9.636E-04	4.976E-06
$(9,3)$	64	5.731E-05	2.162E-07
$(10,3)$	72	8.824E-06	1.319E-08

functions corresponding to

$$\begin{aligned} U &= \{\underbrace{0,...,0}_{5}, \tfrac{1}{4}, \tfrac{1}{4}, \tfrac{1}{4}, \tfrac{1}{2}, \tfrac{1}{2}, \tfrac{1}{2}, \tfrac{3}{4}, \tfrac{3}{4}, \tfrac{3}{4}, \underbrace{1,...,1}_{5}\} \\ V &= \{\underbrace{0,...,0}_{4}, \underbrace{1,...,1}_{4}\} \end{aligned} \tag{41}$$

in u- direction and v-direction respectively. To improve the isogeometric analysis of (**??**) in the angular direction we elevate the degree of B-spline functions with the fixed mesh size $h = \frac{1}{4}$ (the p-refinement).

Table 2: Comparison of relative error in the L^2-norm(%) for IGA-Galerkin and IGA-Collocation methods for crack singularity problem

(p_u, p_v)	dof	IGA-Colloaction	IGA-Galerkin
$(4, 3)$	24	1.314E+00	1.353E-01
$(5, 3)$	32	1.986E-01	8.378E-03
$(6, 3)$	40	3.939E-02	8.915E-04
$(7, 3)$	48	3.513E-03	4.922E-0-5
$(8, 3)$	56	5.945E-04	3.863E-06
$(9, 3)$	64	3.677E-05	1.849E-07
$(10, 3)$	72	5.403E-06	1.147E-08

Table 3: Comparison of computing time(in seconds) for IGA-Galerkin and IGA-Collocation methods for crack singularity problem

(p_u, p_v)	dof	IGA-Colloaction	IGA-Galerkin
$(4, 3)$	24	2.028	24.913
$(5, 3)$	32	3.042	49.561
$(6, 3)$	40	4.726	61.448
$(7, 3)$	48	7.456	142.833
$(8, 3)$	56	10.311	258.539
$(9, 3)$	64	12.947	335.385
$(10, 3)$	72	20.732	635.672

The relative error in the maximum norm(%) for both IGA-Collocation and IGA-Galerkin are listed in Table **??** and the relative error in the L^2-norm(%) for both methods is listed in Table **??**.

Table **??** and Fig. **??** shows the computing time for IGA-Galerkin and IGA-Collocation methods. We can see that as we elevate degree of basis function, time taken for IGA-Collocation increases linearly but for IGA-Galerkin it increases almost quadtratically. Therefore, IGA-Collocation method has advantage over IGA-Galerkin when we are dealing with large scale problems even though the accuracy of solution is less than one order of magnitude than the IGA-Galerkin . It saves lot of time and

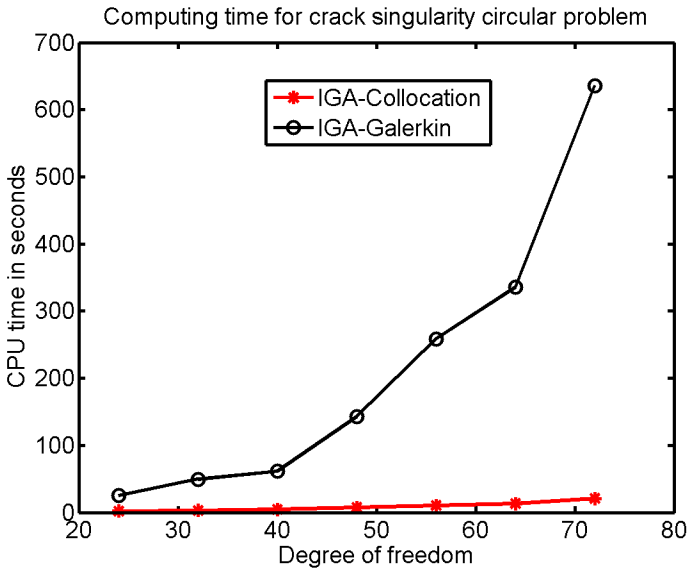


Figure 11: Computing time(in seconds) for IGA-Collocation and IGA-Galerkin

money in computation.

3.2 Partition of Unity(PU) Functions

In literature, several methods for constructions of partition of unity(PU) functions are suggested. In this section, we are going to discuss various types of PU functions applicable to implement in our method. For this purpose, let us first introduce the notations and definitions.

Let $\Omega \subseteq \mathbf{R^d}$. For $m \geq 0$, $C^m(\Omega)$ denotes the space of all functions ϕ with continuous derivatives upto order m. The support of ϕ is defined by

$$\text{supp } \phi = \overline{\{x \in \Omega : \phi(x) \neq 0\}}$$

A family $\{U_k : k \in \mathbf{D}\}$ of open subsets of $\mathbf{R^d}$ is said to be a point finite open covering of $\Omega \subseteq \mathbf{R^d}$ if there is M such that any $x \in \Omega$ lies in at most M of the open sets U_k and $\Omega \subseteq \bigcup_k U_k$.

For a point finite open covering $\{U_k : k \in \mathbf{D}\}$ of a domain Ω, suppose there is a

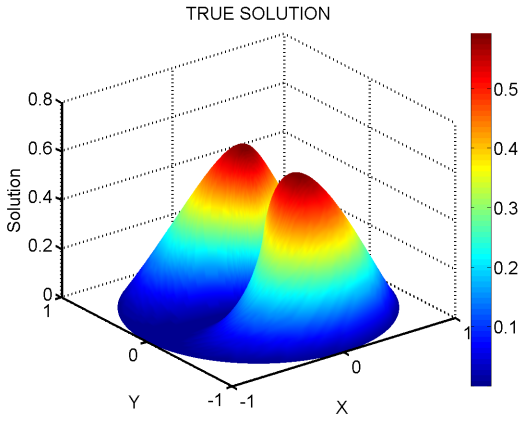


Figure 12: The true Solution given by (**??**)

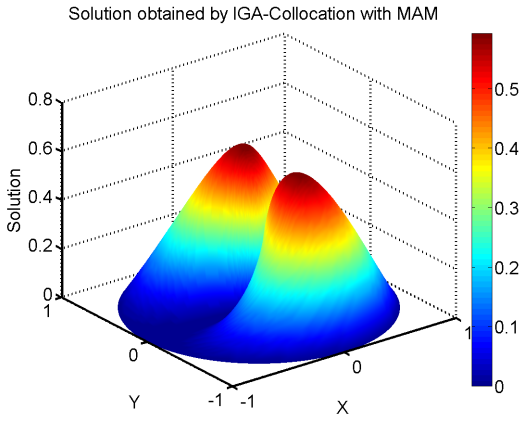


Figure 13: Numerical solutions obtained by IGA-Collocation with mapping technique

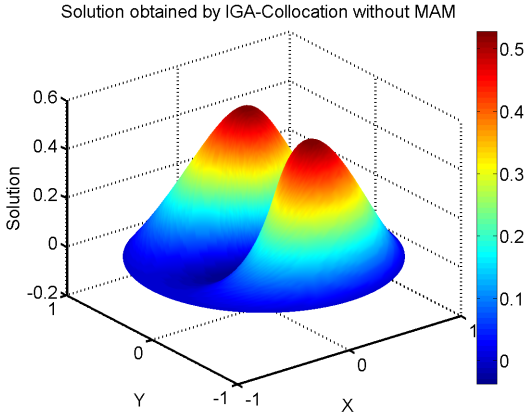


Figure 14: Numerical solutions obtained by IGA-Collocation without mapping technique

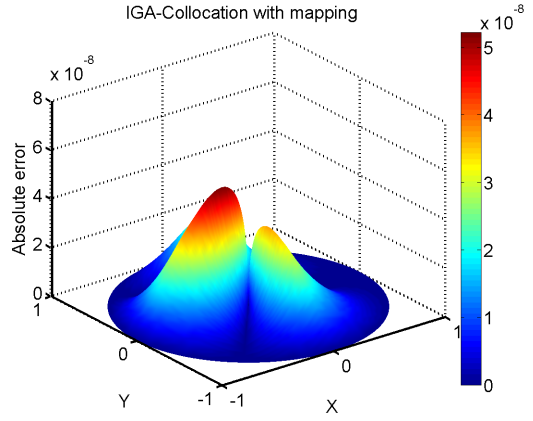


Figure 15: Absolute error obtained by IGA-Collocation with mapping technique

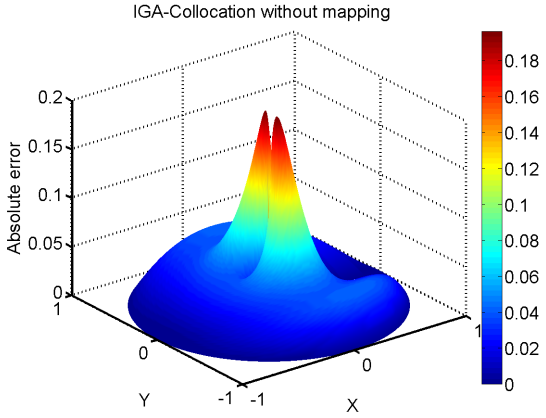


Figure 16: Absolute error obtained by IGA-Collocation without mapping technique

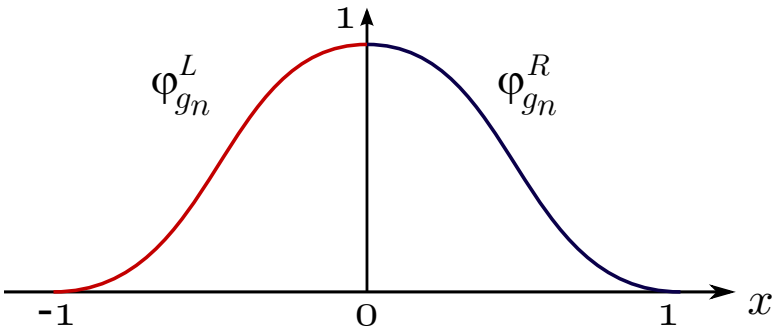


Figure 17: One-dimensional non-flat top PU function

family $\{\psi_k : k \in \mathbf{D}\}$ of Lipschitz functions on Ω satisfying the following conditions:

1.For $k \in \mathbf{D}, 0 \leq \psi_k(x) \leq 1, x \in \mathbf{R^d}$.

2. The supp$(\psi_k) \subseteq \overline{U_k}$, for each $k \in \mathbf{D}$.

3. $\sum_{k\in\mathbf{D}} \psi_k(x) = 1$ for each $x \in \Omega$.

Then $\{\psi_k : k \in \mathbf{D}\}$ is called a **partition of unity (PU)** subordinate to the covering $\{U_k : k \in \mathbf{D}\}$. The covering sets U_k are called **patches**. A **window(or weight) function** is a non-negative continuous function with compact support and is denoted by $w(x)$. We consider the following conical window function in this paper: For $x \in \mathbf{R}$,

$$w(x) = \begin{cases} (1-x^2)^l & \text{if} \quad \mid x \mid \leq 1, \\ 0 & \text{if} \quad \mid x \mid > 1 \end{cases} \tag{42}$$

where l is a positive integer. Then $w(x)$ is a C^{l-1} function and it can be constructed from a one dimensional weight function as $w(x) = \prod_{i=1}^{d} w(x_i)$, where $x = (x_1, x_2, ..., x_d)$. **Normalized window functions** are defined by

$$w^l_\delta(x) = Aw(\frac{x}{\delta}) \tag{43}$$

where $A = \dfrac{(2l+1)!}{2^{2l+1}(l!)^2\delta}$ is a constant that gives $\int_{\mathbf{R}} w^l_\delta(x)dx = 1$.

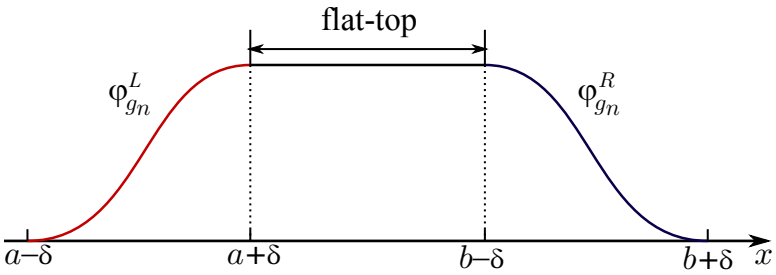


Figure 18: One-dimensional flat top PU function

1- **Shepard PU shape functions:**

Suppose window function is built at every particle x_i for each patch $w_i, i = 1, 2, ..., N$. Then the PU functions $\varphi_i(x)$ associated with particle $x_i, i = 1, 2, ..., N$ is defied by

$$\varphi_i(x) = \frac{w_i(x - x_i)}{\sum_k w_k(x - x_i)}, \quad \text{for all} \quad x \in R \tag{44}$$

2- **One-dimensional non-flat top PU functions[6]:**

This PU function is constructed by C^{n-1} piecewise polynomial $\varphi_n(x)$ for any integer $n \geq 1$

$$\varphi_n(x) = \begin{cases} {\varphi^L}_n(x) := (1+x)^n g_n(x) & \text{if} \quad x \in [-1, 0] \\ {\varphi^R}_n(x) := (1-x)^n g_n(-x) & \text{if} \quad x \in [0, 1] \\ 0 & \text{otherwise} \end{cases} \tag{45}$$

where $g_n(x) = a_{0,n} + a_{1,n}(-x) + a_{2,n}(-x)^2 + ... + a_{n-1,n}(-x)^{n-1}$ is a polynomial of degree $n - 1$. The coefficients $a_{k,n}$'s are defined by

$$a_{k,n}(x) = \begin{cases} 1 & \text{if} \quad k = 0 \\ \sum_{i=0}^{k} a_{i,n-1} & \text{if} \quad 0 < k \leq n-2 \\ 2a_{n-2,n} & \text{if} \quad k = n-1. \end{cases} \tag{46}$$

Using above recurrence formula we get $g_n(x)$ as follows:

$$\begin{cases} g_1(x) = 1 \\ g_2(x) = 1 - 2x \\ g_3(x) = 1 - 3x + 6x^2 \\ g_4(x) = 1 - 4x + 10x^2 - x^3 \\ g_5(x) = 1 - 5x + 15x^2 - 35x^3 + 70x^4 \\ ..... \end{cases} \tag{47}$$

and so on. Since $\varphi_n(x)$ depends on both $(1+x)^n$ and $g_n(x)$ therefore it's $\mathcal{C}^{n-1}$ continuous.

