

**An Optimization Framework for Adaptive Higher-Order  
Discretizations of Partial Differential Equations on  
Anisotropic Simplex Meshes**

by

Masayuki Yano

B.S., Georgia Institute of Technology (2007)

S.M., Massachusetts Institute of Technology (2009)

Submitted to the Department of Aeronautics and Astronautics  
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2012

© Massachusetts Institute of Technology 2012. All rights reserved.

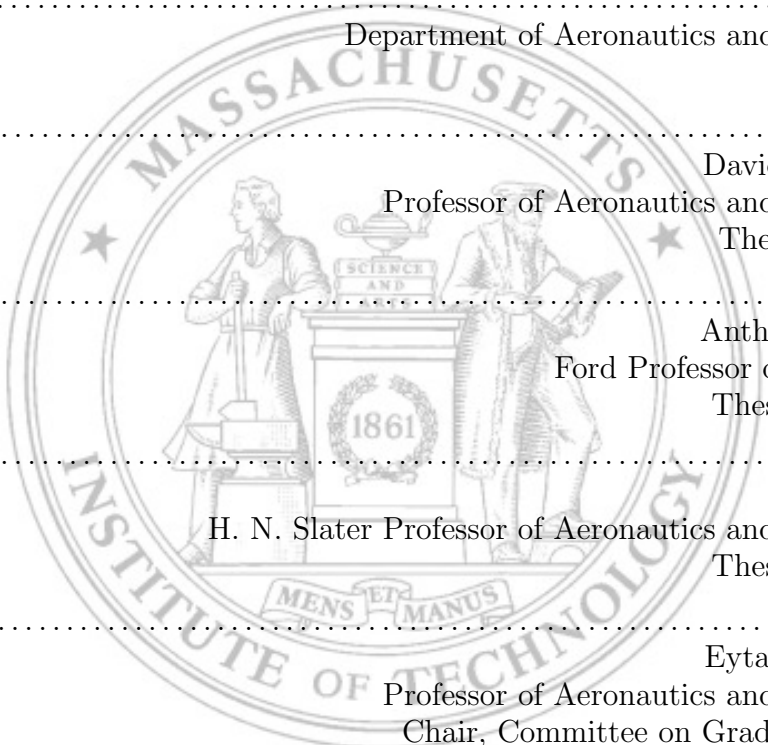
Author .....  
Department of Aeronautics and Astronautics  
May 24, 2012

Certified by .....  
David L. Darmofal  
Professor of Aeronautics and Astronautics  
Thesis Supervisor

Certified by .....  
Anthony T. Patera  
Ford Professor of Engineering  
Thesis Committee

Certified by .....  
Jaime Peraire  
H. N. Slater Professor of Aeronautics and Astronautics  
Thesis Committee

Accepted by .....  
Eytan H. Modiano  
Professor of Aeronautics and Astronautics  
Chair, Committee on Graduate Students

The seal of the Massachusetts Institute of Technology is a large, faint watermark in the background. It is a circular emblem with "MASSACHUSETTS" at the top and "INSTITUTE OF TECHNOLOGY" at the bottom. In the center, there is a shield depicting two figures, a Native American and a European, standing on either side of a pedestal. The pedestal has a scroll with "SCIENCE AND" and a date "1861". Below the shield is a banner with the Latin motto "MENS ET MANUS".







# An Optimization Framework for Adaptive Higher-Order Discretizations of Partial Differential Equations on Anisotropic Simplex Meshes

by

Masayuki Yano

Submitted to the Department of Aeronautics and Astronautics  
on May 24, 2012, in partial fulfillment of the  
requirements for the degree of  
Doctor of Philosophy

## Abstract

Improving the autonomy, efficiency, and reliability of partial differential equation (PDE) solvers has become increasingly important as powerful computers enable engineers to address modern computational challenges that require rapid characterization of the input-output relationship of complex PDE governed processes. This thesis presents work toward development of a versatile PDE solver that accurately predicts engineering quantities of interest to user-prescribed accuracy in a fully automated manner.

