UNIVERSITY OF CALIFORNIA, SAN DIEGO

Adaptive Multilevel Numerical Methods with Applications in Diffusive Biomolecular Reactions

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

in

Mathematics

by

Burak Aksoylu

Committee in charge:

Professor Michael J. Holst, Chair
Professor Randolph E. Bank
Professor Philip E. Gill
Professor J. Andrew McCammon
Professor J. Ben Rosen

2001

The dissertation of Burak Aksoylu is approved, and it is acceptable in quality and form for publication on microfilm:

____

____

____

____

____

Chair

University of California, San Diego

2001

To my wife Türkan, my father Özcan, my mother Meral, and sisters Tuba and Erva.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

This work consists of the highlights of my graduate career as a member of the scientific computation (SCICOMP) group in the Department of Mathematics at the University of California, San Diego. A pure mathematician by training, and an applied mathematician at heart, I have always had great interest in the applications of mathematics in science, in particular life sciences. My interest in science stems from my childhood experiences in my father's dental clinic. Observing numerous surgeries and operations triggered a deep curiosity in science. During the early years of my graduate career, I was in search for a field where I could contribute to science with my pure math background, and quench my thirst to apply mathematics to real life problems through the very "adaptive" nature of mathematics.

My curiosity has yielded a journey in the interface of mathematics and science. I can say wholeheartedly that the most remarkable incident recorded in the captain's log is getting to know my advisor, Mike Holst. Mike, my mentor and dear friend, has always been an inspirational lighthouse. Two characteristics, seldomly found in the same person simultaneously, nourish his dynamism and scientific intellect. These are his willingness to receive: to explore and learn; and his passion to give: to guide and teach. These amount to his succinct statement "I would like to contribute to the human understanding of mathematics."

In my opinion, Mike carries the two noblest values of a true scientist: honesty and humbleness. I have observed his hard work and done my best to keep up with the "Holst work unit", a unit which he established while in graduate school. His positive, reinforcing, and endlessly promoting attitude has kept me afloat. I have always enjoyed his thought provoking conversation, and often chased after him for more interaction. We have shared many delights ranging from Turkish coffee to Linux.

Mike generously shared his software, the Finite Element ToolKit (FEtk), with me. He donated his ten years of coding effort, in the form of this excellent piece of software. It would suffice to understand his dedication to science by his decision that FEtk should be distributed freely. We firmly believe that the ever evolving FEtk will continue to make an impact in the understanding of the phenomena in general relativity, solid mechanics, and most importantly life sciences. The ancestor and foundation of the FEtk, MCLab (Manifold Code in MATLAB), has been a blessing, by which I tested all the algorithms and solver prototypes. I am pleased to be giving back to the source I have benefitted from. I am excited that the insight I gained from the implementation of the hierarchical basis methods, will fundamentally change the solver component of the FEtk, and will help the FEtk to be a truly adaptive code.

I feel extremely fortunate and immensely proud to be the first math Ph.D. student under Mike's supervision. I see our three year interaction as an initial mesh, and look forward to enrichening it by further refinement, so that I can fully appreciate his "multilevel" impact on my academic career, and in my life at large.

It was a great opportunity to be located several offices down the hall from Dr. Randy Bank, the pioneer of hierarchical basis methods. As a senior scientist in the hierarchy of the SCICOMP group, Dr. Bank has always encouraged me to ask questions without making me feel any "hierarchy". We benefitted from his vast mathematical insight, and his numerous years of software (PLTMG) development experience.

I have always found Dr. Philip Gill being pleasant, caring, and supportive towards all students. His frequent stop-bys, and his sincere interest in students' lives, including health problems, sets a benchmark for how much a professor should care about students. I have always admired his social skills, articulate language, and his detail oriented nature.

I wish to express my gratitude to the La Jolla Interfaces in Science (LJIS) training program for sponsoring me for the last two years. LJIS was the gateway to interdisciplinary work with the McCammon research group. The group's unique atmosphere has opened new horizons, introduced me to a different way of thinking, and last but not least, helped me to interact with life sciences where my heart has always been. I came to understand the "diffusive" nature of biochemistry in almost all faculties of life. Dr. McCammon appreciates how mathematics and mathematical software are influential in life sciences. This aspect makes him an exceptional scientist. I am thankful to Dr. McCammon for giving me the chance to fulfill my yearning to contribute to the understanding of life science phenomena, and to potential treatment of devastating disorders such as Amyotrophic Lateral Sclerosis (I recently lost my primary school teacher who was very dear to me due to this illness), and Multiple Sclerosis (from which my aunt suffers).

I wish to thank Dr. Ben Rosen for attending my talks, being a vigorous listener, and asking me challenging questions.

I enjoyed the caring environment of the SCICOMP group. I will certainly miss the SCICOMP lab where I spent endless hours. I wish to thank my office mates Roummel Marcia and Jennifer Erway. Roummel has been a supportive colleague and a caring buddy. Our lengthy conversations have cooled off the research pressure on our shoulders. At times, he contributed to my proofs. The newest member of the SCICOMP group, Dr. Steve Bond, a Linux guru, was very helpful and always willing to listen to me. We "bonded" well.

My special thanks go to Dr. Panayot Vassilevski. The short visit I made to the Lawrence Livermore National Laboratory was a thrilling experience. His welcoming character helped me a lot in understanding the wavelet literature and implementing the wavelet modified hierarchical basis method. I also want to thank Dr. Petr Vanek for hosting me at UCLA and for his philosophical comments on algebraic multigrid.

This work would have not be possible without my wife, Türkan. Her cherishing character, everyday patience, and bearing with my ups and downs are the integral factors behind my career. Her eagerness to listen matured my articulation in conveying research ideas. I want to thank her with all my heart.

I would like to acknowledge the continuous support and encouragement of my family. My father, Özcan Aksoylu, being a perfectionist, was hardly satisfied with ordinary work. He encouraged me to fulfill my potentials, and has pushed me to be a frontier in whatever I do. My mother, Meral Aksoylu, adventurist in spirit, has always motivated me to come up with innovative ideas. She was the driving force for my graduate career. My sister Tuba, has provided me with a home in Chicago, and was always willing to talk to me on the phone. And sincere thanks go to my little sister Erva, for the support she provided from İstanbul.

I wish to thank numerous friends, colleagues, and the Turkish community. They were kind enough to carry me and my work.

VITA

October 30, 1971	Born, İzmir, TURKEY
1989-1993	B.Sc., Science Education, Mathematics Teaching, Middle East Technical University, Ankara, TURKEY
1991-1993	B.Sc. (double major), Mathematics, Middle East Technical University, Ankara, TURKEY
1994-1996	M.Sc., Applied Mathematics, University of California, Irvine
1996-2001	Ph.D., Mathematics, University of California, San Diego

ABSTRACT OF THE DISSERTATION

Adaptive Multilevel Numerical Methods with Applications in Diffusive Biomolecular Reactions

by

Burak Aksoylu
Doctor of Philosophy in Mathematics

University of California San Diego, 2001

Professor Michael J. Holst, Chair

We consider adaptive multilevel numerical methods to solve nonlinear elliptic partial differential equations (PDEs) by using the finite element method (FEM). Our motivation is to provide high performance, provably good numerical solvers for the Poisson-Boltzmann equation (PBE), a three-dimensional (3D) second order nonlinear symmetric elliptic PDE arising in biophysics. The PBE is central to the Brownian dynamics simulations of diffusion-influenced biomolecular reactions. Interaction between enzymes and substrates are examples of such reactions, and they are of great interest, partially due to their roles in diseases like Acquired-Immuno-Deficiency Syndrome, and Alzheimer's Disease.

The core of this work is the performance analysis of preconditioners on adaptive meshes. We present the approximation theory framework and extend two-dimensional (2D) optimality of the Bramble-Pasciak-Xu (BPX) preconditioner to the 3D red-green refinement. By using K-functionals, a simple suboptimal BPX lower bound proof is given. The hierarchical basis (HB) methods are superior to the multigrid methods in local refinement scenarios where the HB methods promise optimal computational and storage complexity. We concentrate on the wavelet modified hierarchical basis (WMHB) method, which is a stabilization of the HB method in the sense that it provides a stable Riesz basis. We outline the

convergence theory of the WMHB, and prove that it is an optimal preconditioner for the 2D red refinement. Without the presence of the continuously differentiable PDE coefficients, we provide suboptimal bounds by using the H^1-stability of the L_2-projection.

Local quasiuniformity of the mesh is the integral part of H^1-stability. We analyze conditions on locally refined meshes that would give an H^1-stable L_2-projection. We study geometric mesh properties related to local quasiuniformity such as simplex size and neighbor relations. We prove that two of the well-known refinement routines, 2D newest vertex and 3D marked tetrahedron bisections, provide locally quasiuniform meshes.

We give explicit formulation of the WMHB block stabilization and discuss the sparse matrix multiplication algorithms. After a careful discussion of algebraic and logical complexity, we point out that the implementation of the WMHB method can only be optimal for symmetric PDEs. We exhibit numerical experiments of the above methods by giving iteration counts.

Chapter 1

Introduction

1.1 Motivation: promoting the quantitative study of biological problems

Recent breakthroughs in biological sciences have showed that computational methods can play an extremely important role in drug development. Progress depends heavily on understanding and predicting the complex behavior and function of biomolecules. A partial list of the devastating disorders in which a small number of biomolecules play key roles includes Acquired Immuno-Deficiency Syndrome (AIDS), Amyotrophic Lateral Sclerosis (ALS), and Multiple Sclerosis (MS), as well as Alzheimer's Disease.

The human immuno-deficiency virus (HIV), believed to cause AIDS, uses the so-called HIV protease for reproduction. In drug therapy, protease inhibitors are designed to interfere with the function of the HIV protease, thereby disrupting viral replication. By modeling the *electrostatic behavior* of the HIV protease on a computer, scientists can predict the effectiveness of a candidate drug molecule by simulating its interaction with the HIV protease. Molecules which exploit electrostatics to aid in binding to the active site in the protease would likely be the basis of the most effective drug. Other modeling candidates include superoxidedismutase, which has been implicated in ALS, and acetylcholinesterase (AChE), known to have a role in Alzheimer's disease. It has also been suggested recently that dismylination of the central and peripheral nervous systems in humans, which leads to MS, might be understood through the use of electrostatic models, coupled to elasticity models of the myelin sheath.

The biochemistry community is intensely interested in the complicated range of interactions between enzymes and their substrates such as reactions between AChE and acetylcholine (ACh), due to their role in these and other diseases. These reactions can be characterized as *diffusion-influenced bimolecular reactions.* To simulate such complex systems, researchers rely on a number of techniques, including molecular dynamics and Brownian dynamics (BD) simulations. In BD, the stochastic trajectory of a particle is simulated by a series of small displacements chosen from a distribution which is the short time solution of Smoluchowski equation. This method is sufficiently general to model hydrodynamic interactions and arbitrary inter- and intramolecular forces. In order to make such simulations possible, the discrete nature of the ionic solutions in which these interactions take place is often approximated with continuum mechanics, leading to the *Poisson-Boltzmann* equation (PBE). In many cases, the accuracy of the simulations is limited by the enormous amount of computer time required to solve the PBE, in order to account for the long-range electrostatic forces in the simulation. Recent progress has been made on improving the efficiency of numerical methods for the PBE [14, 22], but much work remains.

High performance numerical solvers are needed to handle large sparse systems arising from the discretization of the PBE and complex geometry around the enzymes. This work covers the design and implementation of adaptive multilevel finite element methods with *provably good* complexity and error bounds. They provide *orders of magnitude* improvement in solution time and space over current methods. These methods are implemented as the class library extensions to the Finite Element ToolKit (FEtk).

1.2 Motivational problem: the PBE

The PBE is a second order nonlinear PDE, fundamental to Debye-Hückel continuum electrostatic theory [32]. The PBE [3, 21] determines a dimensionless potential $u(x) = e_c\Phi(x)k_B^{-1}T^{-1}$ around a charged biological structure immersed in a salt solution, where $\Phi(x)$ is the electrostatic potential at $x \in \mathbb{R}^d$, with $d = 2$ or $d = 3$. For a $1:1$ electrolyte, the PBE can be written as

$$-\nabla \cdot (\epsilon(x)\nabla u(x)) + \bar{\kappa}^2(x)\sinh(u(x)) = \left(\frac{4\pi e_c^2}{k_B T}\right)\sum_{i=1}^{N_m} z_i\delta(x - x_i), \qquad u(\infty) = 0.$$

In this equation, the dielectric ϵ jumps by one or two orders of magnitude at the interface between the charged structure and the solvent, and the modified Debye-Hückel parameter $\bar{\kappa}$ (a function of the ionic strength of the solvent) is discontinuous at an ion exclusion region surrounding the biological structure. The structure itself (e.g., a biological molecule, or a membrane) is represented implicitly by ϵ and $\bar{\kappa}$, as well as explicitly by the N_m point charges $q_i = z_i e_c$ at the positions $\bar{x}_i$ (with the distributions $\delta(\cdot)$ reflecting the point-charge behavior). A comprehensive numerical and theoretical treatment of the PBE can be found in [22]. This work contains the analysis of multilevel methods with discontinuous coefficients, methods for nonlinear equations, and discussion of the linearization of the PBE.

1.3 Model problem and preconditioning

The following model problem satisfactorily encapsulates the fundamental features of the PDEs under consideration and suffices to cover the general case.

$$-\nabla \cdot (p\ \nabla u) + q\ u = f, \quad u \in H_0^1(\Omega). \tag{1.1}$$

Here, $f \in L_2(\Omega)$, p, $q \in L_\infty(\Omega)$ and $p : \Omega \to L(\mathbb{R}^d, \mathbb{R}^d)$, $q : \Omega \to \mathbb{R}$, where p is a symmetric positive definite (SPD) matrix, and q is nonnegative. Let $\mathcal{T}_0$ be a shape regular and quasiuniform initial partition of Ω into a finite number of

$d+1$ simplices, and generate $\mathcal{T}_1, \mathcal{T}_2, \ldots$ by locally or uniformly refining the initial partition.

Let $\mathcal{S}_j$ be the simplicial linear C^0 finite element (FE) space corresponding to $\mathcal{T}_j$ equipped with zero boundary values. The set of nodal basis functions for $\mathcal{S}_j$ is denoted by $\left\{\phi_i^{(j)}\right\}_{i=1}^{N_j}$ where $N_j = \dim\, \mathcal{S}_j$ is equal to the number of interior nodes in $\mathcal{T}_j$. We have a nested sequence of FE spaces

$$\mathcal{S}_0 \subset \mathcal{S}_1 \subset \ldots \subset \mathcal{S}_j \subset \ldots \subset H_0^1(\Omega).$$

$\mathcal{S}_j$ is used within the framework of conforming FE methods for discretizing (1.1). We denote the bilinear form and the functional acting on the forcing function as

$$a(u,v) = \int_\Omega (p\ \nabla u \cdot \nabla v) + q\ u\ v)\ dx, \quad b(v) = \int_\Omega f\ v\ dx, \quad u, v \in H_0^1(\Omega),$$

and consider the following approximate problem: find $u \in \mathcal{S}_j$, such that

$$a(u,v) = b(v), \quad \forall v \in \mathcal{S}_j. \tag{1.2}$$

If the coefficient vector of u, with respect to the nodal basis $\left\{\phi_i^{(j)}\right\}_{i=1}^{N_j}$ is represented by the vector $(u_1, \ldots, u_{N_j})$,

$$u = \sum_{i=1}^{N_j} u_i\ \phi_i^{(j)},$$

then (1.2) reduces to a linear system

$$A\ u = b, \tag{1.3}$$

with a SPD sparse matrix A, consisting of the entries $A_{kl} = a(\phi_k^{(j)}, \phi_l^{(j)})$, and $b_k = b(\phi_k^{(j)})$.

The matrix A is ill-conditioned, with the condition number $\kappa(A)$ of the system (1.3) growing $O(2^{2j})$ as $j \to \infty$. It is imperative to improve the condition of (1.3) by transforming the system to an equivalent one, namely by preconditioning

$$(C\ A)\ u = C\ b,$$

where $\kappa(C^{1/2}\ A\ C^{1/2}) \ll \kappa(A)$. Moreover, the preconditioning matrix C has to be positive definite, and in some sense simple. One possible way to implement the above strategy is to determine a positive definite matrix B, having the following two properties:

- B^{-1} can efficiently be computed (usually, $O(N_j)$ is the desirable bound for the number of arithmetical operations when solving a linear system with coefficient matrix B),

- A and B are "almost" spectrally equivalent, i.e.

$$\lambda_B(Bu, u) \leq (Au, u) \leq \Lambda_B(Bu, u), \quad u \in \mathbb{R}^{N_j}, \tag{1.4}$$

  with two positive constants λ_B, Λ_B with a small ratio $\frac{\Lambda_B}{\lambda_B}$.

Since $\kappa(B^{-1/2}\ A\ B^{-1/2}) \leq \frac{\Lambda_B}{\lambda_B}$, then $C = B^{-1}$ will be a good choice. Observe that

$$\begin{aligned}(Au, u) &= \sum_{k,l=1}^{N_j} a(\phi_k^{(j)}), \phi_l^{(j)})\ u_k\ u_l \\ &= a(u, u) \approx \|u\|_{H^1}^2.\end{aligned}$$

Consequently, if (Bu, u) can also be interpreted in terms of some norm of u, then (1.4) reduces to a comparison of different norms on $\mathcal{S}_j$, a problem which can be dealt with by function-theoretic methods. We establish optimality of preconditioners by constructing an equivalence between the norm defined by the preconditioner and the underlying H^1-norm.

1.4 Contributions of this dissertation

The driving force behind the dissertation research is to numerically solve the PBE with optimal computational and storage complexity, by employing the FE method with an adaptive refinement strategy. Adaptive refinement becomes the major requirement because of the factors such as complex geometry of the mesh (molecules), discontinuities in the coefficient functions, and the delta functions of the forcing function in the definition of the PBE. Conventional multilevel solvers like multigrid (MG), employed in the uniform refinement scenario, lose both computational and optimal storage complexity due to adaptive refinement.

Next, we focus on multilevel numerical methods with various adaptive refinement strategies that would preserve optimality. We are not simply interested in empirical optimality, but rather we intend to exhibit *provably good* numerical methods. This means that rigorous complexity statements and error estimates are proved about the method. In Chapter 2, our contribution is the proof of the optimality of the BPX preconditioner for the 3D red-green refinement routine. Optimality is demonstrated by the equivalence of the BPX-norm and the underlying H^1-norm. This rigorous proof involves tools from the approximation theory, primarily introduced to the FE community by the German school. We give a summary of the relations between the FE space and approximation spaces and also exhibit equivalence between modulus of smoothness and K-functionals. In 3D red-green refinement routine, we prove that two tetrahedra sharing a common face will be at most 1 level apart in generation. By shape regularity of the mesh, this enables us to conclude generation bounds for edge and vertex adjacent tetrahedra. The lower bound in the norm equivalence poses more subtle technicality than that of the upper one. With the presence of an H^1-stable L_2-projection, we showed the canonical suboptimal lower bound,

$$\frac{c}{J} \, \|u\|_{BPX}^2 \leq \|u\|_{H^1}^2.$$

However, by upgrading to an H^s-stable L_2-projection, $s > 1$, we obtained the optimal lower bound by using a simple proof technique which involves flexible K-functionals from the approximation theory. Here it is ironical that the original BPX proof was upgraded to optimal bounds by realizing the fact that the Jackson estimate holds for H^s, $s > 1$. By noticing the role of $s > 1$, we might as well speculate about a potential hidden relationship between H^s-stability of the L_2-projection and the Jackson estimate. From the computational point of view, optimality of the BPX preconditioner means that the number of iterations in the solver do not increase as the problem size increases. The following issues are storage complexity and algebraic cost of the action of the preconditioner. These are proved to be optimal by Xu and Qin (see page 180 in [40]) for the quasiuniform case. However, the extension of the same result for the locally refined meshes does not seem to be present in the literature.

In order to retain both computational and storage optimality, MG methods require a sequence of subspaces which increase geometrically in dimension. This becomes impossible to attain in a local refinement strategy where only $O(1)$ many unknowns are introduced at every level. In Chapter 3, we turn our attention to the most promising candidate, namely the HB methods. The superiority of the HB methods stem from the fact that they exploit the hierarchy of the degrees of freedom. This means that only new unknowns are used, which are introduced at the very level j. This provides $O(n_j)$ storage for the $O(n_j)$ work estimate per cycle at every level, where $n_j = N_j - N_{j-1}$. However, the condition number suffers from a growth of $O(j^2)$ and $O(2^j)$ in 2D and 3D respectively. $O(n_j)$ storage per level gives rise to overall optimal storage $O(N)$, $N = N_J$, which is usually critical for biological applications where storage can easily become a primary concern. Hence, one should appreciate the breakthrough in storage provided by the HB methods. The condition number complication of the HB method is remedied by utilizing a stabilization through modifying HB functions with wavelets. The WMHB method has proven to be optimal for 2D and 3D quasiuniform meshes. In Chapter 3, we outline the convergence and stability theory behind the WMHB methods. We extend the optimality of the method to locally refined meshes by using 2D red refinement. The proof technique employed relies on the fundamental multilevel estimate, i.e. Cauchy-Schwarz (C-S) like inequality. We carefully detected the problems created by the proof technique of the C-S like estimate. Firstly, due to integration by parts, the coefficient matrix $p(x)$ must be continuously differentiable. This property is generally violated by the jumps in the definition of $p(x)$. Then we hope for the next best thing, which is the canonical suboptimal lower bound. For the suboptimal bound, H^1-stability of the L_2-projection in local refinement plays a crucial role. Secondly, the boundary strip argument employed in the proof of the C-S like estimate may not be present in red-green refinement. This is why we restrict ourselves solely to red refinement and a continuously differentiable coefficient matrix for the sake of optimality. We should emphasize that the red refinement without the green closure gives rise to FE spaces which are true subspaces of the uniformly refined FE space. Then, the optimality is established, by borrowing the estimates like inverse inequality and the C-S like estimate from the uniform refinement case.

It is well known that the L_2-projection is H^1-stable on quasiuniform meshes. It would not be unreasonable to claim that the same stability holds for locally refined meshes if certain mesh criterion is satisfied. In Chapter 4, we analyze BPS- and C-criteria for locally refined meshes with a unifying approach. Such mesh criteria relies on the fundamental assumption that the mesh is locally quasiuniform. Here, "locally" means patchwise, "patch" being the support of a basis function. We proved that both the 3D marked tetrahedron bisection routine by Joe-Liu [24] and the 2D newest vertex bisection routine by Mitchell [26] give rise to patchwise quasiuniform meshes. We should emphasize that the 3D marked tetrahedron bisection is maybe the most sophisticated routine which not only boasts to have an extremely solid mathematical foundation, but also enjoys the power of handling realistic mesh difficulties. This routine has a very strong potential to be the refinement routine for realistic problems like the PBE. We gave a survey of the well known refinement routines which do not seem to be present in the literature. We compare their performance in various categories. We discuss the number of similarity classes of the simplices together with possible shape degeneracy due to degrading of the minimum angle by repeated subdivisions. We outline conformity routines that would stop in finitely many iterations. We also analyze local quasiuniformity which constitutes the foundation of an H^1-stable L_2-projection.

In the implementation Chapter 5, our focus is the HB methods, especially the WMHB method. We carefully formalize all the submatrices used for the stabilization process. We pay close attention to the subtleties on the storage style to keep the storage complexity optimal. On the computational complexity front, we have demonstrated that the WMHB is not optimal for nonsymmetric problems. Here, what we mean exactly is the fact that the WMHB has optimal algebraic complexity which conforms to the popular belief. However, its logical complexity is suboptimal. Of course, the bigger complexity, namely, logical complexity, determines the overall computational complexity. This complication arises when we compute the $\hat{A}_{12}$ block. If the PDE is symmetric, as in the case of the PBE, the $\hat{A}_{12}$ block does not need to be stored due to symmetry. In short, we rigorously exhibited that the WMHB is computationally optimal for symmetric problems and suboptimal for nonsymmetric problems. We explicitly state three different algorithms to calculate the matrix-matrix products of various sparse matrices used in stabilization. These algorithms are specially designed for the hierarchical setting, and show little resemblance to the algorithms used for the uniform refinement case (e.g. MG algorithms). Only with this technology can one guarantee optimal computational complexity for the implementation of the HB methods.

Chapter 2

Optimality of the BPX preconditioner

In this chapter, we outline the development of optimal multilevel methods, in particular the BPX preconditioner. We give the historical development and the fundamentals of the preconditioners.

The finite element community has witnessed remarkable contributions to the construction of preconditioners for iterative solvers over the last two decades. It can be said that this era has been initiated by the two pioneers in 1980, Bank and Dupont, in an unpublished technical report, introduced a multiplicative Schwarz method called the *hierarchical basis multigrid* (HBMG) solver. In 1986, Yserentant [41] independently studied similar notions, and introduced the additive version of the same method. In 1988, Bank, Dupont, and Yserentant [7] together gave an in depth study of the HBMG method. The HB solver is conceptually simple and especially well suited to nonuniform meshes. The HB methods offer $O(n_j)$ computational complexity for each level j, and an overall $O(N)$ storage complexity. Such favorable complexities make them attractive for a broad range of realistic applications where adaptive refinement is imperative, especially in the simulation of biomolecular reactions. For a survey of the HB methods, one can see Yserentant [43].

2.1 Historical development, the fundamentals, and our contribution

Given a sequence $\mathcal{S}_0, \mathcal{S}_1, \ldots$ of successively refined FE spaces, the HB methods are based on the splitting

$$u = I_0 u + \sum_{j=1}^{J} (I_j - I_{j-1}) u \tag{2.1}$$

of the functions in a final space $\mathcal{S} = \mathcal{S}_J$. $I_j u$ denotes the FE interpolation of u. Unfortunately, the HB methods can only perform suboptimally. The FE community was in the search of an optimal solver. In 1990, Bramble, Pasciak, and Xu developed the idea to substitute the above decomposition by

$$u = Q_0 u + \sum_{j=1}^{J} (Q_j - Q_{j-1}) u, \tag{2.2}$$

where Q_j is the L_2-projection onto $\mathcal{S}_j$. This was a breakthrough which allowed the construction of optimal additive methods working not only for 2D but also for 3D.

Definition 2.1 *The BPX preconditioner C_J^{-1}, and the corresponding BPX norm are defined as*

$$(C_J^{-1} u, u') = (Q_0 u, Q_0 u') + \sum_{j=1}^{J} 2^{2jk} ((Q_j - Q_{j-1}) u, (Q_j - Q_{j-1}) u'), \tag{2.3}$$

$$\|u\|_{BPX}^2 = \|Q_0 u\|_{L_2}^2 + \sum_{j=1}^{J} 2^{2jk} \|(Q_j - Q_{j-1}) u\|_{L_2}^2, \quad k = 1. \tag{2.4}$$

The underlying Sobolev space is H^1 due to the second order elliptic PDEs under consideration. Then $k = 1$ becomes the proper choice for norm equivalence purposes. It is worth mentioning that the factors 2^{2j} are not dimension dependent. They reflect the dyadic refinement. If one reduces the element sizes by a factor of 3, then the factors would be 3^{2j}.

The lower and the upper bound in the norm equivalence to the H^1-norm determine the efficiency of the preconditioner. If both bounds come with absolute constants $c_1,\ c_2$,

$$c_1\ \|u\|_{BPX}^2 \leq \|u\|_{H^1}^2 \leq c_2\ \|u\|_{BPX}^2, \tag{2.5}$$

then we arrive at the conclusion that the preconditioner is optimal.

With regularity assumptions, Bramble, Pasciak, and Xu [13] in their original work, proved the suboptimal result

$$\frac{c_1}{J}\ \|u\|_{BPX}^2 \leq \|u\|_{H^1}^2 \leq c_2 J\ \|u\|_{BPX}^2. \tag{2.6}$$

They were able to improve the lower bound to $c_1\|u\|_{BPX}^2 \leq \|u\|_{H^1}^2$ for H^2 regular problems. Later, Zhang [45] attained the optimal upper bound. Yserentant [42] generalized (2.6) to more flexible meshes popularized by PTLMG and carefully analyzed the influence of nonuniform initial triangulations. Eventually, both Xu [39] and Zhang [45] achieved optimality results by using the FE tools.

In an upcoming section, based on the work of Bornemann [11], we will discuss exactly why the original work failed to prove optimality. This subtle analysis involves the relation of interpolation spaces with Sobolev spaces.

The German school has been the most influential in establishing optimality results. They had superior tools stemming from the approximation theory. These are well established tools that have a long history. Oswald [28, 30], a prominent member of the German school, first proved the optimality result for both 2 and 3 dimensional problems. Achieving this within the same year the BPX preconditioner was introduced, he has pioneered the usage of Besov space techniques in the FE theory. Dahmen and Kunoth [17], well known for their work in wavelets, and also Oswald, based their proofs on the fact that H^1 is the Sobolev space that coincides with certain Besov spaces and the corresponding norms are equivalent. They both utilize the so-called *Jackson estimates*. This simply means that the error of the best approximation (exact solution minus the FE solution) can be described by the second order modulus of smoothness involved in the definition of the Besov spaces.

When dealing with quasiuniform meshes, one has well defined element sizes, inverse estimates, and some other convenient facts to use. In the presence of such convenience, the conventional FE tools become easily employable. That is why most of the theoretical work dedicated to the BPX preconditioner revolves around quasiuniform meshes. In local refinement scenarios, it is impossible to talk about quasiuniformity of the mesh. However, it is our very interest to analyze the BPX preconditioner for locally refined meshes in 2D or 3D. The work of Dahmen and Kunoth [17] is significant because they are the only ones who carefully examine a 2D local refinement routine (i.e. red-green refinement) and rigorously prove the optimal estimates. As far as our motivation is concerned, we find ourselves closer to their exposition. In fact, we have extended the 2D optimal result to the 3D one in the local refinement case, in section 2.9 studying an analog of red-green refinement routine in 3D. Bey [9] gave an in depth analysis of the routine. He also proposed a more realistic version that would eliminate some of the inherit restrictions resulting from the green (irregular) refinement.

To the author's knowledge, our exposition in section 2.9 is the first 3D optimal BPX result where full detail is given. In particular, we provide rigorous neighbor relations in the tetrahedralization. Our contribution will serve as a basis to create provably good preconditioners for realistic 3D problems and refinement routines.

A noteworthy contribution from the German school came from Bornemann and Yserentant [10] in 1993, where we borrow some of the historical comments. They have a self contained presentation where flexible K-functionals known from the interpolation spaces are utilized instead of a moduli of smoothness. They do not pay attention to the refinement details except for one edge-adjacency relation. They can bypass such details because they always relate estimates in the local refinement case to the uniform refinement one. We will exhibit various connections between Besov and the interpolation spaces, K-functionals and modulus of smoothness, etc. We will also report an optimal and suboptimal result in section 2.8 by using K-functionals and the H^s-stability of the L_2-projection, $s > 1$ and $s = 1$ respectively.

2.2 Tools from approximation theory

2.2.1 Modulus of continuity

Measuring the smoothness of a function by differentiability is too crude for many purposes in approximation. More subtle measurements are provided by the moduli of continuity and the moduli of smoothness. For further reference, see pages 40-51 in [19], and page 7 in [18].

Definition 2.2 *The modulus of continuity $\omega(f,t,\Omega)$ of a function f on $\Omega \subset \mathbb{R}$ is defined as*

$$\omega(f,t,\Omega) = \sup_{\substack{x,y\in\Omega \\ |x-y|\le t}} |f(x)-f(y)|, \quad 0 \le t.$$

2.2.2 Moduli of smoothness

If a continuous function f on an interval $\Omega = [a,b]$ satisfies $\omega(f,t,\Omega) = o(t)$, then f is constant. Thus, the modulus of continuity is not useful for measuring higher smoothness. We shall use moduli of smoothness which are connected with differences of higher orders.

We are dealing with measuring L_p-smoothness. Classical results use Lipschitz and Hölder continuity conditions that explore first order differences and fill the gap between C^0 and C^1. A natural generalization to L_p and arbitrary smoothness are the *moduli of smoothness.*

Let $\Omega \subset \mathbb{R}^d$ be open, for arbitrary $k = 1,2,\ldots$ and $h \in \mathbb{R}^d$, define the subset

$$\Omega_{k,h} = \left\{x \in \mathbb{R}^d : [x, x+kh] \subset \Omega\right\}.$$

Define directional k-th order difference of $f \in L_p(\Omega)$ as

$$\left(\Delta_h^k f\right)(x) = \sum_{r=0}^{k} \begin{pmatrix} k \\ r \end{pmatrix} (-1)^{k-r} f(x+rh), \quad x,h \in \mathbb{R}^d.$$

This is a sampling of the function f in the direction of h. It is also the exact expression in the k-th derivative of f. We will remove the dependence on the direction by taking the sup over all possible directions. This gives rise to a finer scale of smoothness than the differentiability;

Definition 2.3 *L_p-modulus of smoothness*

$$\omega_k(f, t, \Omega)_p = \sup_{|h| \leq t} \|\Delta_h^k f\|_{L_p(\Omega_{k,h})}.$$

2.2.3 K-functionals

Moduli of smoothness were introduced prior to K-functionals. In contrast to the historical evolution of this part of the approximation theory, the K-functional approach seems to be more convenient and often preferred [10]. K-functionals are used to define the interpolation spaces, which play a somewhat equivalent role in approximation theory as Besov spaces.

Definition 2.4

$$K(f, t; L_p(\Omega), W_p^k(\Omega)) = \inf_{g \in W_p^k(\Omega)} \left\{ \|f - g\|_{L_p(\Omega)} + t|g|_{W_p^k(\Omega)} \right\}, \tag{2.7}$$

is called the K-functional for $f \in L_p(\Omega)$.

Both modulus of smoothness and K-functionals give direct information of the smoothness of f. In fact, they are equivalent by the following.

Theorem 2.1 *Let Ω satisfy the uniform cone condition. Then, for any $k = 1, 2, \ldots,$ $1 \leq p < \infty$, and $f \in L_p(\Omega)$, we have*

$$\omega_k(f, t, \Omega)_p \approx K(f, t^k; L_p(\Omega), W_p^k(\Omega)), \quad t > 0. \tag{2.8}$$

Proof. See for example Theorem 1, page 29 in Oswald [31], or Theorem 2.4, page 177 in DeVore *et. al.* [19]. □

2.3 Spaces from approximation theory

The Besov space, the approximation space and the interpolation space are the three crucial spaces we borrow from the approximation theory for the norm estimates. In order to avoid quasinorms in the estimates, we will restrict $1 \leq p, q$. The domain Ω is fixed through the discussion. If there is no danger of confusion, we will skip reference to Ω in the notation. As a matter of fact, everything works for $\Omega = \mathbb{R}^d$.

2.3.1 Besov spaces

The Besov space becomes the central working ground by realizing the Sobolev space as a Besov space (see (2.10)). The analysis needed for functions in Sobolev space is done in the Besov space. The main motivation of employing the Besov space is due to the need of an upgraded scale of smoothness. Because the characterization of functions which have a given upper bound for the error of approximation sometimes calls for a finer scale of smoothness. The Lipschitz space, the next powerful space in measuring smoothness, cannot answer this need.

Definition 2.5 *Besov space is the collection of functions $f \in L_p(\Omega)$ with a finite Besov norm where the norm is defined by*

$$\|f\|_{B^s_{p,q}(\Omega)} = \|f\|_{L_p(\Omega)} + |f|_{B^s_{p,q}(\Omega)},$$

where the seminorm is given by

$$|f|_{B^s_{p,q}(\Omega)} = \|\{2^{sj}\omega_k(f, 2^{-j}, \Omega)_p\}_{j\in\mathbb{N}_0}\|_{l_q},$$

and k is any fixed integer larger than s.

2.3.2 Approximation spaces

The functions with finite BPX norm (2.4), $\|u\|_{BPX} < \infty$, form a space. This class corresponds to a general class in the approximation theory which is referred to as the approximation space.

Definition 2.6 *The approximation space is the collection of functions $f \in L_p(\Omega)$ with a finite approximation space norm*

$$\|f\|_{A^{s,J}_{p,q}(\Omega)} = \|\{2^{sj}\|(Q_j - Q_{j-1})f\|_{L_p(\Omega)}\}_{j=0}^{J}\|_{l_q},$$
$$\|f\|_{A^{s}_{p,q}(\Omega)} = \|\{2^{sj}\|(Q_j - Q_{j-1})f\|_{L_p(\Omega)}\}_{j\in\mathbb{N}_0}\|_{l_q}.$$

2.3.3 Interpolation spaces

Definition 2.7 *Let X and Y be two Banach spaces. Then the interpolation space $(X,Y)_{\sigma,q}$ is the collection of functions $f \in X+Y$ with a finite interpolation norm where the norm is defined by*

$$\|f\|_{(X,Y)_{\sigma,q}} = \|f\|_X + |f|_{(X,Y)_{\sigma,q}},$$

and the seminorm is given by

$$|f|_{(X,Y)_{\sigma,q}} = \|\{2^{\sigma j}K(f,2^{-j};X,Y)\}_{j\in\mathbb{N}_0}\|_{l_q}.$$

The example of interest is when $X = L_p(\Omega),\ Y = W_p^s(\Omega)$.

2.4 Estimates from approximation theory

The Bernstein estimate and inverse inequality respectively, describe the difference and the differentiability of functions from subspaces used in approximation spaces $\mathcal{S}_j$. They are used for the upper bound in the norm estimate. In particular, the Bernstein estimate (2.11) also dictates what needs to be done for the lower bound (2.26). The remedy prescribed by the Bernstein estimate involves the error of the best approximation. This is the exact location where the Jackson estimate comes into the play in order to pin down the lower bound to an absolute constant. Then we can say that the Bernstein (inverse) estimates and the Jackson (approximation) estimates complement each other for the purpose of achieving the norm equivalence (2.5). The intricate, maybe subtle part of the norm equivalence is the involvement of Jackson like estimates. Utmost attention is required in the selection of the smoothness scale. The choice of H^1 as the working ground for the Jackson estimate rather than $H^{1+\epsilon}$, will give rise to the suboptimal lower bound as in the original proof of the BPX preconditioner [13]. This is discussed in section 2.5.

2.4.1 The Bernstein estimate and inverse inequality

Both the Bernstein estimate and the inverse inequality bound a fine quantity (usually a fine norm) by the corresponding coarse quantity. This is the reverse interpretation of the Jackson estimate or approximation inequality. The treatment of these estimates in the discussion of the norm equivalence (2.5) is two fold. Traditionally, the first treatment used is the approximation-Besov space relation. We exhibit the embedding (2.12) of the approximation space $A^s_{p,q}$ into the Besov space $B^s_{p,q}$ by using the Bernstein estimate (2.11).

The initial step that is traditionally taken towards the establishment of norm equivalence (2.5) is to observe the well known fact

$$B^s_{p,\min\{p,2\}} \subset W^s_p \subset B^s_{p,\max\{p,2\}}, \quad s > 0,\ p \geq 1. \tag{2.9}$$

Especially, realizing the Sobolev space as a Besov space by (2.9),

$$H^s(\Omega) \cong B^s_{2,2}(\Omega), \quad s > 0, \tag{2.10}$$

(see Theorem 8, page 52 in Oswald [31]) opens the door of the approximation theory where numerous well established effective tools are at our disposal. The BPX norm is directly associated to the approximation space $A^s_{p,q}$, hence we will construct bridges between the approximation and Besov spaces through embeddings. Since the working ground is a Besov space, such embeddings call for estimates that involve modulus of smoothness. The central requirement is called the *Bernstein* estimate.