3- **One-dimensional convolution flat-top PU functions[6]:**

Suppose domain $\Omega = [a, b]$ is partitioned uniformly (or non-uniformly) such that

$$x_1 = a - \delta < a < x_2 < ... < x_n < b < x_{n+1} = b + \delta \tag{48}$$

Using non-flat PU functions we can construct PU functions with flat top whose support is $[a - \delta, b + \delta]$ with $(a + \delta) < b - \delta$ in the following way:

$$\psi_{[a,b]}^{(\delta,n-1)}(x) = \begin{cases} \varphi_n^L(\frac{x-(a+\delta)}{2\delta}) & \text{if} \quad x \in [a-\delta, a+\delta] \\ 1 & \text{if} \quad x \in [a+\delta, b-\delta] \\ \varphi_n^R(\frac{x-(b-\delta)}{2\delta}) & \text{if} \quad x \in [b-\delta, b+\delta] \\ 0 & \text{if} \quad x \notin [a-\delta, b+\delta] \end{cases} \tag{49}$$

where φ_n^L and φ_n^R are defined by (**??**).

Here, in order to make a PU function have a flat-top, we assume $\delta \leq \frac{(b-a)}{3}$. Actually, $\psi_{[a,b]}^{(\delta,n-1)}(x)$ is the convolution, $\chi_{Q_k}(x) * w_\delta^{n-1}$, of the characteristic

function $\chi_{Q_k}(x)$ and the scaled window function w_δ^{n-1}, defined by (**??**). Let $Q_k = [x_k, x_{k+1}]$ be an interval with $\mid x_{k+1} - x_k \mid \geq 3\delta$ for $k = 1, 2, ..., n$ then the characteristic function $\chi_{Q_k}(x)$ is defined by

$$\chi_{Q_k}(x) = \begin{cases} 1 & \text{if} \quad x \in [x_k, x_{k+1}], \\ 0 & \text{if} \quad x \notin [x_k, x_{k+1}] \end{cases} \tag{50}$$

Since $\sum_{k=1}^{n} \chi_{Q_k}(x) = 1$, for all $x \in \Omega$ except for nodal points which implies $\sum_{k=1}^{n} \psi_k^{(\delta,n-1)}(x) = 1$ for all $x \in \Omega$.

4- **Two-dimensional flat-top PU functions using B-Splines:**

Suppose

$$N_{i,k}(u), 1 \leq i \leq m - k$$

and

$$M_{j,k'}(v), 1 \leq j \leq n - k'$$

are B-spline functions corresponding to open knot vectors

$$U = \{u_1, u_2, u_3, ..., u_m\}$$

and

$$V = \{v_1, v_2, v_3, ..., v_n\}$$

in u-direction and in v-direction respectively. Supports of B-spline functions $N_{i,k}(u)$ and $M_{j,k'}(v)$ are $[u_i, u_{i+k})$ and $[v_j, v_{j+k'})$ respectively. We can construct PU functions with flat top $\phi_i(u)$ and $\phi_j(v)$ using B-spline functions $N_{i,k}(u)$ and

$M_{j,k'}(v)$ in the following way. Let us consider knot vector

$$U = \{\underbrace{0, ..., 0}_{4}, 0.35, 0.35, 0.4, 0.4, 0.6, 0.6, 0.65, 0.65, \underbrace{1, ..., 1}_{4}\}$$

This knot vector will have twelve cubic B-Spline functions in $u-$direction with following support:

$$\begin{aligned}
\text{Supp}(N_{1,4}(u)) = [0, 0.35], \quad \text{Supp}(N_{2,4}(u)) = [0, 0.35], \\
\text{Supp}(N_{3,4}(u)) = [0, 0.4], \quad \text{Supp}(N_{4,4}(u)) = [0, 0.4], \\
\text{Supp}(N_{5,4}(u)) = [0.35, 0.6], \quad \text{Supp}(N_{6,4}(u)) = [0.35, 0.6], \\
\text{Supp}(N_{7,4}(u)) = [0.4, 0.65], \quad \text{Supp}(N_{8,4}(u)) = [0.4, 0.65], \\
\text{Supp}(N_{9,4}(u)) = [0.6, 1], \quad \text{Supp}(N_{10,4}(u)) = [0.6, 1], \\
\text{Supp}(N_{11,4}(u)) = [0.65, 1], \quad \text{Supp}(N_{12,4}(u)) = [0.65, 1]
\end{aligned}$$

We will construct $\phi_i(u)$ function using middle section of B-Spline fucntions

$$\phi_i(u) = N_{5,4}(u) + N_{6,4}(u) + N_{7,4}(u) + N_{8,4}(u) = 1 \quad \text{if} \quad x \in [0.4, 0.6] \tag{51}$$

Hence,

$$\phi_i(u) = \begin{cases} N_{5,4}(u) + N_{6,4}(u) & \text{if} \quad x \in [0.35, 0.4), \\ 1 & \text{if} \quad x \in [0.4, 0.6], \\ N_{7,4}(u) + N_{8,4}(u) & \text{if} \quad x \in (0.6, 0.65], \\ 0 & \text{otherwise.} \end{cases} \tag{52}$$

Similarly for knot vector $V = \{\underbrace{0, ..., 0}_{4}, 0.2, 0.2, 0.3, 0.3, \underbrace{1, ..., 1}_{4}\}$ there will be eight cubic B-Spline functions in $v-$direction with following support:

$$\text{Supp}(M_{1,4}(u)) = [0, 0.2], \quad \text{Supp}(M_{2,4}(u)) = [0, 0.2],$$

$$\text{Supp}(M_{3,4}(u)) = [0, 0.3], \quad \text{Supp}(M_{4,4}(u)) = [0, 0.3],$$
$$\text{Supp}(M_{5,4}(u)) = [0.2, 1], \quad \text{Supp}(M_{6,4}(u)) = [0.2, 1],$$
$$\text{Supp}(M_{7,4}(u)) = [0.3, 1], \quad \text{Supp}(M_{8,4}(u)) = [0.3, 1].$$

we will construct $\phi_j(v)$ function using first section of B-Spline functions

$$\phi_j(v) = M_{1,4}(v) + M_{2,4}(v) + M_{3,4}(v) + M_{4,4}(v) = 1 \quad \text{if} \quad x \in [0, 0.2] \tag{53}$$

Hence,

$$\phi_j(v) = \begin{cases} 1 & \text{if} \quad x \in [0, 0.2], \\ M_{3,4}(v) + M_{4,4}(v) & \text{if} \quad x \in (0.2, 0.3), \\ 0 & \text{if} \quad x \in [0.3, 1]. \end{cases} \tag{54}$$

We can construct two-dimensional flat top PU function by taking tensor product of $\phi_i(u)$ and $\phi_j(v)$ functions.

$$\psi_{i,j}(u, v) = \phi_i(u) \times \phi_j(v) = \sum_{i=5}^{8} \sum_{j=1}^{4} N_{i,4}(u) M_{j,4}(v) \tag{55}$$

It's not difficult to see that $\psi_{i,j}(u, v)$ is a unit function on the rectangle $[0.4, 0.6] \times [0, 0.2]$.

3.3 Numerical Results

To show the effectiveness of our method we test enrichment method for nonsingular as well as for singular problems.

3.3.1 An elliptic equation with smooth solution

Consider second order boundary value problem containing no singularity.

$$\begin{cases} -u''(x) = f & \text{for} \quad x \in (0,1) \\ u(0) = u(1) = 0 \end{cases} \tag{56}$$

which has the exact solution:

$$u(x) = x^2(x-1) \tag{57}$$

We considered knot vector

$$U = \{0, 0, 0, 0.45, 0.5, 0.55, 1, 1, 1\}$$

for construction of $\mathcal{C}^\infty$-continuous PU functions with flat top. This knot vector will generate six quadratic B-Spline functions with following supports:

$$\begin{aligned} &\text{Supp}(N_{1,3}(u)) = [0, 0.45], \quad \text{Supp}(N_{2,3}(u)) = [0, 0.5], \\ &\text{Supp}(N_{3,3}(u)) = [0, 0.55], \quad \text{Supp}(N_{4,3}(u)) = [0.45, 1], \\ &\text{Supp}(N_{5,3}(u)) = [0.5, 1], \quad \text{Supp}(N_{6,3}(u)) = [0.55, 1]. \end{aligned}$$

Let $\phi_1(u)$ and $\phi_2(u)$ be $\mathcal{C}^\in$-continuous PU function constructed by B-spline functions as follows:

$$\phi_1^L(u) = \begin{cases} 1 & \text{if} \quad x \in [0, 0.45] \\ N_{1,3}(u) + N_{2,3}(u) + N_{3,3}(u) & \text{if} \quad x \in [0.45, 0.55] \\ 0 & x \in [0.55, 1] \end{cases} \tag{58}$$

Table 4: The relative errors in the maximum norm in percentage for numerical solutions of one dimensional non-singular problem obtained by enriched IGA-Collocation

Degree of 1st segment	Degree of 2nd segment	IGA-Collocation
$p = 6$	$p = 3$	2.01E-15
$p = 6$	$p = 6$	8.43E-16
$p = 10$	$p = 7$	1.59E-15

and,

$$\phi_2^R(u) = \begin{cases} 0 & \text{if} \quad x \in [0, 0.45] \\ N_{4,3}(u) + N_{5,3}(u) + N_{6,3}(u) & \text{if} \quad x \in [0.45, 0.55] \\ 1 & x \in [0.55, 1]. \end{cases} \tag{59}$$

These $\phi_1(u)$ and $\phi_2(u)$ are flat top PU-functions with non flat-top on $[0.45, 0.55]$. Let

$$B_k(\xi) = nC_k(1-\xi)^{n-k}\xi^k, k = 0, 1, 2, ...n$$

be Bernstein polynomials(Bézier functions) of degree n. Let $T_1 : [0, 1] \longrightarrow [0, 0.55]$ and $T_2 : [0, 1] \longrightarrow [0.45, 1]$ be bijective linear mapping. We construct $\mathcal{C}^2$-continuous basis functions on $[0, 0.55]$ and $[0.45, 1]$ as follows:

$$\mathcal{V}_1 = \{B_k(T_1^{-1}(x)) \times \phi_1(x) | k = 1, 2, ..., n_1\} \tag{60}$$

$$\mathcal{V}_2 = \{B_k(T_2^{-1}(x)) \times \phi_2(x) | k = 1, 2, ..., n_2\}. \tag{61}$$

We define approximation space $\mathcal{V}$ on $[0, 1]$ by

$$\mathcal{V} = \text{span}(\mathcal{V}_1 \cup \mathcal{V}_2)$$

. Relative errors in the maximum norm in percentage with respect to various combinations of p-degree are shown in Table **??** incase where true solution is smooth.

Table 5: The relative errors in the maximum norm in percentage for numerical solutions of one dimensional problem containing singularity obtained by enriched IGA-Collocation

Degree of 1st segment	Degree of 2nd segment	IGA-Collocation
$p = 6$	$p = 3$	5.10E-2
$p = 6$	$p = 6$	7.93E-3
$p = 10$	$p = 7$	1.46E-3

3.3.2 Problem with Monotone Singularity of type x^λ

Consider a model second order boundary value problem with singularity,

$$\begin{cases} -u''(x) = f \qquad \text{for} \quad x \in (0,1) \\ u(0) = u(1) = 0 \end{cases} \tag{62}$$

which has the exact solution:

$$u(x) = x^{1.7}(x-1). \tag{63}$$

This elliptic boundary value problem containing singularity was solved in the same way problem containing no singularity was solved. Relative errors in the maximum norm in percentage with respect to various combinations of p-degree are shown in Table **??**.

3.3.3 Problem with Oscillating Singularity

This test problem is on the domain [0,1]

$$\begin{cases} -u''(x) = f \qquad \text{for} \quad x \in (0,1) \\ u(0) = u(1) = 0 \end{cases} \tag{64}$$

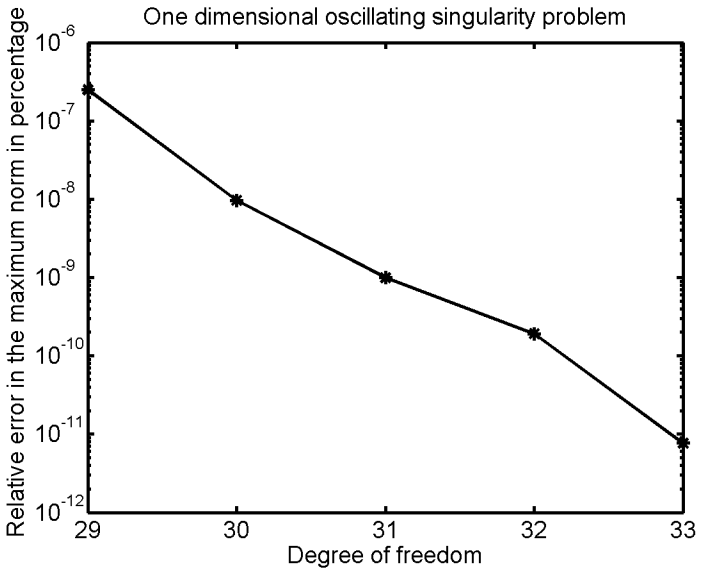


Figure 19: Relative error in the max-norm(%) for oscillating singularity problem using IGA-Collocation method

Table 6: The relative errors in the maximum norm in percentage of numerical solutions of one dimensional second order equation with oscillating singularity obtained by IGA-Collocation

Degree of 1st segment	Degree of 2nd segment	DOF	IGA-Collocation
$p = 5$	$p = 6$	29	2.49E-07
$p = 5$	$p = 7$	30	9.63E-09
$p = 5$	$p = 8$	31	9.91E-10
$p = 5$	$p = 9$	32	1.91E-10
$p = 5$	$p = 10$	33	7.69E-12

which has the exact solution:

$$u(x) = x^{0.65} \sin(0.1 \log x) \tag{65}$$

To solve this problem, domain [0,1] was divided into two overlapping subdomains [0, 0.55] and [0.45, 1]. An enrichment function $x^{0.65} \sin(0.1 \log x)$ is introduced in the singularity part to capture singularity.

$$\mathcal{V}_1^s = \{x^{0.65} \sin(0.1 \log x) \times \phi^L(x)\} \cup \mathcal{V}_1$$

$$\mathcal{V} = \text{span}(\mathcal{V}_1^s \cup \mathcal{V}_2).$$

Relative errors in the maximum-norm(%) of the problem with oscillating singularity obtained by IGA-Collocation using enriched functions in $\mathcal{V}$ are displayed in Table **??**.