We develop an anisotropic adaptation framework that works with any localizable error estimate, handles any discretization order, permits arbitrarily oriented anisotropic elements, robustly treats irregular features, and inherits the versatility of the underlying discretization and error estimate. Given a discretization and any localizable error estimate, the framework iterates toward a mesh that minimizes the error for a given number of degrees of freedom by considering a continuous optimization problem of the Riemannian metric field. The adaptation procedure consists of three key steps: sampling of the anisotropic error behavior using element-wise local solves; synthesis of the local errors to construct a surrogate error model based on an affine-invariant metric interpolation framework; and optimization of the surrogate model to drive the mesh toward optimality. The combination of the framework with a discontinuous Galerkin discretization and an *a posteriori* output error estimate results in a versatile PDE solver for reliable output prediction.

The versatility and effectiveness of the adaptive framework are demonstrated in a number of applications. First, the optimality of the method is verified against anisotropic polynomial approximation theory in the context of  $L^2$  projection. Second, the behavior of the method is studied in the context of output-based adaptation using advection-diffusion problems with manufactured primal and dual solutions. Third, the framework is applied to the steady-state Euler and Reynolds-averaged Navier-Stokes equations. The results highlight the importance of adaptation for high-order discretizations and demonstrate the robustness and effectiveness of the proposed method in solving complex aerodynamic flows exhibiting a wide range of scales. Fourth, fully-unstructured space-time adaptivity is realized, and its competitiveness is assessed for wave propagation problems. Finally, the framework is applied to enable spatial error control of parametrized PDEs, producing universal optimal meshes applicable for a wide range of parameters.

Thesis Supervisor: David L. Darmofal

Title: Professor of Aeronautics and Astronautics







## Acknowledgments

I would like to thank all those who made this work possible. First, I would like to thank my adviser, Prof. David Darmofal, for giving me the opportunity to work with him and for his inspiration and encouragement throughout my graduate study. In addition, I would like to thank my committee members, Prof. Anthony Patera and Prof. Jaime Peraire, for sharing their experienced vision on computational research in continuum mechanics, which led to many improvements in my PhD work. I would also like to recognize Dr. Frédéric Alauzet, Prof. Krzysztof Fidkowski, and Dr. Ralf Hartmann for their critical feedback on the initial draft of this thesis. I am indebted to Dr. Steven Allmaras, not only for his insightful comments on the thesis, but also for his dedication to the weekly meetings. I would also like to thank Bob Haimes for all his honest advices on variety of topics.

I would like to thank the entire ProjectX team, past and present, for the many contributions that enabled this work: Chris, Garrett, and Todd, for laying the foundation of what ProjectX is today; Laslo, for his numerous contributions to the code; Josh, for providing a bridge to the Boeing groups and bringing industry perspectives; Eric, for his work on speeding up the code; and Julie, for reminding me the joy of teaching something I am passionate about. Special thanks goes to Huafei, who has been my office mate for my entire graduate career, always willing to discuss various topics, research related or otherwise. I am indebted to JM for many probing questions that spurred various ideas presented in this thesis and, most of all, for treating my problem as if it was his own when going get tough. I am fortunate to be surrounded by such a passionate group of colleagues and great friends.

I would also like to thank many others who have shaped the past five years of my graduate life. I must first thank the MIT2.086/SUTD303 curriculum development team — Prof. Patera, Debbie, and James; taking part in the undergraduate curriculum development effort has been one of the most rewarding experiences of my graduate career. I would also like to thank everyone in ACDL for creating a productive yet fun environment to conduct research, in particular Andrew, Chad, Chelsea, David L, David M, Eric, Hemant, Matt, and Xun. In addition, Jean, Robin, Sue, and Meghan deserve recognition for helping me schedule meetings with busy professors. I have enjoyed continuing friendships from my undergraduate years at Tech, especially with Justin, Jamie, James, Gaurav, Mike, and Bev. Lastly, I would like to thank JM and Mora for teaching me many things outside of lab and making my past five years a lot of fun.

I would like to thank my parents, Shigeyuki and Keiko, and my brother, Hiroyuki, for all their continuous support, without which I would not have gotten this far. I would also like to thank my grandparents — Kazushige, Michiko, Kaoru, and Yoshiko — for their encouragements.

Finally, I would like to acknowledge the financial support provided by the Boeing Company under technical monitor Dr. Mori Mani and by MIT through the AeroAstro Department Fellowship and the Singapore-MIT Fellowship in Computational Engineering.