Definition 2.8 *The Bernstein estimate*

$$\omega_{k+1}(u,t)_p \leq c\ (\min\{1, t2^J\})^\beta \|u\|_{L_p}, \quad u \in \mathcal{S}_J, \tag{2.11}$$

where c is independent of u and J. Usually k = degree of the element and $\beta > k$.

The immediate implication of the Bernstein estimate is the following embedding which we usually refer to as the upper bound in the norm equivalence (see (2.26)).

Theorem 2.2 *If the Bernstein estimate (2.11) holds with $k = 1$*

$$A^s_{p,q} \hookrightarrow B^s_{p,q}, \quad 0 < s < \beta. \tag{2.12}$$

Proof. For linear elements, see Theorem 2, page 50 in Oswald [28] for $\beta = 1 + 1/p$ and also see Proposition 2, page 1617 in Oswald [29] for $\beta = 2 + 1/p$. □

The second approach is both novel and enlightening, and is inspired by the equivalence of modulus of smoothness and K-functionals. It helps us to identify the mechanics hidden behind the approximation theory, especially the issues pertaining to the Jackson estimate (see section 2.5). In this approach, the approximation-Besov space association is viewed as an approximation-interpolation space engagement. Here we exhibit a similar embedding (2.15) to (2.12) with a slightly modified assumption. This is the embedding of the approximation space into the interpolation space $(L_p(\Omega), Y)_{\sigma,q}$ by using the inverse inequality (2.14). Y is some "nice" space with

$$\mathcal{S}_J \subset Y \subset L_p(\Omega), \quad \forall J. \tag{2.13}$$

Inverse inequality is the counterpart of the Bernstein estimate in the interpolation space, and is defined as follows.

Definition 2.9 *Let $X = L_2(\Omega)$ and Y be as in (2.13) with $q = 2$. We say that inverse inequality holds, if there exists $\beta > 0$ satisfying*

$$\|u\|_Y \le c\ (2^J)^\beta \|u\|_{L_2}, \quad u \in \mathcal{S}_J,\ \forall J. \tag{2.14}$$

Theorem 2.3 *If the inverse inequality (2.14) holds, then the following embedding holds*

$$A^{\beta\sigma}_{2,q} \hookrightarrow (L_2, Y)_{\sigma,q}, \quad 0 < \sigma < 1. \tag{2.15}$$

Proof. See Theorem 1, page 3 Bornemann [11] for $q = 2$. □

Remark 2.1 In standard FE analysis, the Bernstein estimate (inverse inequality) is substituted by the strengthened Cauchy-Schwarz inequality.

2.4.2 The Jackson estimates

Both the Jackson estimate and approximation inequality characterize the approximation error in a coarser norm than the given function. For a given subset $M \subset L_p(\Omega)$, consider the best approximation:

$$E_M(f)_p = \inf_{g \in M} \|f - g\|_{L_p}, \quad f \in L_p(\Omega).$$

The study of these best approximations is the heart of classical approximation theory. We are interested in getting estimates from above for $E_M(f)_p$ when M coincides with a FE subspace. In the Besov space setting, we characterize E by modulus of smoothness.

Definition 2.10 *The Jackson estimate for Besov spaces*

$$E_{\mathcal{S}_J}(f)_p \leq c\ \omega_\alpha(f, 2^{-J})_p, \quad f \in L_p(\Omega). \tag{2.16}$$

The Jackson estimate

$$E_{\mathcal{S}_J}(f)_p \leq c\ (2^{-J})^\alpha |f|_{W_p^\alpha}, \quad f \in W_p^\alpha(\Omega), \tag{2.17}$$

where c is a constant independent of f and J, and α is an integer.

The second definition is more general in the sense that it can be used both in the Besov and interpolation space settings. Moreover (2.17) $\Rightarrow$ (2.16):

$$\begin{aligned} E_{\mathcal{S}_J}(f)_p &\leq \|f - g\|_{L_p} + E_{\mathcal{S}_J}(g)_p \\ &\leq c\ (\|f - g\|_{L_p} + (2^{-J})^l |g|_{W_p^l}) \quad \text{(by (2.17))} \\ &\leq c\ \omega_\alpha(f, 2^{-J})_p \quad \text{by (2.8).} \end{aligned}$$

The Jackson estimate implies the embedding in the other direction. Establishing the lower bound will fill the remaining piece in the norm equivalence (see (2.26) and (2.27)).

Theorem 2.4 *If the Jackson estimate (2.16) for $\alpha = 2$ holds, then*

$$B^s_{p,q} \hookrightarrow A^s_{p,q}, \quad 0 < s < 2. \tag{2.18}$$

Proof. See Proposition 2, page 1617 in Oswald [29]. □

Since we do not relate E to a modulus of smoothness, α can be real in the definition below.

Definition 2.11 *Let $X = L_2(\Omega)$ and Y be as in (2.13). We say that the approximation inequality holds, if there exists $\alpha > 0$ satisfying*

$$E_{\mathcal{S}_J}(u)_p \leq c\ (2^{-J})^{\alpha}|u|_Y, \quad u \in Y. \tag{2.19}$$

Theorem 2.5 *For $\alpha > 0$, if an approximation property like the Jackson estimate (2.19) holds for $p = 2$ and $\forall J$, then the following embedding holds*

$$(L_2(\Omega), Y)_{\sigma,q} \hookrightarrow A^{\alpha\sigma}_{2,q}, \quad 0 < \sigma < 1. \tag{2.20}$$

Proof. See Theorem 1, page 3 in Bornemann [11] for $q = 2$. □

The Jackson estimate plays a pivotal role in reducing the lower bound to an absolute constant.

2.5 Interpolation space and linear finite element space

The embeddings (2.15) and (2.20) make it possible to characterize the approximation space by the interpolation space.

Corollary 2.6 *If both (2.17) and (2.14) hold, then $\alpha \leq \beta$. For $\alpha = \beta$ we get the isomorphism*

$$A^{\alpha\sigma}_{2,2} \cong (L_2(\Omega), Y)_{\sigma,2}, \quad 0 < \sigma < 1. \tag{2.21}$$

Keeping in mind that we always aim for the norm equivalence (2.5), our exposition takes an interesting turn. Is it possible to acquire H^1 by the interpolation spaces? In the light of isomorphism (2.21), the answer is yes,

$$H^1(\Omega) \cong (L_2(\Omega), Y)_{\sigma,2}, \tag{2.22}$$

by carefully prescribing Y and σ which we have left vague so far.

By the standard interpolation technique, we can attain (2.22) by choosing a smoother space than H^1. The choice is

$$Y = H^s(\Omega), \quad 1 < s < \infty, \quad \sigma = \frac{1}{s}, \tag{2.23}$$

so that $0 < \sigma < 1$ as required. With some domain related technicality (i.e. minimal domain smoothness and extension operator) we conclude that

$$H^1(\Omega) \cong (L_2(\Omega), H^s(\Omega))_{\frac{1}{s},2}, \quad 1 < s < \infty. \tag{2.24}$$

We are dealing with linear FEs. One should recall that we are only able to move to the interpolation space arena as long as the inverse (2.14) and approximation (2.19) inequalities are satisfied. We then try to achieve the inequalities within the linear FE framework. One crude observation is that smoothness of linear FEs can vary between 0 and 3/2, meaning that

$$\mathcal{S}_J \subset H^s(\Omega), \quad 0 \leq s \leq 3/2.$$

$1 < s < \infty$ dictates the precise choice of $s = 1 + \epsilon, \quad 0 < \epsilon \leq 1/2$. Therefore the inverse (2.14) and approximation (2.19) inequalities must be established in $H^{1+\epsilon}$, not in H^1. This means that linear finite elements are a little bit smoother than one may presume. By the analysis of the original BPX proof, we observe how the original proof failed to prove optimality. In the original proof of the suboptimal BPX result [13], the corresponding estimates in H^1 have been the working ground. We want to emphasize that optimality can only be achieved by viewing the Jackson estimate in $H^{1+\epsilon}$, $\epsilon > 0$ arbitrarily small. This subtle point that is hidden one way or another in all proofs of optimality of the BPX preconditioner, has been pointed out by Bornemann [11].

In the rest of this section we will borrow results from the FE theory for the 2D quasiuniform mesh. By definition of the L_2-projection and the Bramble-Hilbert Lemma, the approximation (2.19) inequality holds for $s = 0$ and $s = 2$. Interpolating between $s = 0$ and $s = 2$, one can show that approximation (2.19) inequality holds for $\beta = s = 1 + \epsilon$.

$$\|u - Q_J u\|_{L_2} \leq c\ h_J^s \|u\|_{H^s} \leq c\ 2^{-Js} \|u\|_{H^s}, \quad \forall u \in H^s(\Omega).$$

The Bernstein estimate holds for $\alpha = s = 1 + \epsilon$,

$$\|u\|_{H^s} \leq c\ h_J^s \|u\|_{L_2} \leq c\ 2^{Js} \|u\|_{L_2}, \quad \forall u \in \mathcal{S}_J,$$

by using the Sobolev-Slobodeckii norm

$$\|u\|_{H^{1+\epsilon}}^2 \cong \|u\|_{H^1}^2 + \int \int_\Omega \frac{|\nabla u(x) - \nabla u(y)|^2}{|x-y|^{2+2\epsilon}}\ dx\ dy.$$

Corollary 2.6 for $\alpha = \beta = 1/\sigma = s = 1 + \epsilon$ implies the big result:

Theorem 2.7 *For linear FE, we have*

$$A_{2,2}^1 \cong H^1(\Omega).$$

2.6 Main norm equivalence theorems

The fundamental norm equivalence between the Besov space $B_{2,2}^1$ and the approximation space $A_{2,2}^1$ is established in this section. The following estimate $v_J^{(p)}$ will help us to prescribe the lower bound in the presence of the Bernstein estimate (2.11). Also with the help of this estimate, we will use both the Bernstein estimate and the Jackson estimate (2.17) to achieve an optimal lower bound. For the norm equivalence theorems, we follow the exposition presented in Dahmen and Kunoth

[17]. We define $v_{j,J}^{(p)}$ to characterize how well one can reach the Jackson estimate (2.17) for Besov spaces by using functions from $\mathcal{S}_J$.

$$v_{j,J}^{(p)} = \sup_{u \in \mathcal{S}_J} \frac{\|u - Q_j u\|_{L_p}}{\omega_{k+1}(u, 2^{-j}, \Omega)}, \quad v_J^{(p)} = \max\left\{1, v_{j,J}^{(p)} : j = 0, \ldots, J\right\}. \tag{2.25}$$

Theorem 2.8 *Suppose the Bernstein estimate (2.11) holds for some real number $\beta > k$. For each $0 < s < \min\{\beta, k+1\}$, there exists constants $0 < c_1,\ c_2 < \infty$ independent of $u \in \mathcal{S}_J,\ J = 0, 1, \ldots$, such that*

$$\frac{c_1}{v_J^{(p)}} \|u\|_{A_{p,q}^s} \le \|u\|_{B_{p,q}^s} \le c_2 \|u\|_{A_{p,q}^s}, \quad u \in \mathcal{S}_J. \tag{2.26}$$

Proof. See Theorem 4.1, page 323 in [17]. □

Proposition 2.9 *If the Jackson estimate (2.17) holds for some integer $\alpha > k$, then we have*

$$v_J^{(p)} = O(1), \quad J \to \infty. \tag{2.27}$$

Proof. See Proposition 4.1, page 325 in [17]. □

Theorem 2.10 *Suppose there exists $\beta > k$ such that the Bernstein estimate (2.11) holds. Then for the preconditioner defined by (2.3)*

$$\kappa(C_J^{1/2} A_J C_J^{1/2}) = O((v_J^{(2)})^2), \quad J \to \infty. \tag{2.28}$$

Proof. See the Proof of Theorem 3.1, page 324 in [17]. □

Theorem 2.11 *Suppose together with the Bernstein estimate (2.11), the Jackson estimate (2.17) holds for an integer $\alpha > k$ and $p = 2$. Then*

$$\kappa(C_J^{1/2} A_J C_J^{1/2}) = O(1), \quad J \to \infty. \tag{2.29}$$

Proof. See Theorem 3.2, page 321 in [17]. □

2.7 Summary and comparison of the spaces

We insert the following diagram that would shed some light to the connections between the interpolation and approximation spaces.

$$\begin{array}{ccccc}
 & & & \begin{array}{c}\text{Approximation}\\ \text{Spaces}\end{array} & \\
 & & \updownarrow (B) & & \updownarrow (C) \\
\text{Sobolev Spaces} & \overset{(A)}{\cong} & \text{Besov Spaces} & \overset{(D)}{\longleftrightarrow} & \text{Interpolation Spaces} \\
 & & \updownarrow (E) & & \updownarrow (F) \\
 & & \text{Moduli of smoothness} & \overset{\text{Thm 2.1}}{\approx} & K-\text{functionals}
\end{array}$$

(A): Isomorphism (2.10)

$$H^s \cong B^s_{2,2}, \quad s > 0.$$

(B): Embeddings (2.12) and (2.18) imply the isomorphism

$$A^s_{p,q} \cong B^s_{p,q}, \quad 0 < s < 1 + 1/p.$$

In general, Theorem 9.2, page 235 in DeVore *et. al* [19] implies

$$A^s_{p,q} \cong B^s_{p,q}, \quad 0 < s < \infty, \;\; 1 \le p < \infty, \;\; 0 < q \le \infty.$$

(C): Embeddings (2.15) and (2.20), imply the isomorphism (2.24)

$$A^1_{2,q} \cong (L_2, H^s)_{\frac{1}{s},q}, \quad 1 < s < \infty.$$

In general, if inverse inequality and approximation inequality hold for spaces X, Y, as in Theorem 9.1, page 235 in DeVore *et. al* [19], then

$$A^s_{p,q} \cong (X, Y)_{s/r,q}, \quad 0 < s < r, \;\; 0 < q \le \infty.$$

(D): See Example 3, page 196 in DeVore *et. al* [19].

$$B^r_{p,q} \cong (L_p, W^s_p)_{\sigma,q}, \quad r = s\sigma.$$

(E): Definition 2.6.

(F): Definition 2.7.

2.8 H^s-stability of the L_2-projection and the lower bound

It is well known that the L_2-projection is H^s-stable on quasiuniform methods, i.e.

$$|Q_j u|_{H^s} \leq c\ |u|_{H^s}, \quad 0 \leq s \leq 1,$$

also approximation inequality (2.19)

$$\|f - Q_j f\|_{L_2} \leq c\ (2^{-j})^s |f|_{H^s}, \quad f \in H^s, \quad 0 \leq s \leq 2,$$

holds. In order to obtain the canonical suboptimal lower bound for the BPX preconditioner $\frac{1}{J}\|u\|^2_{BPX} \leq c|u|^2_{H^1}$, we present two approaches. The first approach only uses the approximation inequality (2.19)

$$\begin{aligned} \|Q_j u - Q_{j-1}u\|_{L_2} &\leq \|u - Q_{j-1}u\|_{L_2} \quad \text{(a property of } Q_j\text{)} \\ &\leq c\ (2^{-j})^s |u|_{H^s}. \end{aligned}$$

The second approach uses both (2.19) and H^s-stability of the L_2-projection

$$\begin{aligned} \|Q_j u - Q_{j-1}u\|_{L_2} &= \|Q_j u - Q_{j-1}Q_j u\|_{L_2} \\ &\leq c\ (2^{-j})^s |Q_j u|_{H^s} \\ &\leq c\ (2^{-j})^s |u|_{H^s}. \end{aligned}$$

By defining $Q_{-1} = 0$, we can rewrite the BPX norm

$$\|u\|^2_{BPX} = \sum_{j=0}^{J} 2^{2js} \|(Q_j - Q_{j-1})u\|^2_{L_2}, \quad s = 1.$$

By using either of the above approaches, we get the canonical suboptimal lower bound for quasiuniform meshes

$$\begin{aligned} \|u\|^2_{BPX} &\leq c \sum_{j=0}^{J} |u|^2_{H^s} \\ &= c\ J\ |u|^2_{H^s}, \quad s = 1. \end{aligned}$$

Next, we present a novel approach by using K-functionals to obtain the optimal and suboptimal lower bound for quasiuniform meshes. We begin with a lemma that characterizes the terms of the BPX norm in terms of the K-functional.

Lemma 2.12 *Let the approximation inequality (2.19) hold, then*

$$\|(Q_j - Q_{j-1})u\|_{L_2} \leq c\ K(u, 2^{-(j-1)s}; L_2, H^s). \tag{2.30}$$

Proof. Let $v \in H^s$, then

$$\begin{aligned}
\|(Q_j - Q_{j-1})u\|_{L_2} &\leq \|u - Q_{j-1}u\|_{L_2} \\
&\leq \|u - v\|_{L_2} + \|v - Q_{j-1}u\|_{L_2} \\
&\leq \|u - v\|_{L_2} + \|v - Q_{j-1}v\|_{L_2} + \|Q_{j-1}(u - v)\|_{L_2} \\
&\leq 2\|u - v\|_{L_2} + \|v - Q_{j-1}v\|_{L_2} \\
&\leq c\ (\|u - v\|_{L_2} + (2^{-(j-1)})^s |v|_{H^s})\ \text{(using approximation inequality)} \\
&\leq c\ K(u, 2^{-(j-1)s}; L_2, H^s)\ (\text{taking } \inf_{v \in H^s}).
\end{aligned}$$

For an alternative proof on a uniformly refined mesh, see Lemma 7.1, page 471 in [10]. □

If one can establish the H^s-stability, $s > 1$, of the L_2-projection, then we will show that the absolute lower bound can be obtained. This simple approach has not been taken in the literature before. However, we should also say that H^s-stability, $s > 1$, of the L_2-projection has not been carefully studied. In addition, we emphasize that H^1-stability of the L_2-projection can also be established on locally refined meshes which satisfy certain mesh criterion. The following theorem can be extended to certain locally refined meshes. However, Theorem 2.13 relies on Lemma 2.12. Lemma 2.12 requires the approximation inequality which cannot be satisfied on locally refined meshes for the reason stated in the beginning of subsection 2.9.1. If the requirement $v \in H^s$ in the proof of Lemma 2.12 can be replaced by $v \in \mathcal{S}_J$ with another proof technique, then we can extend the optimal lower bound to locally refined meshes which satisfy certain mesh criterion.

Theorem 2.13 *If the L_2-projection is H^s-stable $|Q_{j-1}u|_{H^s} \leq c\ |u|_{H^s}$, then for*

- $s = 1 + \epsilon$, *we have* $\|u\|^2_{BPX} \leq c|u|^2_{H^s}$,
- $s = 1$, *we have* $\frac{1}{J}\|u\|^2_{BPX} \leq c|u|^2_{H^s}$.

Proof. $Q_{j-1} : L_2 \mapsto \mathcal{S}_{j-1}$, also we know that $\mathcal{S}_{j-1} \subset H^s$.

$$\begin{aligned} K(u, 2^{-(j-1)s}; L_2, H^s) &\leq \|u - Q_{j-1}u\|_{L_2} + (2^{-(j-1)})^s |Q_{j-1}u|_{H^s}) \\ &\leq c\ 2^{-(j-1)s}|u|_{H^s} \end{aligned}$$

The last step is due to the approximation inequality and the H^s-stability of the L_2-projection.

$$\begin{aligned} \sum_{j=0}^{J} 2^{2j}\|(Q_j - Q_{j-1})u\|^2_{L_2} &\leq c\ \sum_{j=0}^{J} 2^{2j} K(u, 2^{-(j-1)s}; L_2, H^s)^2 \text{ (by (2.30))} \\ &\leq c\ 2^{2s} \left\{ \sum_{j=0}^{J} 2^{2j(1-s)} \right\} |u|^2_{H^s} \\ &\leq c\ |u|^2_{H^s} \quad (s = 1 + \epsilon > 1) \\ \text{or } &\leq c\ \frac{1}{J}|u|^2_{H^s} \quad (s = 1). \end{aligned}$$

□

It is ironical that the original BPX proof was upgraded to optimal bounds by realizing the fact that the Jackson estimate holds for H^s, $s > 1$. By noticing the role of $s > 1$, we see a similar pattern between H^s-stability of the L_2-projection and the Jackson estimate. We might guess a hidden relation between the H^s-stability of the L_2-projection and the Jackson estimate, which can be a potential subject for future research.

2.9 3D red-green refinement and optimality of the BPX preconditioner

As noted in previous sections, the BPX is a provably good optimal preconditioner for quasiuniform meshes in 2D and 3D. We turn our attention to the optimality of the BPX preconditioner for locally refined meshes in 3D. Dahmen and Kunoth (section 5 in [17]) established an optimality statement for the BPX preconditioner using the *red-green* refinement routine in 2D. This routine is popularized by Bank's PLTMG [5] package.

We prove that the BPX preconditioner still preserves its optimality for an analog routine in 3D. In fact, it is the counterpart of red-green refinement of the PLTMG in 3D. The details of the routine can be found in [9].

We list the geometric assumptions about the mesh. Let $\Omega \subset \mathbb{R}^3$ be a polyhedral domain. We assume that the triangulation $\mathcal{T}_j$ of Ω at level j is a collection of tetrahedra with mutually disjoint interiors which cover Ω

$$\Omega = \bigcup_{\tau \in \mathcal{T}_j} \tau.$$

We want to generate successive refinements $\mathcal{T}_0, \mathcal{T}_1, \ldots$ which satisfy the following conditions.

Assumption 2.1 Nestedness: *Each tetrahedron (son) $\tau \in \mathcal{T}_j$ is covered by exactly one tetrahedron (father) $\tau' \in \mathcal{T}_{j-1}$, and any corner of τ is either a corner or an edge midpoint of τ'.*

Assumption 2.2 Conformity: *The intersection of any two tetrahedra $\tau, \tau' \in \mathcal{T}_j$ is either empty, a common vertex, a common edge or a common face.*

Assumption 2.3 Nondegeneracy: *The interior angles of all tetrahedra in the refinement sequence $\mathcal{T}_0, \mathcal{T}_1, \ldots$ are bounded away from zero.*

A regular (red) refinement subdivides a tetrahedron τ into 8 equal volume subtetrahedra. We connect the edges of each face as the 2D regular refinement. Then, we cut off four subtetrahedra at the corners which are congruent to τ. An octahedron with three parallelograms remains in the interior. Cutting octahedron along the two faces of these parallelograms, we obtain four more subtetrahedra which are not necessarily congruent to τ. We choose the diagonal of the parallelogram so that the successive refinements always preserve nondegeneracy.

If a tetrahedron is marked for regular refinement, the resulting triangulation violates conformity Assumption 2.2. Then nonconformity is remedied by irregular (green) refinement. In 3D, there are altogether $2^6 = 64$ possible edge refinements, of which 62 are irregular. One should pay extra attention to irregular refinement in the implementation. Bey [9] gives an organized handling of irregular cases. Using symmetry arguments, the 62 irregular cases can be divided into 9 different types. On the other hand, as we refine irregularly, we are repeatedly dividing angles which can cause degeneracy. To ensure that the interior angles remain bounded away from zero, we enforce these additional conditions,

Assumption 2.4 *Irregular tetrahedra are not refined further.*

Assumption 2.5 *Only tetrahedra $\tau \in \mathcal{T}_j$ with $L(\tau) = j$ are refined for the construction of $\mathcal{T}_{j+1}$, where*

$$L(\tau) = \min \{j : \tau \in \mathcal{T}_j\} \tag{2.31}$$

denotes the level of τ.

For the possible 62 irregular refinements, we force tetrahedra to obey the following:

Assumption 2.6 *If three or more edges are refined and do not belong to a common face, then the tetrahedron is refined regularly.*

Lemma 2.14 *Let τ and τ' be two tetrahedra in $\mathcal{T}_j$ sharing a common face η. Then*

$$|L(\tau) - L(\tau')| \leq 1. \tag{2.32}$$

Proof. If $L(\tau) = L(\tau')$, then $0 \leq 1$, there is nothing to show. Without loss of generality, assume that $L(\tau) < L(\tau')$. Proof requires a detailed and systematic analysis. To show the line of reasoning, we prefer to list the facts used in the proof:

1. $L(\tau') \leq j$ because by assumption $\tau' \in \mathcal{T}_j$.

2. $L(\tau) < j$ $(L(\tau) < L(\tau') \leq j)$.

3. By assumption $\tau \in \mathcal{T}_j$, meaning that τ was never refined from the level it was born $L(\tau)$ to level j.

4. Let τ'' be the father of τ'. Then $L(\tau'') = L(\tau') - 1 < j$.

5. $L(\tau) < L(\tau')$ by assumption, implying $L(\tau) \leq L(\tau'')$.

6. By (3), τ belongs to all the triangulations from $L(\tau)$ to j, in particular $\tau \in \mathcal{T}_{(\tau'')}$, where by (4) $L(\tau'') < j$.

η is the common face of τ and τ' on level j. If τ' is obtained by regular refinement of its father τ'', then η is still the common face of τ and τ''. But, by (6) both $\tau,\ \tau'' \in \mathcal{T}_{L(\tau'')}$. Then, this would violate conformity assumption (2.2) for level $= \mathcal{T}_{L(\tau'')}$. Hence, τ' must have been irregular.

On the other hand, $L(\tau) \leq L(\tau') - 1 = L(\tau'')$. Next, we proceed by eliminating the possibility that $L(\tau) < L(\tau'')$. If so, we repeat the above reasoning, and τ'' becomes irregular. τ'' is already the father of the irregular τ', contradicting Assumption 2.4 for level $L(\tau'')$. Hence, $L(\tau) = L(\tau'') = L(\tau') - 1$, proving the lemma. $\square$

By Assumptions 2.4 and 2.5, every tetrahedron at any $\mathcal{T}_j$ is geometrically similar to some tetrahedron in $\mathcal{T}_0$ or a tetrahedron arising from an irregular refinement of some tetrahedron in $\mathcal{T}_0$. Then, there exist absolute constants $c_1,\ c_2$ such that

$$c_1 \ \text{size}(\bar{\tau})\ 2^{-L(\tau)} \leq \text{size}(\tau) \leq c_2 \ \text{size}(\bar{\tau})\ 2^{-L(\tau)}, \tag{2.33}$$

where $\bar{\tau}$ is the father of τ in the initial mesh.

Lemma 2.15 *Let τ and τ' be two tetrahedra in $\mathcal{T}_j$ sharing a common edge (two vertices). Then there exists a finite number E depending on the shape regularity such that*

$$|L(\tau) - L(\tau')| \leq E. \tag{2.34}$$

Proof. Start with τ, get the face-adjacent neighbor τ_2 (either of the two faces), and then get the face-adjacent tetrahedron, repeat this process E times until you reach $\tau' = \tau_E$. After you pick one of the two faces, the direction of the face-adjacent tetrahedra is determined. E is always a finite number due to shape regularity. By face-adjacent neighbor relation (2.32), we conclude the lemma. □

Lemma 2.16 *Let τ and τ' be two tetrahedra in $\mathcal{T}_j$ sharing a common vertex. Then there exists a finite number V depending on the shape regularity such that*

$$|L(\tau) - L(\tau')| \le V. \tag{2.35}$$

Proof. Take one edge from τ and τ' where the edges meet at the common vertex. By shape regularity, there exist $e < \infty$ many edge-adjacent tetrahedra between $\tau_1 = \tau$ and $\tau_e = \tau'$. By the above construction in Lemma 2.15, there exist $E_{1,2}$ many face-adjacent tetrahedra between τ_1 and τ_2. We repeat this process until we place $E_{e-1,e}$ many tetrahedra between τ_{e-1} and τ_e. Hence, there exist $V = \sum_{i=1}^{e} E_{i-1,i}$ face-adjacent tetrahedra between τ and τ'. Again, face-adjacent neighbor relation (2.32) yields the lemma. □

Lemma 2.17 *There is a constant depending on the shape regularity of $\mathcal{T}_j$ and the quasiuniformity of $\mathcal{T}_0$, such that*

$$\frac{\text{size}(\tau)}{\text{size}(\tau')} \le c, \quad \forall \tau, \tau' \in \mathcal{T}_j, \quad \tau \cap \tau' \neq \emptyset. \tag{2.36}$$

Proof. τ and τ' are either face-adjacent (three vertices), edge-adjacent (two vertices), or vertex-adjacent, and are handled by (2.32), (2.34), (2.35) respectively.

$$\begin{aligned}
\frac{\text{size}(\tau)}{\text{size}(\tau')} &\le c \frac{2^{-L(\tau)}\ \text{size}(\bar{\tau})}{2^{-L(\tau')}\ \text{size}(\bar{\tau'})} \quad \text{(by (2.33))} \\
&\le c\ 2^{|L(\tau)-L(\tau')|}\ \frac{\text{size}(\bar{\tau})}{\text{size}(\bar{\tau'})} \\
&\le c\ 2^{\max\{1,E,V\}}\ \gamma^{(0)} \quad \text{(by (2.32), (2.34), (2.35) and quasiuniformity of } \mathcal{T}_0\text{)}
\end{aligned}$$

□

Our motivation now is to form a stable basis in the sense of Oswald's assumption (A1). Then this assumption will guarantee the Bernstein estimate (2.11) which is needed for the upper bound of the norm equivalence.

A basis function corresponding to newer vertices will have smaller support than the ones corresponding to old vertices. For a stable basis, we have to supplement the small supports by an appropriate scaling of the basis function to keep the $\|\phi_v^{(j)}\|_{L_2(\Omega)}$ roughly the same for all basis functions.

$$\phi_v^{(j)}(w) = 2^{3/2L_{j,v}}\ \delta_{vw}, \quad v, w \in \mathcal{N}_j, \tag{2.37}$$

where

$$L_{j,v} = \min\{L(\tau) : \tau \in \mathcal{T}_j,\ v \in \tau\}. \tag{2.38}$$

By calculating the element mass matrix, we give the following important formula.

$$\int_\tau |g(x,y,z)|^2\ dxdydz = \frac{\text{volume}(\tau)}{(d+1)(d+2)} \left(\sum_{i=1}^{d+1} g(v_i)^2 + \left[\sum_{i=1}^{d+1} g(v_i) \right]^2 \right), \tag{2.39}$$

where, $i = 1, \ldots, d+1$ and v_i is a vertex of τ, $d = 3$. We utilize (2.39) to arrive at

$$\int_\Omega |\phi_v^{(j)}(x,y,z)|^2\ dxdydz = 2^{3L_{j,v}}\ \frac{\text{volume}(\text{supp } \phi_v^{(j)})}{20}.$$

Since min in (2.38) is attained, then there exists at least one $\tau \in \text{supp } \phi_v^{(j)}$ such that $L(\tau) = L_{j,v}$. By (2.33) we have

$$2^{L_{j,v}} \sim \frac{\text{size}(\tau)}{\text{size}(\bar{\tau})}. \tag{2.40}$$

Also,

$$\text{volume}(\text{supp } \phi_v^{(j)}) \sim \sum_{i=1}^{M} \text{size}^3(\tau_i), \quad \tau_i \in \text{supp } \phi_v^{(j)}. \tag{2.41}$$

By (2.36), we have

$$\text{size}(\tau_i) \sim \text{size}(\tau). \tag{2.42}$$

Combining (2.41) and (2.42), we conclude

$$\text{volume}(\text{supp } \phi_v^{(j)}) \sim M\text{size}^3(\tau) \tag{2.43}$$

Eventually (2.40) and (2.43) yield

$$2^{3L_{j,v}}\text{volume}(\text{supp } \phi_v^{(j)}) \sim M \frac{1}{\text{size}^3(\bar{\tau})}. \tag{2.44}$$

Final comments on the constants in the above inequality are as follows: Firstly, M is a uniformly bounded constant by shape regularity. Secondly, we already know by (2.3) and (2.4) that every tetrahedron at any $\mathcal{T}_j$ is geometrically similar to some tetrahedron in $\mathcal{T}_0$ or a tetrahedron arising from an irregular refinement of some tetrahedron in $\mathcal{T}_0$, hence, to some tetrahedron of a fixed finite collection. Therefore we can view the size of any tetrahedron in $\mathcal{T}_0$ as a constant.

Finally, we have proved that

$$\|\phi_v^{(j)}\|_{L_2(\Omega)} \sim 1, \quad v \in \mathcal{N}_j. \tag{2.45}$$

Let $g = \sum_{v\in\mathcal{N}_j} c_v\phi_v^{(j)} \in \mathcal{S}_j$ and for any $\tau \in \mathcal{T}_j$ we have

$$\|g\|^2_{L_2(\tau)} \leq c \sum_{v\in\mathcal{N}_{j,\tau}} |c_v|^2\|\phi_v^{(j)}\|^2_{L_2(\Omega)}, \tag{2.46}$$

where $\mathcal{N}_{j,\tau} = \{v \in \mathcal{N}_j : v \in \tau\}$, which is uniformly bounded in $\tau \in \mathcal{T}_j$ and $j \in \mathbb{N}_0$. By the scaling (2.37), we get the equality in the below estimate. The inequality is a standard inverse estimate where one bounds $g(v)$ by using the formula (2.39) and handles the volume in the formula by (2.33)

$$|c_v|^2 = 2^{-3L_{j,v}}|g(v)|^2 \leq c\ 2^{-3L_{j,v}}2^{3L_{j,v}}\|g\|^2_{L_2(\tau)}. \tag{2.47}$$

We sum up over $\tau \in \mathcal{T}_j$ in (2.46) and (2.47), and by using (2.45) we achieve stability for our basis:

$$\|\sum_v c_v\phi_v^{(j)}\|_{L_2(\Omega)} \sim \|\{c_v\}_{v\in\mathcal{N}_j}\|_{l_2}. \tag{2.48}$$

Lemma 2.18 *The Bernstein estimate (2.11) holds for $\beta = 3/2$ for the scaled basis (2.37).*

Proof. (2.48) asserts that the scaled basis (2.37) is stable, satisfying Oswald's assumption (A1) (see page 15 in [31]). Hence (by Theorem 4, page 33 in [31]), we get the Bernstein estimate. One can also refer to Lemma 2 page 50 in [28] for the same proof. Here, one should note that the proof actually works independent of the dimension. □

2.9.1 Lower bound in the norm equivalence

In a locally refined mesh, the Jackson estimate (2.17) holds only for functions that are somehow obeying the mesh geometry. For instance, if a mesh is designed to pick up the singularity at $x = 0$ of $y = 1/x$, then on the same mesh we will not be able to recover a singularity at $x = 1$ of $y = 1/(x-1)$ well enough. Hence the Jackson estimate (2.17) cannot hold for any function $f \in W_p^k$. As pointed out above, some kind of a Jackson estimate (2.16) can hold for special functions. In order to get the lower bound, we focus on estimating $v_J^{(2)}$ directly.

Let $\tau \in \mathcal{T}_j$ be a tetrahedron with vertices u, v, w, x. Clearly the restrictions of $\phi_{v'}^{(j)}$ to τ are linearly independent over τ. Then, there is a set of unique linear polynomials $\psi_u^{(j),\tau}, \psi_v^{(j),\tau}, \psi_w^{(j),\tau}, \psi_x^{(j),\tau}$ such that

$$\int_\tau \phi_{v'}^{(j)}(x,y,z)\psi_{v''}^{(j),\tau}(x,y,z)dxdydz = \delta_{v'v''}, \quad v', v'' \in \{u, v, w, x\}. \tag{2.49}$$

For $v \in \mathcal{N}_j$ and $\tau \in \mathcal{T}_j$, define a function for $v \in \tau$

$$\xi_v^{(j)}(x,y,z) = \begin{cases} \frac{1}{M_v}\psi_v^{(j),\tau}(x,y,z), & x \in \tau \\ 0, & x \notin \text{supp } \phi_v^{(j)} \end{cases}, \tag{2.50}$$

where M_v is the number of tetrahedra in $\mathcal{T}_j$ in supp $\phi_v^{(j)}$. Obviously by (2.49) and (2.50), we obtain

$$(\xi_w^{(j)}, \phi_v^{(j)}) = \int_\Omega \xi_w^{(j)}(x,y,z)\phi_v^{(j)}(x,y,z)dxdydz = \delta_{vw}, \quad v, w \in \mathcal{N}_j. \tag{2.51}$$

Now we can define a quasi-interpolant, in fact a *projector* onto $\mathcal{S}_j$ such that

$$(\tilde{Q}_j f)(x,y,z) = \sum_{v\in\mathcal{N}_j} (f, \xi_v^{(j)})\phi_v^{(j)}(x,y,z). \tag{2.52}$$

Claim 2.1 *For $g \in \mathcal{S}_j$, $\tilde{Q}_j g = g$.*

Proof. Let $g(x,y,z) = \sum_{v'\in\mathcal{N}_j} g_{v'}\phi_{v'}^{(j)}(x,y,z)$. Then,

$$\begin{aligned}(\tilde{Q}_j g)(x,y,z) &= \sum_{v\in\mathcal{N}_j}\sum_{v'\in\mathcal{N}_j} g_{v'}\ (\xi_v^{(j)}, \phi_{v'}^{(j)})\phi_v^{(j)}(x,y,z)\\ &= \sum_{v\in\mathcal{N}_j} g_v\phi_v^{(j)} = g.\end{aligned}$$

□

Claim 2.2 *$\tilde{Q}_j$ fixes polynomials of degree at most 1; $\Pi_1(\mathbb{R}^3)$, i.e.*

$$(\tilde{Q}_j P)(x,y,z) = P(x,y,z), \quad P \in \Pi_1(\mathbb{R}^3).$$

Proof. We can uniquely represent a linear function on a tetrahedron by basis functions corresponding to its vertices

$$P\ |_\tau = \alpha_1\phi_1 + \alpha_2\phi_2 + \alpha_3\phi_3 + \alpha_4\phi_4.$$

For $v \in \tau$

$$\tilde{Q}_j P\ |_\tau = (P,\xi_1)\phi_1 + (P,\xi_2)\phi_2 + (P,\xi_3)\phi_3 + (P,\xi_4)\phi_4 = \alpha_1\phi_1 + \alpha_2\phi_2 + \alpha_3\phi_3 + \alpha_4\phi_4 = P.$$

□

One can easily observe by (2.45) and (2.51) that

$$\|\xi_v^{(j)}\|_{L_2(\Omega)} \sim 1, \quad v \in \mathcal{N}_j,\ j \in \mathbb{N}_0. \tag{2.53}$$

Letting $\Omega_{j,\tau} = \bigcup\{\tau' \in \mathcal{T}_j :\ \tau\cap\tau' \neq \emptyset\}$, we can conclude from (2.45) and (2.53) that

$$\|\tilde{Q}_j f\|_{L_2(\tau)} = \|\sum_{v\in\mathcal{N}_{j,v}} (f,\xi_v^{(j)})\phi_v^{(j)}\|_{L_2(\tau)} \le c\|f\|_{L_2(\Omega_{j,\tau})}. \tag{2.54}$$

We define a subset of the triangulation where the refinement activity stops. Meaning that, all tetrahedra in $\mathcal{T}_j^*$, $j \leq m$ also belong to $\mathcal{T}_m$.

$$\mathcal{T}_j^* = \{\tau \in \mathcal{T}_j : \ L(\tau) < j, \ \Omega_{j,\tau} \cap \tau' = \emptyset, \ \forall \tau' \in \mathcal{T}_j \text{ with } L(\tau') = j\} \tag{2.55}$$

Since there is no refinement activity on $\mathcal{T}_j^*$, the FE spaces are frozen in the refinement hierarchy. That is, $g \in \mathcal{S}_m$ implies $g \in \mathcal{S}_j$ for $j \leq m$. By Claim 2.1 $\tilde{Q}_j g = g$ on $\mathcal{T}_j^*$. Hence,

$$\|g - \tilde{Q}_j g\|_{L_2(\tau)} = 0, \quad \tau \in \mathcal{T}_j^*. \tag{2.56}$$

The fact that $\tilde{Q}_j$ fixes polynomials (i.e. Claim 2.2), and (2.54) imply

$$\begin{aligned} \|g - \tilde{Q}_j g\|_{L_2(\tau)} &\leq \|g - P\|_{L_2(\tau)} + \|\tilde{Q}_j(P - g)\|_{L_2(\tau)} \\ &\leq c \, \|g - P\|_{L_2(\Omega_{j,\tau})}, \quad \tau \in \mathcal{T}_j \setminus \mathcal{T}_j^*. \end{aligned} \tag{2.57}$$

We would like to bound the right hand side of (2.57) in terms of a modulus of smoothness in order to reach a Jackson type estimate. At this point, we employ a Whitney type estimate for the best local polynomial approximation. We utilize a modified modulus of smoothness (see Lemma 5.1, page 185 in DeVore *et. al* [19] for a 1 dimensional result)

$$\tilde{\omega}_k(f, t, \Omega)_p = \left(t^{-s} \int_{[-t,t]^s} \|\Delta_h^k f\|_{L_p(\Omega_{k,h})}^p \, dh \right)^{1/p}.$$

For τ a simplex in $\mathbb{R}^d$ and $t = \text{size}(\tau)$, a Whitney estimate shows that

$$\inf_{P \in \Pi_{l-1}(\mathbb{R}^d)} \|f - P\|_{L_p(\tau)} \leq c \tilde{\omega}_k(f, t, \tau)_p, \tag{2.58}$$

where c depends only on the smallest angle of τ but not on f and t. The reason why $\tilde{Q}_j$ works well for tetrahedralization in 3D is the fact that the Whitney estimate (2.58) remains valid for any spatial dimension. $\mathcal{T}_j \setminus \mathcal{T}_j^*$ is the part of the tetrahedralization $\mathcal{T}_j$ where refinement is active at every level. Then, in view of (2.36)

$$\text{size}(\Omega_{j,\tau}) \sim 2^{-j}, \quad \tau \in \mathcal{T}_j \setminus \mathcal{T}_j^*. \tag{2.59}$$

Taking the inf over $P \in \Pi_{k-1}(\mathbb{R}^3)$ in (2.57) and using the Whitney estimate (2.58) we conclude

$$\|g - \tilde{Q}_j g\|_{L_2(\tau)} \leq c\tilde{\omega}_k(g, 2^{-j}, \Omega_{j,\tau})_2. \tag{2.60}$$

Remembering (2.56) and summing over $\tau \in \mathcal{T}_j \setminus \mathcal{T}_j^*$ gives rise to

$$\|g - \tilde{Q}_j g\|_{L_2(\Omega)} \leq c\tilde{\omega}_2(g, 2^{-j}, \Omega)_2. \tag{2.61}$$

It is time to switch from the modified modulus of smoothness to the standard one. In fact, they are equivalent: there exist $0 < c_1, c_2 < \infty$ such that

$$c_1\tilde{\omega}_k(f, t, \Omega)_p \leq \omega_k(f, t, \Omega)_p \leq c_2\tilde{\omega}_k(f, t, \Omega)_p.$$

Consequently,

$$\|g - Q_j g\|_{L_2(\Omega)} \leq \|g - \tilde{Q}_j g\|_{L_2(\Omega)} \leq c\ \omega_2(g, 2^{-j}, \Omega)_2, \tag{2.62}$$

and (2.25) yields

$$v_J^{(2)} = O(1), \quad J \to \infty. \tag{2.63}$$

Remark 2.2 Our construction works for any d-dimensional setting where the basis is scaled in the following fashion:

$$\phi_v^{(j)}(w) = 2^{d/2L_{j,v}}\ \delta_{vw}, \quad v, w \in \mathcal{N}_j.$$

However, we cannot define face-adjacency relations for $d > 3$. If such relations can be defined through some topological or geometrical abstraction, our framework naturally extends to d-dimensional local refinement strategies, and hence the optimality of the BPX preconditioner is guaranteed in $\mathbb{R}^d$.