CHAPTER 4: MODIFICATION OF BASIS FUNCTIONS

For collocation method the basis functions should have continuous derivatives (i.e, $\mathcal{C}^1$). If a problem is solved element wise like in finite element method then we need to modify functions at the patch to make them $\mathcal{C}^1$ continuous.

4.1 Modification of Bézier Polynomials in One-dimension

If we divide physical domain into several patches and assemble B-spline functions constructed on each patch in a patchwise manner, then the derivatives of assembled B-spline functions could be discontinuous along the patch boundaries. So we need to do some modifications in order to make them continuous [18]. These modified Bézier functions are linearly independent and their first derivatives are zero at both ends except for second and second last functions.

By theorem 2.1[18], for $2 \leq k \leq n-1$ the first function $N_{1,n+1}(u)$ and the last function $N_{n+1,n+1}(u)$ can be altered as shown in Table **??** and these alterations are called **Nodal Alteration**. The alterations to the second function $N_{2,n+1}(u)$ and to the second last function $N_{n,n+1}(u)$ are called **Side Alteration**. Applying this technique to degree 5 Bézier functions and taking $s = 2$ we will get these modified

Table 7: Original and Modification of B-Spline Functions

Index of B-spline function	Original function	Modified function
First function	$N_{1,k}(u)$	$N_{1,s}(u)(1+sx)$
Second function	$N_{2,k}(u)$	$-N_{2,k}(u)J(\phi)$
Second last function	$N_{m-k-1,k}(u)$	$N_{m-k-1,k}(u)J(\phi)$
Last function	$N_{m-k,k}(u)$	$N_{m-k,s}(u)(1+s-sx)$

Bézier functions.

$$\begin{cases} N_{1,6}(u) = (1-u)^2(1+2u), \\ N_{2,6}(u) = -5u(1-u)^4 \left|J(\varphi_k(u))\right|, \\ N_{3,6}(u) = 10u^2(1-u)^3, \\ N_{4,6}(u) = 10u^3(1-u)^2, \\ N_{5,6}(u) = 5u^4(1-u) \left|J(\varphi_k(u))\right|, \\ N_{6,6}(u) = u^2(3-2u), \end{cases} \tag{66}$$

where $\varphi_k(u)$ is a linear patch mapping from reference domain $\Omega = [0,1]$ to physical subdomain $\Omega_k = [x_k, x_{k+1}]$ and $J(\varphi_k(u))$ is the Jacobian of $\varphi_k(u)$. This mapping $\varphi_k(u) : \Omega \to \Omega_k$ is defined by

$$\varphi_k(u) = (x_{k+1} - x_k)u + x_k. \tag{67}$$

4.2 Two-dimensional Extension of Modification

This modification technique can be extended to two dimensions also.Consider mesh sizes $h_i = x_{i+1} - x_i$ and $k_j = y_{j+1} - y_j$ of $[a,b]$ and $[c,d]$ respectively. Two dimensional linear patch mapping $\varphi_{i,j}(u,v) : \Omega \to \Omega_{i,j}$ is defined by

$$\varphi_{i,j}(u,v) = \{h_i u + x_i, k_j v + y_j\} \tag{68}$$

Like one dimension case, we will modify Bézier polynomials $N_{i,k}(u)$ and $M_{j,k'}(v)$ to get sets of modified Bézier basis functions in both u and v directions. Tensor product of these modified functions will give the reference shape functions for two dimension. If we denote altered Bézier polynomials by $\tilde{N}_{i,k}(u)$ and $\tilde{M}_{j,k'}(v)$ and take degree 4 polynomials in both directions then the tensor product will give us 25 reference shape functions.

$$\text{Nodal Alterations: } \begin{cases} \tilde{N}_{1,5}(u) \times \tilde{M}_{1,5}(v) & \tilde{N}_{1,5}(u) \times \tilde{M}_{5,5}(v) \\ \tilde{N}_{5,5}(u) \times \tilde{M}_{1,5}(v) & \tilde{N}_{5,5}(u) \times \tilde{M}_{5,5}(v) \end{cases} \tag{69}$$

$$\text{Side Alterations: } \begin{cases} \tilde{N}_{1,5}(u) \times \tilde{M}_{2,5}(v) & \tilde{N}_{1,5}(u) \times M_{3,5}(v) & \tilde{N}_{1,5}(u) \times M_{4,5}(v) \\ \tilde{N}_{2,5}(u) \times \tilde{M}_{1,5}(v) & N_{3,5}(u) \times \tilde{M}_{1,5}(v) & N_{4,5}(u) \times \tilde{M}_{1,5}(v) \\ \tilde{N}_{5,5}(u) \times \tilde{M}_{2,5}(v) & \tilde{N}_{5,5}(u) \times M_{3,5}(v) & \tilde{N}_{5,5}(u) \times M_{4,5}(v) \\ \tilde{N}_{2,5}(u) \times \tilde{M}_{5,5}(v) & N_{3,5}(u) \times \tilde{M}_{5,5}(v) & N_{4,5}(u) \times \tilde{M}_{5,5}(v) \end{cases} \tag{70}$$

$$\text{Inside Alterations: } \begin{cases} \tilde{N}_{2,5}(u) \times \tilde{M}_{2,5}(v) & \tilde{N}_{2,5}(u) \times M_{3,5}(v) & \tilde{N}_{2,5}(u) \times M_{4,5}(v) \\ N_{3,5}(u) \times \tilde{M}_{2,5}(v) & N_{3,5}(u) \times M_{3,5}(v) & N_{3,5}(u) \times M_{4,5}(v) \\ N_{4,5}(u) \times \tilde{M}_{2,5}(v) & N_{4,5}(u) \times M_{3,5}(v) & N_{4,5}(u) \times M_{4,5}(v) \end{cases} \tag{71}$$

4.3 Global Basis Numbering Used for Assembling Local Stiffness Matrices

When the modified Bézier polynomials of degree 4 is applied to elliptic PDE on a rectangular domain consisted of nine rectangular patches, numbering of global basis functions constructed by push-forwards of the 25 modified Bézier polynomials onto nine patches are as follows[7b]:

$$
\begin{array}{cccccccccccc}
⑬ & 97 & 96 & 95 & ⑭ & 95 & 109 & 108 & ⑮ & 108 & 117 & 116 & ⑯ \\
98 & 103 & 104 & 105 & 94 & 105 & 112 & 113 & 107 & 113 & 120 & 121 & 115 \\
99 & 100 & 101 & 102 & 93 & 102 & 110 & 111 & 106 & 111 & 118 & 119 & 114 \\
69 & 74 & 75 & 76 & 65 & 76 & 83 & 84 & 78 & 84 & 91 & 92 & 86 \\
⑨ & 68 & 67 & 66 & ⑩ & 66 & 80 & 79 & ⑪ & 79 & 88 & 87 & ⑫ \\
69 & 74 & 75 & 76 & 65 & 76 & 83 & 84 & 78 & 84 & 91 & 92 & 86 \\
70 & 71 & 72 & 73 & 64 & 73 & 81 & 82 & 77 & 82 & 89 & 90 & 85 \\
26 & 35 & 36 & 37 & 22 & 37 & 49 & 50 & 42 & 50 & 62 & 63 & 55 \\
⑤ & 25 & 24 & 23 & ⑥ & 23 & 44 & 43 & ⑦ & 43 & 57 & 56 & ⑧ \\
26 & 35 & 36 & 37 & 22 & 37 & 49 & 50 & 42 & 50 & 62 & 63 & 55 \\
27 & 32 & 33 & 34 & 21 & 34 & 47 & 48 & 41 & 48 & 60 & 61 & 54 \\
28 & 29 & 30 & 31 & 20 & 31 & 45 & 46 & 40 & 46 & 58 & 59 & 53 \\
① & 17 & 18 & 19 & ② & 19 & 38 & 39 & ③ & 39 & 51 & 52 & ④
\end{array}
\tag{72}
$$

Here ⓚ represents the nodal basis function corresponding to the kth node.

Local numbering for the 25 shape functions listed in (**??**), (**??**), (**??**), are as follows:

$$
\begin{array}{ccccc}
④ & 13 & 12 & 11 & ③ \\
14 & 23 & 24 & 25 & 10 \\
15 & 20 & 21 & 22 & 9 \\
16 & 17 & 18 & 19 & 8 \\
① & 5 & 6 & 7 & ②
\end{array}
\tag{73}
$$

Here the shape functions listed in (**??**), (**??**), (**??**), respectively, is assigned the fol-

lowing local numbers:

$$(Nodal)\begin{bmatrix} 4 & 3 \\ 1 & 2 \end{bmatrix}, \qquad (Side)\begin{bmatrix} 14 & 15 & 16 \\ 11 & 12 & 13 \\ 8 & 9 & 10 \\ 5 & 6 & 7 \end{bmatrix}, \qquad (Internal)\begin{bmatrix} 23 & 24 & 25 \\ 20 & 21 & 22 \\ 17 & 18 & 19 \end{bmatrix}.$$

4.4 Numerical Results

In this section we discuss few one and two dimensional problems, where modified B-spline basis functions were used.

4.4.1 One-dimensional Non-singular Problem

The first test problem is on the domain [0,1]

$$\begin{cases} -u''(x) = f \qquad \text{for} \quad x \in (0,1) \\ u(0) = u(1) = 0 \end{cases} \tag{74}$$

which has the exact solution:

$$u(x) = x^2(x-1) \tag{75}$$

This problem was numerically solved using the IGA-Collocation method. The basis functions were altered as discussed at the beginning of this chapter. The knot vectors were chosen as $U = \{\underbrace{0,...,0}_{7},\underbrace{1,...,1}_{7}\}$ to get Bézier functions of degree 6. We used Gauss points for collocation.

Physical domain [0,1] was first divided into 4 unequal mesh size to solve this problem using collocation method. The relative error in the max-norm in percentage was

Table 8: Comparison of FEM-Galerkin and FEM-Collocation methods for one dimensional non-singular problem

Degree	FEM-Galerkin	FEM-Collocation
$p = 6$	9.36E-16	1.87E-15
$p = 7$	9.36E-16	5.24E-15
$p = 8$	1.49E-15	1.87E-15
$p = 9$	9.36E-16	4.21E-15
$p = 10$	1.12E-15	8.43E-16

approximately 10^{-15} which is almost true solution. We also compared this result with FEM-Galerkin method. The comparison between FEM-Collocation and FEM-Galerkin for different degrees of basis functions are shown in Table **??**. Increasing degree of Bézier polynomials will not make much difference so we can use less degrees of freedom(DOF) to get the same result.

4.4.2 One-dimensional Problem with Monotone Singularity of type x^λ

Consider a model poisson equation,

$$\begin{cases} -u''(x) = f \qquad \text{for} \quad x \in (0,1) \\ u(0) = u(1) = 0 \end{cases} \tag{76}$$

which has the exact solution:

$$u(x) = x^{1.7}(x-1). \tag{77}$$

Like non singular problem, in this problem also physical domain [0,1] was first divided into 4 unequal mesh size. The relative error in the max-norm in percentage for IGA-Collocation and IGA-Galerkin for different degrees of basis functions are shown in Table **??**.

Table 9: Comparison of IGA-Galerkin and IGA-Collocation method for one dimensional singular problem

Degree	DOF	FEM-Galerkin	FEM-Collocation
$p = 6$	22	4.28E-05	3.32E-04
$p = 7$	26	2.49E-05	2.20E-04
$p = 8$	30	1.56E-05	1.24E-04
$p = 9$	34	1.05E-05	6.33E-05
$p = 10$	38	7.41E-06	4.21E-05

Table 10: Comparison of FEM-Galerkin and FEM-Collocation for two dimensional problem

Method	degree	$\|u - u^h\|_{\infty,rel(\%)}$	$\|u - u^h\|_{L^2,rel(\%)}$	$\|u - u^h\|_{eng,rel(\%)}$
IGA-Galerkin	$p = 4$	1.44E-15	6.06E-14	1.74E-06
IGA-Collocation	$p = 4$	5.76E-13	1.21E-12	N/A

4.4.3 Two-dimensional Non-singular Problem

This test problem is on the domain $\Omega = [0, \frac{3}{2}] \times [0, \frac{3}{2}]$

$$\begin{cases} -\Delta u = f & \text{in} \quad \Omega \\ u = 0 & \text{on} \quad \partial\Omega \end{cases} \tag{78}$$

which has the exact solution:

$$u(x, y) = x^2 y^2 (x - \frac{3}{2})(y - \frac{3}{2}) \tag{79}$$

To solve this 2-D problem, the physical domain was partitioned into 3 equal sized mesh in both directions resulting in 9 rectangular grid. The B-spline functions were modified in both u and v directions. We chose $p = 4$ for this problem. There were total 25 altered basis functions in each grid. We first created the local stiffness matrix then followed the numbering technique mentioned in previous section to create the global stiffness matrix.