# Contents

<b>1</b>	<b>Introduction</b>	<b>21</b>
1.1	Motivation . . . . .	21
1.2	Thesis Objective . . . . .	23
1.2.1	Mathematical Description of the Objective . . . . .	24
1.3	Background . . . . .	25
1.3.1	High-Order Discretizations for General Geometries . . . . .	26
1.3.2	Error Estimation . . . . .	28
1.3.3	Adaptation Mechanics . . . . .	30
1.4	Thesis Overview . . . . .	38
<b>2</b>	<b>Discretization, <i>A Posteriori</i> Output Error Estimation, and Continuous Mesh Framework</b>	<b>41</b>
2.1	Discretization . . . . .	41
2.2	Dual-Weighted Residual Method . . . . .	43
2.2.1	Error Estimation . . . . .	43
2.2.2	Error Localization . . . . .	44
2.3	Continuous Mesh Framework . . . . .	45
2.3.1	Metric-Conforming Meshes . . . . .	46
2.3.2	Mesh-Conforming Metric Fields . . . . .	47
2.3.3	Metric-based Representation of Polynomial Approximation Errors . . . . .	48
2.3.4	<i>A Priori</i> Metric-based Representation of the Output Error . . . . .	49
<b>3</b>	<b>Mesh Optimization via Error Sampling and Synthesis</b>	<b>53</b>
3.1	Output Error Minimization Problem . . . . .	53
3.1.1	Problem Definition and Continuous Relaxation . . . . .	53
3.1.2	Error and Cost Functionals . . . . .	54
3.1.3	Design Criteria and Approach . . . . .	56
3.2	Optimization Algorithm: MOESS . . . . .	57
3.2.1	Metric Manipulation Framework . . . . .	57
3.2.2	Local Error Sampling . . . . .	59
3.2.3	Local Error Model Synthesis . . . . .	62



3.2.4	Local Cost Model . . . . .	65
3.2.5	Optimization of the Surrogate Model . . . . .	66
3.3	Properties of MOESS . . . . .	70
3.4	Practical Considerations and Data Reported . . . . .	71
<b>4</b>	<b><math>L^2</math> Projection and Error Control</b>	<b>73</b>
4.1	Introduction . . . . .	73
4.2	Conditions for the Optimal Approximant . . . . .	74
4.3	$r^\alpha$ -Type Corner Singularity . . . . .	75
4.3.1	Analytical Solution . . . . .	76
4.3.2	Numerical Results . . . . .	80
4.4	2d Boundary Layer . . . . .	81
4.4.1	Optimality Conditions for Functions with No Mixed Partial . . . . .	82
4.4.2	Analytical Solution to the 2d Boundary Layer Problem . . . . .	84
4.4.3	Numerical Results . . . . .	85
4.5	3d Boundary Layer . . . . .	87
4.5.1	Analytical Solution . . . . .	88
4.5.2	Numerical Results . . . . .	88
4.6	Conclusion . . . . .	90
<b>5</b>	<b>Advection-Diffusion Equation</b>	<b>91</b>
5.1	Governing Equation and Problem Setup . . . . .	91
5.2	Results . . . . .	94
5.2.1	Assessment Procedure . . . . .	94
5.2.2	Primal-Dual Boundary Layer . . . . .	94
5.2.3	Dual-Only Boundary Layer . . . . .	96
5.2.4	Primal-Only Boundary Layer . . . . .	97
5.3	Conclusions . . . . .	100
<b>6</b>	<b>Compressible Navier-Stokes Equations</b>	<b>101</b>
6.1	Governing Equations . . . . .	102
6.1.1	Euler and Navier-Stokes Equations . . . . .	102
6.1.2	Reynolds-Averaged Navier-Stokes Equations . . . . .	103
6.2	The Importance of Mesh Adaptation for Higher-Order Discretizations of Aerodynamic Flows . . . . .	105
6.2.1	NACA 0012 Subsonic Euler . . . . .	106
6.2.2	RAE 2822 Subsonic RANS-SA . . . . .	109
6.3	Assessment of MOESS Applied to Aerodynamic Flows . . . . .	111
6.3.1	Assessment Procedure . . . . .	111
6.3.2	Laminar Flat Plate . . . . .	112
6.3.3	RAE 2822 Transonic RANS-SA . . . . .	113