Chapter 3

The Hierarchical Basis Methods

There seems to be no substantial challenge to the classical multiplicative and additive multigrid (MG) methods in the uniform refinement case, as MG and BPX preconditioners sit in their throne looking down onto the rest of the preconditioners. They simply exploit the advantage of the uniform refinement as they require a sequence of subspaces which increase geometrically in dimension.

In this case, they are optimal preconditioners, meaning that they have work estimates per cycle proportional to the number of unknowns N. However, the throne is shaken in the local refinement arena because such a sequence is sometimes difficult or even impossible to achieve. In this arena, the most promising candidate for the crown seems to be the hierarchical basis (HB) methods. For any distribution of unknowns among levels, work per cycle remains proportional to N. In other words, the computational complexity is $O(N)$.

The V-cycle MG exactly becomes HBMG if the smoothing is restricted to newly introduced degrees of freedom. With the same spirit, it suffices to store the data for the newly introduced unknowns for the implementation of HB methods. Once again, the storage complexity of HB methods accomplishes $O(N)$, whereas MG methods lose storage optimality as they require to store and use all the unknowns per cycle, giving rise to $O(JN)$ storage.

Storage becomes the biggest obstacle for realistic problems where adaptive refinement is used, for instance, in biomolecular simulations. Then HB methods become the only remedy due to their optimal storage complexity.

So far, we have not mentioned the condition number $\kappa(A^{(j)})$. Unfortunately, HB methods suffer from a growth of order J^2 and 2^J in 2D and 3D respectively. In contrast, in MG and BPX methods, the generalized condition is usually bounded independent of the number of unknowns. We are interested in the stabilization of HB methods where the condition number complications are remedied. The remedy of interest is the so-called wavelet modified HB (WMHB) method introduced by Vassilevski and Wang [36, 37].

This method stabilizes the classical HB by taking away a piece from the HB function. The piece taken away is the approximate L_2-projection on coarse levels of the HB function. The WMHB is viewed as a stabilization of the HB in the sense that it provides a stable Riesz basis in the Sobolev space $H^s(\Omega), \quad s \in (0,1]$. We closely follow the exposition in Vassilevski and Wang [36, 37].

3.1 Matrix conditioning and stable Riesz bases

Our goal is to shed some light on and figure out the connection between the condition number of $A^{(j)}$ and the selection of the basis $\{\phi_i^{(j)}\}_{i=1}^{N_j}$ for $\mathcal{S}_j$. $\mathcal{S}_j$ usually denotes the linear FE element space in our exposition. Here we insert more abstraction to $\mathcal{S}_j$; that is, it is a finite dimensional Hilbert space. In fact, we do not see a need to change the notation, because eventually $\mathcal{S}_j$ will return to its original definition. Let H be a Hilbert space and let $a(\cdot,\cdot)$ be a bounded bilinear form defined on $H \times H$ satisfying the inf-sup condition.

Definition 3.1 *Inf-sup condition: There exists $\beta > 0$ such that*

$$\beta \|v\| \leq \sup_{w \in H} \frac{a(v,w)}{\|w\|} \quad \forall v \in H. \tag{3.1}$$

The Galerkin method finds $u_j \in \mathcal{S}_j$ such that

$$a(u_j, v_j) = b(v_j), \quad \forall v_j \in \mathcal{S}_j. \tag{3.2}$$

Let $\Phi^{(j)} = \{\phi_i^{(j)} : \ i = 1, 2, \ldots, N_j\}$ be a computational basis for $\mathcal{S}_j$. The approximate solution $v^{(j)} \in \mathcal{S}_j$ can be written in terms of this basis as the expansion

$$v^{(j)} = \sum_{i=1}^{N_j} v_i \phi_i^{(j)},$$

where $v = (v_1, \ldots, v_{N_j})^T$ denotes the coordinates of v with respect to the basis $\Phi^{(j)}$.

We introduce a new inner product in the vector setting by

$$< v, w >_{j,0} = \sum_{i=1}^{N_j} v_i w_i = v \cdot w,$$

and we denote the norm induced by $< \cdot, \cdot >_{j,0}$ as $\|v\|_{j,0}^2 = v \cdot v$. Now, our preparation is complete to build a bridge to the condition number with the following theorem.

Theorem 3.1 *If there exist positive constants $\rho_{j,1}$ and $\rho_{j,2}$ satisfying*

$$\rho_{j,1}\|v\|_{j,0} \leq \|A^{(j)}v\|_{j,0} \leq \rho_{j,2}\|v\|_{j,0}, \tag{3.3}$$

then

$$\kappa(A^{(j)}) \leq \frac{\rho_{j,2}}{\rho_{j,1}}.$$

The best estimate for $\rho_{j,1}$ and $\rho_{j,2}$ is given by

$$\rho_{j,1} = \inf_{v \in \mathcal{S}_j, \ \|v\|_{j,0}=1} \|A^{(j)}v\|_{j,0},$$

$$\rho_{j,2} = \sup_{v \in \mathcal{S}_j, \ \|v\|_{j,0}=1} \|A^{(j)}v\|_{j,0}.$$

Moreover, $\kappa(A^{(j)}) = \frac{\rho_{j,2}}{\rho_{j,1}}$ for the best estimates of $\rho_{j,1}$ and $\rho_{j,2}$.

Utilizing the inf-sup condition (3.1), we will reach the bridge (3.3). Initially, let us observe the following fact for any $v \in \mathcal{S}_j$

$$\|A^{(j)}v\|_{j,0} = \sup_{w \in \mathcal{S}_j, \ \|w\|_{j,0}=1} < A^{(j)}v, w >_{j,0} .$$

By the definition of $A^{(j)}$ we arrive at the bilinear form

$$< A^{(j)}v, w >_{j,0} = a(\sum_{i=1}^{N_j} v_i \phi_i^{(j)}, \sum_{j=1}^{N_j} w_i \phi_i^{(j)}) = a(v, w).$$

Then it follows that

$$\|A^{(j)}v\|_{j,0} = \sup_{w \in \mathcal{S}_j,\ \|w\|_{j,0}=1} a(v, w).$$

Using the inf-sup condition (3.1), we choose

$$\begin{aligned} \rho_{j,1} &= \inf_{v \in \mathcal{S}_j} \sup_{w \in \mathcal{S}_j} \frac{a(v, w)}{\|v\|_{j,0}\ \|w\|_{j,0}}, \\ \rho_{j,2} &= \sup_{v \in \mathcal{S}_j} \sup_{w \in \mathcal{S}_j} \frac{a(v, w)}{\|v\|_{j,0}\ \|w\|_{j,0}}, \end{aligned}$$

so that we can conclude

$$\begin{aligned} \|A^{(j)}v\|_{j,0} &\leq \rho_{j,2}\|v\|_{j,0}, \\ \rho_{j,1}\ \|v\|_{j,0} &\leq \|A^{(j)}v\|_{j,0}, \end{aligned}$$

and hence (3.3) holds. Now, we choose a basis $\{\phi_i^{(j)}\}_{i=1}^{N_j}$ for $\mathcal{S}_j$ so that $\frac{\rho_{j,2}}{\rho_{j,1}}$ is as small as possible in order to keep $\kappa(A^{(j)})$ small.

If $\{\phi_i^{(j)}\}_{i=1}^{N_j}$ is an orthonormal basis with respect to the inner product of H, then $\|\cdot\|_{j,0}$ is the same as $\|\cdot\|_H$ in $\mathcal{S}_j$. More precisely, let $v = \sum_i v_i \phi_i$. Then

$$\|v\|_H^2 = (\sum_i v_i \phi_i, \sum_j v_j \phi_j)_H = \sum_i v_i^2 = < v, v >_{j,0} = \|v\|_{j,0}^2.$$

Therefore, $\rho_{j,2}$ is bounded from above by the norm of the bilinear form $a(\cdot,\cdot)$, and $\rho_{j,1}$ is bounded from below by the parameter β_j, where β_j is the β corresponding to the finite dimensional version of the inf-sup condition (3.1). In practical problems both the norm of $a(\cdot,\cdot)$ and β_j remain uniformly bounded in j. Thus $\kappa(A^{(j)})$ is uniformly bounded. However, it is not practical to assume the existence of an orthonormal basis which is also computationally feasible. In any Hilbert space, the next best thing to an orthonormal basis is the Riesz basis.

Definition 3.2 *For $v = \sum_{i=1}^{\infty} c_i\psi_i$ if there exist two absolute constants σ_1 and σ_2 satisfying*

$$\sigma_1\|v\|_H^2 \leq \sum_{i=1}^{\infty} c_i^2 \leq \sigma_2\|v\|_H^2, \quad \forall v \in H, \tag{3.4}$$

then $\{\psi_i\}_{i=1}^{\infty}$ is called a Riesz basis.

The condition (3.4) for finite dimensional H_j can be written as

$$\sigma_1^{(j)} \leq \frac{\|v\|_{j,0}^2}{\|v\|_H^2} \leq \sigma_2^{(j)}, \quad \forall v \in H_j. \tag{3.5}$$

Since $\{\phi_i^{(j)}\}_{i=1}^{N_j}$ is a basis for $\mathcal{S}_j$, then (3.5) holds for

$$\sigma_2^{(j)} = \sup_{v\in\mathcal{S}_j,\ \|v\|_H=1} \|v\|_{j,0}^2,$$
$$\sigma_1^{(j)} = \inf_{v\in\mathcal{S}_j,\ \|v\|_H=1} \|v\|_{j,0}^2.$$

Then the crucial task becomes to gain some control over the ratio $\sigma_2^{(j)}/\sigma_1^{(j)}$.

Definition 3.3 *The family $\left\{\Phi^{(j)} \equiv \{\phi_i^{(j)}\}_{i=1}^{N_j}\right\}$ is said to be a uniformly stable family of Riesz bases for $\mathcal{S}_j$ if there exists c independent of j such that*

$$\sigma_2^{(j)}/\sigma_1^{(j)} \leq c, \quad j = 1, 2, \ldots,$$

meaning that Riesz bounds are uniformly bounded as $j \to \infty$.

We used various quantities, and the following theorem describes their relation to the condition number.

Theorem 3.2 *Let $\{\phi_i^{(j)}\}_{i=1}^{N_j}$ be a computational basis of $\mathcal{S}_j$ satisfying (3.5). Then the condition number with respect to this basis is bounded by*

$$\kappa(A^{(j)}) \leq \frac{\|a\|}{\beta_j} \frac{\sigma_2^{(j)}}{\sigma_1^{(j)}}. \tag{3.6}$$

Theorem 3.2 contains two important factors: $\frac{\|a\|}{\beta_j}$ and $\frac{\sigma_2^{(j)}}{\sigma_1^{(j)}}$. The first one depends on the norm of the given bilinear form and the stability constant β_j from the inf-sup condition. In practical computations, the space $\mathcal{S}_j$ must be constructed so that $\beta_j > \beta^* > 0$ is guaranteed for all j. The second factor is basis dependent. More precisely, it represents a characterization of the difference between the discrete norm $\|\cdot\|_{j,0}$ and the continuous one $\|\cdot\|_H$. Here stable Riesz bases are paramount because the corresponding discretization matrices are well conditioned.

From the aspect of computational feasibility, the selection of the basis is very important because one has to represent $A^{(j)}$ as sparse as possible. The standard requirement is, of course, $O(1)$ nonzeros per row. Although the WMHB representation produces more nonzero entries than the nodal basis representation, it still can achieve a $O(1)$ nonzero pattern per row.

3.2 Motivation and the general approach

The basic idea stems from the fact that the L_2-orthonormal basis of wavelets is also H^1-stable in applications to PDEs. Hence, wavelet bases are good candidates in formulating matrix problems for (3.2). Traditional wavelet bases have complicated structures limiting their numerical applicability. We turn our attention to approximations of wavelet bases.

Assume the standard nestedness of subspaces:

$$\mathcal{S}_0 \subset \mathcal{S}_1 \ldots \subset \mathcal{S}_j \subset \ldots .$$

Utilizing the strong topology induced by the L_2-norm for the closure, we have

$$L_2(\Omega) = \overline{\bigcup_{j=0}^{\infty} \mathcal{S}_j}. \tag{3.7}$$

Let $\mathcal{O}_j$ be the slice space as being the L_2-orthogonal complement of $\mathcal{S}_{j-1}$ in $\mathcal{S}_j$ satisfying

$$\mathcal{S}_j = \mathcal{S}_{j-1} \oplus \mathcal{O}_j, \quad j \geq 1, \quad \mathcal{O}_i \perp \mathcal{O}_j, i \neq j. \tag{3.8}$$

Letting $\mathcal{O}_0 = \mathcal{S}_0$, we get

$$\mathcal{S}_J = \bigoplus_{j=0}^{J} \mathcal{O}_j. \tag{3.9}$$

By (3.7) and the two level decomposition of $\mathcal{S}_j$ (3.8), we can view $L_2(\Omega)$ in terms of slice spaces where slices are mutually orthogonal subspaces. Then

$$L_2(\Omega) = \bigoplus_{j=0}^{\infty} \mathcal{O}_j.$$

If it is possible to find an orthonormal basis for each $\mathcal{O}_j$, then a wavelet basis for $L_2(\Omega)$ can be constructed. Such a basis would be ideal for preconditioning the discrete problem (3.2) if it is computationally feasible. In practice, a computable L_2-orthonormal basis is extremely hard to find. We accommodate this complication by relaxing the orthogonality for slice subspaces and only allowing a direct decomposition of the form below, instead of (3.9)

$$\mathcal{S}_J = \mathcal{S}_0 \oplus \mathcal{S}_1^f \oplus \ldots \oplus \mathcal{S}_J^f, \tag{3.10}$$

where S_j^f denotes the space spanned by the basis functions that corresponds to the newly introduced nodes, and it is a complement of $\mathcal{S}_{j-1}$ in $\mathcal{S}_j$ such that

$$\mathcal{S}_j = \mathcal{S}_{j-1} \oplus \mathcal{S}_j^f. \tag{3.11}$$

We should remark that it is important to have some sort of approximate orthogonality among S_j^f in order to achieve the stability of the wavelet basis.

The general approach relies on the existence of computationally feasible projections from a dense subset of H onto $\mathcal{S}_j$. This mechanism is used for deriving the hierarchical complement of each $\mathcal{S}_{j-1}$ in $\mathcal{S}_j$. We assume that $\pi_j\,\psi_j = \psi_j, \;\; \forall\,\psi_j \in \mathcal{S}_j$ giving $\pi_j\pi_i = \pi_i$ for $j \geq i$, if $\mathcal{S}_i \subset \mathcal{S}_j$. Finally, choosing the complement $\mathcal{S}_j^f$ of $\mathcal{S}_{j-1}$ in $\mathcal{S}_j$ as

$$\mathcal{S}_j^f = (I - \pi_{j-1})\mathcal{S}_j,$$

we have the following two level direct decomposition

$$\mathcal{S}_j = \mathcal{S}_{j-1} \oplus (I - \pi_{j-1})\mathcal{S}_j.$$

Definition 3.4 *Let $\{\phi_i^{(j)},\ i = 1, \ldots, N_j\}$ be a computationally feasible basis of $\mathcal{S}_j$. Assume*
$\{\phi_i^{(j-1)},\ i = 1, \ldots, N_{j-1}\} \cup \{\phi_i^{(j)},\ i = N_{j-1}+1, \ldots, N_j\}$ forms a basis of $\mathcal{S}_j$. Then, a multilevel (ML) basis for $\mathcal{S}_j$ is defined as

$$\Phi^{(J)} = \bigcup_{j=0}^{J} \{(I - \pi_{j-1})\phi_i^{(j)},\ i = N_{j-1} + 1, \ldots, N_j\},$$

with the assumption that $\pi_{-1} = 0$ and $N_{-1} = 0$.

After introducing MLHB, we now argue its stability. The following theorem is a guideline for the selection of π_j.

Theorem 3.3 *If $\Phi^{(J)}$ is a stable Riesz basis for $\mathcal{S}_J$, then π_j is uniformly bounded in $\|\cdot\|_H$-norm on $\mathcal{S}_J$ with respect to j and J for $j \leq J$.*

By choosing $\pi_j = I_j$, as the nodal interpolation operator, MLHB corresponds to the basis in the HB method [7, 41]. In the light of Theorem 3.3, one can determine if the HB is stable. In the model problem, the underlying norm is the H^1-norm for $\mathcal{S}_j$. Unfortunately, I_j is not bounded in the H^1-norm [16]. As a result, Theorem 3.3 implies that the HB is not absolutely stable, meaning that the stiffness matrix with respect to the HB will not be well conditioned. However, the HB provides an improved condition number which grows much slower than that of the standard nodal basis, which behaves like $O(h_J^{-2}) = O(2^{2J})$ at level J. As indicated in [4], the 2D condition number is $O(J^2)$ whereas the 3D condition number for a quasiuniform mesh is $O(N^{1/3}) = O((h_J^{-d})^{1/3}) = O(h_J^{-1}) = O(2^J)$.

As the BPX preconditioner has been a breakthrough over the HB methods by changing I_j with Q_j, we are tempted to do the same, namely, we would like to insert Q_j in the choice for π_j. On the other hand, H^1-stability of π_j is a necessary condition for a stable basis. If π_j boils down to Q_j, then the stability expectation from the basis is high, not only in the uniformly refined meshes but also in the locally refined meshes. It is a well known fact that Q_j is H^1-stable. Furthermore, Bramble, Pasciak, and Steinbach [12] and Carstensen [15] formulated mesh conditions for the locally refined cases such that H^1-stability of the L_2-projection is established. With high optimism, we should as well remark that H^1-stability of π_j is not a sufficient condition for basis stability. So, there is plenty to prove for the locally refined case.

The projection operators of major interest are defined as

$$\pi_k = \prod_{j=k}^{J-1}(I_j + Q_j(I_{j+1} - I_j)), \tag{3.12}$$

with $\pi_J = I$. π_k reduces to the exact L_2-projection, by the following observation

$$\pi_k = Q_k \quad I_{k+1}Q_{k+1} \quad I_{k+2}Q_{k+2} \dots I_{J-1}Q_{J-1} \quad I_J = Q_k \dots Q_{J-1} = Q_k,$$

on $\mathcal{S}_J$. Further properties of π_k can be listed as

- $\pi_k^2 = \pi_k$,
- $\pi_{k-1}(I_k - I_{k-1}) = Q_{k-1}(I_k - I_{k-1})$,
- $\pi_k - \pi_{k-1} = (I - Q_{k-1})(I_k - I_{k-1})\pi_k$.

Using the second property, the MLHB reads as

$$\{\phi_i^{(0)},\ i = 1, \dots, N_0\} \bigcup_{j=1}^{J} \{(I - Q_{j-1})\phi_i^{(j)}, i = N_{j-1} + 1, \dots, N_j\}.$$

Since $(I-Q_{j-1})\phi_i^{(j)} = (I-Q_{j-1})(I_j-I_{j-1})\phi_i^{(j)}$, $\phi_i^{(j)} \in \mathcal{S}_j^f$, the above components $\{(I - Q_{j-1})\phi_i^{(j)},\ i = N_{j-1} + 1, \dots, N_j\}$ can be viewed as a modification of the classical HB components $\{(I_j - I_{j-1})\phi_i^{(j)},\ i = N_{j-1}+1, \dots, N_j\}$. The modification is made by subtracting from each nodal HB function $\phi_i^{(j)}$ its L^2-projection onto the coarse level $j-1$. We assert in the following lemma that the modification forms a basis for $\mathcal{S}_j^f$. Since we modify the HB decomposition, it is worth mentioning the HB decomposition shortly. The slice spaces are denoted by $\mathcal{S}_j^{(f)}$ rather than $\mathcal{S}_j^f$ and are defined as

$$\mathcal{S}_j^{(f)} = (I_j - I_{j-1})\mathcal{S}_j,$$

and the corresponding multilevel decomposition is

$$\mathcal{S}_J = \mathcal{S}_0 \oplus \mathcal{S}_1^{(f)} \oplus \ldots \oplus \mathcal{S}_J^{(f)}.$$

Lemma 3.4 *Let $(I - \pi_{j-1})\phi_i^{(j)}$ be the modification of each HB $\phi_i^{(j)} \in \mathcal{S}_j^{(f)}$. Then,*

$$\{(I - \pi_{j-1})\phi_i^{(j)} : \quad x_i \in \mathcal{N}_j^f \setminus \mathcal{N}_{j-1}\},$$

forms a basis for $\mathcal{S}_j^f$. Moreover,

$$\mathcal{S}_j^f \equiv (I - \pi_{j-1})\mathcal{S}_j^{(f)} = (I - \pi_{j-1})(I_j - I_{j-1})\mathcal{S}_j^f,$$

gives the desired multilevel decomposition in (3.10) and the two level decomposition in (3.11).

Proof. See Lemma 3.1, page 110 in Vassilevski and Wang [36]. □

We will depict such modifications in the figures below.

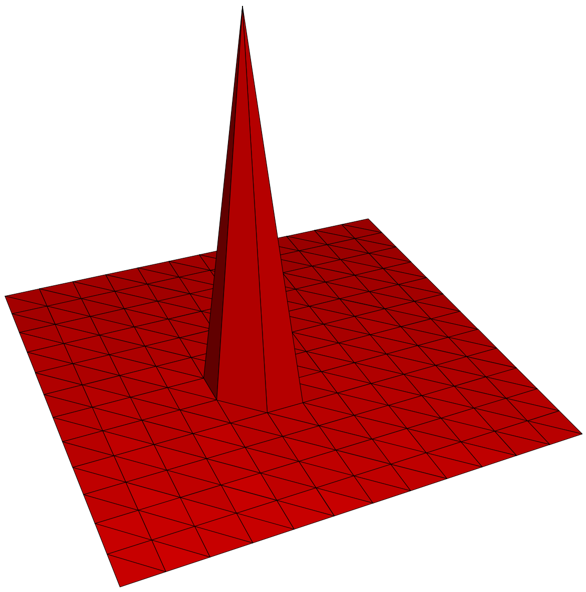

Figure 3.1: Nodal basis function.

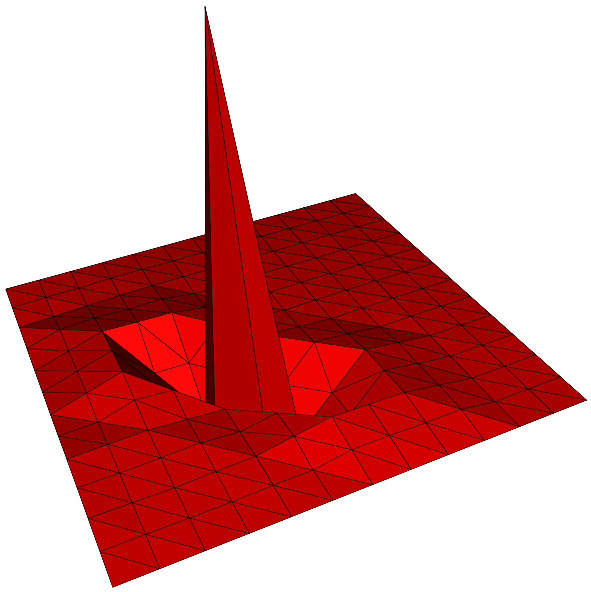

Figure 3.2: Wavelet modified hierarchical basis function with one iteration of symmetric Gauss-Seidel approximation, upper view.

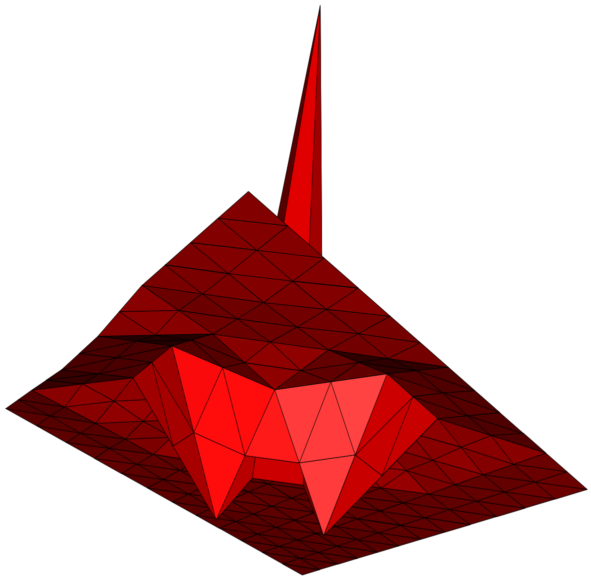

Figure 3.3: Wavelet modified hierarchical basis function with one iteration of symmetric Gauss-Seidel approximation, lower view.

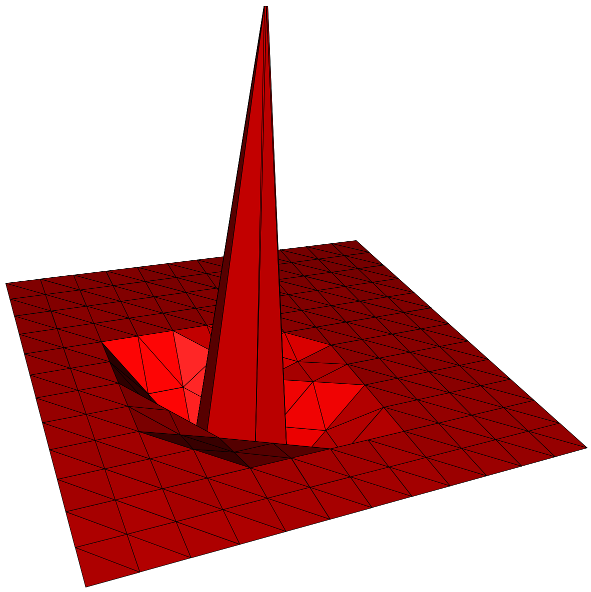

Figure 3.4: Wavelet modified hierarchical basis function with one iteration of Jacobi approximation, upper view.

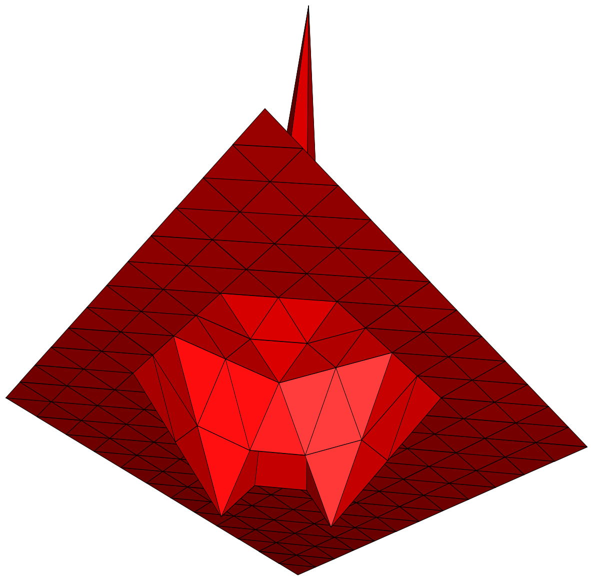

Figure 3.5: Wavelet modified hierarchical basis function with one iteration of Jacobi approximation, lower view.

Observe the larger support of the wavelet modified basis function in Figure 3.2 compared to Figure 3.4. We close this section by noting that modified HB functions are close relatives of the Battle-Lemarie wavelets.

3.3 The WMHB method as function space operators

In the abstract framework, we can analogously interpret the hierarchical structure of the linear algebra setting of HB methods as operator relations in the abstract function space. Namely,

- The discretization operator $A^{(j)} : \mathcal{S}_j \to \mathcal{S}_j$ is defined by

$$(A^{(j)}\varphi, \psi) = a(\varphi, \psi), \quad \forall \varphi, \psi \in \mathcal{S}_j.$$

- The discretization operator $A^{(j)}_{22} : \mathcal{S}^f_j \to \mathcal{S}^f_j$ is the restriction of $A^{(j)}$ onto $\mathcal{S}^f_j$, defined by

$$(A^{(j)}_{22}\psi^f, \varphi^f) = a(\psi^f, \varphi^f) \ \ \forall \varphi^f, \psi^f \in \mathcal{S}^f_j.$$

- The communication operators

$$A^{(j)}_{12} : \mathcal{S}^f_j \to \mathcal{S}_{j-1}, \ \ A^{(j)}_{21} : \mathcal{S}_{j-1} \to \mathcal{S}^f_j,$$

  are given by:

$$(A^{(j)}_{12}\varphi^f, \tilde{\psi}) = (\varphi^f, A^{(j)}_{21}\tilde{\psi}) = a(\varphi^f, \tilde{\psi}) \quad \forall \tilde{\psi} \in \mathcal{S}_{j-1}, \quad \varphi^f \in \mathcal{S}^f_j.$$

Since the decomposition $\mathcal{S}_j = \mathcal{S}_{j-1} \oplus \mathcal{S}^f_j$ in (3.11) is direct, $A^{(j)}$ can be represented by a two-by-two block form:

$$A^{(j)} = \begin{array}{cl} \left[\begin{array}{cc} A^{(j-1)} & A^{(j)}_{12} \\ A^{(j)}_{21} & A^{(j)}_{22} \end{array} \right] & \begin{array}{l} \} \ \mathcal{S}_{j-1} \\ \} \ \mathcal{S}^f_j \end{array} \end{array}. \tag{3.13}$$

3.4 High level spectral equivalence result

We assume the existence of given approximations $B_{22}^{(j)}$, SPD in $\mathcal{S}_j^f$, to the operators $A_{22}^{(j)}, \quad j = 1, 2, \ldots, J$

Assumption 3.1

$$(A_{22}^{(j)}v^f, v^f) \le (B_{22}^{(j)}v^f, v^f) \le (1 + b_1)(A_{22}^{(j)}v^f, v^f) \quad \forall v^f \in \mathcal{S}_j^f. \tag{3.14}$$

Definition 3.5 *Let the preconditioner $B^{(j)}$ be defined as*

$$B^{(j)} \equiv \begin{bmatrix} B^{(j-1)} & A_{12}^{(j)} \\ 0 & B_{22}^{(j)} \end{bmatrix} \begin{bmatrix} I & 0 \\ B_{22}^{(j)^{-1}} A_{21}^{(j)} & B_{22}^{(j)} \end{bmatrix} = \begin{bmatrix} B^{(j-1)} + A_{12}^{(j)} B_{22}^{(j)^{-1}} A_{21}^{(j)} & A_{12}^{(j)} \\ A_{21}^{(j)} & I \end{bmatrix}.$$

Using the two-by-two block definition of $A^{(j)}$ as in (3.13), we define the error operator $E^{(j)}$ as

$$E^{(j)} \equiv B^{(j)} - A^{(j)} = \begin{bmatrix} B^{(j-1)} - A^{(j-1)} + A_{12}^{(j)} B_{22}^{(j)^{-1}} A_{21}^{(j)} & 0 \\ 0 & B_{22}^{(j)} - A_{22}^{(j)} \end{bmatrix} \begin{matrix} \} & \mathcal{S}_{j-1} \\ \} & \mathcal{S}_j^f \end{matrix}.$$

To realize the action of the error operator, we decompose $v \in \mathcal{S}_j$ as $v = \tilde{v} + v^f$, where $\tilde{v} \in \mathcal{S}_{j-1}, \; v^f \in \mathcal{S}_j^f$. The action of $E^{(j)}$ then can be characterized as follows:

$$(E^{(j)}v, v) = ((B_{22}^{(j)} - A_{22}^{(j)})v^f, v^f) + (E^{(j-1)}\tilde{v}, \tilde{v}) + (B_{22}^{(j)^{-1}} A_{21}^{(j)}\tilde{v}, A_{21}^{(j)}\tilde{v}).$$

Our intention is to formalize the spectral equivalence of $A^{(j)}$ and $B^{(j)}$ in terms of $E^{(j)}$. Next, we verify standard requirements and characterize $E^{(j)}$ by utilizing the direct decomposition (3.11).

Claim 3.1 *The operator $E^{(j)}$ is positive semidefinite.*

Proof. This is true for $j = 0$, because $E^{(0)} = 0$. Assume, $0 \le (E^{(j-1)}v, v)$. Then, by (3.14) $0 \le ((B_{22}^{(j)} - A_{22}^{(j)})v^f, v^f)$. Also, $B_{22}^{(j)}$, SPD in $\mathcal{S}_j^f$, implies $0 \le (B_{22}^{(j)^{-1}} A_{21}^{(j)}\tilde{v}, A_{21}^{(j)}\tilde{v})$. $\square$

By (3.14) we get;

$$(E^{(j)}v,v) \le b_1(A_{22}^{(j)})v^f,v^f) + (E^{(j-1)}\tilde{v},\tilde{v}) + (B_{22}^{(j)^{-1}}A_{21}^{(j)}\tilde{v},A_{21}^{(j)}\tilde{v}).$$

In general, $v^{(j)} \in \mathcal{S}_j$ has the decomposition

$$v^{(j)} = v^{(j-1)} + v^{(j)^f}, \quad v^{(j-1)} \in \mathcal{S}_{j-1}, \ v^{(j)^f} \in \mathcal{S}_j^f.$$

Then,

$$(E^{(j)}v,v) - (E^{(j-1)}v^{(j-1)},v^{(j-1)}) \le b_1(A_{22}^{(j)}v^{(j)^f},v^{(j)^f}) + (B_{22}^{(j)^{-1}}A_{21}^{(j)}v^{(j-1)},A_{21}^{(j)}v^{(j-1)}).$$

Summing over j, with $v = v^{(J)}$

$$(E^{(J)}v,v) \le b_1\sum_{j=1}^{J}(A_{22}^{(j)}v^{(j)^f},v^{(j)^f}) + \sum_{j=1}^{J}(B_{22}^{(j)^{-1}}A_{21}^{(j)}v^{(j-1)},A_{21}^{(j)}v^{(j-1)}). \tag{3.15}$$

We have presented the error estimate $E^{(J)}$ as an operator acting on a function space. In order to relate the sums appearing in (3.15) to $(A^{(J)}v,v)$, we will employ inequalities (3.16) and (3.17) respectively.

$$\sum_{j=1}^{J}(A_{22}^{(j)}v^{(j)^f},v^{(j)^f}) \le \rho_1(A^{(J)}v,v), \tag{3.16}$$

$$\sum_{j=1}^{J}(B_{22}^{(j)^{-1}}A_{21}^{(j)}v^{(j-1)},A_{21}^{(j)}v^{(j-1)}) \le \rho_2(A^{(J)}v,v). \tag{3.17}$$

In this abstract setting, the norm equivalence between the H^1-norm and the norm corresponding to the WMHB preconditioner is equivalent to analyzing the spectral estimates of the operator $A^{(J)}$ and $B^{(J)}$. With all the three assumptions in place, we analyze the spectrum of the $A^{(J)}$ and $B^{(J)}$ operators.

Theorem 3.5 *If Assumption 3.1, and inequalities (3.16) and (3.17) hold true, then*

$$(A^{(J)}v,v) \le (B^{(J)}v,v) \le (b_1\rho_1 + \rho_2)(A^{(J)}v,v) \quad \forall v \in \mathcal{S}_J. \tag{3.18}$$

Proof. The first inequality is attained by Claim 3.1. The second one follows from inequalities 3.16 and 3.17. □

3.5 The hierarchical decomposition and well conditioned $A_{22}^{(j)}$

As mentioned before, finding a computable L_2-orthonormal basis is difficult. This has forced us to replace the orthogonality in the two level decomposition (3.8) by the direct sum in (3.11). As we refine the mesh, in order to reach to a better solution for the discrete system (3.2), we naturally introduce a hierarchical ordering for the nodal points in the mesh. Denoting the newly introduced nodal points as $\mathcal{N}_j^f$, one can observe the direct decomposition of the nodal points at level j

$$\mathcal{N}_j = \mathcal{N}_{j-1} \bigcup \mathcal{N}_j^f. \tag{3.19}$$

Reflecting the hierarchical ordering of nodal points onto the corresponding nodal basis functions, we obtained the direct sum of the space $\mathcal{S}_j$ in (3.11). In fact, this two level decomposition as originally proposed by Bank and Dupont [6], is the spirit of HB methods. Applying the two level decomposition (3.11) may not give a stable splitting of $\mathcal{S}_j$. In the linear algebra setting, this means that the submatrix $A^{(j-1)}$ may not be well conditioned. This difficulty can be overcome by repeating the above procedure so that $\mathcal{S}_j$ can be represented by slice spaces. Such splitting will turn out not only to be stable, but it will also provide the advantage of having well conditioned submatrices $A_{22}^{(j)}$. The above considerations now justify the decomposition (3.10).