We solved this problem with both FEM-Galerkin and FEM-Collocation method and the error estimate is shown in Table **??**. In FEM-Collocation method everything was calculated in the reference domain. Computation of higher order derivatives were done in the following way:

Consider a mapping $\Phi : \hat{\Omega} \to \Omega$ from reference space to physical space.

$$\hat{f} = f \circ \Phi, \quad \text{where } f(u, v) = (x, y).$$

By chain rule,

$$\frac{\partial \hat{f}}{\partial u} = \frac{\partial f}{\partial x}\frac{\partial x}{\partial u} + \frac{\partial f}{\partial y}\frac{\partial y}{\partial u},$$
$$\frac{\partial \hat{f}}{\partial v} = \frac{\partial f}{\partial x}\frac{\partial x}{\partial v} + \frac{\partial f}{\partial y}\frac{\partial y}{\partial v}$$

. Hence,

$$(\nabla_{xy} f) \circ \Phi = J(\Phi)^{-1} \nabla_{uv}(f \circ \Phi) \tag{80}$$

where $J_{11} = \dfrac{\partial x}{\partial u}; J_{12} = \dfrac{\partial y}{\partial u}; J_{21} = \dfrac{\partial x}{\partial v}; J_{22} = \dfrac{\partial y}{\partial v}$. For second derivative,

$$(\nabla_{xy} f_x) \circ \Phi = J(\Phi)^{-1} \nabla_{uv}(f_x \circ \Phi), (\nabla_{xy} f_y) \circ \Phi = J(\Phi)^{-1} \nabla_{uv}(f_y \circ \Phi). \tag{81}$$

Therefore,

$$f_{xx} \circ \Phi = J(\Phi)^{-1} \frac{\partial((J_{11})^{-1} \hat{f}_u + (J_{12})^{-1} \hat{f}_v)}{\partial u},$$
$$f_{xy} \circ \Phi = J(\Phi)^{-1} \frac{\partial((J_{11})^{-1} \hat{f}_u + (J_{12})^{-1} \hat{f}_v)}{\partial v},$$
$$f_{yx} \circ \Phi = J(\Phi)^{-1} \frac{\partial((J_{21})^{-1} \hat{f}_u + (J_{22})^{-1} \hat{f}_v)}{\partial u},$$
$$f_{yy} \circ \Phi = J(\Phi)^{-1} \frac{\partial((J_{21})^{-1} \hat{f}_u + (J_{22})^{-1} \hat{f}_v)}{\partial v}$$

. Since inverse mapping $\Phi^{-1} : \Omega \to \hat{\Omega}$

$$f = \hat{f} \circ \Phi^{-1},$$

$$(\partial_{xx} f) \circ \Phi = (J_{11})^{-1} \frac{\partial}{\partial u}\{(J_{11})^{-1} \frac{\partial}{\partial u}\hat{f} + (J_{12})^{-1} \frac{\partial}{\partial v}\hat{f}\} + (J_{12})^{-1} \frac{\partial}{\partial v}\{(J_{11})^{-1} \frac{\partial}{\partial u}\hat{f} + (J_{12})^{-1} \frac{\partial}{\partial v}\hat{f}\},$$

$$(\partial_{yy} f) \circ \Phi = (J_{21})^{-1} \frac{\partial}{\partial u}\{(J_{21})^{-1} \frac{\partial}{\partial u}\hat{f} + (J_{22})^{-1} \frac{\partial}{\partial v}\hat{f}\} + (J_{22})^{-1} \frac{\partial}{\partial v}\{(J_{21})^{-1} \frac{\partial}{\partial u}\hat{f} + (J_{22})^{-1} \frac{\partial}{\partial v}\hat{f}\},$$

$$(\partial_{xy} f) \circ \Phi = (J_{21})^{-1} \frac{\partial}{\partial u}\{(J_{11})^{-1} \frac{\partial}{\partial u}\hat{f} + (J_{12})^{-1} \frac{\partial}{\partial v}\hat{f}\} + (J_{22})^{-1} \frac{\partial}{\partial v}\{(J_{11})^{-1} \frac{\partial}{\partial u}\hat{f} + (J_{12})^{-1} \frac{\partial}{\partial v}\hat{f}\},$$

$$(\partial_{yx} f) \circ \Phi = (J_{11})^{-1} \frac{\partial}{\partial u}\{(J_{21})^{-1} \frac{\partial}{\partial u}\hat{f} + (J_{22})^{-1} \frac{\partial}{\partial v}\hat{f}\} + (J_{12})^{-1} \frac{\partial}{\partial v}\{(J_{21})^{-1} \frac{\partial}{\partial u}\hat{f} + (J_{22})^{-1} \frac{\partial}{\partial v}\hat{f}\},$$

$$\Delta_{xy} f \circ \Phi = \Delta_{xy}(\hat{f} \circ \Phi^{-1}) \circ \Phi = (\partial_{xx} f) \circ \Phi + (\partial_{yy} f) \circ \Phi$$

4.4.4 Two-dimensional Problem with Singularity

This test problem is the Poisson equation on the semi-circular domain of radius $\frac{1}{2}$ and centered at origin.

$$\begin{cases} -\Delta u = f \quad \text{in} \quad \Omega \\ \\ u = 0 \quad \text{on} \quad \partial\Omega \end{cases} \tag{82}$$

which has the exact solution:

$$u(r, \theta) = \sqrt{r}(\frac{1}{2} - r) \sin\theta. \tag{83}$$

To map parameter space to physical space we used smooth non-NURBS mapping $F : \hat{\Omega} \to \Omega$ and $F(u, v) = (x(u, v), y(u, v))$ where

$$F(u, v) = \begin{cases} x(u, v) = \frac{(v^2)}{2} \cos(\pi(1 - u)) \\ \\ y(u, v) = \frac{(v^2)}{2} \sin(\pi(1 - u)). \end{cases} \tag{84}$$

The degree of the Bézier functions in v direction was fixed to 3 whereas the degree in u direction was elevated. We used modified Bézier polynomial as basis functions.

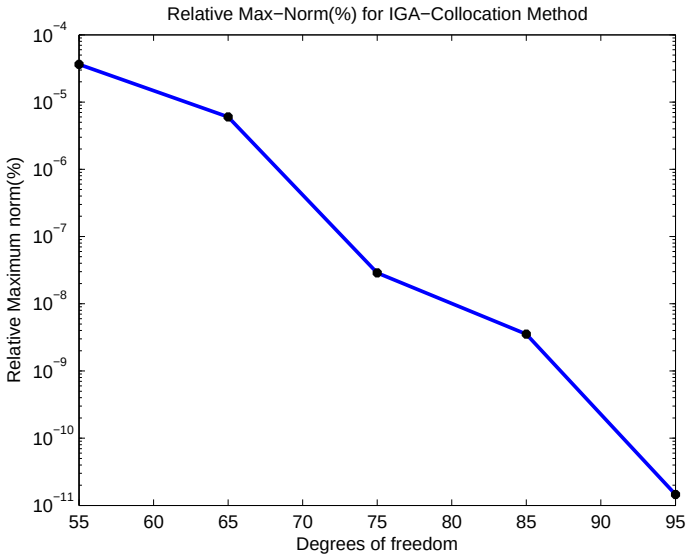


Figure 20: Relative error in the max-norm (%) of numerical solutions obtained by IGA-Collocation

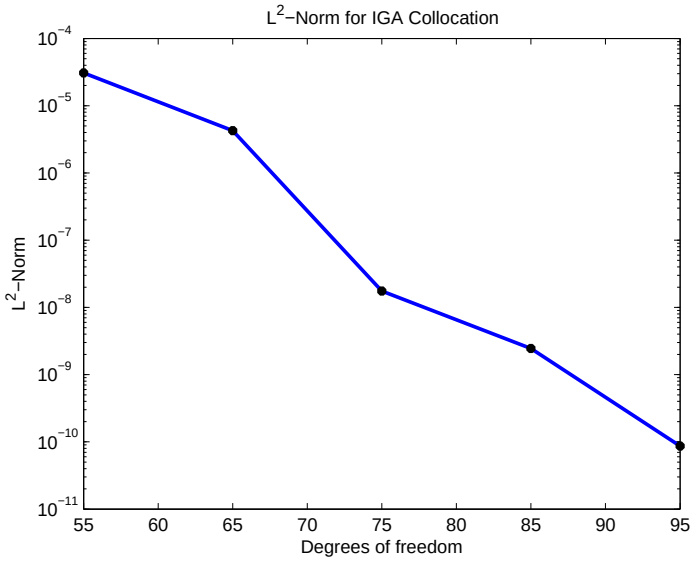


Figure 21: Relative error in for L^2-Norm (%) of numerical solutions obtained by IGA-Collocation

Table 11: Error estimate for IGA-Collocation method for two-dimensional problem on a semicircular domain with singularity.

(p_u, p_v)	$\|u - u^h\|_{\infty,rel(\%)}$	$\|u - u^h\|_{L^2,rel(\%)}$
(6,3)	3.63E-05	3.08E-05
(7,3)	6.02E-06	4.26E-06
(8,3)	2.88E-08	1.75E-08
(9,3)	3.54E-09	2.45E-09
(10,3)	1.45E-11	8.64E-12

The relative error estimate is displayed in Table **??** and the graphs of relative error in the maximum norm(%) and L2-norm(%) are shown in Fig. **??**and **??** respectively.

CHAPTER 5: SCHWARZ ALTERNATING METHOD IN THE FRAMEWORK OF IGA-COLLOCATION

Schwarz alternating method was introduced by H. A. Schwarz[19] in 1870. A modification of this method is known as parallel Schwarz method. In Schwarz alternating method, the domain is divided into two overlapping subdomains and the iterative procedure starts by taking one initial guess for the boundary of first subproblem. This method involves solving the boundary value problem on each of the two subdomains in turn, taking always the last values of the approximate solution as the next boundary conditions. It is important to note that in Schwarz alternating method, the solution of the first problem is required before the second problem can be solved. In parallel Schwarz method, the domain is divided into two overlapping subdomains and the iterative procedure starts by taking initial guesses on each subdomain. In this case the subproblem can be solved independently in each subdomain.

5.1 Schwarz Alternating Method

The classical Schwarz alternating method in the framework of IGA-Collocation is explained in this section. Consider the Poisson problem

$$\begin{cases} -\Delta u = f \quad \text{in} \quad \Omega, \\ u = 0 \quad \text{on} \quad \partial\Omega, \end{cases} \tag{85}$$

on a bounded Lipschitz region Ω with homogeneous(zero) boundary condition on boundary $\partial\Omega$. This domain is divided into two subdomains Ω_1 and Ω_2 with artificial

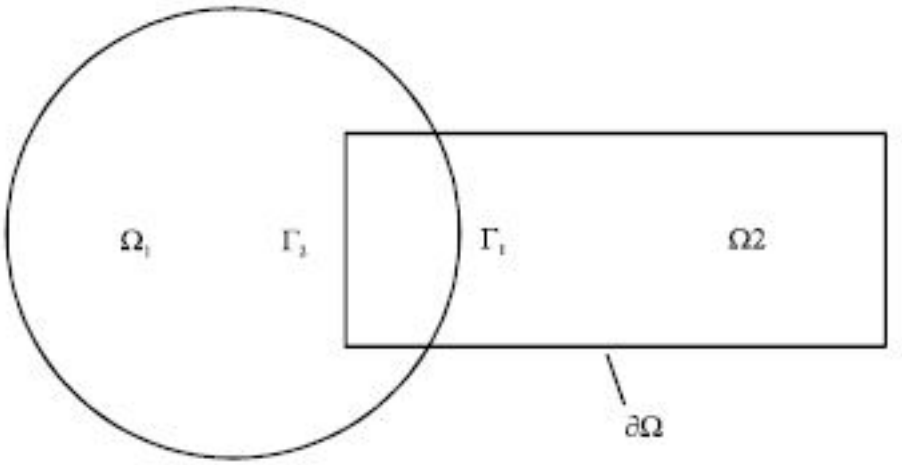


Figure 22: Overlapping subdomains with artificial boundaries

boundaries Γ_1 and Γ_2 respectively,as shown in Fig.**??**. Define $\phi_i(x, y)$ and $\phi_j(x, y)$ basis functions for subdomains Ω_1 and Ω_2 respectively. Assume,

$$u_1(x, y) \approx u_h(x, y) = \sum_{i=1}^{N} c_i \phi_i(x, y), \tag{86}$$

$$u_2(x, y) \approx v_h(x, y) = \sum_{j=1}^{M} c_j \phi_j(x, y) \tag{87}$$

In IGA-Collocation method, $\phi_i(x, y)$ and $\phi_j(x, y)$ are NURBS basis functions.

This subdivision gives us two subproblems to solve using IGA-Collocation method[22].

$$\begin{cases} -\Delta u_1^{n+1} = f \quad \text{for} \quad \Omega_1 \\ u_1^{n+1} = 0 \quad \text{on} \quad \partial\Omega_1 \setminus \Gamma_1 \\ u_1^{n+1} = u_2^n \quad \text{on} \quad \Gamma_1 \end{cases} \tag{88}$$

$$\begin{cases} -\Delta u_2^{n+1} = f \quad \text{for} \quad \Omega_2 \\ u_2^{n+1} = 0 \quad \text{on} \quad \partial\Omega_2 \setminus \Gamma_2 \\ u_2^{n+1} = u_1^{n+1} \quad \text{on} \quad \Gamma_2 \end{cases} \tag{89}$$

where n denotes the number of iterations. To start the iterative process, subproblem(**??**) is first solve for $n = 0$ with some initial guess $u_2^0 = g(x, y)$ on artificial boundary Γ_1.

We will keep solving this problem by iterating steps (**??**) and (**??**) while updating $u_1^{n+1}(x, y)$ and $u_2^{n+1}(x, y)$ by most updated values of $u_2(x, y)$ and $u_1(x, y)$ respectively. The iterations are performed until certain convergence conditions are met. Throughout this dissertation we consider the relative error in the maximum norm(%) as the termination condition. Least squares method is used to determine unknowns c_i's and c_j's along artificial boundaries Γ_1 and Γ_2.

5.2 The Parallel Schwarz Method

Pierre-Louis Lions[20] proposed parallel Schwarz method by doing small but essential modification in Schwarz alternating method which made the problem perfect for parallel computing. Lions modified subproblems (**??**)-(**??**) in the following way:

$$\begin{cases} -\Delta u_1^{n+1} = f \quad \text{for} \quad \Omega_1 \\ u_1^{n+1} = 0 \quad \text{on} \quad \partial\Omega_1 \setminus \Gamma_1 \\ u_1^{n+1} = u_2^{n} \quad \text{on} \quad \Gamma_1 \end{cases} \tag{90}$$

$$\begin{cases} -\Delta u_2^{n+1} = f \quad \text{for} \quad \Omega_2 \\ u_2^{n+1} = 0 \quad \text{on} \quad \partial\Omega_2 \setminus \Gamma_2 \\ u_2^{n+1} = u_1^{n} \quad \text{on} \quad \Gamma_2 \end{cases} \tag{91}$$

To start this parallel process, subproblems (**??**)-(**??**) are solved together for $n = 0$ step with two initial guesses $u_2^0 = g(x, y)$ and $u_1^0 = h(x, y)$ on artificial boundaries Γ_1 and Γ_2 respectively.