6.3.4	NACA 0006 Euler Supersonic Shock Propagation . . . . .	116
6.3.5	Multi-Element Supercritical 8 Transonic RANS-SA . . . . .	119
6.3.6	Laminar Flow over a Delta Wing . . . . .	126
6.3.7	Computational Cost . . . . .	127
6.4	Conclusions . . . . .	129
<b>7</b>	<b>Fully-Unstructured Space-Time Adaptivity for Wave Propagation Problems</b>	<b>131</b>
7.1	Introduction . . . . .	131
7.2	The Wave Equation and Discretization . . . . .	133
7.3	Energy Error Estimate . . . . .	134
7.4	Results: The Wave Equation . . . . .	136
7.4.1	Assessment Procedure . . . . .	136
7.4.2	1+1d Wave Propagation . . . . .	138
7.4.3	2+1d Wave Propagation . . . . .	142
7.5	Nonlinear Waves: Space-Time Euler Equations . . . . .	145
7.5.1	2+1d Vortex Convection . . . . .	147
7.5.2	1+1d Riemann Problem . . . . .	150
7.6	Conclusions . . . . .	154
<b>8</b>	<b>Adaptation for Parametrized Partial Differential Equations</b>	<b>157</b>
8.1	Introduction . . . . .	157
8.1.1	Mathematical Description of the Problem . . . . .	158
8.2	Space-Parameter Galerkin Method . . . . .	158
8.2.1	Formulation . . . . .	158
8.2.2	Stability of the Space-Parameter Formulation . . . . .	160
8.2.3	Spatial Error Estimation and Control . . . . .	162
8.2.4	Practical Considerations . . . . .	163
8.3	Space-Galerkin Parameter-Collocation Method . . . . .	163
8.3.1	Formulation . . . . .	163
8.3.2	Spatial Error Estimate and Error Control . . . . .	164
8.4	Numerical Results . . . . .	165
8.4.1	RAE 2822 Subsonic RANS-SA . . . . .	165
8.4.2	Three-Element MDA High-Lift Airfoil RANS-SA . . . . .	171
8.5	Conclusions . . . . .	178
<b>9</b>	<b>Conclusions</b>	<b>181</b>
9.1	Summary and Conclusions . . . . .	181
9.2	Future Work . . . . .	183



<b>A</b>	<b>Discontinuous Galerkin Method</b>	<b>187</b>
A.1	Discontinuous Galerkin Discretization . . . . .	188
A.1.1	Nonlinear Discontinuity Regularization . . . . .	190
A.1.2	Solution Method . . . . .	191
A.2	Output Evaluation . . . . .	192
<b>B</b>	<b>Comparison of Vector- and Tensor-Based Element Sizing for the Shock PDE</b>	<b>195</b>
B.1	Comparison on a Fixed Mesh . . . . .	195
B.2	Effects on Adaptation . . . . .	198
<b>C</b>	<b>Regularization of Surface Quantity Distributions</b>	<b>199</b>
C.1	Formulation . . . . .	199
C.2	Results . . . . .	200
<b>D</b>	<b>Metric-based <i>A Priori</i> Error Bounds</b>	<b>203</b>
D.1	Anisotropic Polynomial Interpolation Theory . . . . .	203
D.1.1	Notation . . . . .	204
D.1.2	Volume Inequalities . . . . .	204
D.2	Output Error Bounds . . . . .	209
<b>E</b>	<b>On DWR Error Estimates for <math>p</math>-Dependent Discretizations</b>	<b>219</b>
E.1	$p$ -Dependence of DG Discretizations . . . . .	219
E.2	The Dual-Weighted Residual Error Estimation . . . . .	223
E.2.1	Problem Setup . . . . .	223
E.2.2	Local and Global Consistency Results . . . . .	224
E.2.3	DWR Error Estimates . . . . .	226
E.2.4	Assessment of the Error Estimates . . . . .	229
E.3	<i>A Priori</i> Error Analysis . . . . .	230
E.3.1	Assumptions . . . . .	230
E.3.2	Useful Relationships . . . . .	230
E.3.3	<i>A Priori</i> Error Analysis of the True Output Error . . . . .	235
E.3.4	<i>A Priori</i> Error Analysis of Output Error Estimate 3 . . . . .	236
E.3.5	<i>A Priori</i> Error Analysis of Output Error Estimate 1 . . . . .	237
E.3.6	<i>A Priori</i> Error Analysis of Output Error Estimate 2 . . . . .	239
E.3.7	Summary of <i>A Priori</i> Error Analysis . . . . .	242
E.4	Numerical Results . . . . .	243
E.4.1	True Output Error . . . . .	244
E.4.2	Output Error Estimate 1 . . . . .	245
E.4.3	Output Error Estimate 2 . . . . .	245
E.4.4	Output Error Estimate 3 . . . . .	248