Lemma 3.6 *$A_{22}^{(j)}$ is well conditioned:*
If $\lambda_{j,min}^f$ and $\lambda_{j,max}^f$ are the smallest and largest eigenvalues of $A_{22}^{(j)}$, then there exist constants c_1 and c_2 both independent of j such that

$$c_1 2^{2j} \leq \lambda_{j,min}^f \leq \lambda_{j,max}^f \leq c_2 2^{2j}. \tag{3.20}$$

We leave the proof of Lemma 3.6 for the next subsection because it requires some norm equivalence results.

3.5.1 Norm equivalence for modified HB

The goal here is to verify the well-conditionedness of the matrix $A_{22}^{(j)}$ that was claimed in Lemma 3.6. We first establish a norm equivalence for the modified hierarchical basis.

Lemma 3.7 *Let $\mathcal{S}_j^f = (I - \pi_{j-1})\mathcal{S}_j^{(f)}$ be the modified hierarchical subspace of level j. Then, there are constants c_1 and c_2 independent of j such that for any $\psi^f = (I - \pi_{j-1})\phi^f \in \mathcal{S}_j^f$ with $\phi^f \in \mathcal{S}_j^{(f)}$,*

$$c_1\|\phi^f\|_{H^1}^2 \leq \|\psi^f\|_{H^1}^2 \leq c_2\|\phi^f\|_{H^1}^2. \tag{3.21}$$

Proof. We extend the existing proof of Vassilevski and Wang (Lemma 6.1, page 121 in [37]) for quasiuniform meshes to the locally refined ones. The strengthened C-S inequality is central to the proof: there exists $\delta \in (0,1)$ independent of the mesh size or the level j such that

$$|(\nabla\phi^f, \nabla\tilde{\phi})| \leq \delta(\nabla\tilde{\phi}, \nabla\tilde{\phi})^{1/2}(\nabla\phi^f, \nabla\phi^f)^{1/2}, \quad \forall\tilde{\phi} \in \mathcal{S}_{j-1}, \phi^f \in \mathcal{S}_j^{(f)}.$$

We derive an equivalent C-S like inequality

$$\begin{aligned}
(\nabla(\tilde{\phi} + \phi^f), \nabla(\tilde{\phi} + \phi^f)) &= (\nabla\tilde{\phi}, \nabla\tilde{\phi}) + 2(\nabla\tilde{\phi}, \nabla\phi^f) + (\nabla\phi^f, \nabla\phi^f) + (\nabla\phi^f, \nabla\phi^f)\\
&\geq (\nabla\tilde{\phi}, \nabla\tilde{\phi}) - 2\delta(\nabla\tilde{\phi}, \nabla\tilde{\phi})^{1/2}(\nabla\phi^f, \nabla\phi^f)^{1/2} + (\nabla\phi^f, \nabla\phi^f)\\
&= (\nabla\tilde{\phi}, \nabla\tilde{\phi}) - 2\delta(\nabla\tilde{\phi}, \nabla\tilde{\phi})^{1/2}(\nabla\phi^f, \nabla\phi^f)^{1/2} +\\
&\quad \delta^2(\nabla\phi^f, \nabla\phi^f) + (1-\delta^2)(\nabla\phi^f, \nabla\phi^f)\\
&= \left\{(\nabla\tilde{\phi}, \nabla\tilde{\phi})^{1/2} + \delta(\nabla\phi^f, \nabla\phi^f)^{1/2}\right\}^2 + (1-\delta^2)(\nabla\phi^f, \nabla\phi^f)\\
&\geq (1-\delta^2)(\nabla\phi^f, \nabla\phi^f).
\end{aligned}$$

Repeat the same procedure for

$$|(\tilde{\phi}, \phi^f)| \leq \delta(\tilde{\phi}, \tilde{\phi})^{1/2}(\phi^f, \phi^f)^{1/2},$$

then this implies

$$(1-\delta^2)\|\phi^f\|_{L_2}^2 \leq \|\tilde{\phi} + \phi^f\|_{L_2}^2.$$

This and the above result give

$$(1-\delta^2)\|\phi^f\|_{H^1}^2 \leq \|\tilde{\phi}+\phi^f\|_{H^1}^2.$$

Choosing $\tilde{\phi} = -\pi_{j-1}\phi^f$, then

$$(1-\delta^2)\|\phi^f\|_{H^1}^2 \leq \|\psi^f\|_{H^1}^2.$$

Then, the left side of the inequality holds for

$$c_1 = 1-\delta^2.$$

The red refinement gives rise to $\mathcal{S}_j$ which is a true subspace of $\tilde{\mathcal{S}}_j$. Then, $\mathcal{S}_j^f \subset \mathcal{S}_j \subset \tilde{\mathcal{S}}_j$. Hence, the local inverse inequality (3.27) holds, and it can easily be extended to a global inverse inequality. Then,

$$\begin{aligned}\|\psi^f\|_{H^1}^2 &\leq c_0 2^{2j}\|\psi^f\|_{L_2}^2 = c_0 2^{2j}\|-\pi_{j-1}\phi^f+\phi^f\|_{L_2}^2 \\ &\leq c_0 2^{2j}\left(1+\|\pi_{j-1}\|_{L_2}\right)^2\|\phi^f\|_{L_2}^2 \leq c2^{2j}\|\phi^f\|_{L_2}^2.\end{aligned}$$

We have used the L_2 boundedness of π_{j-1}. We remark that the above inverse inequality argument also holds for the red-green refinement for $\mathcal{S}_j^f$. The region Ω_j^f corresponding to $\mathcal{S}_j^f$ is exposed to uniform refinement, hence elements in that region have $L(\tau) = j$. Then we can relate to their fathers like in the case uniform refinement; there exists c_1, c_2 such that

$$c_1 \text{ size}(\overline{\tau})\ 2^{-j} \leq \text{size}(\tau) \leq c_2 \text{ size}(\overline{\tau})\ 2^{-j}.$$

Inverse inequality is then guaranteed on $\mathcal{S}_j^f$ by the quasiuniformity of the region $\Omega_j^{(f)}$. The slice space $\mathcal{S}_j^f$ is oscillatory. Then there exists c such that $\|\phi^f\|_{L_2}^2 \leq c2^{-2j}\|\phi^f\|_{H^1}^2$. This observation can be verified by the BPX lower bound in (a.i), namely,

$$2^{2j}\|(Q_j-Q_{j-1})\phi^f\|_{L_2}^2 = 2^{2j}\|\phi^f\|_{L_2}^2 \leq c\|\phi^f\|_{H^1}^2.$$

Then, $\|\psi^f\|_{H^1}^2 \leq c\|\phi^f\|_{H^1}^2$. □

Proof. Proof of Lemma 3.6

For any $\psi^f = (I - \pi_{j-1})\phi^f \in \mathcal{S}_j^f$ since

$$(A_{22}^{(j)}\psi^f, \psi^f) = a(\psi^f, \psi^f),$$

and the bilinear form $a(\cdot,\cdot)$ is equivalent to the H^1-inner product, there exists c_1, c_2 such that

$$c_1\|\psi^f\|_{H^1}^2 \leq (A_{22}^{(j)}\psi^f, \psi^f) \leq c_2\|\psi^f\|_{H^1}^2.$$

Using the norm equivalence in (3.21),

$$c_1 2^{2j}\|\phi^f\|_{L_2}^2 \leq (A_{22}^{(j)}\psi^f, \psi^f) \leq c_2 2^{2j}\|\psi^f\|_{L_2}^2.$$

Since π_{j-1} is L_2-bounded

$$c_1 2^{2j}\|\psi^f\|_{L_2}^2 \leq (A_{22}^{(j)}\psi^f, \psi^f) \leq c_2 2^{2j}\|\psi^f\|_{L_2}^2,$$

which implies the lemma. □

3.6 Assumptions of the WMHB

The realization of the inequalities (3.16) and (3.17) determine the spectral equivalence (2.4). We proceed with verifying these assumptions for the refinement strategy under consideration. We move to a higher abstraction layer by inserting assumptions that relate to the ingredients of the splitting and the angles among the FE spaces.

As in the BPX splitting, the main ingredient is the L_2-projection in the WMHB splitting. Hence, the stability of the BPX splitting is still important in the WMHB splitting. We assume the difficult part in the BPX norm equivalence, namely the lower bound.

Our objective is to establish the inequalities (3.16) and (3.17). The argument is based on two fundamental inequalities in the multilevel theory. Namely, there exists a constant σ independent of J satisfying,

Assumption 3.2

$$(a.i) \quad \|Q_0 u\|_{H^1}^2 + \sum_{j=1}^{J} 2^{2j} \|(Q_j - Q_{j-1})u\|_{L_2}^2 \le \sigma \|u\|_{H^1}^2 \quad \forall u \in \mathcal{S}_J.$$

Assumption 3.3

$$(a.ii) \quad |a(u_i, u_j)|^2 \le \sigma \delta^{2(j-i)} 2^{2j} a(u_i, u_i) \|u_j\|_{L_2}^2, \quad \forall u_i \in \mathcal{S}_i,\ u_j \in \mathcal{S}_j,\ j \ge i,$$

where $\delta \in (0, 1)$ *for* $i = 1, \ldots, J$.

The assumption (a.ii) establishes a Cauchy-Schwarz (C-S) like estimate between the FE spaces.

Lemma 3.8 *If (a.i) holds, then there exists a constant c such that*

$$\sum_{j=1}^{J} (A_{22} v^{(j)^f}, v^{(j)^f}) \le c\ (A^{(J)} v, v) \quad \forall v \in \mathcal{S}_J. \tag{3.22}$$

Proof. See the Proof of Lemma 4.1, page 114 in Vassilevski and Wang [36]. □

Lemma 3.9 *If (a.i) and (a.ii) hold true, then there exists a constant c such that*

$$\sum_{j=1}^{J} 2^{-2j} \|A_{21} v^{(j-1)}\|_{L_2}^2 \le c\ a(v, v) \quad \forall v \in \mathcal{S}_J. \tag{3.23}$$

Proof. See the Proof of Lemma 4.2, page 114 in Vassilevski and Wang [36]. □

Let us verify Assumption 3.1. Lemma 3.6 indicates that $A_{22}^{(j)}$ is well conditioned. Thus, one may choose a diagonal preconditioner $B_{22}^{(j)} = \alpha 2^{2j} I$ for the matrix $A_{22}^{(j)}$. Here α is a parameter which should be adjusted so that Assumption 3.1 is satisfied for some b_1. With the above selection of $B_{22}^{(j)}$, and by using (3.23), we get

$$\sum_{j=1}^{J} (B_{22}^{(j)^{-1}} A_{21}^{(j)} v^{(j-1)}, A_{21}^{(j)} v^{(j-1)}) \le c \sum_{j=1}^{J} 2^{-2j} \|A_{21} v^{(j-1)}\|_{L_2}^2 \le c\ (A^{(J)} v, v).$$

By assuming (a.i) and (a.ii), we are in a position to satisfy inequalities (3.16) and (3.17) by (3.22) and (3.23) respectively, in the wavelet basis. Hence, the spectral equivalence follows as in the below theorem:

Theorem 3.10 *If (a.i) and (a.ii) hold, then there exists a constant c independent of j such that*

$$(A^{(j)}v, v) \leq (B^{(j)}v, v) \leq c\ (A^{(j)}v, v) \quad \forall v \in \mathcal{S}_j,$$

for $j = 0, 1, \ldots, J$. The constant c depends only on b_1 from (3.14), δ from (a.ii), and σ from (a.i), (a.ii).

3.7 2D red refinement

As mentioned in the BPX section 2.1, the lower bound has been proven by many people for the uniform refinement. There have also been contributions for the local refinement. Bornemann and Yserentant [10] showed that the BPX preconditioner is optimal for 2D red refinement. Dahmen and Kunoth [17] showed the same result for 2D red-green refinement. We extended their result to 3D. In short, if we want to extend WMHB optimal estimates to locally refined cases, then potential candidates are 2D red refinement, 2D and 3D reed-green refinements, or some other local refinement routine. Assumption (a.ii) establishes a C-S like estimate, basically quantifying the angle between the FE spaces. Unfortunately, we have to eliminate the 2D and 3D red-green refinements from the candidate list because the proof technique poses the following technical difficulties.

The important step in the proof in order to get the term $\delta^{2(j-i)}$, is the boundary strip S of the triangle τ that allows the decomposition

$$w = w_{bdry} + w_{intr}, \tag{3.24}$$

where w_{bdry} live on S and w_{intr} live in the interior (complement of S). If one uses the same technique for local refinement, one has to make sure that S is nonempty to utilize the decomposition (3.24). The resulting strip is contained by the strip $\tilde{S}$ arising in the uniformly refined case (i.e. $S \subset \tilde{S}$). The proof technique aims to obtain a ratio

$$\frac{\text{area}(\tilde{S})}{\text{area}(\tau)} \leq c\ \delta^{2(j-i)}.$$

If $S \subset \tilde{S}$, we then get

$$\frac{\text{area}(S)}{\text{area}(\tau)} \leq \frac{\text{area}(\tilde{S})}{\text{area}(\tau)} \leq c\ \delta^{2(j-i)}.$$

The red-green refinement can be discussed from this aspect. We close the resulting nonconforming (hanging) nodes by green refinement after we are finished with the red refinement. This closure process is chaotic in the sense that area(S) does not decrease as j increases and it would be impossible to obtain the upper bound $\delta^{2(j-i)}$. For this reason we give up on the idea of verifying assumption (a.ii) for red-green refinement. Instead, we focus on a simpler routine, which is 2D red refinement without the green closure. Next, we exhibit the red refinement formalism.

The triangulation $\mathcal{T}_0$ is refined several times, giving a family of nested, possibly nonconforming triangulations $\mathcal{T}_0, \mathcal{T}_1, \ldots$. A triangle of $\mathcal{T}_{j+1}$ is either an element of $\mathcal{T}_j$ or generated by red refinement, i.e. subdivided into 4 congruent subtriangles. The level of a triangle is defined exactly the same as in the case of red-green refinement in (2.31). The mesh may contain unrefined triangles from all lower levels, and thus may be highly nonuniform.

We enforce Assumption 1.5. This means that if initial $\mathcal{T}_0$ and final triangulations $\mathcal{T}_j$ are given, we can uniquely reconstruct the intermediate triangulations $\mathcal{T}_1, \ldots, \mathcal{T}_{j-1}$. We require that the level of two triangles in a triangulation $\mathcal{T}_j$, which have at least one common point, differs at most by one. We have verified the same result for 3D red-green refinement in (2.32). The vertices of the triangles in $\mathcal{T}_j$ are called nodes. The set $\mathcal{N}_j$ is divided into two categories as free (degrees of freedom) and nonfree (hanging) nodes. A function $u \in \mathcal{S}_j$ is determined by its values at the free nodes $x \in \mathcal{N}_j$. The values of u at the free nodes can be given arbitrarily. Under these assumptions, nonfree nodes are always midpoints of edges connecting two free nodes. The values at the nonfree nodes can be computed by linear interpolation along these edges from the values at the corresponding neighboring free nodes. The interpolation process addresses the nonconformity at the nonfree nodes. This completes the red refinement strategy.

In addition to triangulations $\mathcal{T}_j$, we will utilize the uniformly refined triangulations $\mathcal{T}_j$. Again, as in BPX Assumption 2.6, the triangles in $\tilde{\mathcal{T}}_{j+1}$ are generated by subdividing all triangles in $\mathcal{T}_j$ into four congruent subtriangles. We observe that any element in $\mathcal{T}_j$ can be expressed as a union of elements in $\tilde{\mathcal{T}}_j$

$$\tau \in \mathcal{T}_j, \quad \tau = \bigcup_{\tilde{\tau} \in \tilde{\mathcal{T}}_j} \tilde{\tau}. \tag{3.25}$$

In other words, elements in $\tilde{\mathcal{T}}_j$ play the role of the building blocks for elements in $\mathcal{T}_j$. This property is no more valid if the red refinement is complemented by the green refinement.

The triangle relation (3.25) gives rise to the most interesting feature of the red refinement. Namely, basis function that live on elements of $\mathcal{T}_j$ can be uniquely expressed by the basis functions which live on the elements of $\tilde{\mathcal{T}}_j$.

$$u = \sum_{i=1}^{N_j} u_i \phi_i^{(j)} \in \mathcal{S}_j \Rightarrow u = \sum_{i=1}^{\tilde{n}_j} \tilde{u}_i \tilde{\phi}_i^{(j)} \in \tilde{\mathcal{S}}_j. \tag{3.26}$$

So, $\mathcal{S}_j$ is a true subspace of $\tilde{\mathcal{S}}_j$. The fact that $\mathcal{S}_j \subset \tilde{\mathcal{S}}_j$ brings a lot of convenience simply because the standard estimates such as inverse inequality and C-S like estimates which naturally hold for uniformly refined case $\tilde{\mathcal{S}}_j$ can be inherited for $\mathcal{S}_j$. Since $\mathcal{S}_j \subset \tilde{\mathcal{S}}_j$, establishing inverse inequality and C-S like estimates for $\tilde{\mathcal{S}}_j$ will suffice to conclude the same result for the red refinement. The most fascinating fact is that we literally put no effort into extending the uniform refinement estimates to local (red) refinement estimates. Although the resulting meshes from the red refinement is somewhat less flexible than that of red-green refinement, they are sufficiently general to handle any desired type of singularity. As mentioned before, (3.25) does not hold for the red-green refinement. Therefore, it is impossible to employ inverse and C-S like estimates designed for the uniformly refined case. We are merely interested in the red refinement.

3.8 Verification of the assumptions

Bornemann and Yserentant [10] basically dedicated the paper for the establishment of the norm equivalence (2.5). Since (a.i) exactly corresponds to the lower bound the norm equivalence (2.5), we automatically verify (a.i) for 2D red refinement. Now, we will verify the assumption (a.ii) for the uniform refinement. That will suffice to verify the red uniform refinement case as explained in section 3.7. Closely following [10], we report some technical lemmas necessary for verifying (a.ii).

Lemma 3.11 *Inverse inequality: For all functions $v \in \tilde{\mathcal{S}}_j$ and all triangles $\tau \in \tilde{\mathcal{T}}_j$,*

$$|v|^2_{H^1(\tau)} \leq c_0 2^{2j} \|v\|^2_{L_2(\tau)}, \tag{3.27}$$

c_0 depends only on the shape regularity of the triangles in $\mathcal{T}_0$.

Lemma 3.12 *Let τ be a triangle which is a subset of a triangle in $\mathcal{T}_0$. Let $\bar{v}$ be a linear function taking the same values as v at one or two vertices of τ and the value 0 at the remaining vertices of τ. Then,*

$$\|\bar{v}\|^2_{L_2(\tau)} \leq \frac{3}{2} \|v\|^2_{L_2(\tau)}. \tag{3.28}$$

Proof. Let w take the values w_1, w_2, w_3 on vertices $(x_1, y_1), (x_2, y_2), (x_3, y_3)$ of τ respectively, and let $\phi_1(x, y)$, $\phi_2(x, y)$, and $\phi_3(x, y)$ be basis functions corresponding to these vertices. Then, we can write $w(x, y) = w_1\phi_1(x, y) + w_2\phi_2(x, y) + w_3\phi_3(x, y)$.

$$\int_\tau |w(x, y)|^2 \, dxdy = (w, w) =$$

$$\begin{bmatrix} w_1 & w_2 & w_3 \end{bmatrix} \begin{bmatrix} (\phi_1, \phi_1) & (\phi_1, \phi_2) & (\phi_1, \phi_3) \\ (\phi_2, \phi_1) & (\phi_2, \phi_2) & (\phi_2, \phi_3) \\ (\phi_3, \phi_1) & (\phi_3, \phi_2) & (\phi_3, \phi_3) \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix}.$$

Here, the matrix is the mass matrix. We can compute the mass matrix on the reference element by changing the variables with the following rule:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} x_2 - x_1 & x_3 - x_1 \\ y_2 - y_1 & y_3 - y_1 \end{bmatrix} \begin{bmatrix} \hat{x} \\ \hat{y} \end{bmatrix} + \begin{bmatrix} x_1 \\ y_1 \end{bmatrix}.$$

Let $Jac = \begin{bmatrix} x_2 - x_1 & x_3 - x_1 \\ y_2 - y_1 & y_3 - y_1 \end{bmatrix}$. Then, $|det(Jac)| = 2 * \text{area}(\tau)$. We can calculate the entries of Jac on the master triangle, for $d = 2$ yielding to:

$$\begin{bmatrix} (\phi_1, \phi_1) & (\phi_1, \phi_2) & (\phi_1, \phi_3) \\ (\phi_2, \phi_1) & (\phi_2, \phi_2) & (\phi_2, \phi_3) \\ (\phi_3, \phi_1) & (\phi_3, \phi_2) & (\phi_3, \phi_3) \end{bmatrix} = \frac{\text{area}(\tau)}{(d+1)(d+2)} \begin{bmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{bmatrix}.$$

Hence,

$$\int_\tau |w(x,y)|^2 \, dxdy = \frac{\text{area}(\tau)}{12} \left(w_1^2 + w_2^2 + w_3^2 + (w_1 + w_2 + w_3)^2 \right).$$

If $\bar{v}$ agrees with v on two vertices, and the other value is 0, then

Claim I:

$$F(v_1, v_2, 0) \leq \frac{3}{2} F(v_1, v_2, v_3) \quad \forall v_1, v_2, v_3, \tag{3.29}$$

where $F(v_1, v_2, v_3) = v_1^2 + v_2^2 + v_3^2 + (v_1 + v_2 + v_3)^2$.

Pf: Proof by contradiction. Assume there exist v_1, v_2, v_3 such that $F(v_1, v_2, 0) > \frac{3}{2} F(v_1, v_2, v_3)$.

$$v_1^2 + v_2^2 + (v_1 + v_2)^2 > \frac{3}{2} \left(v_1^2 + v_2^2 + v_3^2 \right) + \frac{3}{2} (v_1 + v_2 + v_3)^2.$$

Equivalently,

$$0 > \frac{1}{2}(v_1^2 + v_2^2) + \frac{3}{2} v_3^2 + \frac{1}{2}(v_1 + v_2)^2 + \frac{3}{2}(v_1 + v_2) v_3,$$

giving us a contradiction

$$0 > \frac{1}{2}(v_1^2 + v_2^2) + \left(\frac{\sqrt{3}}{\sqrt{2}} v_3 + \frac{\sqrt{3}}{2\sqrt{2}} (v_1 + v_2) \right)^2 + \frac{1}{8}(v_1 + v_2)^2.$$

Claim II: $\frac{3}{2}$ is the sharpest bound for (3.29). In other words, if there exists α such that

$$F(v_1, v_2, 0) \leq \alpha F(v_1, v_2, v_3), \quad \forall v_1, v_2, v_3 \Rightarrow \frac{3}{2} \leq \alpha.$$

Pf: Observe that $F(v_2, v_2, 0) = \frac{3}{2}F(v_2, v_2, -v_2), \quad v_2 \neq 0$. Then by this particular case of the assumption, we have

$$\frac{3}{2}F(v_2, v_2, -v_2) = F(v_2, v_2, 0) \leq \alpha F(v_2, v_2, -v_2).$$

By (3.29), if $\bar{v}$ agrees with v on one vertex, and the rest of the values are 0, then

$$v_1^2 + v_1^2 \leq \frac{3}{2}\left(v_1^2 + v_2^2 + v_3^2 + (v_1 + v_2 + v_3)^2\right).$$

□

Lemma 3.13 *There is a constant c depending only on the shape regularity of the triangles under consideration with*

$$D(v,w)|_\tau \leq c\left(\frac{1}{\sqrt{2}}\right)^{j-i} |v|_{H^1(\tau)}\ 2^j\|w\|_{L_2(\tau)}, \quad \forall \tau \in \tilde{\mathcal{T}}_i \quad \forall v \in \tilde{\mathcal{S}}_i, \forall w \in \tilde{\mathcal{S}}_j, j > i. \tag{3.30}$$

Proof. Case I: $j = i+1$, $v \in \mathcal{S}_i$, $w \in \mathcal{S}_j$. By Lemma 3.11

$$D(v,w)|_\tau \leq |v|_{H^1(\tau)}|w|_{H^1(\tau)} \leq |v|_{H^1(\tau)}c_0^{1/2}2^j\|w\|_{L_2(\tau)}.$$

Then $c = \sqrt{2c_0}$.

Case II: We decompose $w \in \mathcal{S}_j$ into two parts as in (3.24) $w = w_{bdry} + w_{intr}$ where w_{bdry} and w_{intr} live on the boundary and strip in the interior respectively. Let $w_0 \in \tilde{\mathcal{S}}_j$ be given as

$$w_{bdry}(x) = \begin{cases} w(x), & x \in \tilde{\mathcal{N}}_j \cap \partial\tau \\ 0, & x \in \tilde{\mathcal{N}}_j \backslash \partial\tau \end{cases}.$$

Let $w_{intr} = w - w_{bdry}$, then $D(v,w)|_\tau = D(v, w_{bdry})|_\tau + D(v, w_{intr})|_\tau$. Since $w_{intr}|_{\partial\tau} = 0$ and v is linear, integration by parts gives

$$D(v, w_{intr})|_\tau = 0.$$

Hence,

$$D(v,w)|_\tau = D(v,w_{bdry})|_\tau.$$

At level i, $\frac{x^{(j)}}{x^{(i)}} = \frac{y^{(j)}}{y^{(i)}} = \left(\frac{1}{2}\right)^{j-i}$. For the sake of simplicity, consider a right angled triangle $\tau \in \tilde{\mathcal{T}}_k$ with sides $x^{(i)}$ and $y^{(i)}$. The following relation holds for S and τ.

$$\begin{aligned}\frac{\text{area}(S)}{\text{area}(\tau)} &= \frac{x^{(i)}y^{(i)} - (x^{(i)} - 3x^{(j)})(y^{(i)} - 3y^{(j)})}{x^{(i)}y^{(i)}}\\ &= \frac{x^{(i)}y^{(i)} - x^{(i)}(1 - 3\frac{x^{(j)}}{x^{(i)}})y^{(i)}(1 - 3\frac{y^{(j)}}{y^{(i)}})}{x^{(i)}y^{(i)}}\\ &= 1 - \left(1 - 3\left(\frac{1}{2}\right)^{j-i}\right)^2 \le 6\left(\frac{1}{2}\right)^{j-i},\end{aligned}$$

where S is a boundary strip of τ out of which w_{bdry} vanishes. Then,

$$D(v,w_{bdry})|_\tau = D(v,w_{bdry})|_S.$$

By Cauchy-Schwarz,

$$D(v,w_{bdry})|_S \le |v|_{H^1(S)}|w_{bdry}|_{H^1(S)}.$$

By the area argument above and utilizing that the first derivatives of v are constant,

$$|v|_{H^1(S)}|w_{bdry}|_{H^1(S)} \le \sqrt{6}\left(\frac{1}{\sqrt{2}}\right)^{j-i}|v|_{H^1(\tau)}|w_{bdry}|_{H^1(S)}.$$

By the inverse inequality (3.27) and w_{bdry} vanishes outside of S,

$$|w_{bdry}|^2_{H^1(S)} \le c_0 4^j \|w_{bdry}\|^2_{L_2(S)} = c_0 4^j \|w_{bdry}\|^2_{L_2(\tau)}.$$

Finally by Lemma 3.12,

$$c_0 4^j \|w_{bdry}\|^2_{L_2(\tau)} \le \frac{3}{2} c_0 4^j \|w\|^2_{L_2(\tau)}.$$

Incorporating all inequalities, result holds with $c = 3\sqrt{c_0}$ □

Remark 3.1 The local estimate (3.30) implies the global estimate

$$\begin{aligned}D(v,w)^2 = \sum_{\tau\in\tilde{\mathcal{T}}_i} D(v,w)^2|_\tau &\le c\left(\frac{1}{\sqrt{2}}\right)^{2(j-i)} 2^{2j}\sum_{\tau\in\tilde{\mathcal{T}}_i}|v|^2_{H^1(\tau)}\ \|w\|^2_{L_2(\tau)}\\ &= c\left(\frac{1}{\sqrt{2}}\right)^{2(j-i)} 2^{2j}|v|^2_{H^1(\Omega)}\ \|w\|^2_{L_2(\Omega)}\end{aligned}$$

$$D(v,w) \le c\left(\frac{1}{\sqrt{2}}\right)^{(j-i)} |v|_{H^1(\Omega)}\ 2^j\|w\|_{L_2(\Omega)} \quad \forall v \in \tilde{\mathcal{S}}_i, \forall w \in \tilde{\mathcal{S}}_j, j > i \tag{3.31}$$

Consequently, the global estimate (3.31) holds for $v \in \mathcal{S}_i, w \in \mathcal{S}_j, j > i$, because $\mathcal{S}_i \subset \tilde{\mathcal{S}}_i, \quad \mathcal{S}_j \subset \tilde{\mathcal{S}}_j$.

Remark 3.2 If we generalize (3.31) to

$$a(v,w) \le c\left(\frac{1}{\sqrt{2}}\right)^{(j-i)} a(v,v)\ 2^j\|w\|_{L_2(\Omega)} \quad \forall v \in \tilde{\mathcal{S}}_i, \forall w \in \tilde{\mathcal{S}}_j, j > i,$$

then due to integration by parts, the coefficient matrix $p(x,y)$ has to be continuously differentiable. Then the only difference in the proof is that instead of $D(v, w_{intr})|_\tau = 0$ we have

$$\begin{aligned} a(v, w_{intr})|_\tau &\le c|v|_{H^1(\tau)}\ 2^j\|w_{intr}\|_{L_2(\tau)}, \\ &\le c\sqrt{3/2}|v|_{H^1(\tau)}\ 2^j\|w\|_{L_2(\tau)}. \end{aligned}$$

The difficulty in the proof of (a.ii) arises when the coefficient matrix is not C^1. Without assuming (a.ii), one can derive the following suboptimal estimate.

Lemma 3.14 *If (a.i) holds, then there exists c such that*

$$\sum_{j=1}^{J} 2^{-2j}\|A_{21}^{(j)}v^{(j-1)}\|_{L_2}^2 \ \le c\ J\ a(v,v), \quad \forall v \in\ \mathcal{S}_j. \tag{3.32}$$

Proof. For uniform refinement, see Lemma 4.4, page 116 in Vassilevski and Wang [36]. □

Our analysis takes an interesting turn by observing the role of H^1-stability of the L_2-projection in the proof of Lemma 3.14. In local refinement strategies, even for the sake of a suboptimal estimate, one has to prove plenty of results in order to keep the L_2-projection H^1-stable. This is where the literature that pertains to *mesh conditions* comes into the play [12, 15]. Such mesh conditions simply require that the element sizes do not change drastically between active regions of refinement and passive ones. In other words, the transition must be smooth between the highly refined and relatively less refined regions with respect to the element size. In this context, local quasiuniformity in a patch becomes crucial. A patch is usually the support of a basis function. Local quasiuniformity requires neighbor generation relations in the patch and shape regularity of the mesh.

We close this section by reporting the suboptimal norm equivalence results.

Theorem 3.15 *If (a.i) holds, then there exists a constant c independent of j such that*

$$(A^{(j)}v, v) \leq (B^{(j)}v, v) \leq c\ (1 + j)(A^{(j)}v, v) \quad \forall v \in \mathcal{S}_j,$$

for $j = 0, 1, \ldots, J$. The constant c depends only on b_1 from (3.14), δ from (a.ii), and σ from (a.i), and the H^1-norm of the L_2-projection Q_j for $j = 1, \ldots, J$.

Griebel and Oswald [20] improved the theorem above as follows.

Theorem 3.16 *If (a.i) holds, then there exists a constant c independent of j such that*

$$(A^{(j)}v, v) \leq (B^{(j)}v, v) \leq c\ (1 + \log_2(1 + j))(A^{(j)}v, v) \quad \forall v \in \mathcal{S}_j,$$

for $j = 0, 1, \ldots, J$. The constant c depends only on b_1 from (3.14), δ from (a.ii), and σ from (a.i), and the H^1-norm of the L_2-projection Q_j for $j = 1, \ldots, j$.

3.8.1 Approximate wavelet basis spectral estimates

π_k in (3.12) plays a crucial role in the stabilization process. L_2-projections present in the definition of π_k are responsible for the success of the stabilization. We have witnessed a similar success story as the BPX turned out to be a breakthrough over the HB methods. The WMHB exploits the best of both worlds. It not only utilizes the hierarchical structure but also has a stabilizing effect with the help of L_2-projections. Our motivation is to discuss the computational feasibility of these projections onto the coarse level FE space $\mathcal{S}_{j-1}$. For the sake of the computational feasibility, we are willing to accommodate the original operators by the approximate ones as long as the theoretical framework does not break down. Recall that the basic idea lies behind approximating the wavelets without deteriorating their stability, thereby creating a stable Riesz basis for the FE space under consideration.

Let $\mathbf{v}_2$ and $\mathbf{w}_2$ be the nodal coefficient vectors of $Q_{j-1}v$ and $w \in \mathcal{S}_{j-1}$ respectively. Then the definition of Q_{j-1}

$$(Q_{j-1}v, w) = (v, w), \quad \forall \in \mathcal{S}_{j-1},$$

can be represented in matrix-vector form as follows:

$$\mathbf{w}_2^T M^{(j-1)} \mathbf{v}_2 = (P_{j-1}^j \mathbf{w}_2)^T M^{(j)} \mathbf{v}, \quad \forall \mathbf{w}_2,$$

where P_{j-1}^j is the prolongation operator from level $j-1$ to j. Then we need to solve

$$M^{(j-1)} \mathbf{v}_2 = (P_{j-1}^j)^T M^{(j)} \mathbf{v},$$

implying that the exact representation of $Q_{j-1}v$ is given by

$$\left(M^{(j-1)}\right)^{-1} (P_{j-1}^j)^T M^{(j)} \mathbf{v}.$$

Inverting $M^{(j-1)}$ is costlier than solving the whole algebraic system. We replace Q_{j-1} by an approximate one Q_{j-1}^a, usually a polynomial approximation to preserve the sparsity. One can naturally use Jacobi or Gauss-Seidel approximations. Although, Gauss-Seidel is an effective approximation, we would like to emphasize that it literally destroys the sparsity. One can observe the sparsity complications in Figures 5.18 and 5.19. Instead of π_k in (3.12), we redefine it as

$$\pi_k = \prod_{j=k}^{J-1} (I_j + Q_j^a (I_{j+1} - I_j)).$$

To control the approximation quality of Q_{j-1}^a we always assume a small fixed tolerance c_Q satisfying

$$\|(Q_j^a - Q_j)v\|_{L_2} \leq c_Q \|Q_j v\|_{L_2}, \quad \forall v \in L_2(\Omega).$$

The general approach of Theorem 3.10 will be employed to yield the same type of spectral estimates for approximate wavelet basis. The key point here is again to verify Assumption 3.1, (3.16), and (3.17). Notice that $v^{(j)} \in \mathcal{S}_j$ admits the unique decomposition:

$$v^{(j)} = v^{(j-1)} + v^{(j)^f},$$

where

$$\begin{aligned} v^{(j-1)} &= Q^a_{j-1} v^{(j)} + (I - Q^a_{j-1}) I_{j-1} v^{(j)} \in \mathcal{S}_{j-1}, \\ v^{(j)^f} &= (I - Q^a_{j-1})(I_j - I_{j-1}) v^{(j)} \in \mathcal{S}^f_j \equiv (I - Q^a_{j-1}) \mathcal{S}^{(f)}_j. \end{aligned}$$

Let $e_j = v^{(j)} - Q_j v$ be the deviation of $v^{(j)}$ from $Q_j v$. Since Q^a_j is an approximation of Q_j, it is reasonable to believe that the deviation e_j can be neglected in the argument. The following lemma provides a rigorous estimate on this perturbation.

Lemma 3.17

$$e_{j-1} = [Q_{j-1} + R_{j-1}]\, e_j + R_{j-1}(Q_j - Q_{j-1}) v, \tag{3.33}$$

where $R_{j-1} = (Q_{j-1} - Q^a_{j-1})(I_{j-1} - I_j)$.

The above perturbation estimate enables us to characterize $\|e_{j-1}\|_{L_2}$ in terms of $\|(Q_j - Q_{j-1})v\|_{L_2}$ as follows.

$$\sum_{j=1}^{J} 2^{2(j-1)} \|e_{j-1}\|^2_{L_2} \le c(c_Q) \sum_{j=1}^{J} 2^{2j} \|(Q_j - Q_{j-1})v\|^2_{L_2}. \tag{3.34}$$

(3.34) will play a central role in the spectral analysis. In particular, the following two lemmas are proved using (3.34). We carry on with the spectral analysis to verify (3.16) and (3.17). Since the treatment is quite similar to the case covered for the exact L_2-projection, we skip the details and merely report the results.

Lemma 3.18 *If (a.i) holds and c_Q is sufficiently small (but fixed), then (3.16) holds for some constant ρ_1.*

Proof. For the quasiuniform setting, see Lemma 5.2, page 120 in [36]. □

We point out that Lemma 3.6 is valid for the operator $A_{22}^{(j)}$ obtained from the approximate wavelet basis. Again, it is preferable to choose a diagonal preconditioner $B_{22}^{(j)} = \alpha 2^{2j} I$, where α is adjusted to satisfy (3.14).

Lemma 3.19 *If (a.i) and (a.ii) hold, c_Q is sufficiently small (but fixed), then (3.17) holds for some constant ρ_2.*

Proof. The first sum is estimated by using (3.34) and (a.i), and the second sum is estimated by Lemma 3.6. For the quasiuniform setting, see Lemma 5.3, page 120 in [36]. □

Theorem 3.20 *Let $B^{(j)}$ be the approximate wavelet preconditioner constructed as in Definition 3.5.*

- *If (a.i), (a.ii), and c_Q is sufficiently small, then $B^{(j)}$ is spectrally equivalent to $A^{(j)}$.*
- *Without assuming the inequality (a.ii), $B^{(j)}$ is suboptimally spectral equivalent to $A^{(j)}$ as in Theorems 3.15, 3.16 with sufficiently small c_Q.*

3.9 Overview of the discussed preconditioners

We close the preconditioner chapters by giving an overview of the preconditioners discussed. First, we highlight the space decomposition corresponding to each method. We also give some matrix formulation of the preconditioners, reported in Vassilevski and Wang [38].

$$\mathcal{S} = \mathcal{S}_J = \mathcal{S}_0 \oplus \mathcal{S}_1^f \oplus \cdots \oplus \mathcal{S}_J^f.$$

$$\begin{array}{llcl}
\text{HB:} & \mathcal{S}_j^f & = & (I_j - I_{j-1})\mathcal{S}, \\
\text{BPX:} & \mathcal{S}_j^f & = & (Q_j - Q_{j-1})\mathcal{S}, \\
\text{WMHB:} & \mathcal{S}_j^f & = & (I - Q_{j-1}^a)(I_j - I_{j-1})\mathcal{S}.
\end{array}$$

- The HBMG method:

$$\hat{A}^{(j)} = \begin{bmatrix} \hat{A}_{11}^{(j)} & \hat{A}_{12}^{(j)} \\ \hat{A}_{21}^{(j)} & \hat{A}_{22}^{(j)} \end{bmatrix}$$

$$\hat{B}^{(j)} = \begin{bmatrix} I & 0 \\ \hat{A}_{21}^{(j)} & D_{22}^{(j)} - L_{22}^{(j)} \end{bmatrix} \begin{bmatrix} B^{(j-1)} & 0 \\ 0 & D_{22}^{(j)^{-1}} \end{bmatrix} \begin{bmatrix} I & \hat{A}_{12}^{(j)} \\ 0 & D_{22}^{(j)} - U_{22}^{(j)} \end{bmatrix}$$

$$Y_1^{(j)} = \begin{bmatrix} I \\ R^{(j)} \end{bmatrix}, \qquad Y_2^{(j)} = \begin{bmatrix} 0 \\ I \end{bmatrix}$$

$A_{22}^{(j)} = D_{22}^{(j)} - L_{22}^{(j)} - U_{22}^{(j)}$.