5.3 One-dimensional Problems

Several problems were tested in one-dimensional cases to see the efficiency of iterative method in IGA-Collocation. We test performance of Schwarz alternating method with respect to the following three combinations:

[I] IGA-Galerkin on Ω_1 and IGA-Galerkin on Ω_2

[II] IGA-Galerkin on Ω_1 and IGA-Collocation on Ω_2

[III] IGA-Collocation on Ω_1 and IGA-Collocation on Ω_2

One-dimension equation with singularity was also tested by Schwarz alternating method together with mapping techniques.

5.3.1 One-dimensional Problems Whose Solutions are Smooth

Consider the one dimension Poisson equation

$$\begin{cases} -u''(x) = f \quad \text{in} \quad \Omega \\ u(x) = 0 \quad \text{on} \quad \partial\Omega \end{cases} \tag{92}$$

which has smooth solution:

$$u(x) = x^2 - x^3 \tag{93}$$

Problem (**??**) is solved with respect to various sizes of the overlapping subdomains with initial guess 0 on artificial boundary Γ_1. Relative errors in the maximum

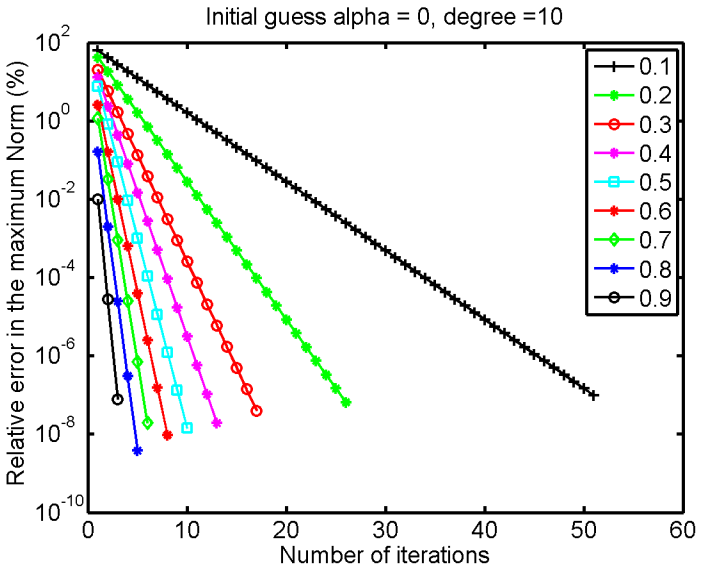


Figure 23: Relative error in the maximum norm(%) of numerical solutions of second order equation with smooth solution $u(x) = x^2 - x^3$

Table 12: Various overlapping sizes, number of iterations, ratio of relative errors and slope of line of convergence for problem(**??**).

Overlapping Size	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
Iterations	51	26	17	13	10	8	6	5	3
$(err2/err1)$	0.66	0.44	0.29	0.17	0.11	0.05	0.03	0.01	0.002
$\lambda \approx \frac{\log(err2/err1)}{N_2 - N_1}$	-0.42	-0.82	-1.24	-1.77	-2.21	-2.99	-3.51	-4.61	-6.21

norm(%) versus the sizes of overlapping subdomains is depicted in Fig. **??**. Table **??** shows that the larger overlapping subdomains are, the less number of iterations are.

The location of artificial boundaries did not matter for the rate of convergence when the solution is smooth. If the size of the overlapping region was increased then the solution acquired in the first step was very close to true solution so it required less number of iterations and thus had smaller convergence rate as shown in Table **??**.

Since relative errors versus number of iterations in semi-log scale are straight lines for various sizes of overlapping subdomains, we expect the following lemma.

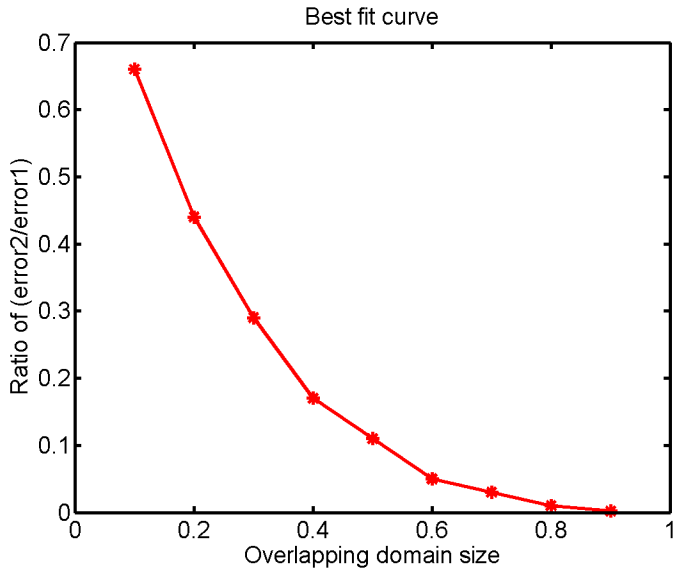


Figure 24: Ratio of relative errors in the maximum norm(%) versus sizes of overlapping subdomains for problem (**??**)

Lemma- *Let N be the number of iterations in the alternating method and $\|err\|_\infty$ be the relative errors in the maximum norm. Then we expect*

$$\|err\|_\infty \leq e^{\lambda N}, \tag{94}$$

where λ is the slope of lines in Table **??**.

Therefore,

$$\log \|Err_1\|_\infty \leq \lambda N_1,$$

$$\log \|Err_2\|_\infty \leq \lambda N_2,$$

$$\log \|\frac{Err_2}{Err_1}\|_\infty \approx \lambda(N_2 - N_1)$$

$$\lambda \approx \frac{\log \|\frac{Err_2}{Err_1}\|_\infty}{(N_2 - N_1)}$$

where N is the number of iteration. If we plot ratio of relative errors versus the sizes of overlapping subdomains from Table **??** in xy-axis then the convergence profile makes

Table 13: CPU time comparison for IGA-Galerkin-Galerkin, IGA-Galerkin-Collocation and IGA-Collocation-Collocation iterative methods

Iterative Method	CPU time(in seconds)
IGA-Galerkin and IGA-Galerkin	15.864
IGA-Galerkin and IGA-Collocation	7.175
IGA-Collocation and IGA-Collocation	0.6069

a quartic curve which is given by:

$$y = ax^4 + bx^3 + cx^2 + dx + e \tag{95}$$

where

$$a = 1.63353,$$

$$b = -4.87451,$$

$$c = 6.05185,$$

$$d = -3.78608,$$

$$e = 0.990304.$$

We also solved the same problem w.r.to three combinations IGA-Galerkin-Galerkin, IGA-Galerkin-Collocation and IGA-Collocation-Collocation with B-spline basis functions corresponding to following open knot vector.

$$U = \{\underbrace{0, ..., 0}_{10}, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, \underbrace{1, ..., 1}_{10}\}$$

Now the domain $[0, 1]$ is subdivided into two overlapping subdomains [0, 0.6] and [0.4, 1]. All three methods gave same results and required 26 number of iterations to reach solution of accuracy 6.1E-08 but the time taken for IGA-Collocation-Collocation was much less compared with other two combinations. In other words, IGA-Collocation-

Collocation is the most cost efficient method. CPU time taken for all three methods is listed in Table **??**.

5.3.2 One-dimensional Problem with Monotone Singularity

We test the iterative method to the following equation:

$$\begin{cases} -u''(x) = f & \text{for} \quad x \in (0,1), \\ u(0) = u(1) = 0, \end{cases} \tag{96}$$

which has the exact solution

$$u(x) = x^{\alpha} - x \tag{97}$$

containing a monotone singularity with intensity $\alpha = 0.65$. To solve this problem, domain [0,1] was subdivided into overlapping subdomains $\Omega_1 = [0, \frac{3}{5}]$ and $\Omega_2 = [\frac{2}{5}, 1]$ with artificial boundaries $\Gamma_1 = \frac{3}{5}$ and $\Gamma_2 = \frac{2}{5}$. To apply Schwarz alternating method, the initial guess on artificial boundary Γ_1 was chosen as 0. IGA -Galerkin method was used to solve subproblem (**??**) and IGA -Collocation method was used to solve subproblem (**??**).

Problem(**??**) was solved in two different ways.

1- With using mapping techniques to solve subproblem(**??**) on subdomain Ω_1

2- Without using mapping techniques to solve subproblem(**??**) on subdomain Ω_1

For Ω_1 the subdomain of Ω which contains the singularity of u, the auxiliary mapping[16] $\varphi^{\beta} : \widehat{\Omega}_1 \longrightarrow \Omega_1$ is defined by:

$$\varphi^{\beta}(\xi) = (\xi)^{\beta}, \tag{98}$$

Table 14: Comparing relative error in the maximum norm(%) for $u(x) = x^\alpha - x$, when the intensity of singularity is $\alpha = 0.65$

p-degree	DOF	without mapping	with mapping
$p = 5$	21	0.085	0.003
$p = 6$	25	0.041	0.0009
$p = 7$	29	0.014	0.0007
$p = 8$	33	0.006	0.00056
$p = 9$	37	0.004	0.00054
$p = 10$	41	0.003	0.000064

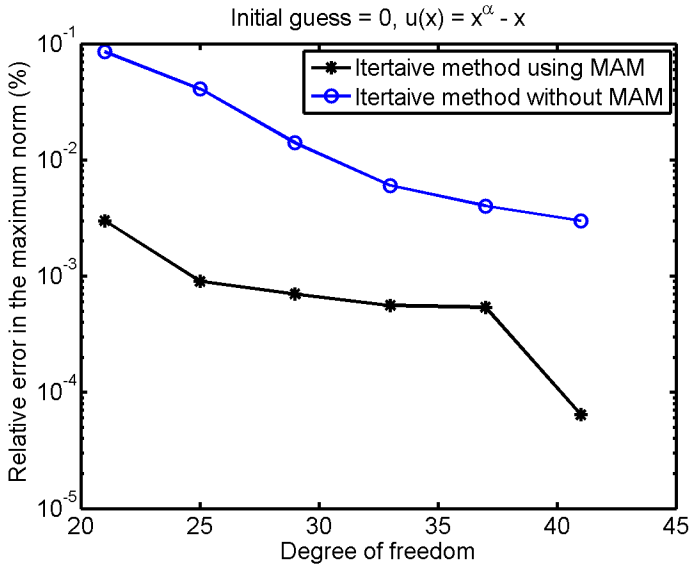


Figure 25: Relative error in the maximum norm(%) of numerical solutions of one-dimensional second order equation with nonregular solution $u(x) = x^\alpha - x$, when the intensity of singularity is $\alpha = 0.65$

where ξ denotes the coordinate of the points in the transformed domain $\widehat{\Omega}_1$. Here β is called the **mapping size** of the auxiliary mapping. In this problem for $u(x) = x^\alpha - x$, where $\alpha = 0.65$. $u \circ \varphi^\beta = (\xi)^{\alpha\beta}$ is much smoother than x^α. In particular, if $\beta = 1/\alpha$, then $u \circ \varphi^\beta$ is smooth. Integrals in bilinear form were computed as discussed in remark-2.1 of [16]. Table **??** shows relative errors in the maximum norm(%) with respect to these methods. We kept the degree of B-spline basis functions in subdomain Ω_2 fixed while degree of B-spline basis functions in subdomain Ω_1 is elevated.

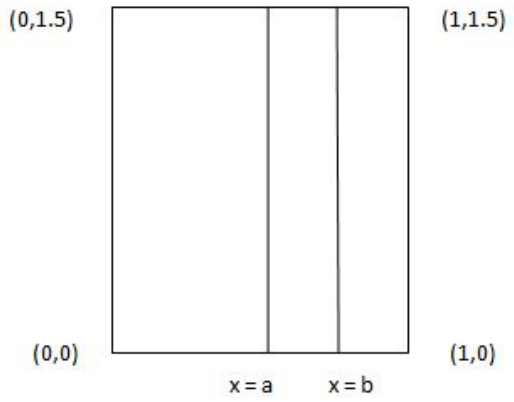


Figure 26: Rectangular subdomains $\Omega_1 = [0, b] \times [0, \frac{3}{2}]$ and $\Omega_2 = [a, 1] \times [0, \frac{3}{2}]$

5.4 Two-dimensional Problems

Now we test IGA-Collocation iterative method to two-dimensional problems with smooth as well as singular solutions.

5.4.1 The Poisson Equation on a Rectangular Domain

Model problem is elliptic boundary value problem on the rectangle $\Omega = [0, \frac{3}{2}] \times [0, \frac{3}{2}]$

$$\begin{cases} -\Delta u = f \quad \text{for} \quad \Omega \\ u = 0 \quad \text{on} \quad \partial\Omega \end{cases} \tag{99}$$

which has the exact solution:

$$u(x, y) = x^2 y^2 (x - \frac{3}{2})(y - \frac{3}{2}). \tag{100}$$

Mapping F_1 and F_2 are smooth linear mappings which maps parameter space $\widehat{\Omega} = [0, 1] \times [0, 1]$ onto physical spaces $\Omega_1 = [0, b] \times [0, \frac{3}{2}]$ and $\Omega_2 = [a, 1] \times [0, \frac{3}{2}]$ respectively, where a and b are the locations of artificial boundaries on the $x-$axis.