E.5	Conclusion . . . . .	249
<b>F</b>	<b>Properties of the Adaptation Algorithm</b>	<b>251</b>
F.1	Relationship between Step Matrix and the Change in Approximability . . .	251
F.2	Inclusion of the Isotropic Error Model . . . . .	252
F.3	Invariance of the Sampling Quality . . . . .	253
F.4	Invariance under Coordinate Transformation . . . . .	254
<b>G</b>	<b>On Gradient Descent in the Metric Tensor Space</b>	<b>259</b>
G.1	The Choice of Metric . . . . .	259
G.1.1	Frobenius Norm . . . . .	260
G.1.2	Log-Euclidean Framework . . . . .	260
G.1.3	Affine-Invariant Framework . . . . .	262
G.2	Single Step Descent Test . . . . .	263
G.2.1	Action from the Identity Tensor . . . . .	264
G.2.2	Action from an $\mathcal{R} = 5$ Tensor . . . . .	265
G.2.3	Action from an $\mathcal{R} = 50$ Tensor . . . . .	266
G.3	Multi- Step Descent Test . . . . .	267
G.3.1	From the Identity Tensor to an $\mathcal{R} = 2$ Tensor . . . . .	267
G.3.2	From the Identity Tensor to an $\mathcal{R} = 20$ Tensor . . . . .	268
G.3.3	From an $\mathcal{R} = 5$ Tensor to an $\mathcal{R} = 20$ Tensor . . . . .	268







# List of Figures

1-1	Illustration of the information flow in an automated PDE solver. . . . .	25
1-2	An example of a metric-mesh pair. . . . .	32
1-3	An example of anisotropic quadrilateral mesh generated by hierarchical subdivisions (reproduced with permission from [40]). . . . .	34
1-4	The original, anisotropic split, and isotropic split configurations for quadrilateral-based hierarchical subdivision strategy. . . . .	37
2-1	Illustration of the transformation of the reference element $\hat{\kappa}$ into a physical element $\kappa$ . The metric tensor associated with each element is shown in dashed lines. . . . .	47
3-1	The original, edge split, and uniformly split configurations used to sample the local error behavior in two dimensions. The metrics implied by the sampled configurations are shown in dashed lines. . . . .	60
3-2	The original and edge split configurations used to sample the local error behavior in three dimensions. . . . .	61
3-3	Sequence of adapted meshes for the 2d boundary layer $L^2$ error control problem. . . . .	72
3-4	Variation in the degrees of freedom and error with the adaptation iterations. The samples used for assessment are marked in red boxes. . . . .	72
4-1	The optimized meshes for the corner singularity problem. Each mesh contains approximately 200 elements. . . . .	80
4-2	The element size $h$ vs. the distance of the element centroid from the corner $r$ for the optimized meshes for the corner singularity problem with $\alpha = 2/3$ . The lines and coefficients shown result from least-squares fit in $\log(h)$ vs. $\log(r)$ . . . . .	81
4-3	Examples of optimized boundary layer meshes for $p = 1$ and $p = 3$ . Each mesh contains approximately 200 elements. . . . .	85
4-4	The element size in the perpendicular direction, $h_1$ , and the aspect ratio distribution, $\mathcal{R} = h_2/h_1$ , for the 2d boundary layer problem with $\epsilon = 0.01$ and $\beta = 2^{p+1}$ . . . . .	86