- The WMHB method:

$$\hat{B}^{(j)} = \begin{bmatrix} B^{(j-1)} & \hat{A}_{12}^{(j)} \\ 0 & \hat{B}_{22}^{(j)} \end{bmatrix} \begin{bmatrix} I & 0 \\ \left(\hat{B}_{22}^{(j)}\right)^{-1} \hat{A}_{21}^{(j)} & I \end{bmatrix}$$

$$Y_1^{(j)} = P_{j-1}^{j} = \begin{bmatrix} I \\ R^{(j)} \end{bmatrix}, \qquad Y_2^{(j)} = \left(I - P_{j-1}^{j} \tilde{M}_{inv}^{(j-1)} {P_{j-1}^{j}}^{T} M^{(j)}\right) \begin{bmatrix} 0 \\ I \end{bmatrix}$$

$\hat{B}_{22}^{(j)} \equiv B_{22}^{(j)} = (D_{22}^{(j)} - L_{22}^{(j)}) D_{22}^{(j)^{-1}} (D_{22}^{(j)} - U_{22}^{(j)})$.

- A preconditioner spectrally equivalent to the MG method:

$$\hat{B}^{(j)} = \begin{bmatrix} I & 0 \\ Y_2^{(j)^T} A^{(j)} & D^{(j)} - L^{(j)} \end{bmatrix} \begin{bmatrix} B^{(j-1)} & 0 \\ 0 & D^{(j)^{-1}} \end{bmatrix} \begin{bmatrix} I & A^{(j)} Y_2^{(j)} \\ 0 & D^{(j)} - U^{(j)} \end{bmatrix}$$

$$Y_1^{(j)} = P_{j-1}^{j}, \qquad Y_2^{(j)} = I$$

$$A^{(j)} = D^{(j)} - L^{(j)} - U^{(j)}.$$

- The BPX method:

  The additive version of the above MG-like method is commonly referred to as the BPX preconditioner. The additive form is obtained by a simple deletion of the $\hat{A}_{12}^{(j)}$ and $\hat{A}_{21}^{(j)}$ blocks, or $Y_2^{(j)^T} A^{(j)}$ and $A^{(j)} Y_2^{(j)}$.

The inverse of the preconditioner is given by

$$(B^{(j)})^{-1} = \begin{bmatrix} Y_1^{(j)} & Y_2^{(j)} \end{bmatrix} (\hat{B}^{(j)})^{-1} \begin{bmatrix} Y_1^{(j)} & Y_2^{(j)} \end{bmatrix}^T,$$

which does not require the inversion of $\begin{bmatrix} Y_1^{(j)} & Y_2^{(j)} \end{bmatrix}$. We direct the reader to sections 5.1, 5.2 for explicit derivation of the matrices.

Chapter 4

Local Quasiuniformity

The efficiency of a preconditioner is dictated by the stability of the splitting of the FE space $\mathcal{S}_j$, or equivalently stability of the Riesz basis $\Phi^{(j)}$ of $\mathcal{S}_j$. The splitting of $\mathcal{S}_j$ is done by operators, for instance: Q_j, the L_2-projection; I_j, the nodal interpolant as in the case of BPX and HB preconditioners respectively. Naturally, stability of the operators affects the stability of the $\Phi^{(j)}$. Since the working ground is H^1, then the stability of an operator is measured with respect to the H^1-norm. As indicated in Theorem 3.3, H^1-stability of the operator becomes a necessary condition for the stability of the Riesz basis.

Actually, we have seen a practical usage of Theorem 3.3, by detecting the reason why the HB preconditioner cannot be optimal. The reason simply is that I_j is not H^1-stable, i.e. the H^1-norm of I_j is not bounded uniformly with respect to the difference $J - j \to \infty$ or the ratio of the mesh size $\frac{\text{size}(\tau^{(j)})}{\text{size}(\tau^{(J)})}$.

H^1-stability of the L_2-projection is especially vital in our uniform refinement context. Firstly, without the presence of the assumption $(a.ii)$, we are only able to derive a suboptimal lower bound in the norm equivalence estimate for the WMHB preconditioner, by the help of H^1-stability of the L_2-projection (see Lemma 3.14, Theorem 3.15).

Secondly, a similar scenario holds for the BPX lower bound. Using the K-functionals and H^1-stability of the L_2-projection, we exhibited the canonical suboptimal lower bound in Theorem 2.13

$$c\frac{1}{J}\|u\|_{BPX}^2 \leq \|u\|_{H^1}^2.$$

Interestingly, if we ask for a stronger H^s-stability, $s > 1$, then the lower bound can be upgraded to a constant which is also shown in Theorem 1.13 for uniform refinement. There is striking irony in the fact that $s > 1$ gives rise to the optimal lower bound not only in the Jackson estimate for H^s, but also in H^s-stability of the L_2-projection setting for the BPX preconditioner. We demand a Jackson estimate in H^s after realizing that $\mathcal{S}_j$ is smoother than one would usually think. This observation improves the lower bound to an optimal level. Likewise, H^s-stability of the L_2-projection is the key for an optimal lower bound. We may be able to speculate on the hidden relation between the Jackson estimate and H^s-stability of the L_2-projection by means that they are associated to the optimal lower bound argument. However, to the author's knowledge, there seems to be no literature on H^s-stability of the L_2-projection for $s > 1$. This may well be a potential subject for further research.

Whatever is stated about the lower bounds in uniform refinement can be extended to local refinement because H^1-stability of the L_2-projection can even be established on locally refined meshes. Two separate works exist in the literature. The early work by Bramble, Pasciak, and Steinbach [12] (BPS-criterion) is fundamental in the sense that explicit matrix and eigenvalue analysis is presented. The later work by Carstensen [15] (C-criterion) is more comprehensive that it involves general and stronger results. They both formulate mesh conditions for local refinement that guarantee the H^s-stability of the L_2-projection. In spirit, both analyses call for local quasiuniformity of the mesh, and stability is established as long as neighboring elements do not change drastically in size. Local quasiuniformity is generally enforced patchwise, patch usually being the support of a basis function. Patchwise quasiuniformity was utilized to show that the appropriately scaled basis functions have a constant L_2-norm (2.45).

In the first section, we will formulate the tools needed for patchwise quasiuniformity. We will report two of the author's results. These prove patchwise quasiuniformity for the well known refinement routines: 2D newest vertex bisection due to Mitchell [26] and 3D marked tetrahedron bisection due to Joe and Liu [24].

4.1 Patchwise quasiuniformity

We start with the definition of patchwise quasiuniformity.

Definition 4.1 *A patch P consists of elements*

$$P = \bigcup_{i \in I} \tau_i,$$

where I is the index set that contains the elements in the support of a basis function. P is called quasiuniform if there exists a constant c such that

$$\frac{\max_{i\in I} \operatorname{size}(\tau_i)}{\min_{i\in I} \operatorname{size}(\tau_i)} \leq c.$$

Here, we analyze 3D patchwise quasiuniformity. Because the 2D one is a simple reduction of the 3D analysis. The main ingredient is a generation bound for the simplices that share a face (3 vertices).

Assumption 4.1 *Let τ and τ' in patch P share a common face. Then exists F such that*

$$|L(\tau) - L(\tau')| \leq F. \tag{4.1}$$

By shape regularity, for any $\tau, \tau' \in P$ there exists G such that $\#\{\tau :\ \tau \in P\} \leq G$. Therefore, combining (4.1) we get,

$$|L(\tau) - L(\tau')| \leq FG, \quad \forall \tau, \tau', \in P. \tag{4.2}$$

The above generation bound is also handled in Lemmas 2.15 and 2.16 by the constants E and V both depending on shape regularity. We always assume that the coarsest mesh $\mathcal{T}_0$ is quasiuniform.

Assumption 4.2 *Quasiuniformity: There exists an absolute constant* $\gamma^{(0)}$ *such that*

$$\frac{\text{size}(\tau_{\max}^{(0)})}{\text{size}(\tau_{\min}^{(0)})} \leq \gamma^{(0)}, \tag{4.3}$$

where $\text{size}(\tau_{\max}^{(0)})$ *denotes the element with the largest size in* $\mathcal{T}_0$*. One can use an equivalent definition*

$$\frac{\text{size}(\tau_i^{(0)})}{\text{size}(\tau_j^{(0)})} \leq \gamma^{(0)}, \quad \forall \tau_i^{(0)}, \tau_j^{(0)} \in \mathcal{T}_0. \tag{4.4}$$

Minimal assumption regarding the size is as follows:

Assumption 4.3 *For* $l \geq k$*, there exists* c *such that*

$$c\ (\frac{1}{\delta})^{l-k} \text{size}(\underline{\tau}^{(k)}) \leq \text{size}(\tau_{\min}^{(l)}), \quad k = l, \ldots, \underline{k},$$

where $\underline{k} = \min\{L(\tau) : \tau \in P\}$.

A patch typically consists of elements from various levels because of the local refinement. Usually, relatively big elements in a patch are from coarse levels, whereas small ones tend to come from finer levels. One of the two following scenarios will occur in a patch:

Case I: $l \geq k,\ \delta > 1$

$$\begin{aligned}\frac{\text{size}(\tau_{\max}^{(k)})}{\text{size}(\tau_{\min}^{(l)})} &\leq c \frac{\text{size}(\tau_{\max}^{(k)})}{\left(\frac{1}{\delta}\right)^{l-k} \text{size}(\underline{\tau}^{(k)})} \\ &= \delta^{(l-k)} \frac{\text{size}(\tau_{\max}^{(k)})}{\text{size}(\underline{\tau}^{(k)})}.\end{aligned}$$

Case II: $l \leq k$

$$\frac{\text{size}(\tau_{\max}^{(k)})}{\text{size}(\tau_{\min}^{(l)})} \leq \frac{\text{size}(\overline{\tau}^{(l)})}{\text{size}(\tau_{\min}^{(l)})}.$$

$\overline{\tau}^{(0)}, \underline{\tau}^{(0)}$ denote the level 0 fathers of $\tau_{\max}^{(m)}, \tau_{\min}^{(m)}$ respectively. We can express the two cases as one case by introducing $m = \min(l, k)$, and using (4.2)

$$\begin{aligned}\frac{\text{size}(\tau_{\max}^{(k)})}{\text{size}(\tau_{\min}^{(l)})} &\leq c\delta^{|l-k|} \frac{\text{size}(\overline{\tau}^{(m)})}{\text{size}(\underline{\tau}^{(m)})} \\ &\leq c\delta^{FG} \frac{\text{size}(\overline{\tau}^{(m)})}{\text{size}(\underline{\tau}^{(m)})}.\end{aligned}$$

Furthermore, if we assume quasiuniformity for level $= m$

Assumption 4.4 *There exists an absolute constant $\gamma^{(m)}$ such that*

$$\frac{\text{size}(\tau_{\max}^{(m)})}{\text{size}(\tau_{\min}^{(m)})} \leq \gamma^{(m)}, \tag{4.5}$$

then we will establish the patchwise quasiuniformity

$$\frac{\text{size}(\tau_{\max}^{(k)})}{\text{size}(\tau_{\min}^{(l)})} \leq c\delta^{FG}\gamma^{(m)}. \tag{4.6}$$

If we assume a size equivalence between the coarsest father and the son

$$c_1\delta^m \text{size}(\overline{\tau}^{(0)}) \leq \text{size}(\tau^{(m)}) \leq c_2\delta^m \text{size}(\overline{\tau}^{(0)}), \tag{4.7}$$

then

$$\frac{\text{size}(\tau_{\max}^{(m)})}{\text{size}(\tau_{\min}^{(m)})} \leq c\,\frac{\text{size}(\overline{\tau}^{(0)})}{\text{size}(\underline{\tau}^{(0)})} \leq c\,\gamma^{(0)},$$

level $= m$ quasiuniformity is guaranteed. Moreover Assumption 4.3 is automatically satisfied.

4.2 3D marked tetrahedron bisection routine

We establish patchwise quasiuniformity for the marked tetrahedron routine [24]. We will cite two important theorems in order to construct the the generation bound (4.2). We should remark that the generation bound for face-adjacent simplices as in Theorem 4.1 suffices to obtain (4.2). This bisection routine provides strong quantitative relations such as generation bounds for adjacent simplices (Theorems 4.1, 4.2), size measures (4.8), (4.11), and shape measure between son and father (4.9). These relations make the 3D marked tetrahedron bisection routine shine among the rest of the routines. Even an amazing generation bound for edge-adjacent simplices holds as in Theorem 4.2.

Theorem 4.1 *For any interior face f in $\mathcal{T}_j$, if $\tau \cap \tau' = f$, then*

$$|L(\tau) - L(\tau')| \leq 2,$$

and the upper bound is tight.

Proof. See Theorem 4.3, page 1279 in [24]. □

Theorem 4.2 *For any edge e in $\mathcal{T}_j$, if $\tau \cap \tau' = e$, then*

$$|L(\tau) - L(\tau')| \leq 4,$$

and the upper bound is tight.

Proof. See Theorem 4.4, page 1280 in [24]. □

By construction, an immediate father-son relation is at our disposal

$$\mathrm{size}(\tau_i^{(j)}) \leq c_2(1/2)^{j/3}\mathrm{size}(\tau^{(0)}). \tag{4.8}$$

Definition 4.2 *$\eta(\tau)$ denotes the shape measure of a tetrahedron τ, defined by*

$$\eta(\tau) = \frac{12(3v)^{2/3}}{\sum_{i=1}^{6} l_i^2},$$

where v is the volume of τ and l_i are the lengths of the edges of τ. As η gets closer to 1, the shape quality increases.

The next theorem indicates that the shape of the refined tetrahedra does not decrease arbitrarily as the refinement proceeds.

Theorem 4.3 *For any tetrahedron $\tau^{(j)} \in \mathcal{T}_j$ we have*

$$\eta(\tau^{(j)}) \geq c\eta(\tau^{(0)}), \tag{4.9}$$

where c is independent of $\tau^{(0)}$.

Proof. See Theorem 5, page 150 in [23]. □

After j levels of bisection, the shape measure of a subtetrahedron is given as

$$\eta(\tau^{(j)}) = \frac{12\ (1/2)^{2j/3}\ (3v)^{2/3}}{\sum_{i=1}^{6} l_{i,j}^2}. \tag{4.10}$$

We define the size of a simplex to be the length of the largest edge:

$$\text{size}(\tau^{(j)}) = l_{\max,j}.$$

Let $\eta_{\min}$ denote the poorest shape measure in $\mathcal{T}_0$. Letting $c_{\min} = 1/\eta_{\min}(\tau^{(0)})$, then $c_{\min} \geq 1/\eta(\tau^{(0)})$. This yields $c_{\min}\eta(\tau^{(0)}) \geq 1 \geq \eta(\tau^{(j)})$, shortly $c_{\min}\ \eta(\tau^{(0)}) \geq \eta(\tau^{(j)})$. Using the Definition 4.2, (4.9), and (4.10), we get

$$c_{\min} \sum_{i=1}^{6} l_{i,j}^2 \geq (1/2)^{2j/3} \sum_{i=1}^{6} l_{i,0}^2,$$

$$6\ c_{\min}\ \text{size}^2(\tau^{(j)}) \geq c_{\min} \sum_{i=1}^{6} l_{i,j}^2 \geq (1/2)^{2j/3} \sum_{i=1}^{6} l_{i,0}^2 \geq (1/2)^{2j/3}\text{size}^2(\tau^{(0)}).$$

Finally, we can conclude that there exist c_1 depending only on the initial mesh such that

$$\text{size}(\tau^{(j)}) \geq (1/2)^{j/3}\text{size}(\tau^{(0)})c_1. \tag{4.11}$$

Putting (4.8) and (4.11) together, we have the size equivalence:

$$c_1(1/2)^{j/3}\ \text{size}(\tau^{(0)}) \leq \text{size}(\tau^{(j)}) \leq c_2(1/2)^{j/3}\ \text{size}(\tau^{(0)}).$$

Having the size equivalence in place, and using the generation bounds in Theorems 4.1, 4.2 we are ready to prove patchwise quasiuniformity.

$$\begin{aligned}
\frac{\text{size}(\tau_{\max}^{(k)})}{\text{size}(\tau_{\min}^{(l)})} &\leq \frac{c_2(1/2)^{k/3}\text{size}(\overline{\tau}^{(0)})}{c_1(1/2)^{l/3}\text{size}(\underline{\tau}^{(0)})} \\
&= \frac{c_2}{c_1} 2^{|k-l|/3} \frac{\text{size}(\overline{\tau}^{(0)})}{\text{size}(\underline{\tau}^{(0)})} \\
&\leq \frac{c_2}{c_1} 2^{FG} \gamma^{(0)}.
\end{aligned}$$

4.3 Introduction to the 2D newest vertex bisection

We are going to establish patchwise quasiuniformity for the 2D newest vertex bisection routine which is a central topic in Mitchell's dissertation [26]. In bisection subdivision, a triangle is divided to form two new triangles by connecting one of the vertices, called the *peak*, to the midpoint of the opposite side called the *children*. The children are said to have *generation* $n+1$ where n is the generation of the parent. The initial triangle is assigned generation 0. The new vertex created at the midpoint of the base is assigned to be the peak of the children.

It is important that the angles are bounded away from 0 and π. Notice that by assigning the peak this way, no angle is divided more than once. Intuitively, this should prevent the angles from getting too small. Indeed, it is shown by Sewell [35] that there are only four similarity classes of triangles created by this method, and hence the angles are bounded.

The triangle of type 1 gives rise to triangles of type 2 and 3. Both triangles of type 2 and 3 give rise to triangle of type 1 and 4. Type 4 gives rise to triangles of type 2 and 3. Therefore in a uniform refinement case; after the coarsest level 0, at level 1 (in general $level = 2n-1,\ n \geq 1$) we only have triangles of type 2 and 3. At level 2 (in general $level = 2n,\ n \geq 1$), we only have triangles of type 1 and 4. We can conclude an equivalent statement about the types of triangles of the same generation $2n$ or $2n-1, \quad n \geq 1$, since same generation triangles are the subset of the same level uniformly refined triangles.

Since no angle is divided more than once, the mesh after any number of refinement is going to be shape regular. In fact, for shape regular meshes, it can be shown that the diameter of the circumscribing circle is equivalent to the longest edge of the triangle. For convenience, we will use the longest edge of a triangle as its size.

4.4 Edges and sizes of triangles after bisection

Definition 4.3 *Let τ have edges l_1, l_2, and l_3.*

$$\text{size}(\tau) = \max\{l_1, l_2, l_3\}. \tag{4.12}$$

Let k be the edge in the child triangle bisecting the edge l_3. Then by parallelogram law,

$$k = \frac{(2l_1^2 + 2l_2^2 - l_3^2)^{1/2}}{2}.$$

We will also use $K^2(l_1, l_2, l_3) \equiv 2l_1^2 + 2l_2^2 - l_3^2$, where $K = 2k$. Also parallelogram law indicates

$$K^2 + l_3^2 = 2l_1^2 + 2l_2^2.$$

In the following section, for $m = 2n,\ n \geq 1$, we establish equivalence between

- (A): $\text{size}(\tau_2^{(m-1)}) = \max\left\{l_1, \frac{l_3}{2}, \frac{K}{2}\right\}$,
- (B): $\text{size}(\tau_3^{(m-1)}) = \max\left\{l_2, \frac{l_3}{2}, \frac{K}{2}\right\}$,
- (C): $\text{size}(\tau_4^{(m)}) = \max\left\{\frac{l_1}{2}, \frac{l_2}{2}, \frac{K}{2}\right\}$,

and

- $\text{size}(\tau_1^{(m-2)}) = \max\{l_1, l_2, l_3\}$.

The following fact is useful for our analysis

$$\max\{a, b\} = \left(\max\left\{a^2, b^2\right\}\right)^{1/2}, \quad a, b \geq 0.$$

We would rather establish the relationship, say between $\text{size}(\tau_4^{(m)})$ and $\text{size}(\tau_1^{(m-2)})$, by using K^2. Hence,

$$\text{size}(\tau_4^{(m)}) = \max\left\{\frac{l_1}{2}, \frac{l_2}{2}, \frac{K}{2}\right\} = \frac{1}{2}\max\{l_1, l_2, K\} = \frac{1}{2}\left(\max\left\{l_1^2, l_2^2, K^2\right\}\right)^{1/2}.$$

So, we have

$$\max\left\{l_1^2, l_2^2, K^2\right\} = 4\ \text{size}(\tau_4^{(m)})^2. \tag{4.13}$$

If there exist two positive, absolute constants c_1 and c_2 such that

$$c_1 \max\left\{l_1^2, l_2^2, K^2\right\} \leq \max\left\{l_1^2, l_2^2, l_3^2\right\} \leq c_2 \max\left\{l_1^2, l_2^2, K^2\right\}, \tag{4.14}$$

then by using (4.13),

$$4\ c_1 \mathrm{size}(\tau_4^{(m)})^2 \leq \mathrm{size}(\tau_1^{(m-2)})^2 \leq 4\ c_2 \mathrm{size}(\tau_4^{(m)})^2.$$

Finally, we get

$$2\ \sqrt{c_1}\ \mathrm{size}(\tau_4^{(m)}) \leq \mathrm{size}(\tau_1^{(m-2)}) \leq 2\ \sqrt{c_2}\ \mathrm{size}(\tau_4^{(m)}). \tag{4.15}$$

4.5 Technical results

4.5.1 Equivalence between $\mathrm{size}(\tau_4^{(m)})$ and $\mathrm{size}(\tau_1^{(m-2)})$

Lemma 4.4

$$\mathrm{size}(\tau_4^{(m)}) \leq \mathrm{size}(\tau_1^{(m-2)}) \leq 4\mathrm{size}(\tau_4^{(m)}).$$

Proof. Case 1: $l_1 \geq l_2$, $l_1 \geq l_3$ or $l_1^2 = \max\{l_1^2, l_2^2, l_3^2\}$

1. $l_1^2 \leq \max\{l_1^2, l_2^2, K^2\}$.

2.
   - $K^2 = 2l_1^2 + 2l_2^2 - l_3^2 \leq 2l_1^2 + 2l_2^2 \leq 4l_1^2$,
   - $l_1^2 \leq 4l_1^2$,
   - $l_2^2 \leq 4l_2^2 \leq 4l_1^2$;

   $\max\{l_1^2, l_2^2, K^2\} \leq 4l_1^2$.

$$\frac{1}{4}\max\left\{l_1^2, l_2^2, K^2\right\} \leq l_1^2 \leq \max\left\{l_1^2, l_2^2, K^2\right\}.$$

Case 2: $l_2 \geq l_1$, $l_2 \geq l_3$ or $l_2^2 = \max\{l_1^2, l_2^2, l_3^2\}$.
Symmetry of l_1 and l_2 in the expression of K, that is $K(l_1, l_2, l_3) = K(l_2, l_1, l_3)$, the conclusion above follows. However, we do the proof for clarity.

1. $l_2^2 \leq \max\{l_1^2, l_2^2, K^2\}$.

2. - $K^2 = 2l_1^2 + 2l_2^2 - l_3^2 \leq 2l_1^2 + 2l_2^2 \leq 4l_2^2$,
   - $l_1^2 \leq 4l_1^2 \leq 4l_2^2$,
   - $l_2^2 \leq 4l_2^2$;

   $\max\{l_1^2, l_2^2, K^2\} \leq 4l_2^2$.

$$\frac{1}{4}\max\left\{l_1^2, l_2^2, K^2\right\} \leq l_2^2 \leq \max\left\{l_1^2, l_2^2, K^2\right\}.$$

Case 3: $l_3 \geq l_1$, $l_3 \geq l_2$ or $l_3^2 = \max\{l_1^2, l_2^2, l_3^2\}$

1. - $K^2 = 2l_1^2 + 2l_2^2 - l_3^2 \leq 3l_3^2$,
   - $l_1^2 \leq l_3^2 \leq 3l_3^2$,
   - $l_2^2 \leq 3l_2^2 \leq 3l_3^2$;

   $\max\{l_1^2, l_2^2, K^2\} \leq 3l_3^2$.

2. Case 3a: $l_1 \geq l_2$ i.e. $l_3 \geq l_1 \geq l_2$.
   Then, $\max\{l_1^2, l_2^2, K^2\} = \max\{l_1^2, K^2\}$ and $l_1 > l_3/2$.
   Possibility 1: $l_1^2 \leq K^2$ or $\max\{l_1^2, K^2\} = K^2$

   - $K^2 = \max\{l_1^2, K^2\} \geq l_1^2 > l_3^2/4$,
   - $K^2 = 2l_1^2 + 2l_2^2 - l_3^2 \leq 2l_3 + 2l_3^2 - l_3^2 = 3l_3^2$;

   $1/3K^2 < l_3^2 < 4K^2$.
   Possibility 2: $K^2 \leq l_1^2$ or $\max\{l_1^2, K^2\} = l_1^2$

   - $l_1^2 > l_3^2/4$,
   - $l_1^2 \leq l_3^2$;

   $l_1^2 \leq l_3^2 \leq 4l_1^2$.

$$1/3\max\left\{l_1^2, l_2^2, K^2\right\} \leq l_3^2 \leq 4\max\left\{l_1^2, l_2^2, K^2\right\}.$$

The symmetry argument holds again. For the sake of clarity, we do the proof in detail.

3. Case 3b: $l_2 \geq l_1$ i.e. $l_3 \geq l_2 \geq l_1$.
   Then, $\max\{l_1^2, l_2^2, K^2\} = \max\{l_2^2, K^2\}$ and $l_2 > l_3/2$.
   Possibility 1: $l_2^2 \leq K^2$ or $\max\{l_2^2, K^2\} = K^2$

   - $K^2 = \max\{l_2^2, K^2\} \geq l_2^2 > l_3^2/4$,
   - $K^2 = 2l_1^2 + 2l_2^2 - l_3^2 \leq 2l_3 + 2l_3^2 - l_3^2 = 3l_3^2$;

   $1/3K^2 < l_3^2 < 4K^2$.
   Possibility 2: $K^2 \leq l_2^2$ or $\max\{l_2^2, K^2\} = l_2^2$

   - $l_2^2 > l_3^2/4$,
   - $l_2^2 \leq l_3^2$;

   $l_2^2 \leq l_3^2 \leq 4l_2^2$,

$$1/3 \max\left\{l_1^2, l_2^2, K^2\right\} \leq l_3^2 \leq 4 \max\left\{l_1^2, l_2^2, K^2\right\}.$$

Putting all results together;

$$1/4 \max\left\{l_1^2, l_2^2, K^2\right\} \leq \max\left\{l_1^2, l_2^2, l_3^2\right\} \leq 4 \max\left\{l_1^2, l_2^2, K^2\right\}.$$

Lemma follows by using (4.15). $\square$

4.5.2 Equivalence between $\text{size}(\tau_2^{(m-1)})$ and $\text{size}(\tau_1^{(m-2)})$

Lemma 4.5

$$\text{size}(\tau_2^{(m)}) \leq \text{size}(\tau_1^{(m-2)}) \leq 2\text{size}(\tau_2^{(m)}).$$

Proof. Case 1: $l_1 \geq l_2$, $l_1 \geq l_3$ or $l_1^2 = \max\{l_1^2, l_2^2, l_3^2\}$

- $4l_1^2 \geq 2l_1^2 + 2l_2^2 \geq 2l_1^2 + 2l_2^2 - l_3^2 = K^2$,
- $4l_1^2 \geq l_1^2 \geq l_3^2$,

$\max\{4l_1^2, l_3^2, K^2\} = 4l_1^2$;

$$\frac{1}{4}\max\left\{4l_1^2, l_3^2, K^2\right\} \le l_1^2 \le \frac{1}{4}\max\left\{4l_1^2, l_3^2, K^2\right\}.$$

Case 2: $l_2 \ge l_1$, $l_2 \ge l_3$ or $l_2^2 = \max\{l_1^2, l_2^2, l_3^2\}$

1. $l_2^2 \le l_2^2 + (l_2^2 - l_3^2) \le 2l_2^2 - l_3^2 + 2l_1^2 = K^2$. Then $l_2^2 \le K^2 \le \max\{4l_1^2, l_3^2, K^2\}$.

2.
   - $K^2 = 2l_1^2 + 2l_2^2 - l_3^2 \le 2l_1^2 + 2l_2^2 \le 4l_2^2$,
   - $4l_1^2 \le 4l_2^2$,
   - $l_3^2 \le l_2^2 \le 4l_2^2$;

   $\max\{4l_1^2, l_3^2, K^2\} \le 4l_2^2$.

$$\frac{1}{4}\max\left\{4l_1^2, l_3^2, K^2\right\} \le l_2^2 \le \max\left\{4l_1^2, l_3^2, K^2\right\}.$$

Case 3: $l_3 \ge l_1$, $l_3 \ge l_2$ or $l_3^2 = \max\{l_1^2, l_2^2, l_3^2\}$

1. $l_3^2 \le \max\{4l_2^2, l_3^2, K^2\}$.

2.
   - $K^2 = 2l_1^2 + 2l_2^2 - l_3^2 \le 2l_3^2 + 2l_3^2 - l_3^2 = 3l_3^2 \le 4l_3^2$,
   - $4l_1^2 \le 4l_3^2$,
   - $l_3^2 \le 4l_3^2$;

   $\max\{4l_1^2, l_3^2, K^2\} \le 4l_3^2$.

$$\frac{1}{4}\max\left\{4l_1^2, l_3^2, K^2\right\} \le l_3^2 \le \max\left\{4l_1^2, l_3^2, K^2\right\}.$$

Putting all results together:

$$\frac{1}{4}\max\left\{4l_1^2, l_3^2, K^2\right\} \le \max\left\{l_1^2, l_2^2, l_3^2\right\} \le \max\left\{4l_1^2, l_3^2, K^2\right\}.$$

Lemma follows by using (4.15). $\square$

4.5.3 Equivalence between size$(\tau_3^{(m-1)})$ and size$(\tau_1^{(m-2)})$

Lemma 4.6

$$\text{size}(\tau_3^{(m)}) \leq \text{size}(\tau_1^{(m-2)}) \leq 2\text{size}(\tau_3^{(m)}).$$

Proof. Case 1: $l_1 \geq l_2$, $l_1 \geq l_3$ or $l_1^2 = \max\{l_1^2, l_2^2, l_3^2\}$

1. $l_1^2 \leq l_1^2 + (l_1^2 - l_3^2) \leq 2l_1^2 - l_3^2 + 2l_2^2 = K^2$. Then $l_1^2 \leq K^2 \leq \max\{4l_2^2, l_3^2, K^2\}$.

2.
   - $K^2 = 2l_1^2 + 2l_2^2 - l_3^2 \leq 2l_1^2 + 2l_2^2 \leq 4l_1^2$,
   - $4l_2^2 \leq 4l_1^2$,
   - $l_3^2 \leq 4l_3^2 \leq 4l_1^2$;

   $\max\{4l_2^2, l_3^2, K^2\} \leq 4l_1^2$.

$$\frac{1}{4}\max\left\{4l_2^2, l_3^2, K^2\right\} \leq l_1^2 \leq \max\left\{4l_2^2, l_3^2, K^2\right\}.$$

Case 2: $l_2 \geq l_1$, $l_2 \geq l_2$ or $l_2^2 = \max\{l_1^2, l_2^2, l_3^2\}$

- $K^2 = 2l_1^2 + 2l_2^2 - l_3^2 \leq 2l_1^2 + 2l_2^2 = 4l_2^2$,
- $4l_2^2 \leq 4l_2^2$,
- $l_3^2 \leq 4l_3^2 \leq 4l_2^2$;

$\max\{4l_2^2, l_3^2, K^2\} = 4l_2^2$.

$$\frac{1}{4}\max\left\{4l_2^2, l_3^2, K^2\right\} \leq l_2^2 \leq \frac{1}{4}\max\left\{4l_2^2, l_3^2, K^2\right\}.$$

Case 3: $l_3 \geq l_1$, $l_3 \geq l_2$ or $l_3^2 = \max\{l_1^2, l_2^2, l_3^2\}$

1. $l_3^2 \leq \max\{4l_2^2, l_3^2, K^2\}$.

2.
   - $K^2 = 2l_1^2 + 2l_2^2 - l_3^2 \leq 2l_3^2 + 2l_3^2 - l_3^2 = 3l_3^2 \leq 4l_3^2$,
   - $4l_2^2 \leq 4l_3^2$,
   - $l_3^2 \leq 4l_3^2$;

   $\max\{4l_2^2, l_3^2, K^2\} \leq 4l_3^2$.

$$\frac{1}{4}\max\left\{4l_2^2, l_3^2, K^2\right\} \le l_3^2 \le \max\left\{4l_2^2, l_3^2, K^2\right\}.$$

Putting all results together:

$$\frac{1}{4}\max\left\{4l_2^2, l_3^2, K^2\right\} \le \max\left\{l_1^2, l_2^2, l_3^2\right\} \le \max\left\{4l_2^2, l_3^2, K^2\right\}.$$

Lemma follows by using (4.15). □

4.6 Quasiuniformity results for generation $2n$

Type 1 fathers of $\tau_{max}^{(m)}$ and $\tau_{\min}^{(m)}$ at *level* $= m-2$ are denoted by $\overline{\tau}_1^{(m-2)}$ and $\underline{\tau}_1^{(m-2)}$ respectively. We always assume that the coarsest triangulation is quasiuniform as in (4.3). On $m = level = 2n$, we only have triangles of type 1 and 4.

Case 1: $type(\tau_{\max}^{(m)}) = 1$ and $type(\tau_{\min}^{(m)}) = 1$

$$\frac{\text{size}(\tau_{\max}^{(m)})}{\text{size}(\tau_{\min}^{(m)})} = \frac{(1/2)\text{size}(\overline{\tau}_1^{(m-2)})}{(1/2)\text{size}(\underline{\tau}_1^{(m-2)})} = \frac{(1/2)^{m/2}\text{size}(\overline{\tau}_1^{(0)})}{(1/2)^{m/2}\text{size}(\underline{\tau}_1^{(0)})} \le \gamma^{(0)}.$$

Case 2: $type(\tau_{\max}^{(m)}) = 1$ and $type(\tau_{\min}^{(m)}) = 4$

For any triangle of $type = 4$ on $level = m$, by Lemma 4.4 we had

$$\text{size}(\tau_4^{(m)}) \le \text{size}(\tau_1^{(m-2)}) \le 4\ \text{size}(\tau_4^{(m)}).$$

$$\frac{\text{size}(\tau_{\max}^{(m)})}{\text{size}(\tau_{\min}^{(m)})} = \frac{(1/2)\text{size}(\overline{\tau}_1^{(m-2)})}{\text{size}(\tau_{\min}^{(m)})} \le 4\frac{(1/2)\text{size}(\overline{\tau}_1^{(m-2)})}{\underline{\tau}_1^{(m-2)}} = 4\frac{(1/2)^{m/2}\text{size}(\overline{\tau}_1^{(0)})}{(1/2)^{m/2-1}\text{size}(\underline{\tau}_1^{(0)})} \le 2\gamma^{(0)}.$$

Case 3: $type(\tau_{\max}^{(m)}) = 4$ and $type(\tau_{\min}^{(m)}) = 1$

$$\frac{\text{size}(\tau_{\max}^{(m)})}{\text{size}(\tau_{\min}^{(m)})} = \frac{\text{size}(\tau_{\max}^{(m)})}{(1/2)\text{size}(\underline{\tau}_1^{(m-2)})} \le \frac{\text{size}(\overline{\tau}_1^{(m-2)})}{(1/2)\underline{\tau}_1^{(m-2)}} = \frac{(1/2)^{m/2-1}\text{size}(\overline{\tau}_1^{(0)})}{(1/2)^{m/2}\text{size}(\underline{\tau}_1^{(0)})} \le 2\gamma^{(0)}.$$

Case 4: $type(\tau_{\max}^{(m)}) = 4$ and $type(\tau_{\min}^{(m)}) = 4$

$$\frac{\text{size}(\tau_{\max}^{(m)})}{\text{size}(\tau_{\min}^{(m)})} \le 4\frac{\text{size}(\overline{\tau}_1^{(m-2)})}{\text{size}(\underline{\tau}_1^{(m-2)})} = 4\frac{(1/2)^{m/2-1}\text{size}(\overline{\tau}_1^{(0)})}{(1/2)^{m/2-1}\text{size}(\underline{\tau}_1^{(0)})} \le 4\gamma^{(0)}.$$

4.7 Quasiuniformity results for generation $2n-1$

Triangles of *level* $= m-1$ are of type 2 and 3, where $m = 2n,\ n \geq 1$. For any triangle of *type* $= 2$ on *level* $= m-1$, by Lemma 4.5 we had

$$\text{size}(\tau_2^{(m-1)}) \leq \text{size}(\tau_1^{(m-2)}) \leq 2\ \text{size}(\tau_2^{(m-1)}).$$

For any triangle of *type* $= 3$ on *level* $= l$, by Lemma 4.6 we had

$$\text{size}(\tau_3^{(m-1)}) \leq \text{size}(\tau_1^{(m-2)}) \leq 2\ \text{size}(\tau_3^{(m-1)}).$$

Case 1: $type(\tau_{\max}^{(m-1)}) = 2$ and $type(\tau_{\min}^{(m-1)}) = 2$. By Lemma 4.5,

$$\frac{\text{size}(\tau_{\max}^{(m-1)})}{\text{size}(\tau_{\min}^{(m-1)})} \leq 2\frac{\text{size}(\overline{\tau}_1^{(m-2)})}{\text{size}(\underline{\tau}_1^{(m-2)})} = 2\frac{(1/2)^{(m-2)/2}\text{size}(\overline{\tau}_1^{(0)})}{(1/2)^{(m-2)/2}\text{size}(\underline{\tau}_1^{(0)})} \leq 2\gamma^{(0)}.$$

Case 2: $type(\tau_{\max}^{(m-1)}) = 2$ and $type(\tau_{\min}^{(m-1)}) = 3$. By Lemmas 4.5 and 4.6,

$$\frac{\text{size}(\tau_{\max}^{(m-1)})}{\text{size}(\tau_{\min}^{(m-1)})} \leq 2\frac{\text{size}(\overline{\tau}_1^{(m-2)})}{\text{size}(\underline{\tau}_1^{(m-2)})} = 2\frac{(1/2)^{(m-2)/2}\text{size}(\overline{\tau}_1^{(0)})}{(1/2)^{(m-2)/2}\text{size}(\underline{\tau}_1^{(0)})} \leq 2\gamma^{(0)}.$$

Case 3: $type(\tau_{\max}^{(m-1)}) = 3$ and $type(\tau_{\min}^{(m-1)}) = 2$. By Lemmas 4.5 and 4.6,

$$\frac{\text{size}(\tau_{\max}^{(m-1)})}{\text{size}(\tau_{\min}^{(m-1)})} \leq 2\frac{\text{size}(\overline{\tau}_1^{(m-2)})}{\text{size}(\underline{\tau}_1^{(m-2)})} = 2\frac{(1/2)^{(m-2)/2}\text{size}(\overline{\tau}_1^{(0)})}{(1/2)^{(m-2)/2}\text{size}(\underline{\tau}_1^{(0)})} \leq 2\gamma^{(0)}.$$

Case 4: $type(\tau_{\max}^{(m-1)}) = 3$ and $type(\tau_{\min}^{(m-1)}) = 3$. By Lemma 4.6,

$$\frac{\text{size}(\tau_{\max}^{(m-1)})}{\text{size}(\tau_{\min}^{(m-1)})} \leq 2\frac{\text{size}(\overline{\tau}_1^{(m-2)})}{\text{size}(\underline{\tau}_1^{(m-2)})} = 2\frac{(1/2)^{(m-2)/2}\text{size}(\overline{\tau}_1^{(0)})}{(1/2)^{(m-2)/2}\text{size}(\underline{\tau}_1^{(0)})} \leq 2\gamma^{(0)}.$$

For the sake of simplicity, we have always considered the case where the father at level $m-2$ is of type 1. However, a type 4 father is also possible. We also need to establish equivalence between

- $\text{size}(\tau_1^{(m)}) = \max\left\{\frac{l_1}{2}, \frac{l_2}{2}, \frac{l_3}{2}\right\}$,
- $\text{size}(\tau_2^{(m-1)}) = \max\left\{l_1, \frac{l_3}{2}, \frac{K}{2}\right\}$,

- $\mathrm{size}(\tau_3^{(m-1)}) = \max\left\{l_2, \frac{l_3}{2}, \frac{K}{2}\right\}$,

and

- $\mathrm{size}(\tau_4^{(m-2)}) = \max\{l_1, l_2, K\}$.