$F_1 : \hat{\Omega} \to \Omega_1$ and $F_1(u, v) = (x(u, v), y(u, v))$ is defined by

$$F_1(u,v) = \begin{cases} x(u,v) = bu \\ \\ y(u,v) = \frac{3}{2}v. \end{cases} \tag{101}$$

$F_2 : \hat{\Omega} \to \Omega_2$ and $F_2(u, v) = (x(u, v), y(u, v))$ is defined by

$$F_2(u,v) = \begin{cases} x(u,v) = (1.5-a)u + a \\ \\ y(u,v) = \frac{3}{2}v. \end{cases} \tag{102}$$

We first solved the below subproblem using IGA-Collocation method in subdomain Ω_1 with initial guess $u_2^0(x, y) = g(x, y) = 0$

$$\begin{cases} -\Delta u_1^{n+1} = f \quad \text{for} \quad \Omega_1 \\ \\ u_1^{n+1}(x,y) = 0 \quad \text{on} \quad \partial\Omega_1 \setminus \Gamma_1 \\ \\ u_1^{n+1}(x,y) = u_2^n(x,y) \quad \text{on} \quad \Gamma_1 \end{cases} \tag{103}$$

and then we solved the below subproblem for subdomain Ω_2 and kept iterating both steps until we acquired the solution of desired accuracy.

$$\begin{cases} -\Delta u_2^{n+1} = f \quad \text{for} \quad \Omega_2 \\ \\ u_2^{n+1}(x,y) = 0 \quad \text{on} \quad \partial\Omega_2 \setminus \Gamma_2 \\ \\ u_2^{n+1}(x,y) = u_1^{n+1}(x,y) \quad \text{on} \quad \Gamma_2 \end{cases} \tag{104}$$

This problem was also solved for different overlapping regions. Like in one-dimension, the number of iterations required to get the solution of accuracy $\| \, err \, \|_\infty \leq 10^{-12}$ were dependent upon the size of the overlapping region but not on the location of artificial boundaries. The iterative solver was stopped when the solution reached the desired accuracy of $\| \, err \, \|_\infty \leq 10^{-12}$ or if it has gone through 50 iterations, whichever

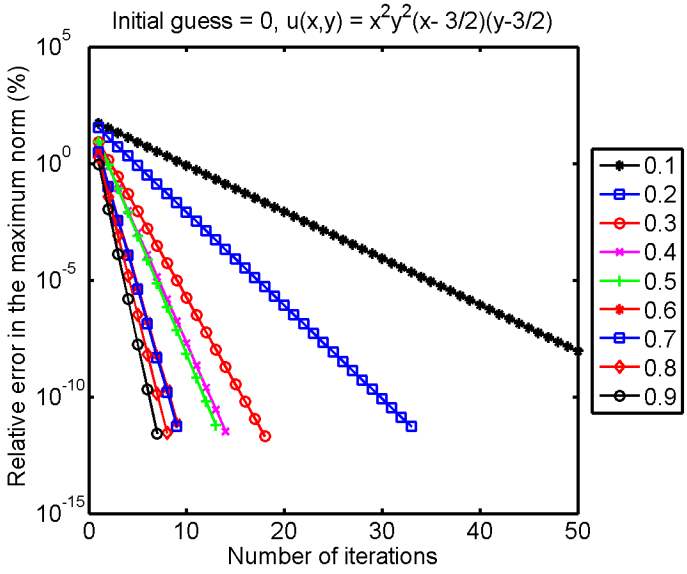


Figure 27: Relative error in the maximum norm(%) for $u(x) = x^2y^2(x - \frac{3}{2})(y - \frac{3}{2})$ using iterative collocation method

Table 15: Size of overlapping region and number of iterations required to get solution of accuracy 10^{-12}

Overlapping Size	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
Iterations	50	33	18	14	13	9	9	8	7

comes first. The result is displayed in Table **??** and the relative error in the maximum norm in (%) for different overlapping sizes are shown in Fig. **??**.

5.4.2 Two-dimensional Elliptic Equations Containing Singularities

The problem discussed here has crack singularity of type $r^{1/2}$ on a circular domain of radius 2 and centered at origin. This circular domain was decomposed into a circle $\Omega_1 = [(r, \theta) : r < 1, 0 < \theta < 2\pi]$ and a annulus $\Omega_2 = [(r, \theta) : 0.5 < r < 2, 0 < \theta < 2\pi]$. Consider the following Poisson equation:

$$\begin{cases} -\Delta u = f \quad \text{in} \quad \Omega \\ u = 0 \quad \text{on} \quad \partial\Omega \end{cases} \tag{105}$$

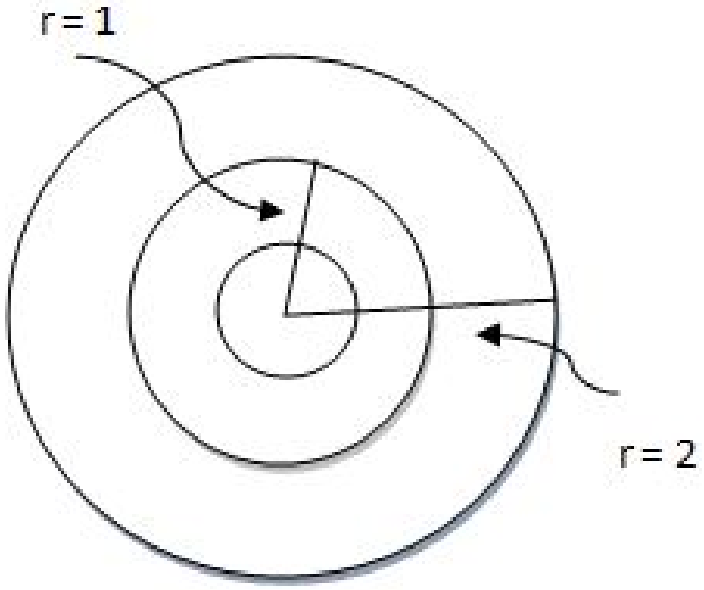


Figure 28: Circular subdomain $\Omega_1 = [(r,\theta) : r \leq 1, 0 < \theta < 2\pi]$ and annular subdomain $\Omega_2 = [(r,\theta) : 0.5 \leq r \leq 2, 0 < \theta < 2\pi]$.

which has the exact solution:

$$u(r,\theta) = \sqrt{r}(1-r)[\sin(\frac{\theta}{2}) + \sin(\frac{3\theta}{2})]. \tag{106}$$

Smooth mapping F_1 maps reference domain Ω onto physical subdomain Ω_1,

$$F_1 : \hat{\Omega} = [0,1] \times [0,1] \longrightarrow \Omega_1 = \{(x,y) : 0 \leq x^2 + y^2 \leq r_1^2\}$$

$$F_1(u,v) = \begin{cases} x(u,v) = v^2 \cos(2\pi(1-u)) \\ \\ y(u,v) = v^2 \sin(2\pi(1-u)). \end{cases} \tag{107}$$

Mapping F_2 maps reference domain Ω onto physical subdomain Ω_2,

$$F_2 : \hat{\Omega} = [0,1] \times [0,1] \longrightarrow \Omega_2 = \{(x,y) : r_2^2 \leq x^2 + y^2 \leq 2\}$$

$$F_2(u,v) = \begin{cases} x(u,v) = \frac{1}{2}(1-v)\cos(2\pi(1-u)) + 2v\cos(2\pi(1-u)) \\ y(u,v) = \frac{1}{2}(1-v)\sin(2\pi(1-u)) + 2v\sin(2\pi(1-u)). \end{cases} \tag{108}$$

where r_1 is radius of singular subdomain and r_2 is radius of inner circle of regular subdomain. Accuracy of solution depends on choice of r_1 and r_2. Since this is the problem with crack singularity of type $r^{\frac{1}{2}}$, initial guess is chosen $u(r,\theta) = \sqrt{r}\sin(\theta/2)$ along artificial boundary Γ_1 of subdomain Ω_1

$$\begin{cases} -\Delta u_1^{n+1} = f \quad \text{in} \quad \Omega_1 \\ u_1^{n+1} = u_2^n \quad \text{on} \quad \Gamma_1 \\ u_1^{n+1} = 0 \quad \text{on} \quad \partial\Omega_1 \cap \partial\Omega. \end{cases} \tag{109}$$

and the above problem was solved with IGA-Collocation. Once the solution for this step was found then the problem was solved for subdomain Ω_2.

$$\begin{cases} -\Delta u_2^{n+1} = f \quad \text{in} \quad \Omega_2 \\ u_2^{n+1} = u_1^{n+1} \quad \text{on} \quad \Gamma_2 \\ u_2^{n+1} = 0 \quad \partial\Omega_2 \cap \partial\Omega. \end{cases} \tag{110}$$

Unlike, non-singular problem the number of iterations required to reach the solution of desired accuracy is dependent upon the choice of radius $[r_2, r_1]$ as well as the size of the overlapping domain. If r_1, the radius of circular subdomain with singularity is very small then it takes more iterations to converge. Relative errors in the maximum norm (%) is displayed in Fig. **??**.

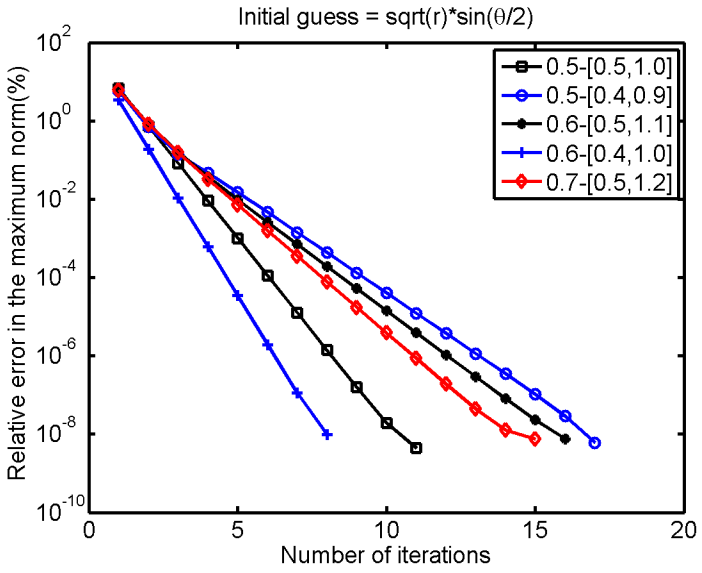


Figure 29: Relative error in the maximum norm(%) for $u(r,\theta) = \sqrt{r}(1-r)[\sin(\frac{\theta}{2}) + \sin(\frac{3\theta}{2})]$ using iterative IGA-Collocation method

CHAPTER 6: ALTERNATING METHOD FOR TWO NON-OVERLAPPING SUBDOMAINS

Previous chapter explained alternating method in the framework of IGA-Collocation when domain was subdivided into two overlapping subdomains. Now we shall see how to impose the same method when domain is subdivided into two non-overlapping subdomains. In non-overlapping methods, the subdomains intersect only on their interface.

Consider the Poisson problem

$$\begin{cases} -\Delta u = f \quad \text{in} \quad \Omega, \\ u = 0 \quad \text{on} \quad \partial\Omega, \end{cases} \tag{111}$$

on a bounded Lipschitz region Ω with homogeneous boundary condition on boundary $\partial\Omega$. Domain Ω is divided into two non-overlapping subdomains Ω_1 and Ω_2 with common internal boundary Γ. Like in overlapping subdomain problem this problem will also be solved by solving two subproblems. These subproblems will be explained by two different methods. In each method the boundary condition on internal boundary is imposed in two different ways as explained in[21].

1. The Dirichlet-Neumann method

2. The Neumann-Neumann method

6.1 The Dirichlet-Neumann Method

In this method for given initial guess λ^0, problem(**??**) will be solved by solving these two subproblems for each $k \geq 0$:

$$\begin{cases} -\Delta u_1^{k+1} = f \quad \text{in} \quad \Omega_1, \\ u_1^{k+1} = 0 \quad \text{on} \quad \partial\Omega_1 \cap \partial\Omega, \\ u_1^{k+1} = \lambda^k \quad \text{on} \quad \Gamma. \end{cases} \tag{112}$$

$$\begin{cases} -\Delta u_2^{k+1} = f \quad \text{in} \quad \Omega_2, \\ u_2^{k+1} = 0 \quad \text{on} \quad \partial\Omega_2 \cap \partial\Omega, \\ \dfrac{\partial u_2^{k+1}}{\partial n} = \dfrac{\partial u_1^{k+1}}{\partial n} \quad \text{on} \quad \Gamma. \end{cases} \tag{113}$$

with

$$\lambda^{k+1} := \theta u_{2|\Gamma}^{k+1} + (1-\theta)\lambda^k, \tag{114}$$

where $\theta, 0 \leq \theta \leq 1$ is a positive accelerated parameter.

Similar alternating procedure can be obtained if we change subproblems in this way

$$\begin{cases} -\Delta u_1^{k+1} = f \quad \text{in} \quad \Omega_1, \\ u_1^{k+1} = 0 \quad \text{on} \quad \partial\Omega_1 \cap \partial\Omega, \\ u_1^{k+1} = u_2^k \quad \text{on} \quad \Gamma. \end{cases} \tag{115}$$

$$\begin{cases} -\Delta u_2^{k+1} = f \quad \text{in} \quad \Omega_2, \\ u_2^{k+1} = 0 \quad \text{on} \quad \partial\Omega_2 \cap \partial\Omega, \\ \dfrac{\partial u_2^{k+1}}{\partial n} = \mu^k \quad \text{on} \quad \Gamma. \end{cases} \tag{116}$$

with

$$\mu^{k+1} := \theta \frac{\partial u_1^{k+1}}{\partial n} + (1-\theta)\mu^k. \tag{117}$$

6.2 The Neumann-Neumann Method

In this case, for eack $k \geq 0$ we need to solve:

$$\begin{cases} -\Delta u_i^{k+1} = f \quad \text{in} \quad \Omega_i, \\ u_i^{k+1} = 0 \quad \text{on} \quad \partial\Omega_i \cap \partial\Omega, \\ u_i^{k+1} = \lambda^k \quad \text{on} \quad \Gamma, \end{cases} \tag{118}$$

and then solve

$$\begin{cases} -\Delta v_i^{k+1} = 0 \quad \text{in} \quad \Omega_i, \\ v_i^{k+1} = 0 \quad \text{on} \quad \partial\Omega_i \cap \partial\Omega, \\ \dfrac{\partial v_i^{k+1}}{\partial n} = \dfrac{\partial u_1^{k+1}}{\partial n} - \dfrac{\partial u_2^{k+1}}{\partial n} \quad \text{on} \quad \Gamma, \end{cases} \tag{119}$$

for i = 1, 2, with

$$\lambda^{k+1} := \lambda^k - \theta\{a_1 v_{1|\Gamma}^{k+1} - a_2 v_{2|\Gamma}^{k+1}\}. \tag{120}$$

As before, θ is a positive accelerated parameter, a_1 and a_2 are two positive averaging coefficients,where λ^0 is initial guess.