4-5	The element size in the perpendicular direction, $h_1$ , and the aspect ratio distribution, $\mathcal{R}_i = h_i/h_1$ , for the 3d boundary layer problem with $\epsilon = 0.01$ and $\beta_2 = 2^{p+1}$ and $\beta_3 = 4^{p+1}$ . . . . .	89
5-1	The domain for the advection-diffusion problems. . . . .	92
5-2	Solutions to the boundary layer problems. . . . .	93
5-3	Output error convergence for the primal-dual boundary layer problem using the $p = 1$ and $p = 2$ discretizations. . . . .	95
5-4	Adapted meshes for the primal-dual boundary layer problem. All $p = 1$ and $p = 2$ meshes have dof = 1000 and dof = 2000, respectively. . . . .	96
5-5	Output error convergence for the dual-only boundary layer problem using the $p = 1$ and $p = 2$ discretizations. . . . .	97
5-6	Adapted meshes for the dual-only boundary layer problem. All $p = 1$ and $p = 2$ meshes have dof = 1000 and dof = 2000, respectively. . . . .	98
5-7	Output error convergence for the primal-only boundary layer problem using the $p = 1$ and $p = 2$ discretizations. . . . .	99
5-8	Adapted meshes for the primal-only boundary layer problem. All $p = 1$ and $p = 2$ meshes have dof = 1000 and dof = 2000, respectively. . . . .	99
6-1	Comparison of the error convergence for uniform and adaptive refinements for the subsonic NACA 0012 Euler flow. . . . .	107
6-2	Comparison of the trailing edge mesh grading and error indicator distribution of the $p = 3$ , dof = 20,000 meshes obtained from uniform and adaptive refinements of the $p = 3$ , dof = 5,000 optimized mesh for the subsonic NACA 0012 Euler flow. The color scale is in $\log_{10}(\eta_\kappa)$ . . . . .	107
6-3	Element size distributions in the vicinity of the trailing edge of the $p = 1$ and $p = 3$ optimized meshes for the subsonic NACA 0012 Euler flow. . . . .	108
6-4	Comparison of the error convergence for uniform and adaptive refinements for the subsonic RAE 2822 RANS-SA flow. . . . .	109
6-5	Comparison of the error indicator distributions of $p = 3$ , dof = 80,000 meshes obtained from uniform and adaptive refinements of the $p = 3$ , dof = 20,000 optimized mesh for the subsonic RAE 2822 RANS-SA flow. The color scale is in $\log_{10}(\eta_\kappa)$ . . . . .	110
6-6	Drag error convergence for the laminar flat plate problem. . . . .	112
6-7	Close views of the meshes for the laminar flat plate problem. ( $p = 1$ , dof = 2,000) . . . . .	113
6-8	Drag error convergence for the RAE 2822 transonic RANS-SA problem. . .	114
6-9	The Mach number, the mass adjoint, and the meshes for the RAE 2822 transonic RANS-SA problem. ( $p = 2$ , dof = 60,000) . . . . .	115



6-10	The regularized $c_p$ and $c_f$ distributions for the transonic RAE 2822 RANS-SA problem computed on $p = 1$ and $p = 2$ adapted meshes obtained using MOESS. . . . .	116
6-11	Pressure line output error convergence for the NACA 0006 Euler shock propagation problem ( $p = 2$ ). . . . .	117
6-12	The pressure, the mass adjoint, and the meshes for the NACA 0006 Euler supersonic shock propagation problem. The pressure line is depicted in a red line. ( $p = 2$ , dof = 40,000) . . . . .	118
6-13	The Mach number distribution and the mass adjoint for the MSC8 transonic RANS-SA problem. . . . .	120
6-14	The initial mesh for the MSC8 transonic RANS-SA problem. . . . .	120
6-15	Drag adaptation histories for the $p = 1$ , dof = 40,000 isotropic-to-RANS mesh transition test. . . . .	121
6-16	The Mach number distribution and the mesh for the fifth adaptation iteration starting from the isotropic mesh in Figure 6-14 using FFMA ( $p = 1$ , dof = 40,000). . . . .	121
6-17	The adapted meshes starting from the isotropic mesh in Figure 6-14 using MOESS. . . . .	122
6-18	Drag error convergence for the MSC8 Transonic RANS-SA problem. . . . .	123
6-19	Drag-adapted meshes for the transonic MSC8 RANS-SA problem. For each subfigure: overview (top left); main-element shock (top right); main-element leading edge (bottom left); and flap-element (bottom right). ( $p = 2$ , dof = 120,000) . . . . .	124
6-20	The $c_p$ and $c_f$ distributions for the transonic MSC8 RANS-SA problem computed on adapted meshes obtained using MOESS. . . . .	125
6-21	The Mach number isosurface, Mach number slices, and the streamlines for the delta wing case. . . . .	126
6-22	Drag error convergence for the laminar delta wing case. “HOW mesh” are the high-order workshop meshes prepared by NLR for the High-Order Workshop [152]. “L&H” is the result reported by Leicht and Hartmann using their hexahedron-based hierarchical subdivision strategy [93]. . . . .	127
6-23	The 26-element initial mesh and the $p = 2$ , dof = 160,000 adapted mesh. The symmetry plane is shown in gray. . . . .	128
7-1	The first component of the primal and dual solutions to the 1+1d wave problem. . . . .	138
7-2	Energy and output error convergence for the 1+1d wave problem. ( $p = 2$ ) .	139
7-3	Adapted meshes for the 1+1d wave problem. ( $p = 2$ ) . . . . .	140
7-4	Scaling of the energy-error-to-dof efficiency with the characteristic length ratio, $1/s$ , for the 1+1d wave problem. ( $p = 2$ ) . . . . .	141