Then, these equivalences can be remedied with the already existing ones for the father of type 1 as follows:

$$\frac{\mathrm{size}(\tau_4^{(m-2)})}{\mathrm{size}(\tau_2^{(m-1)})} = \frac{\mathrm{size}(\tau_4^{(m-2)})}{\mathrm{size}(\tau_4^{(m)})} \frac{\mathrm{size}(\tau_4^{(m)})}{\mathrm{size}(\tau_1^{(m-2)})} \frac{\mathrm{size}(\tau_1^{(m-2)})}{\mathrm{size}(\tau_2^{(m-1)})},$$

the first ratio $= 2$, and the second and third ones are handled by (C) and (A) respectively.

$$\frac{\mathrm{size}(\tau_4^{(m-2)})}{\mathrm{size}(\tau_3^{(m-1)})} = \frac{\mathrm{size}(\tau_4^{(m-2)})}{\mathrm{size}(\tau_4^{(m)})} \frac{\mathrm{size}(\tau_4^{(m)})}{\mathrm{size}(\tau_1^{(m-2)})} \frac{\mathrm{size}(\tau_1^{(m-2)})}{\mathrm{size}(\tau_3^{(m-1)})},$$

the first ratio $= 2$, and the second and third ones are handled by (C) and (B) respectively.

$$\frac{\mathrm{size}(\tau_4^{(m-2)})}{\mathrm{size}(\tau_1^{(m)})} = \frac{\mathrm{size}(\tau_4^{(m-2)})}{\mathrm{size}(\tau_1^{(m-4)})} \frac{\mathrm{size}(\tau_1^{(m-4)})}{\mathrm{size}(\tau_1^{(m)})},$$

the first ratio is handled by (C) and the second one is $= 4$.

Eventually, we conclude that the 2D newest vertex bisection gives rise to quasiuniform meshes for the simplices of generation m. Next, we conjecture that the generation bound is as follows:

$$|L(\tau) - L(\tau')| \leq 1; \quad \tau \cap \tau' = e.$$

Then in order to establish quasiuniformity, one can easily observe the pattern that would satisfy Assumption 4.3.

4.8 Survey of refinement routines

We give a survey of the existing refinement routines on simplicial 2D/3D conforming meshes. We analyze their behavior in four categories: number of similarity classes produced, shape degeneracy, number of steps taken by the conformity routines, and local quasiuniformity.

Any simplex has different possible subdivision configurations. These configurations determine the size and shape of children simplices. We define an equivalence relation in order to categorize simplices into similarity (equivalence) classes. A small number of similarity classes indicates a simple routine in nature which is relatively easy not only to implement, but also to analyze further properties such as conformity and shape quality. We define two $d+1$-simplices to be in the same similarity class, if one can be transformed into the other by translation and uniform scaling (i.e. the scaling is the same for all d coordinates). So any two simplices in the same similarity class can be related to each other by an affine transformation.

If only a fixed finite number of similarity classes arise, then the simplicial shape cannot degenerate as the mesh is refined. Meaning all the solid angles of all descendents remain bounded below by a positive constant depending on the simplices in the initial mesh.

To refine a given conforming mesh, we first subdivide those simplices that have been marked for refinement. This will usually lead to a nonconforming mesh. Nonconformity is then remedied by further refinement until a conforming mesh is obtained. It is of great importance that this procedure will terminate in finitely many steps.

It is always desirable to obtain a locally quasiuniform mesh. This indicates a local harmony in simplex sizes. As will be discussed in (4.17), this is a fundamental requirement for the H^1-stability of the L_2-projection on locally refined meshes.

We discuss local bisection and octasection strategies. Usually octasection is combined with a variety of complementary procedures to preserve conformity. However, bisection can be used alone to create locally refined conforming meshes, and this simplifies the implementation. Moreover, bisection allows finer control over the mesh size. If an error indicator marks a tetrahedron, with bisection we can subdivide the tetrahedron to reduce its volume by a factor of $2, 4, \ldots$ as required. On the other hand, with octasection it is not possible to reduce the volume by a factor less than 8.

4.8.1 Newest vertex bisection

2D newest vertex bisection is one primary focus in Mitchell [26]. Indeed, it is shown by Sewell [35] that this method creates only four similarity classes, independent of the refinement level. This method is relatively simple in a way that no computation is necessary to determine the edge to be bisected. Maubach [25] extended the method to the d-spatial dimension, and also gave a remarkable sharp bound of $d\ d!\ 2^{d-2}$ for the number of similarity classes.

Due to a small number of similarity classes, this method may provide the highest quality shape regular meshes. Mitchell introduced the notion of *compatible divisibility* for a triangle. Conformity is always preserved, once all the triangles in the mesh are compatibly divisible. Given any triangle, one can find a way to assign the initial peaks such that every triangle is compatibly divisible. It would be nice to develop an algorithm which automatically assigns the peaks such that one can get a perfect matching. However, the perfect problem is known to be NP-Complete. Maubach considered a refinement edge assignment procedure similar to compatible divisibility, again NP-Completeness rules out the practical use of this method except for very special initial meshes.

On the other hand, Bänsch in his pioneering paper [8], presented a local tetrahedral mesh refinement algorithm which produces a conforming mesh in a finite number of steps without assuming any special initial mesh (see Proposition 2, page 187 in [8]).

The author showed, in subsections 4.5.2, 4.5.3 that we get locally quasiuniform meshes in 2D, and there seems to be no 3D result present at the moment.

4.8.2 Octasection (quadrasection) combined with bisection

Besides bisection, a tetrahedron may be subdivided by octasection. This approach has been studied by Zhang [44] and Ong [27] to obtain uniformly refined meshes. However, octasection cannot be used alone to create adaptive conforming meshes. The 2D red refinement complemented with green closure to preserve conformity, was introduced by Bank [5]. We facilitate Bey's [9] local octasection (red) with a supplemental bisection (green) closure, again to obtain conformity.

By construction, every tetrahedron in any $\mathcal{T}_j$ is geometrically similar to some tetrahedron in $\mathcal{T}_0$. Bey points out 10 similarity classes of which 9 correspond to irregular refinements. There are only 2 similarity classes in 2D.

Nondegeneracy and conformity are automatically preserved. Dahmen and Kunoth [17] proved the 2D local quasiuniformity, and the author extended the same result to 3D by using analogous techniques.

4.8.3 Longest edge bisection

The mathematical theory of this method first appeared in Rivara [33]. In 2D, Adler [1] proved the existence of a finite number of similarity classes. Although the number may be arbitrarily large depending on the initial triangle. Finiteness of similarity classes is still an open question for 3D.

2D nondegeneracy follows from the finite similarity classes. However, it was independently proven by Stein, who was not aware of the finiteness of the similarity classes. There is a rumor that 3D nondegeneracy has been proved, but it has not appeared in the literature yet. Rivara and Levin [34] conjectured the same fact, but no theoretical basis is given.

It is proven in [34] by a combinatorial approach that iteratively bisecting tetrahedra with a nonconforming edge by the longest edge produces a conforming mesh in a finite number of steps. The longest edge bisection is quite popular and 3D and 2D versions of conformity routines are implemented in the FEtk and PLTMG respectively. There seems to be no local quasiuniformity result in the literature.

4.8.4 Marked tetrahedron bisection

Arnold, Mutherjee, and Pouly (AMP) [2] and Joe and Liu (JL) [24] introduced different methods. Although the two methods show great resemblance, it is hardly possible to conclude exact correspondence. AMP-bisection is relatively simple compared to JL-bisection.

AMP-bisection has a sharp bound of 36 for the number of similarity classes, obeying Maubach's estimates. Whereas, JL-bisection produces at most 168 distinct similarity classes. Both methods propose conformity routines that would terminate at most in $3j$ steps for a tetrahedron in $\mathcal{T}_j$ (see Theorem 3.1 in both papers).

JL-bisection has a sound mathematical framework that enables one to quantify shape quality and element size. With the help of the lower bound provided for a shape measure, the author established local quasiuniformity. On the other hand, there is no known result for AMP-bisection.

4.9 H^1-stability of the L_2-projection

There is widespread interest in the stability of Q_j in $H^1(\Omega)$.

Definition 4.4 *The L_2-projection Q_j onto $\mathcal{S}_j$, is H^1-stable if there exists an absolute constant c such that*

$$\|Q_j u\|_{H^1(\Omega)} \leq c \, \|u\|_{H^1(\Omega)}, \quad \forall u \in H^1(\Omega). \tag{4.16}$$

A partial list of interested fields would include: Galerkin or boundary FE methods for elliptic and parabolic boundary value problems, analysis of a Neumann series corresponding to a second kind boundary integral equation. We are primarily interested in 4.16 because of the lower bound in the norm equivalence (2.5) in the BPX setting. Under the assumption of H^s-stability, $s > 1$, of Q_j, we have given a simple proof of an optimal lower bound. In the WMHB setting, H^s-stability of Q_j becomes precious when the standard C-S like assumption (a.ii) fails to hold. In this case, we only hope for a suboptimal lower bound, and this is made possible through use of the H^1-stability of Q_j. Armed with the quasiuniform mesh, it is well known that Q_j attains H^1-stability. It is striking that (4.16) holds as well for locally quasiuniform meshes, provided that the size of neighboring elements does not change drastically. Bramble, Pasciak, and Steinbach [12] formulate an explicit mesh condition for any spatial dimension, involving element mass matrices which will imply (4.16). These conditions can be easily checked for a given FE mesh, allowing the user to redefine the mesh if necessary. For brevity, we do not give technical details required to establish the H^1-stability of the L_2-projection in (4.16). The details can be found in [12, 15]. We restrict our attention to providing an outline of the mesh conditions as they are given in [12, 15].

4.10 Notation for mesh conditions

Let $\tau_l \in \mathcal{T}_j$ be a simplex whose size is defined as

$$\mathrm{size}(\tau_l) = \mathrm{volume}(\tau_l)^{\frac{1}{d}}.$$

We will omit the superscript (j) which denotes the level. We are interested in locally quasiuniform meshes. Then, we develop tools that control various properties of a patch. Let $I(k)$ denote the index set of elements satisfying $\tau_l \in \mathrm{supp}\ \phi_k$. We define a local mesh size (patchwise) associated to the basis function ϕ_k

$$\mathrm{size}(\mathrm{supp}\ \phi_k) = \frac{1}{\#I(k)} \sum_{l \in I(k)} \mathrm{size}(\tau_l).$$

We assume that the mesh $\mathcal{T}_j$ is patchwise quasiuniform. Then using the equivalent quasiuniform definition in (4.4), elements τ_k, τ_l in any patch P satisfies

$$\frac{\mathrm{size}(\tau_l)}{\mathrm{size}(\tau_k)} \leq \gamma^{(j)}.$$

Then, we get

$$\frac{\mathrm{size}(\mathrm{supp}\ \phi_k)}{\mathrm{size}(\tau_l)} \leq \frac{\mathrm{size}(\tau_{\max})}{\mathrm{size}(\tau_l)} \leq \gamma^{(j)},$$
$$\frac{\mathrm{size}(\tau_l)}{\mathrm{size}(\mathrm{supp}\ \phi_k)} \leq \frac{\mathrm{size}(\tau_l)}{\mathrm{size}(\tau_{\min})} \leq \gamma^{(j)}.$$

Thus, we have verified the *fundamental patch assumption* in the mesh criterion analysis, that is,

$$\frac{1}{\gamma^{(j)}} \leq \frac{\text{size}(\text{supp } \phi_k)}{\text{size}(\tau_l)} \leq \gamma^{(j)}. \tag{4.17}$$

The existence of $\gamma^{(j)}$, which depends only on shape regularity, is only possible in a patchwise quasiuniform mesh. If we demand an H^1-stable Q_j, then only refinement routines which provide patchwise quasiuniformity can be pleasing. This makes it clear why we established patchwise quasiuniformity for the 2D newest vertex and 3D marked tetrahedron bisection routines.

4.11 Explicit mesh condition

We outline the mesh condition used by Bramble, Pasciak, and Steinbach [12]. For each simplex τ_l, we define local matrices of size $(d+1) \times (d+1)$

$$\begin{aligned} M(\tau_l)_{ij} &= (\phi_{l,i}, \phi_{l,j})_{L_2(\tau_l)}, \\ D(\tau_l)_{ij} &= (\phi_{l,i}, \phi_{l,j})_{L_2(\tau_l)} \delta_{ij}, \\ H(\tau_l)_{ij} &= \text{size}(\text{supp } \phi_{l,i}) \delta_{ij}, \quad i, j = 1, \ldots, d+1, \end{aligned}$$

where $\phi_{l,i}$ is the basis function corresponding to the i-th vertex of τ_l, and $M(\tau_l)$ is the element mass matrix.

Definition 4.5 *If there exists c such that*

$$x^T D(\tau_l) x \leq c\, x^T H(\tau_l)^{-1} M(\tau_l) H(\tau_l) x, \quad \forall x \in \mathbb{R}^{d+1}, \tag{4.18}$$

then we say that the BPS-criterion is satisfied.

Next, we give some insight about (4.18) in terms of functions. For that, let H denote the $m \times m$ matrix, where m is the number of basis function $\phi_i \in \mathcal{S}_j, \quad i = 1, \ldots m$

$$H_{ij} = \text{size}(\text{supp } \phi_i)\ \delta_{ij}.$$

For $\underline{u} \in \mathbb{R}^m$, define $\underline{v} = H\underline{u}$ and $\underline{w} = H^{-1}\underline{u}$. The corresponding FE functions are

$$u = \sum_{i=1}^{m} \underline{u}_i \phi_i, \quad v = \sum_{i=1}^{m} \underline{v}_i \phi_i, \quad w = \sum_{i=1}^{m} \underline{w}_i \phi_i.$$

Then the following holds.

$$\begin{aligned}(u, u)_{L_2(\tau_l)} &= x^T M x, \\ (v, w)_{L_2(\tau_l)} &= x^T H^{-1} M H x, \quad \forall x \in \mathbb{R}^{d+1},\end{aligned}$$

where x denotes the components of $\underline{u}$ associated with τ_l. Since both M, D are SPD, there exists c such that

$$x^T M x \leq c \; x^T D x.$$

Then (4.18) is equivalent to

$$||u||^2_{L_2(\tau_l)} \leq c(v, w)_{L_2(\tau_l)}. \tag{4.19}$$

If all the simplices have the same measure, then $H^{-1} M H^{-1} = M$, so that (4.19) trivially holds

$$(u, u)_{L_2(\tau_l)} = (v, w)_{L_2(\tau_l)}.$$

The inequality (4.19) will still hold provided that the size of neighboring simplices do not vary too much. Explicit local conditions on the mesh will be given in the next section.

4.12 Reduction to eigenvalue estimate

By observing

$$x^T \; \frac{1}{2}(HMH^{-1} + H^{-1}MH) \; x = x^T \; H^{-1}MH \; x,$$

we introduce the symmetric matrix

$$M^s = \frac{1}{2}(HMH^{-1} + H^{-1}MH),$$

so that (4.18) is the same as

$$x^T\ D\ x \le c\ x^T\ M^s\ x, \quad \forall x \in \mathbb{R}^{d+1}. \tag{4.20}$$

By the mass matrix calculation in (2.39) and in the proof of Lemma 3.12, we observe that

$$D = \frac{2\ \mathrm{area}(\tau_l)}{(d+1)(d+2)}\ I.$$

Moreover,

$$M^s = \frac{2\ \mathrm{area}(\tau_l)}{4\ (d+1)(d+2)}\ A, \tag{4.21}$$

for $i, j = 1, \dots, d+1$, A is defined as

$$A_{ij} = \begin{cases} 4, & i = j \\ \frac{\mathrm{size}(\mathrm{supp}\ \phi_{l,i})}{\mathrm{size}(\mathrm{supp}\ \phi_{l,j})} + \frac{\mathrm{size}(\mathrm{supp}\ \phi_{l,j})}{\mathrm{size}(\mathrm{supp}\ \phi_{l,i})}, & i \ne j \end{cases}.$$

Proposition 4.7 *Let A be defined as above. Then all the eigenvalues are in the set $\{\lambda_+, \lambda_-, 2\}$, where*

$$\lambda_\pm = 3 + d \pm \left\{ \sum_{i=1}^{d+1} \mathrm{size}^2(\mathrm{supp}\ \phi_{l,i}) \sum_{i=1}^{d+1} \mathrm{size}^{-2}(\mathrm{supp}\ \phi_{l,i}) \right\}^{1/2}. \tag{4.22}$$

Proof. See Proposition 6.1, page 9 in [12]. □

If all eigenvalues of A are strictly positive and bounded away from zero, then by (4.21), so are the eigenvalues of M^s. Hence, we have transformed the mesh criterion (4.20) into an eigenvalue condition. By Proposition 4.7, this is equivalent to

$$3 + d - \left\{ \sum_{i=1}^{d+1} \mathrm{size}^2(\mathrm{supp}\ \phi_{l,i}) \sum_{i=1}^{d+1} \mathrm{size}^{-2}(\mathrm{supp}\ \phi_{l,i}) \right\}^{1/2} \ge c, \tag{4.23}$$

where c is independent of τ_l. For adaptive triangulations, (4.23) provides a mesh constraint. Using the fundamental patch assumption (4.17), we get

$$\begin{aligned} 2 - (d+1)((\gamma^{(j)})^2 - 1) &= 3 + d - (d+1)(\gamma^{(j)})^2 \\ &\le 3 + d - \left\{ \sum_{i=1}^{d+1} \mathrm{size}^2(\mathrm{supp}\ \phi_{l,i}) \sum_{i=1}^{d+1} \mathrm{size}^{-2}(\mathrm{supp}\ \phi_{l,i}) \right\}^{1/2}. \end{aligned}$$

If $2-(d+1)((\gamma^{(j)})^2-1) \geq c$, or equivalently

$$\gamma^{(j)} \leq \sqrt{1+\frac{2-c}{d+1}},$$

holds for $0<c<2$, then (4.23) is satisfied. $0<c<2$ has to be enforced, because the fundamental patch assumption requires $\gamma^{(j)} \geq 1$ in (4.17). In this step, note that

$$\gamma^{(j)} < \sqrt{1+\frac{2}{d+1}}.$$

Then, the construction of meshes with the above bound becomes a challenge. Such $\gamma^{(j)}$ can be achieved only if the neighboring element sizes are controlled in the mesh. We can assert that such a mesh guarantees an H^1-stable L_2-projection. One can see a numerical example of an adaptive 2D mesh where the direct computation

$$c = \max_{\tau_l} \left(5 - \left\{ \sum_{i=1}^{3} \mathrm{size}^2(\mathrm{supp}\ \phi_{l,i}) \sum_{i=1}^{3} \mathrm{size}^{-2}(\mathrm{supp}\ \phi_{l,i}) \right\}^{1/2} \right),$$

is utilized to satisfy the stability condition (4.18) (page 10 in [12]). Next, we examine a similar mesh condition by Carstensen [15].

4.13 A similar mesh condition

We use a different matrix Λ instead of H in order to scale the mass matrix M. For the change, we assume a positive real parameter d_z (weight) associated to each basis function ϕ_z.

Definition 4.6 *The scaling matrix $\Lambda(\tau)$ is defined as*

$$\Lambda(\tau)_{jk} = \frac{\mathrm{size}(\tau)}{d_{z_j}} \delta_{jk}, \tag{4.24}$$

where d_{z_j} is the weight associated to $\phi_{\tau,j}$.

To obtain a bounded scaling, we further assume an absolute constant c such that

$$\mathrm{size}(\tau)/d_z + d_z/\mathrm{size}(\tau) < c. \tag{4.25}$$

The following mesh condition is weaker than (implied by) either the BPS or Crouzeix-Thomee criterion. In fact, we will also show that it is equivalent to the BPS-criterion with some additional assumption.

Definition 4.7 *If there exists c_1 and c_2 such that*

$$c_1 x^T \ \Lambda^2(\tau)M(\tau)\Lambda^2(\tau) \ x \leq x^T \ M(\tau) \ x \leq c_2 x^T \ \Lambda^2(\tau)M(\tau) \ x, \quad \forall x \in \mathbb{R}^{d+1}, \tag{4.26}$$

then we can say that the mesh satisfies the C-criterion.

The first inequality in (4.26) reflects a proper scaling of d_z and size(τ). The second inequality implies that the symmetric part of $\Lambda^2 M$ is positive definite. This is a crucial condition and relates the mass matrix M to neighboring element sizes. If we choose the weights as $d_z =$ size(τ) equal to a global discretization parameter, then (4.26) holds with $c_1 = c_2 = 1$. Then, we can say that (4.26) trivially holds for globally quasiuniform meshes.

Lemma 4.8 *The BPS-criterion (4.18) implies the C-criterion (4.26).*

Proof. If d_{z_l} is chosen to be the arithmetic mean of all elements $\tau_l \in \text{supp } \phi_l$, then Λ in terms of H is formulated as

$$\Lambda = \text{size}(\tau)H^{-1}.$$

$$x^T Dx \leq cx^T H^{-1} MHx = cx^T \text{size}(\tau)^{-1}\Lambda M \Lambda^{-1}\text{size}(\tau)x = cx^T \Lambda M \Lambda^{-1} x, \quad \forall x \in \mathbb{R}^{d+1}.$$

Equivalently, letting $x = \Lambda y$

$$y^T \Lambda D \Lambda y \leq cy^T \Lambda^2 My, \quad \forall y \in \mathbb{R}^{d+1}.$$

Since M and $\Lambda D \Lambda$ are both SPD, then

$$x^T Mx \leq cx^T \Lambda^2 Mx.$$

The first inequality in (4.26) is obtained by the SPD-ness of $\Lambda^2 M \Lambda^2$ and M and the boundedness of the scaling in (4.25). □

4.14 Equivalence of BPS- and C-criteria

Carstensen [15] established equivalence between the BPS- (4.18) and the C-criterion (4.26) by purely using eigenvalue analysis found in [12]. In this section, we present the same result in a more systematic way.

Lemma 4.9 *Let $d_x/d_y \leq \kappa < \sqrt{2}+\sqrt{3}$ for all vertices x and y of $\tau \in \mathcal{T}_j$. Then, the smallest eigenvalue of M^s is positive.*

Proof. By the previous analysis we know that the smallest eigenvalue of M^s is $5-\mu$ where

$$\mu^2 = \sum_{i=1}^{3} \text{size}^2(\text{supp } \phi_i) \sum_{i=1}^{3} \text{size}^{-2}(\text{supp } \phi_i),$$

which can also be written as

$$\mu^2 = \sum_{j,k=1}^{3} \frac{\text{size}^2(\text{supp } \phi_j)}{\text{size}^2(\text{supp } \phi_k)}.$$

The entries of Λ are $\lambda_k = \frac{\text{size}(\tau)}{d_y}$. By assumption $\frac{\lambda_k}{\lambda_j} = \frac{d_x}{d_y} \leq \kappa$. Since $H = \text{size}(\tau)\Lambda^{-1}$, then $\text{size}(\text{supp } \phi_j) = \frac{\text{size}(\tau)}{\lambda_j}$, implying, $\frac{\text{size}(\text{supp } \phi_j)}{\text{size}(\text{supp } \phi_k)} = \frac{\lambda_k}{\lambda_j}$. Then $\mu^2 = \sum_{j,k=1}^{3} \frac{\lambda_k^2}{\lambda_j^2} \leq 3+2(1+\kappa^2+1/\kappa^2) < 25$. Therefore the smallest eigenvalue $5-\mu > 0$. $\square$

Now, we can conclude that all the eigenvalues of M^s are positive, thereby the BPS-criterion (4.18) is satisfied. Under assumption of $d_x/d_y < \sqrt{2}+\sqrt{3}$, we can also satisfy the C-criterion (4.26) as in the corollary below. Finally, with the proper choice of the weights d_x, one can attain both the BPS-criterion and C-criterion, which may be interpreted as an equivalence of the two criteria.

Corollary 4.10 *If $d_x/d_y \leq \kappa < \sqrt{2}+\sqrt{3}$, then C-criterion (4.26) holds.*

Proof. $M^s = 1/2(HMH^{-1} + H^{-1}MH) = 1/2(\Lambda^{-1}M\Lambda + \Lambda M\Lambda^{-1})$ by using $\Lambda = \text{size}(\tau)H^{-1}$ then by Lemma 4.9

$$1/2(\Lambda^{-1}M\Lambda + \Lambda M\Lambda^{-1}) = 1/2\Lambda^{-1}(M\Lambda^2 + \Lambda^2 M)\Lambda^{-1},$$

becomes SPD. Hence, $1/2(M\Lambda^2 + \Lambda^2 M)$ is also SPD. M is SPD, implying that there exists c such that

$$\begin{aligned} x^T M x &\leq c x^T 1/2(M\Lambda^2 + \Lambda^2 M)x \\ &= c x^T \Lambda^2 M x. \end{aligned}$$

Moreover, $\frac{dx}{d_y} \leq \kappa$ gives use to a proper scaling, meaning that $\Lambda^2 M \Lambda^2$ does not blow up for any τ. Using the fact that $\Lambda^2 M \Lambda^2$ is SPD, yields a c such that

$$c\ x^T\ \Lambda^2 M \Lambda^2\ x \leq x^T M x, \quad \forall x \in \mathbb{R}^{d+1}.$$

$\square$

Chapter 5

Implementation

5.1 Matrix representations of the WMHB method

For any $v \in \mathcal{S}_j$, let $\mathbf{v}$ denote its nodal coefficient vector. We decompose v as follows:

$$v = (I - Q^a_{j-1})(I_j - I_{j-1})v + w,$$

where $w \in \mathcal{S}_{j-1}$ is uniquely determined as $w = I_{j-1}v + Q^a_{j-1}(I_j - I_{j-1})v$, and Q^a_{j-1} denotes the approximate L_2-projection onto $\mathcal{S}_{j-1}$. Our goal is to find vector representation for the components of v.

$$\mathbf{v} = Y_1^{(j)}\hat{\mathbf{v}}_1 + Y_2^{(j)}\hat{\mathbf{v}}_2,$$

where

$$Y_1^{(j)} = P^j_{j-1} = \begin{bmatrix} I \\ R^{(j)} \end{bmatrix}, \; Y_2^{(j)} = \left(I - P^j_{j-1}\tilde{M}^{(j-1)}_{inv}{P^j_{j-1}}^T M^{(j)}\right)\begin{bmatrix} 0 \\ I \end{bmatrix},$$

where $\hat{\mathbf{v}} = \begin{bmatrix} \hat{\mathbf{v}}_\mathbf{1} \\ \hat{\mathbf{v}}_\mathbf{2} \end{bmatrix}$ WMHB coefficients of v, and $\tilde{M}^{(j-1)}_{inv}$ denotes the approximate inverse of the mass matrix at level $j-1$.

Armed with the transformation matrix $Y = [Y_1, Y_2]$, we can consider the problem $A^{(j)}\mathbf{v} = \mathbf{d}$. We transform it into WMHB by letting $\mathbf{v} = \mathbf{Y}\hat{\mathbf{v}}$. In other words, we consider

$$Y^T A^{(j)} Y \hat{\mathbf{v}}, \quad \hat{\mathbf{d}} = Y^T \mathbf{d}.$$

Two-by-two block WMHB system:

$$\begin{bmatrix} \hat{A}_{11}^{(j)} & \hat{A}_{12}^{(j)} \\ \hat{A}_{21}^{(j)} & \hat{A}_{22}^{(j)} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{v}}_{\mathbf{1}} \\ \hat{\mathbf{v}}_{\mathbf{2}} \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{d}}_{\mathbf{1}} \\ \hat{\mathbf{d}}_{\mathbf{2}} \end{bmatrix}$$

$$\begin{aligned} \hat{A}_{11}^{(j)} &= Y_1^{(j)^T} A^{(j)} Y_1^{(j)} = P_{j-1}^{j}{}^T A^{(j)} P_{j-1}^{j} = A^{(j-1)}, \\ \hat{A}_{12}^{(j)} &= Y_1^{(j)^T} A^{(j)} Y_2^{(j)}, \\ \hat{A}_{21}^{(j)} &= Y_2^{(j)^T} A^{(j)} Y_1^{(j)}, \\ \hat{A}_{22}^{(j)} &= Y_2^{(j)^T} A^{(j)} Y_2^{(j)}. \end{aligned}$$

5.2 Hierarchical representations of the WMHB blocks

Let us derive the block structure of $Y_2^{(j)}$.

$$\begin{aligned} Y_2^{(j)} &= \left(I - P_{j-1}^{j} \tilde{M}_{inv}^{(j-1)} P_{j-1}^{j}{}^T M^{(j)}\right) \begin{bmatrix} 0 \\ I \end{bmatrix} \\ &= \begin{bmatrix} 0 \\ I \end{bmatrix} - \left(P_{j-1}^{j} \tilde{M}_{inv}^{(j-1)} P_{j-1}^{j}{}^T\right) \begin{bmatrix} M_{11}^{(j)} & M_{12}^{(j)} \\ M_{21}^{(j)} & M_{22}^{(j)} \end{bmatrix} \begin{bmatrix} 0 \\ I \end{bmatrix} \\ &= \begin{bmatrix} 0 \\ I \end{bmatrix} - \left(P_{j-1}^{j} \tilde{M}_{inv}^{(j-1)} \begin{bmatrix} I & R^{(j)^T} \end{bmatrix}\right) \begin{bmatrix} M_{12}^{(j)} \\ M_{22}^{(j)} \end{bmatrix} \\ &= \begin{bmatrix} 0 \\ I \end{bmatrix} - \begin{bmatrix} I \\ R^{(j)} \end{bmatrix} \tilde{M}_{inv}^{(j-1)} \left(M_{12}^{(j)} + R^{(j)^T} M_{22}^{(j)}\right) \\ &= \begin{bmatrix} 0 \\ I \end{bmatrix} - \begin{bmatrix} \tilde{M}_{inv}^{(j-1)} \left(M_{12}^{(j)} + R^{(j)^T} M_{22}^{(j)}\right) \\ R^{(j)} \tilde{M}_{inv}^{(j-1)} \left(M_{12}^{(j)} + R^{(j)^T} M_{22}^{(j)}\right) \end{bmatrix} \\ &= \begin{bmatrix} -\tilde{M}_{inv}^{(j-1)} \left(M_{12}^{(j)} + R^{(j)^T} M_{22}^{(j)}\right) \\ I - R^{(j)} \tilde{M}_{inv}^{(j-1)} \left(M_{12}^{(j)} + R^{(j)^T} M_{22}^{(j)}\right) \end{bmatrix} \end{aligned}$$

$$= \begin{bmatrix} C^{(j)} \\ D^{(j)} \end{bmatrix}.$$

Omitting the superscripts, one can write down the expression for hierarchical blocks by utilizing change of basis matrix;

$$Y = [Y_1 \;\; Y_2], \quad Y_1 = \begin{bmatrix} I \\ R \end{bmatrix}, \quad Y_2 = \begin{bmatrix} C \\ D \end{bmatrix},$$

as follows.

$$\begin{array}{ccccc} \hat{A}_{WMHB} & = & Y^T & A & Y \\ \begin{bmatrix} \hat{A}_{11} & \hat{A}_{12} \\ \hat{A}_{21} & \hat{A}_{22} \end{bmatrix} & = & \begin{bmatrix} I & R^T \\ C^T & D^T \end{bmatrix} & \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} & \begin{bmatrix} I & C \\ R & D \end{bmatrix} \end{array}$$

$$\begin{array}{lclclclcl} \hat{A}_{11} & = & A_{11} & + & A_{12}R & + & R^T A_{21} & + & R^T A_{22} R \\ \hat{A}_{12} & = & A_{11}C & + & A_{12}D & + & R^T A_{21} C & + & R^T A_{22} D \\ \hat{A}_{21} & = & C^T A_{11} & + & C^T A_{12} R & + & D^T A_{21} & + & D^T A_{22} R \\ \hat{A}_{22} & = & C^T A_{11} C & + & C^T A_{12} D & + & D^T A_{21} C & + & D^T A_{22} D \end{array}$$

If $C = 0, D = I$, then the hierarchical blocks correspond to HB blocks, and

$$Y = [Y_1 \;\; Y_2], \quad Y_1 = \begin{bmatrix} I \\ R \end{bmatrix}, \quad Y_2 = \begin{bmatrix} 0 \\ I \end{bmatrix}.$$

Then hierarchical blocks with respect to HB simplify as follows.

$$\begin{array}{ccccc} \hat{A}_{HB} & = & Y^T & A & Y \\ \begin{bmatrix} \hat{A}_{11} & \hat{A}_{12} \\ \hat{A}_{21} & \hat{A}_{22} \end{bmatrix} & = & \begin{bmatrix} I & R^T \\ 0 & I \end{bmatrix} & \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} & \begin{bmatrix} I & 0 \\ R & I \end{bmatrix} \end{array}$$

$$\begin{array}{lclclclcl} \hat{A}_{11} & = & A_{11} & + & A_{12}R & + & R^T A_{21} & + & R^T A_{22} R \\ \hat{A}_{12} & = & & & A_{12} & & & + & R^T A_{22} \\ \hat{A}_{21} & = & & & & & A_{21} & + & A_{22} R \\ \hat{A}_{22} & = & & & & & & & A_{22}. \end{array}$$

5.3 Matrix-matrix products

WMHB claims to be an optimal method for any refinement strategy. For this to happen, one has to make sure that the preprocessing computational cost (i.e. computation of $\hat{A}_{11}, \hat{A}_{12}, \hat{A}_{21}, \hat{A}_{22}$) is $O(n_j)$, where n_j is the number of unknowns corresponding to $\mathcal{S}_j^f$ at level j. In a typical case, n_j is a small constant relative to N and has no relation with N. Hence, in an adaptive refinement scenario, the work required to retain the overall complexity at $O(N)$ must be $O(n_j)$ at each level j.

Define N_j to be the total number of unknowns at level j. Then $N_j = n_0 + n_1 + n_2 + \cdots + n_j$. If the variational conditions are enforced in the conventional way (i.e. $A^{(j-1)} = {P_{j-1}^j}^T A^{(j)} P_{j-1}^j$, without utilizing the block structure), then the storage complexity at level j is $O(N_j)$, due to a constant number of nonzeros in the stiffness matrices. Let us observe below, why the conventional approach brings the overall computational cost up to $O(NJ)$ in the adaptive refinement scenario. Let us assume that the finest level is J, then $N = N_J, \quad N_{J-1} = N - n_J$, and count the total work at each level. Here we take complexity constants to be 1 for simplicity.

$$
\begin{array}{lll}
\textit{level}\ \ J & : & N \\
\textit{level}\ \ J-1 & : & N - n_J \\
\textit{level}\ \ J-2 & : & N - n_J - n_{J-1} \\
\vdots & & \vdots \\
\textit{level}\ \ 2 & : & N - n_J - n_{J-1} - \cdots - n_3 \\
\textit{level}\ \ 1 & : & N - n_J - n_{J-1} - \cdots - n_3 - n_2 \\
\textit{level}\ \ 0 & : & N - n_J - n_{J-1} - \cdots - n_3 - n_2 - n_1.
\end{array}
$$

Adding up the cost at all levels, overall cost is:

$$(J+1)N - (J)n_J - (J-1)n_{J-1} - \cdots - 3n_3 - 2n_2 - n_1. \tag{5.1}$$

Since n_j's are small constants, then (5.1) $\leq N(J+1)$, implying that overall cost is $O(NJ)$.

To get around the extra J term, we have to utilize the block structure of stiffness matrix and clever storage styles for matrices showing up in the calculations. In the case of a symmetric PDE (e.g. the PBE, the elasticity problem), the block expressions simplify, and this dramatically reduces computation for the preprocessing step. So we are only interested in the following blocks:

$$\begin{array}{lcllllllll}
\hat{A}_{22} & = & (C^T A_{11})^* C & + & (C^T A_{21}^T)^* D & + & (D^T A_{21})^* C & + & (D^T A_{22})^* D \\
(\hat{A}_{21})^* & = & (C^T A_{11}) & + & (C^T A_{21}^T) R & + & (D^T A_{21}) & + & (D^T A_{22}) R \\
\hat{A}_{12} & = & (\hat{A}_{21})^T & & & & & & \\
\hat{A}_{11} & = & A_{11} & + & (R^T A_{21})^T & + & (R^T A_{21})^* & + & R^T A_{22} R.
\end{array}$$

$$A = \begin{bmatrix} \begin{bmatrix} A_{11} & \rightarrow & \rightarrow & \rightarrow \\ \rightarrow & \rightarrow & \rightarrow & \rightarrow \\ \rightarrow & \rightarrow & \rightarrow & \rightarrow \\ \rightarrow & \rightarrow & \rightarrow & \rightarrow \end{bmatrix}_{N\times N} & \begin{bmatrix} A_{12} & \times \\ \times & \times \\ \times & \times \\ \times & \times \end{bmatrix}_{N\times n} \\ \\ \begin{bmatrix} A_{21} & \rightarrow & \rightarrow & \rightarrow \\ \rightarrow & \rightarrow & \rightarrow & \rightarrow \end{bmatrix}_{n\times N} & \begin{bmatrix} A_{22} & \rightarrow \\ \rightarrow & \rightarrow \end{bmatrix}_{n\times n} \end{bmatrix},$$

$$Y = \begin{bmatrix} \begin{bmatrix} I & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \end{bmatrix}_{N\times N} & \begin{bmatrix} C & \downarrow \\ \downarrow & \downarrow \\ \downarrow & \downarrow \\ \downarrow & \downarrow \end{bmatrix}_{N\times n} \\ \\ \begin{bmatrix} R & \rightarrow & \rightarrow & \rightarrow \\ \rightarrow & \rightarrow & \rightarrow & \rightarrow \end{bmatrix}_{n\times N} & \begin{bmatrix} D & \rightarrow \\ \rightarrow & \rightarrow \end{bmatrix}_{n\times n} \end{bmatrix},$$

$(\cdot)^*$ indicates a recycled term.