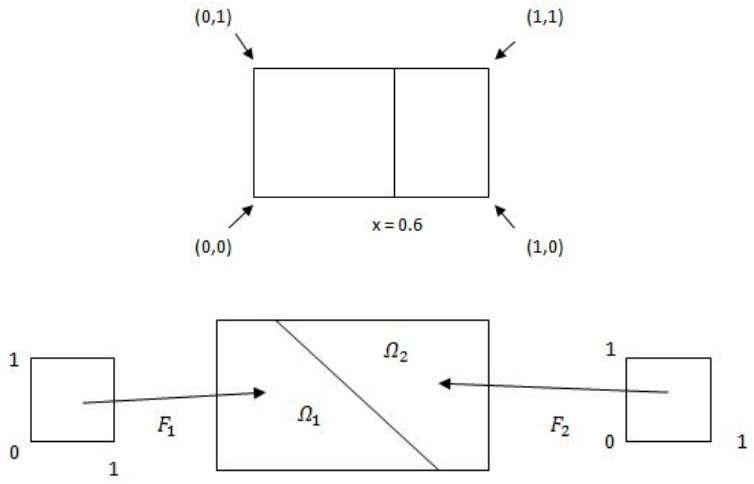


Figure 30: (a) Division by a vertical interface(b) Division by a slanted interface

6.3 Numerical Examples

We tested to two-dimensional elliptic equation on a rectangular domain problem by decomposing the physical domain into two non-overlapping subdomains in two different ways:

- Division by a vertical interface
- Division by a slanted interface

as shown in Fig. **??**. In this dissertation, the Dirichlet-Neumann subproblems(**??**)-(**??**) are used to solve elliptic boundary value problems for non-overlapping subdomains.

6.3.1 Vertical Interface in a Rectangular Domain

Consider two-dimensional elliptic boundary value problem with vertical subdivision of domain $\Omega = [-1, 1] \times [0, 1]$

$$\begin{cases} -\Delta u = f \quad \text{for} \quad \Omega \\ u = 0 \quad \text{on} \quad \partial\Omega \end{cases} \tag{121}$$

which has the exact solution:

$$u(x, y) = e^x(1 - x^2)(y - y^2). \tag{122}$$

To solve problem (**??**) we will divide the domain $\Omega = [-1, 1] \times [0, 1]$ into two non-overlapping subdomains $\Omega_1 = [-1, 0] \times [0, 1]$ and $\Omega_2 = [0, 1] \times [0, 1]$ with common interface at $a = 0$. For this vertical subdivision, two linear mappings, F_1 and F_2 are used to map parameter domain onto subdomains Ω_1 and Ω_2 with common internal boundary Γ at $a = 0$.

$F_1 : \hat{\Omega} \to \Omega_1$ and $F_1(u, v) = (x(u, v), y(u, v))$ is defined by

$$F_1(u, v) = \begin{cases} x(u, v) = (a + 1)u - 1 \\ y(u, v) = v. \end{cases} \tag{123}$$

$F_2 : \hat{\Omega} \to \Omega_2$ and $F_2(u, v) = (x(u, v), y(u, v))$ is defined by

$$F_2(u, v) = \begin{cases} x(u, v) = (1 - a)u + a \\ y(u, v) = v. \end{cases} \tag{124}$$

In order to solve problem(**??**) we solved subproblems(**??**) and (**??**) by assuming initial guess $\lambda^0 = 0.5$. For the subdomain Ω_2, we need separate equations for collocation

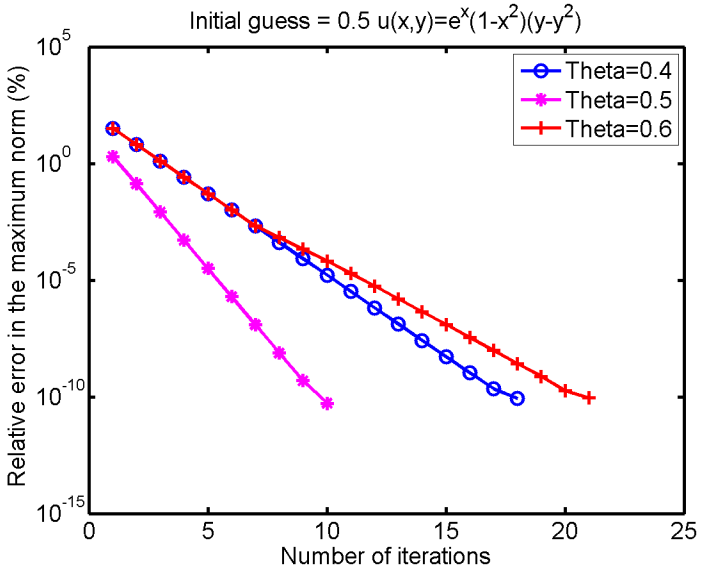


Figure 31: Nonoverlapping domain decomposition method for different values of

points that lie on the interface Γ. No additional equations are needed for collocation points that lie on homogeneous boundary. For points p_i that lie inside the domain we will use equation

$$(-\Delta u)(p_i) = f(p_i). \tag{125}$$

where as for the points that satisfy Neumann boundary condition in subdomain Ω_2 we used equation

$$\frac{\partial u_2^{k+1}}{\partial n}(p_i) = \frac{\partial u_1^{k+1}}{\partial n}(p_i). \tag{126}$$

Non-overlapping alternating method is tested w.r.t various relaxation parameter θ. The results are shown in Table **??**. It required only 10 number of iterations for $\theta = 0.5$ where as for $\theta = 0.4$ and for $\theta = 0.6$ it required more number of iterations to converge. Therefore the optimal value of θ is considered to be $\theta = 0.5$ for this problem with subdivision at $x = 0$.

Table 16: Comparison of relative error in the maximum norm in(%) for nonoverlapping domain problem for different θ values

Iteration	$\theta = 0.4$	$\theta = 0.5$	$\theta = 0.6$
1	3.192E+01	1.94E+00	3.192E+01
2	6.38E+00	1.32E-01	6.38E+00
3	1.27E+00	8.20E-03	1.27E+00
4	2.55E-01	5.09E-04	2.55E-01
5	5.11E-02	3.16E-05	5.11E-02
6	1.02E-02	1.96E-06	1.02E-02
7	2.04E-03	1.22E-07	2.04E-03
8	4.09E-04	7.52E-09	6.60E-04
9	8.17E-05	4.93E-10	2.14E-04
10	1.63E-05	5.26E-11	6.52E-05
11	3.27E-06	-	1.92E-05
12	6.54E-07	-	5.54E-06
13	1.31E-07	-	1.57E-06
14	2.61E-08	-	4.43E-07
15	5.23E-09	-	1.24E-07
16	1.05E-09	-	3.44E-08
17	2.25E-10	-	9.55E-09
18	8.62E-11	-	2.61E-09
19	-	-	7.51E-10
20	-	-	1.80E-10
21	-	-	8.98E-11

6.3.2 Slanted Interface in a Rectangular Domain

We tested the below problem with slanted subdivision at interface on the domain $\Omega = [-1, 1] \times [0, 1]$

$$\begin{cases} -\Delta u = f \quad \text{for} \quad \Omega \\ u = 0 \quad \text{on} \quad \partial\Omega \end{cases} \tag{127}$$

which has the exact solution:

$$u(x, y) = x^3(1 - x^2)(y - y^2) \tag{128}$$

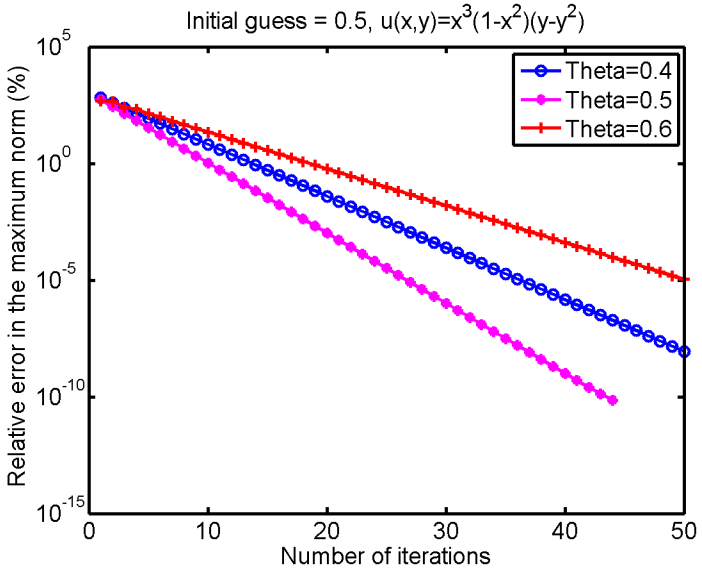


Figure 32: Relative error in the maximum norm(%) for problem with slanted nonoverlapping subdomains

For this subdivision nonlinear mappings F_1 and F_2 were used to map the reference domain $\widehat{\Omega} = [0,1] \times [0,1]$ to physical subdomains Ω_1 and Ω_2 respectively.

$$F_1(u,v) = (0,0)L_1(u,v) + (\frac{1}{2},0)L_2(u,v) + (-\frac{1}{2},1)L_3(u,v) + (0,1)L_4(u,v) \quad (129)$$

$$F_2(u,v) = (\frac{1}{2},0)L_1(u,v) + (1,0)L_2(u,v) + (1,1)L_3(u,v) + (-\frac{1}{2},1)L_4(u,v). \quad (130)$$

where,

$$\begin{aligned} &L_1(u,v) = (1-u)(1-v), \quad L_2(u,v) = u(1-v), \\ &L_3(u,v) = uv, \quad L_4(u,v) = (1-u)v \end{aligned} \quad (131)$$

Like vertical division problem, we solved this problem by solving subproblems(**??**) and (**??**) in the framework of IGA-Collocation and by assuming initial guess $\lambda^0 = 0.5$. Non-overlapping alternating method with slanted subdivision is tested w.r.t various relaxation parameter θ. The results are shown in Fig.**??**. Like in vertical division case, $\theta = 0.5$ was the optimal value and it took 44 iterations to get the solution with

accuracy 10^{-11}.

CHAPTER 7: ALTERNATING METHOD IN IGA COLLOCATION FOR ELASTICITY PROBLEMS

In the previous chapter, alternating method using IGA-Collocation was introduced for one and two dimensional elliptic boundary value problems. in this chapter, this method is extended to linear elasticity and presented how to use the same method when coupled elliptic equations are involved in the problem.

7.1 Preliminaries

In this section, we briefly introduce the notation, terminologies and definitions involved in linear elasticity. For any displacement vector $\{u\} = \{u_1(x, y), u_2(x, y)\}^T$, the stress field is defined by $\{\sigma\} = \{\sigma_x, \sigma_y, \tau_{xy}\}^T$ and the strain field is defined by $\{\varepsilon\} = \{\varepsilon_x, \varepsilon_y, \gamma_{xy}\}^T$.$\sigma_x$ and σ_y are normal stress and τ_{xy} is shear stress. Similarly, ε_x and ε_y are normal strain and γ_{xy} is shear strain. The relation between strain-displacement and stress-strain is given by

$$\{\varepsilon\} = [D]\{u\}, \quad \{\sigma\} = [E]\{\varepsilon\}, \tag{132}$$

where $[D]$ is the differential operator matrix and $[E]$ is the material stiffness matrix given by 3×3 symmetric positive definite matrix of material constants. Matrix $[D]$

is given by

$$[D] = \begin{pmatrix} \frac{\partial}{\partial x} & 0 \\ 0 & \frac{\partial}{\partial y} \\ \frac{\partial}{\partial y} & \frac{\partial}{\partial x} \end{pmatrix} \tag{133}$$

Generally there are two types of problems that are involved in linear elasticity, *plain stress* and *plain strain*. Plane stress is defined to be a state of stress in which normal stress σ_z and shear stresses τ_{xz} and τ_{yz} are assumed to be zero.Plane strain is defined to be a state of strain in which normal strain ε_z and shear strains γ_{xz} and γ_{yz} are assumed to be zero.

Material that have identical values of property in all directions are called Isotropic materials. These materials have two components, Young's modulus(E) and Poisson's ration(ν). The range of Poisson's ratio is $0 \leq \nu \leq 0.5$. The material stiffness matrix $[E]$ for an isotropic elastic body is as follows:

$$[E] = \frac{E}{1-\nu^2} \begin{pmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{pmatrix} \quad \text{for plane stress,} \tag{134}$$

$$[E] = \begin{pmatrix} \zeta + 2\mu & \zeta & 0 \\ \zeta & \zeta + 2\mu & 0 \\ 0 & 0 & \mu \end{pmatrix} \quad \text{for plane strain,} \tag{135}$$

where

$$\mu = \frac{E}{2(1+\nu)}, \quad \zeta = \frac{\nu E}{(1+\nu)(1-2\nu)} \tag{136}$$

The equilibrium equation of elasticity are

$$[D]^T\{\sigma\}(x,y) + \{f\}(x,y) = 0, \quad (x,y) \in \Omega, \tag{137}$$

where $\{f\} = \{f_1(x,y), f_2(x,y)\}^T$ is the vector of internal sources representing the body force per unit area.This equilibrium equation can be written in terms of displacement in the following way,

$$\{\sigma\} = [E]\{\varepsilon\},$$

$$\{\sigma\} = [E][D]\{u\},$$

$$\{\sigma\} = \frac{E}{1-\nu^2}\begin{pmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{pmatrix}\begin{pmatrix} \frac{\partial}{\partial x} & 0 \\ 0 & \frac{\partial}{\partial y} \\ \frac{\partial}{\partial y} & \frac{\partial}{\partial x} \end{pmatrix}\begin{pmatrix} u_1(x,y) \\ u_2(x,y) \end{pmatrix}.$$

The Navier equations for plane stress are

$$\frac{E}{1-\nu^2}[\frac{\partial^2 u_1}{\partial x^2} + \frac{(1+\nu)}{2}\frac{\partial^2 u_2}{\partial x \partial y} + \frac{(1-\nu)}{2}\frac{\partial^2 u_1}{\partial y^2}] + f_1(x,y) = 0 \tag{138}$$

$$\frac{E}{1-\nu^2}[\frac{(1-\nu)}{2}\frac{\partial^2 u_2}{\partial y^2} + \frac{(1+\nu)}{2}\frac{\partial^2 u_1}{\partial x \partial y} + \frac{\partial^2 u_2}{\partial y^2}] + f_2(x,y) = 0 \tag{139}$$

Similarly, the Navier equations for plane strain are

$$(\zeta + 2\mu)\frac{\partial^2 u_1}{\partial x^2} + (\zeta + \mu)\frac{\partial^2 u_2}{\partial x \partial y} + \mu\frac{\partial^2 u_1}{\partial y^2} + f_1(x,y) = 0 \tag{140}$$

$$\mu\frac{\partial^2 u_2}{\partial y^2} + (\zeta+\mu)\frac{\partial^2 u_1}{\partial x \partial y} + (\zeta+2\mu)\frac{\partial^2 u_2}{\partial y^2} + f_2(x,y) = 0 \tag{141}$$

Three dimensional elasticity problems are complex to solve, thus we reduce the dimension in different ways such as planar, axisymmetric, shell, plate beam and bar models. Here we have assumed $u_z = 0$. Therefore in plane stress we have $\sigma_z = \tau_{yz} = \tau_{zx} = 0$ and in the case of plane strain we have $\epsilon_z = \gamma_{yz} = \gamma_{zx} = 0$.