$\rightarrow$ indicates a block stored rowwise.

$\downarrow$ indicates a block stored columnwise.

$\times$ indicates a block which is not stored.

5.4 Types of sparse matrix-matrix products

Sparse matrix storage requires special data structure. Let nz denote the number of nonzeros in matrix $A_{N\times N}$. We list the major components in this storage style.

- `IA` stores the offsets indicating the beginning and end of each row, and is of dimension $1 \times N+1$.
- `JA` stores the column location of each nonzero entry, and is of dimension $1 \times nz$.
- `A` stores the value of nonzero entries, and is of dimension $1 \times nz$.

We are going to give three algorithms to compute the matrices involved in the definition of the stabilized blocks $\hat{A}_{11}, \hat{A}_{12}, \hat{A}_{21}, \hat{A}_{22}$. Since there is no logical check in Algorithm 5.1, and Algorithm 5.2, we are interested only in the algebraic complexity. However in Algorithm 5.3, due to logical checks, one should analyze the algebraic and logical complexities separately. We are going to show that the logical complexity will make the overall computational complexity of the WMHB implementation for the nonsymmetric problem suboptimal.

• Same style storage:

$A_{n\times N}B_{N\times N}, \quad A_{n\times N}B_{N\times n}, \quad A_{n\times n}B_{n\times N}, \quad A_{n\times n}B_{n\times n}: \quad O(n)$

$$\begin{bmatrix} A & \rightarrow & \rightarrow & \rightarrow \\ \rightarrow & \rightarrow & \rightarrow & \rightarrow \end{bmatrix}_{n\times N} \begin{bmatrix} B & \rightarrow & \rightarrow & \rightarrow \\ \rightarrow & \rightarrow & \rightarrow & \rightarrow \\ \rightarrow & \rightarrow & \rightarrow & \rightarrow \\ \rightarrow & \rightarrow & \rightarrow & \rightarrow \end{bmatrix}_{N\times N}, \quad \begin{bmatrix} A & \rightarrow \\ \rightarrow & \rightarrow \end{bmatrix}_{n\times n} \begin{bmatrix} B & \rightarrow & \rightarrow & \rightarrow \\ \rightarrow & \rightarrow & \rightarrow & \rightarrow \end{bmatrix}_{n\times N},$$

$$\begin{bmatrix} A & \rightarrow & \rightarrow & \rightarrow \\ \rightarrow & \rightarrow & \rightarrow & \rightarrow \end{bmatrix}_{n\times N} \begin{bmatrix} B & \rightarrow \\ \rightarrow & \rightarrow \\ \rightarrow & \rightarrow \\ \rightarrow & \rightarrow \end{bmatrix}_{N\times n}, \qquad \begin{bmatrix} A & \rightarrow \\ \rightarrow & \rightarrow \end{bmatrix}_{n\times n} \begin{bmatrix} B & \rightarrow \\ \rightarrow & \rightarrow \end{bmatrix}_{n\times n}.$$

Example:

$$\begin{bmatrix} \rightarrow & \rightarrow & \rightarrow & \rightarrow \\ \rightarrow & \rightarrow & + & \rightarrow \end{bmatrix}_{n\times N} \begin{bmatrix} \rightarrow & \rightarrow & \rightarrow & \rightarrow \\ \rightarrow & \rightarrow & \rightarrow & \rightarrow \\ \bigcirc & \bigcirc & \bigcirc & \bigcirc \\ \rightarrow & \rightarrow & \rightarrow & \rightarrow \end{bmatrix}_{N\times N} = \begin{bmatrix} \rightarrow & \rightarrow & \rightarrow & \rightarrow \\ \bigoplus & \bigoplus & \bigoplus & \bigoplus \end{bmatrix}_{n\times N}.$$

Algorithm 5.1

```
for (i=0;i<n;i++) {
    for (kk=IA[i];kk<IA[i+1];kk++) {
        k = JA[kk];
        for (jj=IB[k];jj<IB[k+1];jj++) {
            j = JB[jj];
            AB[i][j] += A[kk]*B[jj];
        }
    }
}
```

• Different style storage:

$A_{N\times n}B_{n\times N},\quad A_{n\times n}B_{n\times N},\quad A_{n\times n}B_{n\times n}:\quad O(n)$

$$\begin{bmatrix} A & \downarrow \\ \downarrow & \downarrow \\ \downarrow & \downarrow \\ \downarrow & \downarrow \end{bmatrix}_{N\times n} \begin{bmatrix} B & \rightarrow & \rightarrow & \rightarrow \\ \rightarrow & \rightarrow & \rightarrow & \rightarrow \end{bmatrix}_{n\times N}, \quad \begin{bmatrix} A & \downarrow \\ \downarrow & \downarrow \end{bmatrix}_{n\times n} \begin{bmatrix} B & \rightarrow & \rightarrow & \rightarrow \\ \rightarrow & \rightarrow & \rightarrow & \rightarrow \end{bmatrix}_{n\times N},$$

$$\begin{bmatrix} A & \downarrow \\ \downarrow & \downarrow \end{bmatrix}_{n\times n} \begin{bmatrix} B & \rightarrow \\ \rightarrow & \rightarrow \end{bmatrix}_{n\times n}.$$

Example:

$$\begin{bmatrix} \downarrow & \downarrow \\ \downarrow & \downarrow \\ \downarrow & \times \\ \downarrow & \downarrow \end{bmatrix}_{N\times n} \begin{bmatrix} \rightarrow & \rightarrow & \rightarrow & \rightarrow \\ \bigcirc & \bigcirc & \bigcirc & \bigcirc \end{bmatrix}_{n\times N} = \begin{bmatrix} \rightarrow & \rightarrow & \rightarrow & \rightarrow \\ \rightarrow & \rightarrow & \rightarrow & \rightarrow \\ \bigotimes & \bigotimes & \bigotimes & \bigotimes \\ \rightarrow & \rightarrow & \rightarrow & \rightarrow \end{bmatrix}_{N\times N}.$$

Algorithm 5.2

```
for (c=0;c<n;c++) {
   for (ii=IA[c];ii<IA[c+1];ii++) {
      i = JA[ii];
      for (jj=IB[c];jj<IB[c+1];jj++) {
         j = JB[jj];
         AB[i][j] += A[ii]*B[jj];
      }
   }
}
```

• Standard matrix-matrix multiplication:

$A_{n\times N}B_{N\times n}: \quad O(n^2)$

$$\begin{bmatrix} A & \rightarrow & \rightarrow & \rightarrow \\ \rightarrow & \rightarrow & \rightarrow & \rightarrow \end{bmatrix}_{n\times N} \begin{bmatrix} B & \downarrow \\ \downarrow & \downarrow \\ \downarrow & \downarrow \\ \downarrow & \downarrow \end{bmatrix}_{N\times n}.$$

Algorithm 5.3

```
for (i=0;i<n;i++) {
    for(rr=IA[i];rr<IA[i+1];rr++) {
        r = JA[rr];
        for (j=0;j<n;j++) {
            for (cc=IB[j];cc<IB[j+1];cc++) {
                c = JB[cc];
                if (r==c)
                    AB[i][j] += A[rr]*B[cc];
            }
        }
    }
}
```

5.5 Storage style and computing the terms involving A_{11} and C

We have detected a major flaw in the implementation of the WMHB method for nonsymmetric problems. We first discuss the blocks for the symmetric problem, and then return to the nonsymmetric case at the end of this section.

• $C_{N\times n}$:

The biggest subblock of A is A_{11}, we need to handle C^TA_{11} or $A_{11}C$ with $O(n)$ complexity. $O(n)$ complexity restriction also applies to the storage complexity. Then, the rule of thumb is: A matrix has to be stored with respect to the smaller dimension (i.e. n). We store $C_{N\times n}$ columnwise.

• C^TA_{11} or $A_{11}C$:

We want to recycle the existing code as much as possible. In the existing code, results of matrix-matrix products are always produced by an algorithm which is implemented rowwise. In other words, we produce matrix-matrix products by using the row variant of Algorithm 5.1. Here we want to make a choice between the computation of C^TA_{11} and $A_{11}C$. If we choose to compute $A_{11}C$ with $O(n)$ complexity, then A_{11} has to be stored columnwise, forcing us to use column variant of Algorithm 5.1. Due to code recycling, we decided that we could only use row variant of Algorithm 5.1. Therefore, $A_{11}C$ choice becomes impossible. Hence, we compute C^TA_{11} with $O(n)$ cost.

$C^TA_{11} \to \hat{A}_{21}$ by Algorithm 5.1

$$\begin{bmatrix} C^T & \to & \to & \to \\ \to & \to & \to & \to \end{bmatrix}_{n\times N} \begin{bmatrix} A_{11} & \to & \to & \to \\ \to & \to & \to & \to \\ \to & \to & \to & \to \\ \to & \to & \to & \to \end{bmatrix}_{N\times N}.$$

• $(C^TA_{11})C$:

$(C^TA_{11})C$ causes the following difficulty.

$(C^TA_{11})C \to \hat{A}_{22}$ by Algorithm 5.3,

$$\begin{bmatrix} (C^T A_{11}) & \rightarrow & \rightarrow & \rightarrow \\ \rightarrow & & \rightarrow & \rightarrow & \rightarrow \end{bmatrix}_{n\times N} \begin{bmatrix} C & \downarrow \\ \downarrow & \downarrow \\ \downarrow & \downarrow \\ \downarrow & \downarrow \end{bmatrix}_{N\times n},$$

can only be handled with Algorithm 5.3 with $O(n^2)$ logical cost, although algebraic cost is $O(n)$. The overall computational complexity is determined by the costlier of the algebraic and logical complexity. For thorough complexity discussion, we use subscripted variables $O(n_j)$ rather than $O(n)$. We wonder if $O(n_j^2)$ (or $O(n_j^p),\ p > 1$ in general) can give rise to an overall complexity of $O(N^s),\ s > 1$. In a typical adaptive refinement setting, n_j's are constants at each level which are independent of N, J. Moreover, one can assume that the number of newly introduced unknowns at every level is uniformly bounded;

$$1 \leq n_j \leq K, \quad j = 0, \ldots, J.$$

The following lemma proves that $O(n_j^p)$ is still acceptable, meaning that, we can maintain the overall computational complexity at $O(N)$. In very rough language, it can be said that $O(n^p)$ and $O(n)$ are almost the same complexities.

Lemma 5.1 *Assume* $1 \leq n_j \leq K, \quad j = 0, \ldots, J$, *where* K *is an absolute constant. Then a computational complexity of* $O(n_j^p),\ p > 1$ *per level gives rise to an overall complexity of* $O(N)$, *that is*

$$\sum_{j=0}^{J} c_j n_j^p \leq cN,$$

where c_j *is the complexity constant and* c *is an absolute constant.*

Proof. $J = \sum_{j=0}^{J} 1 \leq N = \sum_{j=0}^{J} n_j \leq KJ$, hence, $N \sim J$.

$$\sum_{j=0}^{J} c_j n_j^p \leq K^p \sum_{j=0}^{J} c_j \leq K^p \max_{j=0,\ldots,J} c_j \ J \leq cN.$$

□

5.5.1 Suboptimal logical complexity in the nonsymmetric problem

For the nonsymmetric problem, the stabilized block $\hat{A}_{12}$ has to be calculated because $\hat{A}_{12} \neq \hat{A}_{21}^T$. As in the below formula $\hat{A}_{12}$ contains the term $A_{11}C$;

$$\hat{A}_{12} = A_{11}C + A_{12}D + R^T A_{21}C + R^T A_{22}D.$$

The $A_{11}C$ creates a major computational obstacle. So far, we have decided to store A_{11} rowwise and C must be stored columnwise. Then $A_{11}C$ can only be handled by Algorithm 5.3, unfortunately, with a logical complexity of $O(Nn)$. Although the algebraic complexity remains $O(n)$ due to sparseness of A_{11} and C, we lose the overall optimal complexity because of the logical complexity. Then one attempts to use Algorithm 5.1 (i.e. the same storage style algorithm) to avoid the suboptimal logical complexity. According to the storage criterion, C must be stored columnwise. Algorithm 5.1 dictates same storage style for A_{11}. But A_{11} is already stored rowwise and to convert it into columnwise style takes $O(N)$ algebraic cost. As a result, the attempt to avoid $O(Nn)$ logical complexity fails. At this point we stick with the $A_{11}C$ computation by using Algorithm 5.3. Although the WMHB has optimal algebraic complexity for nonsymmetric problems, it suffers from a suboptimal logical complexity. For nonsymmetric PDEs, the WMHB loses the battle of optimal computational complexity to the suboptimal logical complexity. Since our test case is the PBE, we managed to protect ourselves from the suboptimal logical complexity by taking refuge in the safe haven of symmetric PDEs.

5.6 Computing rest of the terms in the symmetric problem

1. $C^T A_{21}^T$ by Algorithm 5.3

$$\begin{bmatrix} C^T & \rightarrow & \rightarrow & \rightarrow \\ \rightarrow & \rightarrow & \rightarrow & \rightarrow \end{bmatrix}_{n\times N} \begin{bmatrix} A_{21}^T & \downarrow \\ \downarrow & \downarrow \\ \downarrow & \downarrow \\ \downarrow & \downarrow \end{bmatrix}_{N\times n}$$

$(C^T A_{21}^T)R \rightarrow \hat{A}_{21}$ by Algorithm 5.1

$$\begin{bmatrix} (C^T A_{21}^T) & \rightarrow \\ \rightarrow & \rightarrow \end{bmatrix}_{n\times n} \begin{bmatrix} R & \rightarrow & \rightarrow & \rightarrow \\ \rightarrow & \rightarrow & \rightarrow & \rightarrow \end{bmatrix}_{n\times N}$$

$(C^T A_{21}^T)D \rightarrow \hat{A}_{22}$ by Algorithm 5.1

$$\begin{bmatrix} (C^T A_{21}^T) & \rightarrow \\ \rightarrow & \rightarrow \end{bmatrix}_{n\times n} \begin{bmatrix} D & \rightarrow \\ \rightarrow & \rightarrow \end{bmatrix}_{n\times n}.$$

2. $D^T A_{21} \rightarrow \hat{A}_{21}$ by Algorithm 5.2

$$\begin{bmatrix} D^T & \downarrow \\ \downarrow & \downarrow \end{bmatrix}_{n\times n} \begin{bmatrix} A_{21} & \rightarrow & \rightarrow & \rightarrow \\ \rightarrow & \rightarrow & \rightarrow & \rightarrow \end{bmatrix}_{n\times N}$$

$(D^T A_{21})C \rightarrow \hat{A}_{22}$ by Algorithm 5.3

$$\begin{bmatrix} (D^T A_{21}) & \rightarrow & \rightarrow & \rightarrow \\ \rightarrow & \rightarrow & \rightarrow & \rightarrow \end{bmatrix}_{n\times N} \begin{bmatrix} C & \downarrow \\ \downarrow & \downarrow \\ \downarrow & \downarrow \\ \downarrow & \downarrow \end{bmatrix}_{N\times n}.$$

3. $D^T A_{22}$ by Algorithm 5.2

$$\begin{bmatrix} D^T & \downarrow \\ \downarrow & \downarrow \end{bmatrix}_{n\times n} \begin{bmatrix} A_{22} & \rightarrow \\ \rightarrow & \rightarrow \end{bmatrix}_{n\times n}$$

$(D^T A_{22})R \to \hat{A}_{21}$ by Algorithm 5.1

$$\begin{bmatrix} (D^T A_{22}) & \to \\ \to & \to \end{bmatrix}_{n\times n} \begin{bmatrix} R & \to & \to & \to \\ \to & \to & \to & \to \end{bmatrix}_{n\times N}$$

$(D^T A_{22})D \to \hat{A}_{22}$ by Algorithm 5.1

$$\begin{bmatrix} (D^T A_{22}) & \to \\ \to & \to \end{bmatrix}_{n\times n} \begin{bmatrix} D & \to \\ \to & \to \end{bmatrix}_{n\times n}.$$

4. $R^T A_{21} \to \hat{A}_{11}$ by Algorithm 5.2

$$\begin{bmatrix} R^T & \downarrow \\ \downarrow & \downarrow \\ \downarrow & \downarrow \\ \downarrow & \downarrow \end{bmatrix}_{N\times n} \begin{bmatrix} A_{21} & \to & \to & \to \\ \to & \to & \to & \to \end{bmatrix}_{n\times N}.$$

$(R^T A_{21})^T \to \hat{A}_{11}$

5. $A_{22}R$ by Algorithm 5.1 optional, can also do $(R^T A_{22})R$

$$\begin{bmatrix} A_{22} & \to \\ \to & \to \end{bmatrix}_{n\times n} \begin{bmatrix} R & \to & \to & \to \\ \to & \to & \to & \to \end{bmatrix}_{n\times N}$$

$R^T(A_{22}R) \to \hat{A}_{11}$ by Algorithm 5.2

$$\begin{bmatrix} R^T & \downarrow \\ \downarrow & \downarrow \\ \downarrow & \downarrow \\ \downarrow & \downarrow \end{bmatrix}_{N\times n} \begin{bmatrix} (A_{22}R) & \to & \to & \to \\ \to & \to & \to & \to \end{bmatrix}_{n\times N}.$$

5.7 Alternative implementation for the symmetric problem

In a nonsymmetric problem, the sparsity pattern of A is symmetric, whereas, the value of the nonzero entries are not. In a symmetric problem A is truly symmetric. The subblocks A_{11} and A_{22} are symmetric. One can represent $A = U + D + U^T$. The symmetry is also reflected to the stabilized A, namely,

$$\hat{A} = Y^T A Y = Y^T(U + D + U^T)Y = Y^T U Y + Y^T D Y + Y^T U^T Y = \hat{A}^U + \hat{A}^D + \hat{A}^{U^T}.$$

Note that $\hat{A}^{U^T} = Y^T U^T Y = \left(\hat{A}^U\right)^T$. So, we do not need to compute $\left(\hat{A}^U\right)^T$. Since $\text{size}(A_{ii}^{(j)}) = 2\text{size}(U_{ii}^{(j)}) + \text{size}(D_{ii}^{(j)})$ where $i = 1, 2$ and also another symmetry utilization for $\hat{A}^D$ will save almost a 50% of the computation corresponding to the diagonal blocks. There is no computational saving on the offdiagonal blocks $A_{12}^{(j)}$, $A_{21}^{(j)}$. Let us clarify the above algebraic complexity saving with one example.

$$\hat{A} = \hat{A}^U + \hat{A}^D + \hat{A}^{U^T},$$

$$\begin{bmatrix} 2\alpha^U + \alpha^D & a^U + s^D & b^U + X^U + 1 & c^U + Z^U + 2 \\ a^U + s^D & 2\beta^U + \beta^D & d^U + Y^U & e^U \\ b^U + X^U + 1 & d^U + Y^U & 2\gamma^U + \gamma^D & f^U + t^D \\ c^U + Z^U + 2 & e^U & f^U + t^D & 2\theta^U + \theta^D \end{bmatrix} =$$

$$\begin{bmatrix} \alpha^U & a^U & b^U & c^U \\ 0 & \beta^U & d^U & e^U \\ X^U & Y^U & \gamma^U & f^U \\ Z^U & 0 & 0 & \theta^U \end{bmatrix} + \begin{bmatrix} \alpha^D & s^D & 1 & 2 \\ s^D & \beta^D & 0 & 0 \\ 1 & 0 & \gamma^D & t^D \\ 2 & 0 & t^D & \theta^D \end{bmatrix} + \begin{bmatrix} \alpha^U & 0 & X^U & Z^U \\ a^U & \beta^U & Y^U & 0 \\ b^U & d^U & \gamma^U & 0 \\ c^U & e^U & f^U & \theta^U \end{bmatrix}.$$

Now, with this motivation, the explicit matrix representation of the stabilized blocks is as follows.

$$\begin{array}{ccccc}
\hat{A}^U_{WMHB} & = & Y^T & U & Y \\
\begin{bmatrix} \hat{A}^U_{11} & \hat{A}^U_{12} \\ \hat{A}^U_{21} & \hat{A}^U_{22} \end{bmatrix} & = & \begin{bmatrix} I & R^T \\ C^T & D^T \end{bmatrix} & \begin{bmatrix} U_{11} & A_{12} \\ 0 & U_{22} \end{bmatrix} & \begin{bmatrix} I & C \\ R & D \end{bmatrix} \\
\hat{A}^D_{WMHB} & = & Y^T & D & Y \\
\begin{bmatrix} \hat{A}^D_{11} & \hat{A}^D_{12} \\ \hat{A}^D_{21} & \hat{A}^D_{22} \end{bmatrix} & = & \begin{bmatrix} I & R^T \\ C^T & D^T \end{bmatrix} & \begin{bmatrix} D_{11} & 0 \\ 0 & D_{22} \end{bmatrix} & \begin{bmatrix} I & C \\ R & D \end{bmatrix} \\
\hat{A}^{U^T}_{WMHB} & = & Y^T & U^T & Y \\
\begin{bmatrix} \hat{A}^{U^T}_{11} & \hat{A}^{U^T}_{12} \\ \hat{A}^{U^T}_{21} & \hat{A}^{U^T}_{22} \end{bmatrix} & = & \begin{bmatrix} I & R^T \\ C^T & D^T \end{bmatrix} & \begin{bmatrix} U^T_{11} & 0 \\ A^T_{12} & U^T_{22} \end{bmatrix} & \begin{bmatrix} I & C \\ R & D \end{bmatrix}
\end{array}$$

$$\begin{array}{lllllll}
\hat{A}^U_{22} & = & (C^T U_{11})^* C & + & (C^T A_{12})^* D & + & (D^T U_{22})^* D \\
\hat{A}^U_{21} & = & (C^T U_{11}) & + & (C^T A_{12}) R & + & (D^T U_{22}) R \\
\hat{A}^U_{12} & = & U_{11} C & + & A_{12} D & + & (R^T U_{22})^* D \\
\hat{A}^U_{11} & = & U_{11} & + & A_{12} R & + & (R^T U_{22}) R
\end{array}$$

$$\begin{array}{lllllll}
\hat{A}^D_{22} & = & (C^T D_{11})^* C & + & D^T D_{22} D & & \\
\hat{A}^D_{12} & = & (C^T D_{11})^T & + & (R^T D_{22})^* D & & \\
\hat{A}^D_{11} & = & D_{11} & + & (R^T D_{22}) R & & \\
\hat{A}^D_{21} & = & (C^T D_{11}) & + & (D^T D_{22}) R & = & (\hat{A}^D_{12})^T
\end{array}$$

$$\begin{array}{lllllllll}
\hat{A}^{U^T}_{22} & = & (C^T U^T_{11})^* C & + & (D^T A^T_{12})^* C & + & (D^T U^T_{22})^* D & = & (\hat{A}^U_{22})^T \\
\hat{A}^{U^T}_{21} & = & (C^T U^T_{11}) & + & (D^T A^T_{12}) & + & (D^T U^T_{22}) R & = & (\hat{A}^U_{12})^T \\
\hat{A}^{U^T}_{12} & = & U^T_{11} C & + & (R^T A^T_{12})^* C & + & (R^T U^T_{22})^* D & = & (\hat{A}^U_{21})^T \\
\hat{A}^{U^T}_{11} & = & U^T_{11} & + & (R^T A^T_{12}) & + & (R^T U^T_{22}) R & = & (\hat{A}^U_{11})^T.
\end{array}$$

We do not need to calculate $\hat{A}^D_{21}, \hat{A}^{U^T}_{11}, \hat{A}^{U^T}_{12} \hat{A}^{U^T}_{21}, \hat{A}^{U^T}_{22}$ for symmetric problems, due to symmetry in the block representation. All the saving mentioned so far, is related to the algebraic complexity. As discussed in the nonsymmetric problem, the calculation of $A_{11}C$ in the definition of $\hat{A}_{12}$ has $O(Nn)$ logical complexity. The same problem repeats for the term $U_{11}C$ in the definition of $\hat{A}^U_{12}$. Although this approach offers a dramatic saving in algebraic complexity (up to almost 50%), it suffers from an $O(Nn)$ logical complexity even for symmetric problems. In a symmetric problem, if one favors relatively less algebraic cost, then symmetry utilization is the preferred way. On the other hand, the computation time is determined by the bigger complexity, i.e. logical complexity. Then, this alternative method will not be favorable for the WMHB implementation of the symmetric problems. However, it is favorable for the HB implementation of both symmetric and nonsymmetric problems. This method becomes advantageous because HB blocks do not contain matrix multiplications with the A_{11} block.

5.7.1 Implementation draft

$\hat{A}^U$ is not symmetric, is mainly upper triangular but not strictly upper triangular. Its lower triangular and diagonal pieces contain some nonzero entries. The entries contributed to $\hat{A}$ from the lower triangular and diagonal piece of $\hat{A}^U$ are identical to the ones contributed from the upper triangular and diagonal piece of $\hat{A}^{U^T}$. Hence, as we calculate the lower triangular piece of $\hat{A}^U$, we contribute the nonzeros to the upper triangular piece of $\hat{A}$ by transposing the original locations (i.e. contribute with respect to diagonal). For short, $\hat{A}^U$ contributes to $off\hat{U}$ and $\hat{D}$ of $\hat{A}$, and the remaining entries get contributed by $\hat{A}^D$ later on.

$\hat{A}^D$ is symmetric and mainly diagonal. So, we only contribute the upper triangular piece of $\hat{A}^D$ to $off\hat{U}$ and diagonal entries to $\hat{D}$. Eventually, we obtain the upper triangular piece of $\hat{A}$, which is equal to $off\hat{U} + \hat{D}$.

On the finest level, we are given the stiffness matrix in symmetric Bank-YSMP format. So, we can either read this matrix into a linked list or do all the finest level calculations with this matrix. Since we are willing to work with a linked list for coarser levels, let us assume that the finest level stiffness matrix is in linked list format, for the sake of simplicity.

We will store two linked lists: $lnkUin$, and $lnkUout$. We store the upper triangle of the stiffness matrix in $lnkUin$. After each level, we accumulate the $\hat{A}_{11}$ in $lnkUin$. This is actually an implementation restriction, that is, the number of entries in $\hat{A}_{11}$ is $O(N)$. Meaning that, even moving these entries will be of cost $O(N)$. So, we have to leave those entries where they are, and contribute the new entries to that location. Since $\hat{A}_{11}$ was read in $lnkUin$, we are going to contribute to this linked list, and after the contribution is done, this block is going to be the new input for the next coarse level.

$lnkUout$ contains all the stabilized hierarchical blocks on each level. Since the size of the diagonal is always known (i.e. the number of rows/cols of the stiffness matrix), we just allocate a fixed size array, $linkD$. In the end, $lnkUin$ will contain the coarsest level stiffness matrix, and $lnkUout$ will contain all the stabilized hierarchical blocks. We peel off $lnkUout$ as we go down the coarsest level in the hierarchy.

1. $C^T U_{11} \to tmp$ by Algorithm 5.1
   $tmp \overset{U21}{\to} lnkUout$ by Algorithm $readIn$ & $(tmp)C \overset{U22}{\to} lnkUout + diag$ by Algorithm 5.1.

2. $C^T A_{12} \to tmp$ by Algorithm 5.1
   $(tmp)R \overset{U21}{\to} lnkUout$ by Algorithm 5.1 & $(tmp)D \overset{U22}{\to} lnkUout$ by Algorithm 5.1.

3. $D^T U_{22} \to tmp$ by Algorithm 5.2
   $(tmp)R \overset{U21}{\to} lnkUout$ by Algorithm 5.1 & $(tmp)D \overset{U22}{\to} lnkUout$ by Algorithm 5.1.

4. $U_{11}C \overset{U12}{\to} lnkUout$ by Algorithm 5.3.

5. $A_{12}D \overset{U12}{\to} lnkUout$ by Algorithm 5.1.

6. $R^T U_{22} \to tmp$ by Algorithm 5.2
   $(tmp)D \overset{U12}{\to} lnkUout$ by Algorithm 5.1 & $(tmp)R \overset{U11}{\to} lnkUin$ by Algorithm 5.1.

7. $C^T D_{11} \to tmp$ by Algorithm *diag* or by Algorithm 5.1
   $(tmp)^T \overset{D12}{\to} lnkUout$ by Algorithm *readIn* & $(tmp)C \overset{D22}{\to} lnkUout, lnkD$ by Algorithm 5.2.

8. $R^T D_{22} \to tmp$ by Algorithm 5.1
   $(tmp)D \overset{D12}{\to} lnkUout$ by Algorithm 5.1 & $(tmp)R \overset{D11}{\to} lnkUin, lnkD$ by Algorithm 5.1.

9. $D^T D_{22} \to tmp$ by Algorithm 5.1
   $(tmp)D \overset{D22}{\to} lnkUout, lnkD$ by Algorithm 5.2.

$\overset{U21}{\to}$: reflect the lower triangle with respect to diagonal.
$\overset{U22}{\to}$: reflect the lower triangle with respect to diagonal + double the diagonal.
$\overset{U12}{\to}$: leave it as it is.
$\overset{U11}{\to}$: reflect the lower triangle with respect to diagonal + double the diagonal.
$\overset{D12}{\to}$: leave it as it is.
$\overset{D22}{\to}$: reflect the lower triangle with respect to diagonal.
$\overset{D22}{\to}$: reflect the lower triangle with respect to diagonal.
$\overset{D11}{\to}$: reflect the lower triangle with respect to diagonal.

5.8 Related software

Our research software is FEtk. The dominating software contribution to our research is the involvement of FEtk. It provides the backbone and implementation basis for our research. It is an evolving collection of parallel adaptive multilevel finite element software tools for solving PDEs and integral equations. FEtk is designed, developed, and maintained by the Holst Research Group at UC San Diego, with contributions from a number of colleagues at other institutions. Some of the FEtk libraries contain Objective-C, C++, Java, FORTRAN, and MATLAB code, but it is written primarily in an object-oriented form of C called Clean OO C.

Figure 5.1: Sample 3D adaptive mesh generated using FEtk.

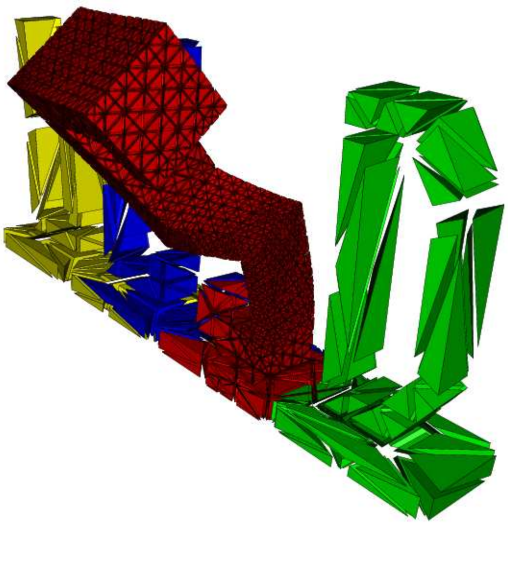

Figure 5.2: Sample mesh for a nonlinear elasticity problem solved using parallel processing and adaptivity, generated using FEtk.

(Images courtesy of Mike Holst.)

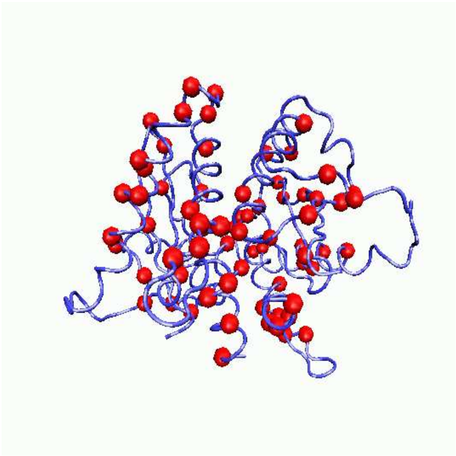

Figure 5.3: Ribbon representation of the HIV integrase.

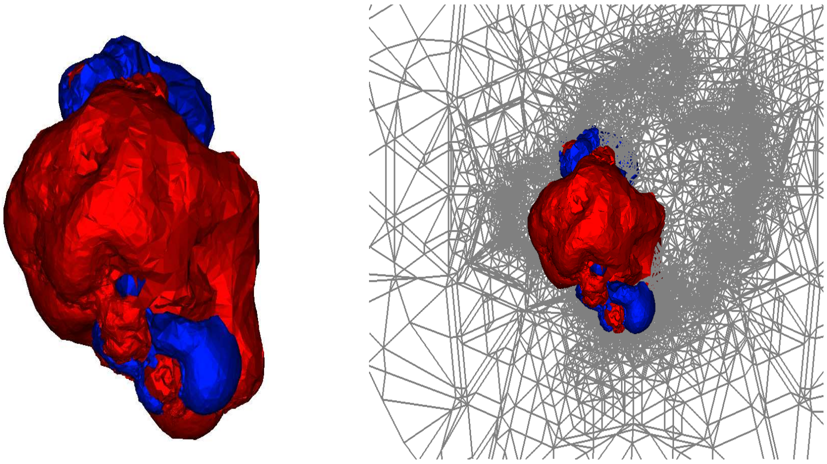

Figure 5.4: On the left, potential contours from solution to linear PBE for HIV integrase are shown (red $=-0.15$, blue$=+0.15$). On the right, a slice through the finite element mesh.

(Images courtesy of Nathan Baker.)

FEtk contains and is centered around a small self-contained adaptive multilevel finite element kernel package called the Manifold Code (MC). MC is written entirely in Clean OO C, and as a result is very portable, efficient, and robust. MC is copylefted and distributed freely to the scientific community in source form under the GNU General Public License.

The HB solvers are implemented as class libraries in the MCX (Manifold Code eXtension) component of FEtk, and prototypes are tested using the MCLab (Manifold Code in MATLAB) component of FEtk.

5.9 Numerical experiments

Below, one can observe the complexity of various methods and corresponding computation times for a fictitious $3D$ problem: *1-Giga flops* sequential computer with unlimited memory; 1000 mesh points in each direction,(i.e. $N = (10^3)^3$); complexity constants taken to be $\equiv 1$ for simplicity.

Table 5.1: Complexity of various methods and corresponding computation times for a fictitious $3D$ problem.

Method	RAM	Computation	Time (sec)	Approx. Time
Dense GE	$O(N^3)$	$O(N^3)$	$\frac{(10^9)^3}{10^9} = 10^{18}$	$\approx 31 \times 10^9$ yrs
Sparse GE	$O(N^2)$	$O(N^2)$	$\frac{(10^9)^2}{10^9} = 10^9$	≈ 31 yrs
Gauss-Seidel	$O(N)$	$O(N^{1.67} \ \log N)$	$\frac{(10^9)^{1.67} \ \log 10^9}{10^9}$	≈ 240 days
Conj. Grad.	$O(N)$	$O(N^{1.33} \ \log N)$	$\frac{(10^9)^{1.33} \ \log 10^9}{10^9}$	≈ 5.6 hours
HBMG	$O(N)$	$O(N^{1.33})$	$\frac{(10^9)^{1.33}}{10^9}$	≈ 0.27 hours
HBMG-PCG	$O(N)$	$O(N^{1.16})$	$\frac{(10^9)^{1.16}}{10^9}$	≈ 27.5 sec
WMHB	$O(N)$	$O(N)$	$\frac{(10^9)^1}{10^9}$	≈ 1 second
MG/DD (unif. mesh)	$O(N)$	$O(N)$	$\frac{(10^9)^1}{10^9}$	≈ 1 second

5.9.1 MCLab numerical experiments

Test problem:

$$
\begin{aligned}
-\nabla \cdot (p\ \nabla u) + q\ u &= f, \quad x \in \Omega \subset \mathbb{R}^2, \\
n \cdot (p\ \nabla u) &= g, \quad \text{on } \Gamma_N, \\
u &= 0, \quad \text{on } \Gamma_D,
\end{aligned}
$$

where

$$
p = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \text{ and } q = 1.
$$

The source term f is constructed so that the true solution is $u = \sin \pi x \sin \pi y$.

• Domain I

$\Omega = [0,1] \times [0,1]$,

$\Gamma_D = \{(x,y) : 0 \le x \le 1, y = 0\} \cup \{(x,y) : 0 \le x \le 1, y = 1\} \cup \{(x,y) : x = 0, 0 \le y \le 1\} \cup \{(x,y) : x = 1, 0 \le y \le 1\}$.

• Domain II

$\Omega = [0,1] \times [0,1]$,

$\Gamma_N = \{(x,y) : x = 0, 0 < y < 1\} \cup \{(x,y) : x = 1, 0 < y < 1\}$,

$\Gamma_D = \{(x,y) : 0 \le x \le 1, y = 0\} \cup \{(x,y) : 0 \le x \le 1, y = 1\}$.

• Domain III

$\Omega = [0,1] \times [0,1]$,

$\Gamma_N = \{(x,y) : 0 \le x \le 1, y = 0\} \cup \{(x,y) : 0 \le x \le 1, y = 1\} \cup \{(x,y) : x = 0, 0 \le y \le 1\} \cup \{(x,y) : x = 1, 0 \le y \le 1\}$.

Stopping criterion: $\|\text{error}\|_A < 10^{-7}$.

Table 5.2: MCLab iteration counts for various methods, red-green refinement on domain I.

Levels	1	2	3	4	5	6	7	8	9
WMHB-sGS	1	1	9	17	18	22	24	24	25
WMHB-Jac	1	1	9	19	22	25	26	27	29
WMHB-exact	1	1	9	17	19	26	28	28	30
HBMG	1	1	6	14	15	20	23	25	27
No. unknowns	4	7	10	25	41	83	159	296	588

Table 5.3: MCLab iteration counts for various methods, red-green refinement on domain II.

Levels	1	2	3	4	5	6	7	8	9
WMHB-sGS	1	8	9	24	24	25	25	25	25
WMHB-Jac	1	8	10	20	22	26	29	30	32
WMHB-exact	1	8	9	26	26	27	27	27	30
HBMG	1	4	10	19	24	36	49	63	79
No. unknowns	4	7	10	25	41	83	159	296	588

Table 5.4: MCLab iteration counts for various methods, red-green refinement on domain III.

Levels	1	2	3	4	5	6	7	8	9
WMHB-sGS	1	17	21	21	26	27	28	28	28
WMHB-Jac	1	25	31	33	38	39	42	42	43
WMHB-exact	1	15	29	19	25	57	58	58	58
HBMG	1	23	29	39	49	51	59	72	88
No. unknowns	4	7	10	25	41	83	159	296	588

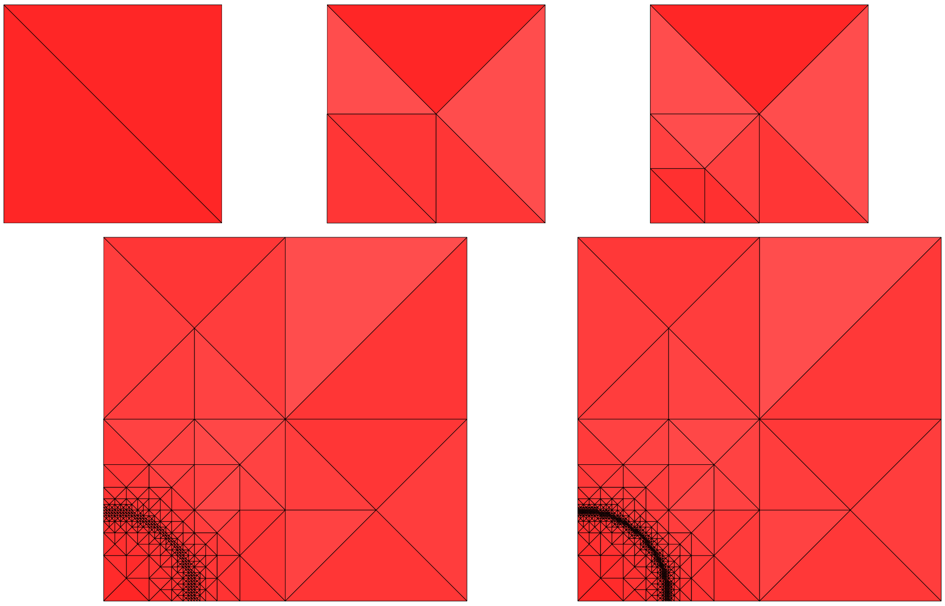

Figure 5.5: 2D red-green refined mesh.

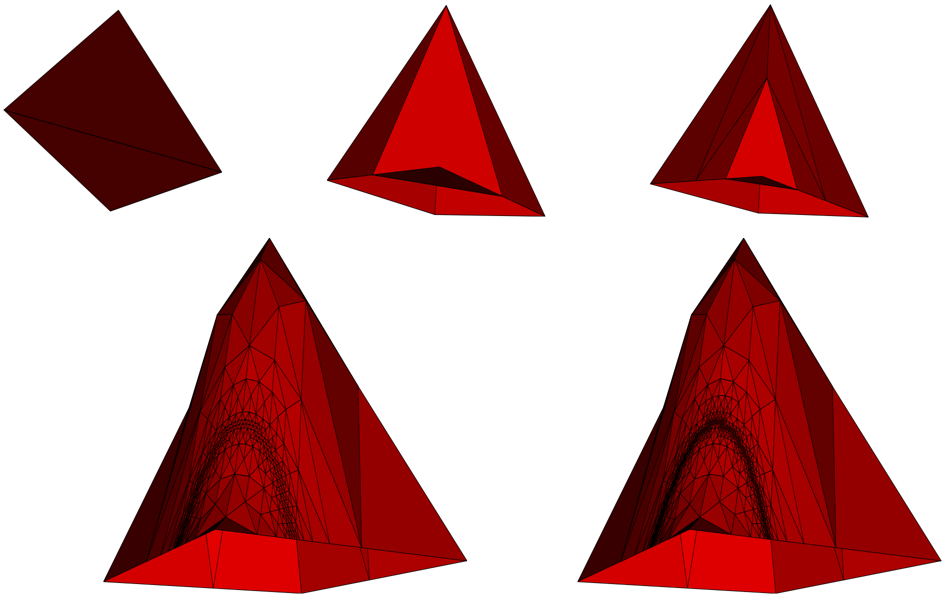

Figure 5.6: Solutions corresponding to the 2D red-green refined mesh.

Table 5.5: MCLab iteration counts for WMHB-sGS, adaptive longest-edge bisection, level 1-8.

Levels	1	2	3	4	5	6	7	8
Domain I	1	1	1	2	2	13	15	16
Domain II	1	1	16	17	17	31	29	29
Domain III	1	41	26	21	22	25	23	22
No. unknowns	4	5	7	8	10	19	25	30

Table 5.6: MCLab iteration counts for WMHB-sGS, adaptive longest-edge bisection, level 9-16.

Levels	9	10	11	12	13	14	15	16
Domain I	16	30	27	28	28	26	26	29
Domain II	29	30	29	31	31	29	29	29
Domain III	22	30	28	27	29	28	29	28
No. unknowns	38	54	72	99	137	184	254	360

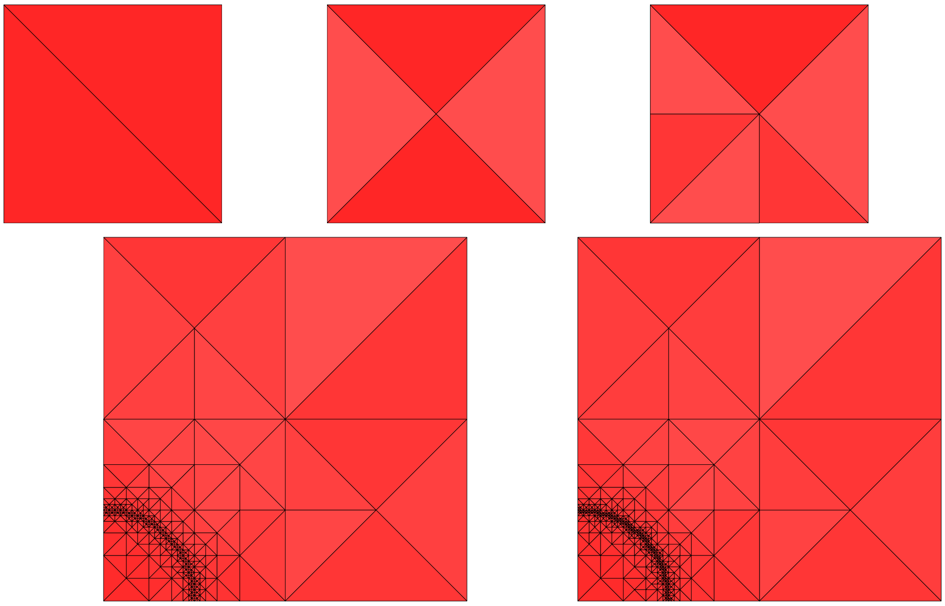

Figure 5.7: 2D longest-edge bisected mesh.

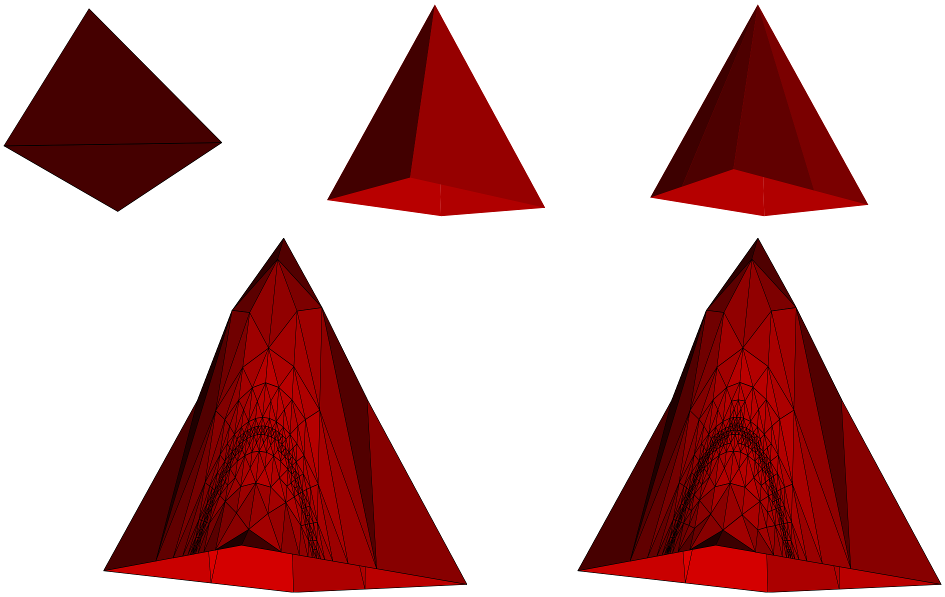

Figure 5.8: Solutions corresponding to the 2D longest-edge bisected mesh

Table 5.7: MCLab iteration counts for various methods, red-green refinement, $N_1 = 16$, domain II.

Levels	4	5	6	7	8	9	10
HBMG	26	30	35	45	56	70	85
WMHB-sGS	17	28	28	28	27	27	27
WMHB-Jac	21	29	31	33	35	35	35
HBMG-PCG	10	12	14	15	15	16	17
WMHB-sGS-PCG	9	11	11	11	12	12	12
No. unknowns	55	117	219	429	835	1655	3301

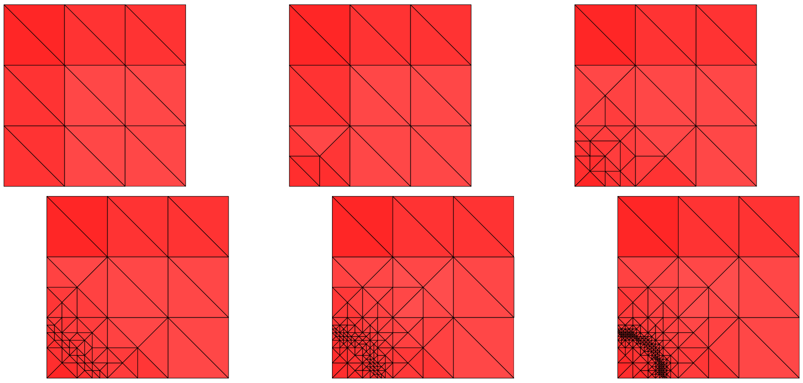

Figure 5.9: Adaptive mesh, $N_1 = 16$, domain II.

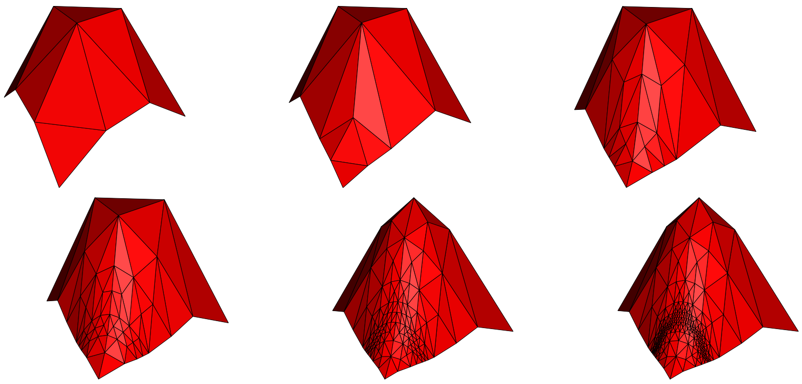

Figure 5.10: Solutions on the adaptive mesh, $N_1 = 16$, domain II.

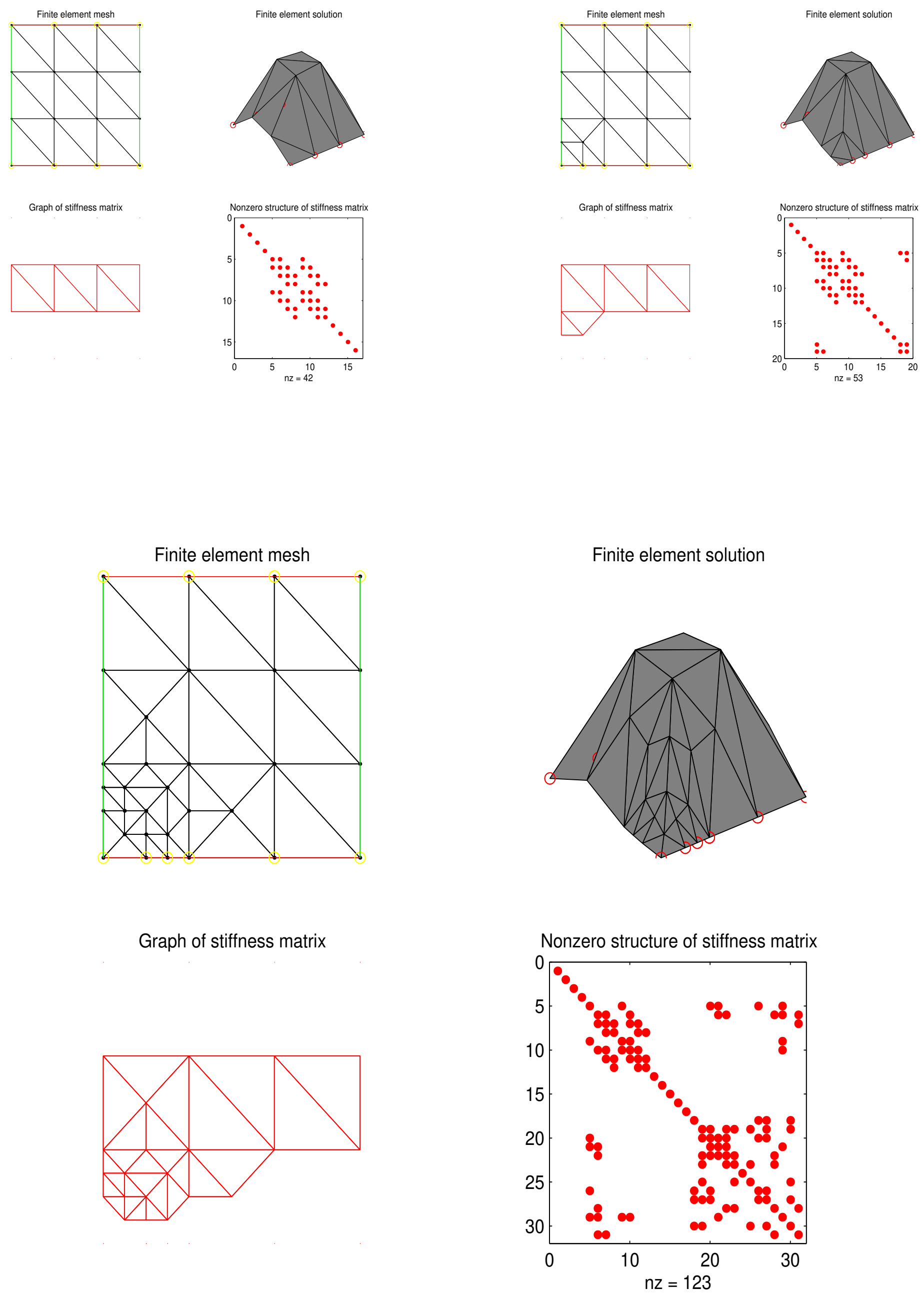

Figure 5.11: Dialog box from MCLab.

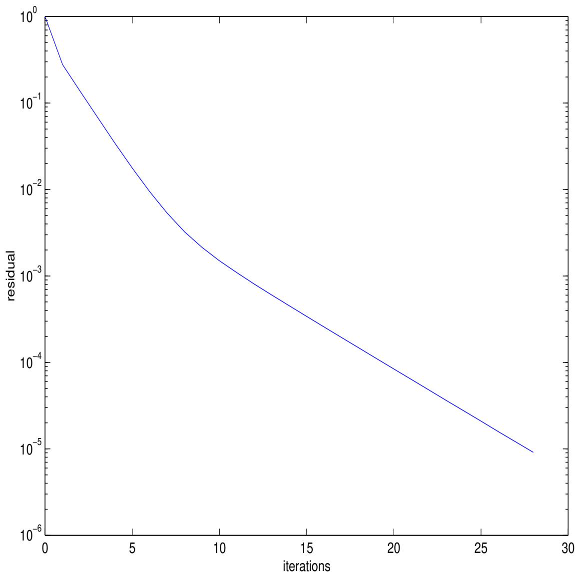

Figure 5.12: Residual reduction vs. iteration count.

5.9.2 FEtk numerical experiments

Test problem:

$$\begin{aligned} -\nabla \cdot (\epsilon \nabla u) + \kappa^2 u &= f, \quad x \in \Omega \subset \mathbb{R}^2, \\ n \cdot (\epsilon \nabla u) &= g, \quad \text{on } \Gamma_N, \\ u &= 0, \quad \text{on } \Gamma_D. \end{aligned}$$

The source term f is constructed so that the solution has local singular behavior:

$$u(x) = \frac{.05}{(.9\|c - x\|_{l_2})^{21}},$$

where

$$c = \begin{bmatrix} \frac{1}{4} \\ \frac{1}{4} \end{bmatrix}.$$

Table 5.8: FEtk iteration counts for various methods (red-green refinement).

Levels	1	2	3	4	5	6	7
HBMG	14	23	28	24	31	36	36
MG	7	8	7	8	9	9	9
No. unknowns	9	25	72	200	593	1699	3907

Table 5.9: FEtk iteration counts for various methods (adaptive longest-edge bisection refinement).

Levels	5	6	7	8	9	10	11	12
HBMG	21	24	22	27	29	30	35	32
MG	5	5	5	5	5	5	5	5
No. unknowns	39	67	105	169	295	504	853	1411

Table 5.10: FEtk iteration counts for various methods (uniform quadrasection refinement).

Levels	1	2	3	4	5	6	7	8
HBMG	14	20	32	43	53	63	73	80
MG	7	7	8	9	9	9	8	8
No. unknowns	9	25	81	289	1089	4225	16,641	66,049

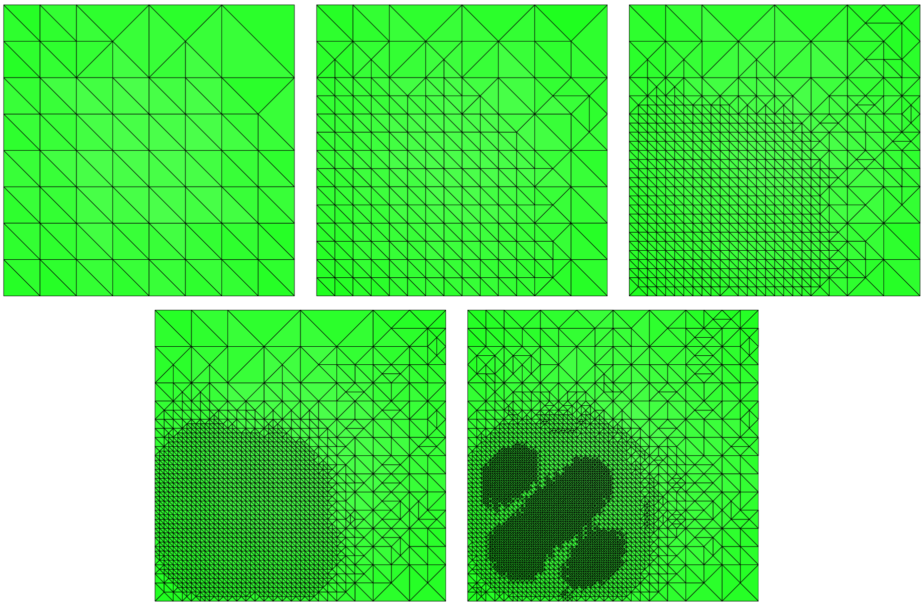

Figure 5.13: FEtk 2D red-green subdivision driven by a residual-based *a posteriori* error estimation.

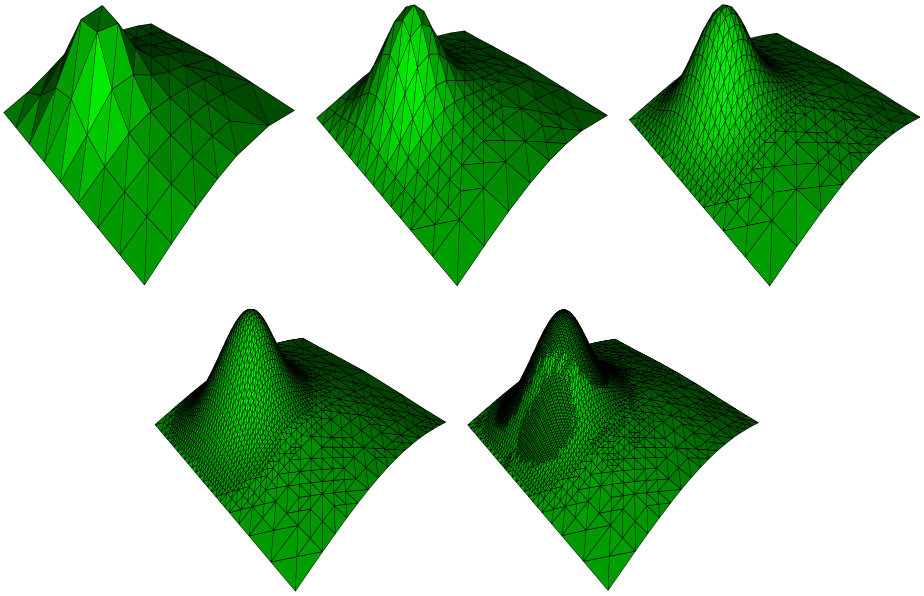

Figure 5.14: Corresponding HBMG solutions in FEtk for 2D red-green subdivision.

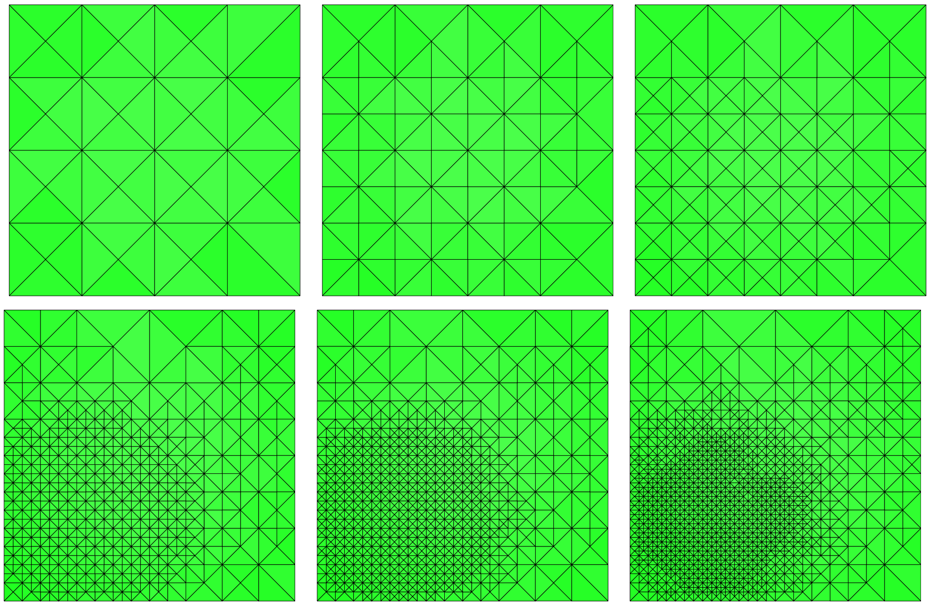

Figure 5.15: FEtk 2D longest-edge bisection subdivision driven by a residual-based *a posteriori* error estimation.

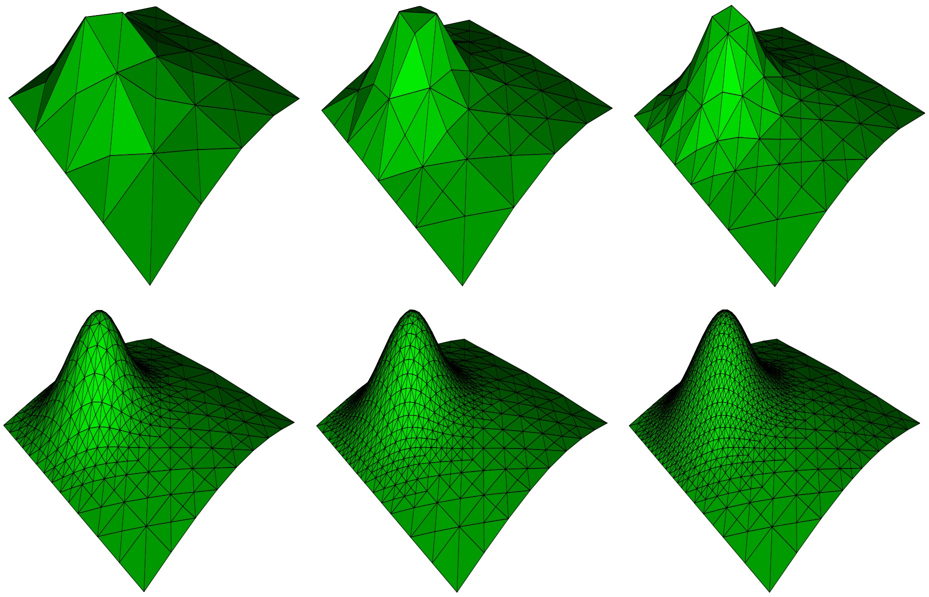

Figure 5.16: Corresponding HBMG solutions in FEtk for 2D longest-edge bisection subdivision.

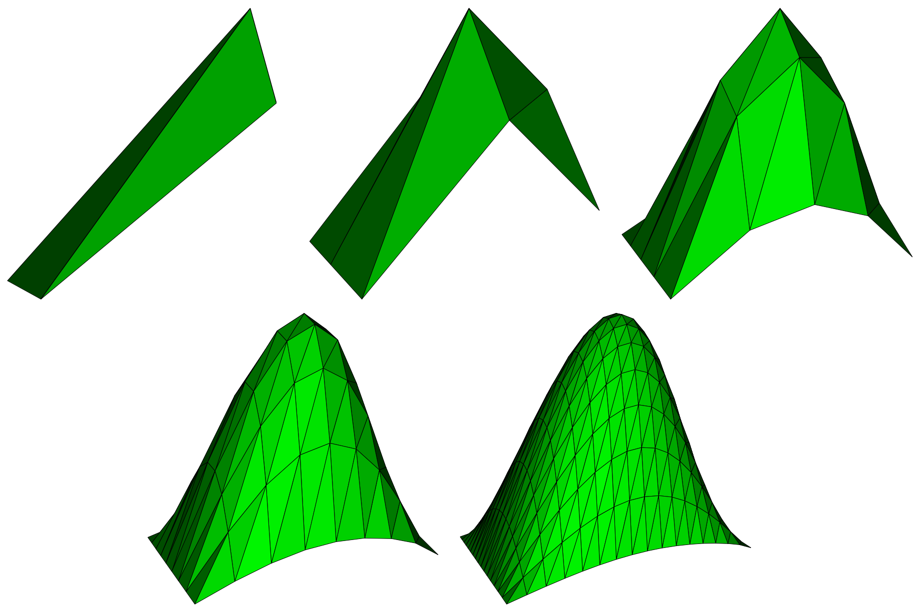

Figure 5.17: HBMG solutions in FEtk for 2D uniform quadrasection subdivision.

Remark 5.1 In the WMHB experiments, we used the symmetric Gauss-Seidel (sGS) and Jacobi (Jac) methods to approximate the inverse of the mass matrix. When the iteration counts for the exact inverse and sGS approximation were compared, we found the sGS method to be more efficient. We speculate that the sGS approximation provides a more stable wavelet modified hierarchical basis. Understanding the exact cause of this improvement in efficiency is a potential area for further research. One should note that, the sparsity of the change-of-basis matrix is destroyed by sGS. Hence, the sparsity can only be preserved if an appropriate polynomial approximation of the inverse can be found, as it is discussed in [38]. The sparsity of the two approaches is compared in Figures 5.18 and 5.19.

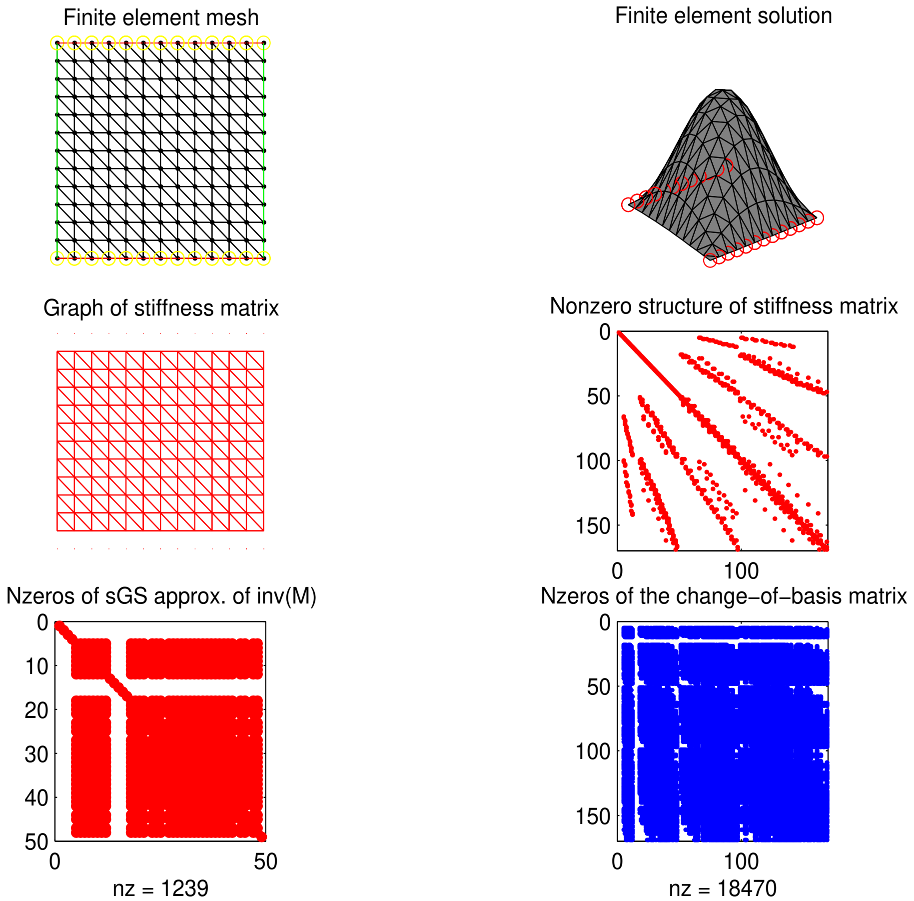

Figure 5.18: Dense structure of the inverse of the coarse mass matrix and the change-of-basis matrix, by using the symmetric Gauss-Seidel approximation (sGS) on a uniform mesh.

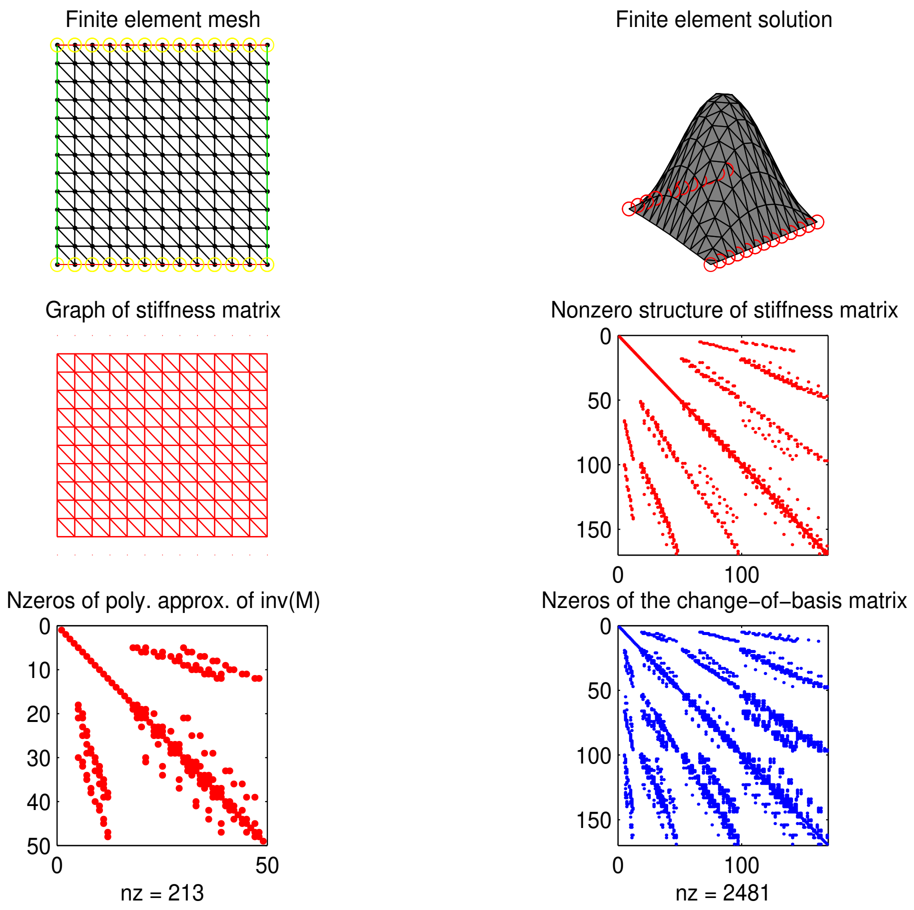

Figure 5.19: Sparse structure of the inverse of the coarse mass matrix and the change-of-basis matrix, by using a polynomial approximation on a uniform mesh.

Bibliography

[1] A. Adler. On the bisection method for triangles. *Math. Comp.*, 40:571–574, 1983.

[2] D. Arnold, A. Mukherjee, and L. Pouly. Locally adapted tetrahedral meshes using bisection. *SIAM J. Sci. Comput.*, 22:431–448., 2000.

[3] N. Baker, M. Holst, and F. Wang. Adaptive multilevel finite element solution of the Poisson-Boltzmann equation II: fast solutions for biomolecular systems based on refinement at solvent accessible surfaces. *J. Comput. Chem.*, 21 Issue 15:1319–1342., 2000.

[4] R. E. Bank. Hierarchical basis and the finite element method. *Acta Numerica*, pages 1–43, 1996.

[5] R. E. Bank. *PLTMG: A Software Package for Solving Elliptic Partial Differential Equations User's Guide 8.0.* Software, Environments, Tools. SIAM, Philadelphia, PA, 1998.

[6] R. E. Bank and T. Dupont. Analysis of a two-level scheme for solving finite element equations. Technical report, Center for Numerical Analysis, University of Texas at Austin, 1980. CNA–159.

[7] R. E. Bank, T. Dupont, and H. Yserentant. The hierarchical basis multigrid method. *Numer. Math.*, 52:427–458, 1988.

[8] E. Bänsch. Local mesh refinement in 2 and 3 dimensions. *Impact of Comp. in Sci. Engrg.*, 3:181–191, 1991.

[9] J. Bey. Tetrahedral grid refinement. *Computing*, 55:271–288., 1995.

[10] F. Bornemann and H. Yserentant. A basic norm equivalence for the theory of multilevel methods. *Numer. Math.*, 64:455–476, 1993.

[11] F. A. Bornemann. Interpolation Spaces and Optimal Multilevel Preconditioner. In D. Keyes and J. Xu, editors, *Proceedings of the 7th International Conference on Domain Decomposition Methods*, pages 3–8, Providence, 1994. CONM Series of the AMS, MR 95j:65155; Zbl. 818.65107.

[12] J. H. Bramble, J. E. Pasciak, and O. Steinbach. On the stability of the L^2 projection in $H^1(\Omega)$. *Math. Comp.*, 2000. accepted.

[13] J. H. Bramble, J. E. Pasciak, and J. Xu. Parallel multilevel preconditioners. *Math. Comp.*, 55:1–22., 1990.

[14] M. E. D. Brock A. Luty and J. A. McCammon. Solving the finite-difference non-linear Poisson-Boltzmann equation. *J. Comput. Chem.*, 13 Issue 9:1114–1118, 1992.

[15] C. Carstensen. Merging the Bramble-Pasciak-Steinbach and the Crouzeix-Thomee criterion for H1-stability of the L2-projection onto finite element spaces. *Math. Comp.*, 2000. Accepted. Berichtsreihe des Mathematischen Seminars Kiel 00-1 (2000).

[16] P. G. Ciarlet. *The Finite Element Method for Elliptic Problems.* North-Holland, New York, NY, 1978.

[17] W. Dahmen and A. Kunoth. Multilevel preconditioning. *Numer. Math.*, 63:315–344, 1992.

[18] P. J. Davis. *Interpolation and Approximation.* Dover Publications, 180 Varvick Street, New York, NY 10014, 1975.

[19] R. A. DeVore and G. G. Lorentz. *Constructive Approximation.* Grundlehren der mathematischen Wissenschaften 303. Springer Verlag, Berlin Heidelberg, 1993.

[20] M. Griebel and P. Oswald. On the abstract theory of additive and multiplicative Schwarz algorithms. *Numer. Math.*, 70:163–180, 1995.

[21] M. Holst, N. Baker, and F. Wang. Adaptive multilevel finite element solution of the Poisson-Boltzmann equation I: theory. *J. Comput. Chem.*, 21 Issue 15:1343–1352., 2000.

[22] M. J. Holst. *The Poisson-Boltzmann Equation, Analysis and Multilevel Numerical Solution.* PhD thesis, Numerical Computing Group, University of Illinois, Urbana-Champaign, IL, 1993.

[23] B. Joe and A. Liu. On the shape of tetrahedra from bisection. *Math. Comp.*, 63:141–154, 1994.

[24] B. Joe and A. Liu. Quality local refinement of tetrahedral meshes based on bisection. *SIAM J. Sci. Comput.*, 16:1269–1291, 1995.

[25] J. M. Maubach. Local bisection refinement for n-simplicial grids generated by reflection. *SIAM J. Sci. Comput.*, 16 No. 1:210–227, 1995.

[26] W. F. Mitchell. *Unified Multilevel Adaptive Finite Element Methods for Elliptic Problems*. PhD thesis, Computer Science, University of Illinois at Urbana-Champaign, Urbana, IL, 1988.

[27] M. E. G. Ong. *Hierarchical basis preconditioners for second order elliptic problems in three dimensions*. PhD thesis, University of Washington, 1989.

[28] P. Oswald. On function spaces related to finite element approximation theory. *Zeitschrift für Analysis und ihre Anwendungen*, 9:43–64, 1990.

[29] P. Oswald. Hierarchical conforming finite element methods for the biharmonic equation. *SIAM J. Numer. Anal.*, 29, No. 6:1610–1625, 1992.

[30] P. Oswald. On discrete norm estimates related to multilevel preconditioners in the finite element method. *In K. G. Ivanov, P. Petrushev, and B. Sendov, editors, Proceedings International Conference on Constructive Theory of Functions, Varna 1991*, pages 203–214, 1992. Publ. House of Bulgarian Academy of Sciences.

[31] P. Oswald. *Multilevel Finite Element Approximation Theory and Applications*. Teubner Skripten zur Numerik. B. G. Teubner, Stuttgart, 1994.

[32] P.Debye and E.Hückel. *Physik. Z.*, volume 24. 1923.

[33] M. C. Rivara. Local modification of meshes for adaptive and/or multigrid finite element methods. *J. Comput. Appl. Math.*, 36:79–89, 1991.

[34] M. C. Rivara and C. Levin. A 3-d refinement algorithm suitable for adaptive and multigrid techniques. *Comm. in App. Num. Meth.*, 8:281–290, 1992.

[35] E. G. Sewell. *Automatic generation of triangulations for piecewise polynomial approximation*. PhD thesis, Department of Mathematics, Purdue University, West Lafayette, IN, 1972.

[36] P. S. Vassilevski and J. Wang. Stabilizing the hierarchical basis by approximate wavelets, I: Theory. *Numer. Linear Alg. Appl.*, 4 Number 2:103–126, 1997.

[37] P. S. Vassilevski and J. Wang. Wavelet-like methods in the design of efficient multilevel preconditioners for elliptic PDEs. In W. Dahmen, A. Kurdila, and P. Oswald, editors, *Multiscale Wavelet Methods For Partial Differential Equations*, chapter 1, pages 59–105. Academic Press, 1997.

[38] P. S. Vassilevski and J. Wang. Stabilizing the hierarchical basis by approximate wavelets, II: Implementation and numerical experiments. *SIAM J. Sci. Comput.*, 20 Number 2:490–514., 1998.

[39] J. Xu. Iterative methods by space decomposition and subspace correction. *SIAM Rev.*, 34:581–613, 1992.

[40] J. Xu and J. Qin. Some remarks on a multigrid preconditioner. *SIAM J. Sci. Comput.*, 15:172–184, 1994.

[41] H. Yserentant. On the multilevel splitting of finite element spaces. *Numer. Math.*, 49:379–412, 1986.

[42] H. Yserentant. Two preconditioners based on the multi-level splitting of finite element spaces. *Numer. Math.*, 58:163–184, 1990.

[43] H. Yserentant. Old and new convergence proofs for multigrid methods. *Acta Numerica*, pages 285–326, 1993.

[44] S. Zhang. *Multilevel iterative techniques.* PhD thesis, Pennsylvania State University, 1988.

[45] X. Zhang. Multilevel Schwarz methods. *Numer. Math.*, 63:521–539, 1992.