7.2 Alternating Method for Elasticity Problems

To solve Navier's equation either in plane stress or plane strain case, the components of the displacement vector are written in terms of basis functions, $\phi_i(x,y), i = 1, ..., N$,

$$\{u(x,y)\} = \begin{cases} u_1(x,y) = \sum_{i=1}^{N} c_i \phi_i(x,y), \\ u_2(x,y) = \sum_{i=1}^{N} c_{i+N} \phi_i(x,y) \end{cases} \tag{142}$$

where $c_i (i = 1, 2, ..., 2N)$ are called the amplitudes of the basis functions and $\phi_i(x,y) = (N_r(u) \times M_s(v)) \circ G^{-1}$, is the tensor product of B-spline basis functions $N_r(u)$ and $M_s(v)$ in the u- and v-direction respectively and $G : [0,1] \times [0,1] \longrightarrow \Omega$ is a geometric map. Collocation points $p_i = G(u_i, v_i)$ are chosen using the tensor-product of *Greville abscissae* defined in chapter 3. Collocation equations for (**??**) are given by

$$[D]^T\{\sigma\}(p_i) + \{f\}(p_i) = 0, \quad \text{for} \quad \text{collocation points} \quad p_i, i = 1, 2, ..., N. \tag{143}$$

We divide a domain into overlapping subdomains and solve (**??**) in each subdomain and at every step keep updating the solution at interior boundaries like it was done in previous chapters.

7.3 Numerical Examples

This alternating method is tested for both singular and nonsingular problems in two dimensions. Non-singular problem was tested on a rectangular domain where as singular problem was tested on a wedge shaped domain with singularity of intensity $\lambda = 0.5$. Plane stress Navier's equations is used for nonsingular case and plane strain Navier's equations is used for singular case.

7.3.1 Non-singular Two-dimensional Problem on Rectangular Domain

This problem is tested on domain $\Omega = [0,1] \times [0,1]$ and the isotropic material is assumed to have material constant $E = 1000$ and $\nu = 0.3$.

$$\begin{cases} [D]^T\{\sigma\}(x,y) + \{f\}(x,y) = 0, \quad (x,y)\epsilon\Omega \\ u_1(x,y) = 0, \quad u_2(x,y) = 0, \quad \text{on} \quad \partial\Omega. \end{cases} \tag{144}$$

which has the exact solution:

$$u(x,y) = \begin{cases} u_1(x,y) = x(1-x)y^3(1-y), \\ u_2(x,y) = y(1-y)x^3(1-x) \end{cases} \tag{145}$$

On the reference domain $\widehat{\Omega}$, we started with 9 basis functions of degree 5 in ξ-direction and 4 basis functions of degree 3 in η-direction which gave us total 36 basis functions by taking their tensor products. For collocation points, Greville absciaase were used in both directions which gave us 36 collocation points by tensor product. Since there are two components in the equation so $2(36) = 72$ collocation points are needed to solve this problem. Therefore these 36 points were used for both components. The size of the stiffness matrix was 72×72.

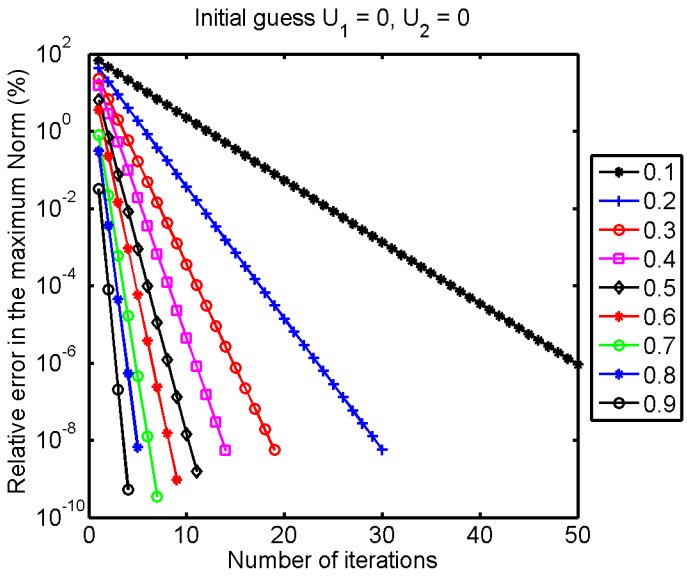


Figure 33: Relative error in the maximum norm in(%) for different overlapping domain sizes for nonsingular elasticity problem

The alternating scheme is repeated for two different overlapping domains with various sizes of overlapping regions. Like in one elliptic boundary value problem, the number of iterations required is dependent upon the size of the overlapping area. Fig. **??** shows that larger the overlapping area, it requires lesser number of iterations.

7.3.2 Two-dimensional Singular Problem on Wedge Shaped Domain

This problem was solved on wedge shaped domain $\Omega = \{(r, \theta) : r < 2, -\alpha \leq \theta \leq \alpha\}, \quad 0 \leq \alpha \leq 90^\circ$ for plane strain case, where α is the wedge angle and the body force was neglected here. The isotropic material was assumed to have material constant $E = 1000$ and $\nu = 0.3$.

$$[D]^T\{\sigma\}(x, y) + \{f\}(x, y) = 0, \quad (x, y)\epsilon\Omega, \tag{146}$$

which has the exact solution in the form:

$$u(x,y) = \begin{cases} u_1(x,y) = u_r \cos\theta - u_\theta \sin\theta, \\ \\ u_2(x,y) = u_r \cos\theta + u_\theta \sin\theta, \end{cases} \tag{147}$$

where

$$\begin{aligned} u_r(r,\theta) &= \tfrac{r^\lambda}{2G}\{-(\lambda+1)f(\theta) \\ u_\theta(r,\theta) &= \tfrac{r^\lambda}{2G}\{-f'(\theta), \end{aligned} \tag{148}$$

and $f(\theta) = \sin(\lambda+1)\theta, \quad \lambda = \frac{90^\circ}{\alpha} - 1$. Here we have imposed non-homogeneous Dirichlet boundary condition along the entire boundary and zero boundary condition at origin for displacement vector.

It's not difficult to show that

$$\frac{\partial^2 u_1}{\partial x^2} = -\frac{\lambda(1+\lambda)(\lambda-1)}{2G} r^{\lambda-2} \sin(\lambda-2)\theta, \tag{149}$$

$$\frac{\partial^2 u_1}{\partial xy} = -\frac{\lambda(1+\lambda)(\lambda-1)}{2G} r^{\lambda-2} \cos(\lambda-2)\theta, \tag{150}$$

$$\frac{\partial^2 u_1}{\partial y^2} = +\frac{\lambda(1+\lambda)(\lambda-1)}{2G} r^{\lambda-2} \sin(\lambda-2)\theta, \tag{151}$$

$$\frac{\partial^2 u_2}{\partial x^2} = -\frac{\lambda(1+\lambda)(\lambda-1)}{2G} r^{\lambda-2} \cos(\lambda-2)\theta, \tag{152}$$

$$\frac{\partial^2 u_2}{\partial xy} = -\frac{\lambda(1+\lambda)(\lambda-1)}{2G} r^{\lambda-2} \sin(\lambda-2)\theta, \tag{153}$$

$$\frac{\partial^2 u_2}{\partial y^2} = +\frac{\lambda(1+\lambda)(\lambda-1)}{2G} r^{\lambda-2} \cos(\lambda-2)\theta \tag{154}$$

which satisfies equation(**??**) if body force vector $\{f\} = 0$. Mapping F_1 maps parameter space $\widehat{\Omega} = [0,1] \times [0,1]$ to wedge subdomain $\Omega_1 = \{(r,\theta) : r < 1, -\alpha \leq \theta \leq \alpha\}$ and F_2 maps it to subdomain $\Omega_2 = \{(r,\theta) : 0.5 < r < 2, -\alpha \leq \theta \leq \alpha\}$, here wedge angle was taken as $\alpha = 60^\circ$.

CHAPTER 8: CONCLUDING REMARKS AND FUTURE WORK

We have applied modified B-spline basis functions to IGA-Collocation method in both one-dimensional non-singular as well as singular problems and got almost true solutions. Also this approach was tested to two-dimensional non-singular as well as singular problems. In the future research work those methods proposed in this dissertation will be extended to solve elliptic PDEs on non-convex domains like L-shaped, cracked domain and polygonal domain.

Similarly, Schwarz alternating method in the framework of IGA-Collocation will also be extended to solve elliptic PDEs on non-convex domains like L-shaped,cracked domain and polygonal domain. So far this method was applied in elasticity for plane stress and plane strain cases. In future, it will be tested for shell, plate problems as well.

Even though the test problems that were tested in this dissertation contained one singularity, we expect that the method can easily be applied to problems with multiple singularities. Schwarz alternating IGA-Collocation method can be extended to deal with oscillating singularities of the type $r^{\lambda}\cos(\varepsilon \log r), \quad 0 < \lambda, \varepsilon < 1$.

Direct solvers were used to solve problems with Schwarz alternating method in the frame work of IGA-Collocation. In future iterative solvers will also be tested. We will also extend IGA-Collocation approach to general domain decomposition method.

REFERENCES

[1] Y. Bazilevs, L. Beirao Da Veiga, J.A. Cottrell, T.J.R. Hughes, and G. Sangalli,Isogeometric analysis: Approximation, stability and error estimates for h-refined meshes, Mathematical Models and Methods in Applied Sciences (2006), 1031-1090.

[2] J.A. Cottrell, T.J.R. Hughes, and Y. Bazilevs, Isogeometric analysis: Toward integration of CAD and FEM, 2009.

[3] T.J.R. Hughes, J.A. Cottrell, T.J.R. Hughes, and Y. Bazilevs, Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement, Comput. Methods Appl. Mech. Engrg.(2005), 4135-4195.

[4] H.-S. Oh, J.W. Jeong, and W.T. Hong, The generalized product partition of unity for the meshless methods, J. Comp. Phy. (2010), 1600-1620.

[5] H.-S. Oh, J.W. Jeong, and J. G. Kim, The reproducing singularity particle shape function for problems containing singularities, Comput Mech (2007), 135-157.

[6] H.-S. Oh, J. G. Kim, and W.T. Hong, The piecewise polynomial partition of unity shape functions for the generalized finite element methods, Comput. Methods Appl. Mech. Engrg. (2008), 3702-3711.

[7] H.-S. Oh, J.G. Kim, and J.W. Jeong, The closed form reproducing polynomial particle shape functions for meshfree particle methods, Comput. Methods Appl.Mech. Engrg. (2007), 3435-3461.

[8] D.F. Rogers, An introduction to NURBS, Academic Press, 2001.

[9] Piegl and Tiller, The NURBS book, second edition, Springer, 1995.

[10] H.-S. Oh and I. Babuška, The method of auxiliary mapping for the finite element solutions of plane elasticity problems containing singularities, Journal of Computational Physics (1995), 193-212.

[11] Höllig, K : Finite Element methods with B-Spline, SIAM, 2003.

[12] Han W, Meng X Error analysis of reproducing kernel particle method. Comput. Methods Appl.Mech. Engrg. (2001) 190:6157-6181.

[13] P. Lancaster, K. Salkauskas, Curve and Surface Fitting, An Introduction, Academic Press, San Diego, 1986.

[14] D. Shepard, A two-dimensional function for irregularly spaced data, in: ACM National Conference, 1968, 517-524.

[15] Krishan P.S. Gahalaut, Isogeometric Analysis:Condition Number Estimates and Fast Solvers in:Dissertation, 2013: Johannes Kepler University Linz, Austria.

[16] Hoonjoo Kim, S-J Lee,H.-S. Oh, Numerical Methods for Accurate Finite Element Solutions of Elliptic Boundary Value Problems Containing Singularities, 2002.

[17] P.G. Ciarlet, Basic error estimates for elliptic problems, North-Holland, 1991.

[18] H.-S. Oh and J.W. Jeong, Construction of C^1-basis functions for numerical solutions of the fourth-order partial differential equations, to appear.

[19] Schwarz H.A. úber einen grenzbergang durch alternierendes verfahren. Vierteljahrsschrift der Naturforschenden Gesellschaft in Zurich, 15: pp. 272-286, 1870.

[20] Pierre-Louis Lions. On the Schwarz alternating method. II. In Tony Chan, Roland Glowinski, Jacques Périaux, and Olof Widlund, editors, Domain Decomposition Methods. Second International Symposium on Domain Decomposition Methods, pages 47-70, Philadelphia, PA, 1989. SIAM. Los Angeles, California, January 14-16, 1988.

[21] Alfio Quarteroni and Alberto Valli, Domain Decomposition Methods for Partial Differential Equations book,Editors G. H. Golub, R. Jeltsch, W. A. Light, K. W Morton, E. Sùli, Oxford, 1999.

[22] A. Toselli and O. Widlund, Domain Decomposition Methods- Algorithms and Theory, pages 21-23, Springer, 2005