7-5	Time slices of the solution to the 2+1d wave problem. The output evaluation point is marked by a red circle. Note that the color scale for $t = 0.0$ is different from that for all the others. . . . .	143
7-6	Energy and output error convergence for the 2+1d wave propagation problem. ( $p = 2$ ) . . . . .	144
7-7	Solution history at $x_1 = 0.0$ , $x_2 = 0.75$ for the 2+1d wave propagation problem. ( $p = 2$ , $\text{dof} \approx 240000$ ) . . . . .	144
7-8	Time slices of the solution to the 2+1d wave problem obtained on the $p = 2$ , $\text{dof} = 240,000$ output-adapted mesh. The output evaluation point is marked by a red circle. (c.f. the reference solution in Figure 7-5) . . . . .	145
7-9	The primal solution, the dual solution, and $p = 2$ , $\text{dof} = 240,000$ adapted meshes. 2+1d view (top row); the $x_2 = 0$ plane (middle row); and the $x_1 = 0$ plane (bottom row). . . . .	146
7-10	The density field of the isentropic vortex convection problem. The solution at $t = 0$ and $t = 20$ (left), and the space-time cut along $x_2 = 0$ (right). . .	148
7-11	The mass adjoint for the momentum perturbation output of the isentropic vortex convection problem. The solution at $t = 0$ and $t = 10$ (left), and the space-time cut along $x_2 = 0$ (right). . . . .	149
7-12	Convergence of the momentum-perturbation output for the isentropic vortex convection problem. . . . .	149
7-13	Space-time adapted mesh for the isentropic vortex convection problem. ( $p = 2$ , $\text{dof} = 80,000$ ) . . . . .	150
7-14	Solution to the shock tube problem. . . . .	151
7-15	Convergence of the two outputs of the shock tube problem. ( $p = 2$ ) . . . .	151
7-16	Adapted meshes for the shock tube problem. ( $p = 2$ , $\text{dof} = 10,000$ ) . . . .	152
7-17	The density and pressure distributions of the shock tube problem at two different time instances. The uniform mesh contains approximately 20,000 degrees of freedom whereas the adapted meshes contain approximately 10,000 degrees of freedom. . . . .	153
8-1	Parameter expansion mode strengths of the first two modes for select solution fields of the RAE 2822 case. The output is the mean drag. . . . .	166
8-2	The lift curve and drag polar for the RAE 2822 case on a fixed $p = 2$ , $\text{dof} = 10,000$ mesh. . . . .	167
8-3	Variation in the $c_d$ error with the parameter expansion degree, $s$ , for the RAE 2822 case. The reference solution is computed using the $s = 8$ expansion. Solutions computed on a fixed $p = 2$ , $\text{dof} = 10,000$ mesh. . . . .	168
8-4	Optimized meshes for the RAE 2822 subsonic RANS case. Overview (left) and zoom of the leading edge region $[-0.03c, 0.03c]^2$ (right). . . . .	169



8-5	Variation in the $c_d$ error for the RAE 2822 case over $\alpha \in [0^\circ, 6^\circ]$ using $\alpha = 0^\circ$ , $\alpha = 4^\circ$ , and $\alpha \in [0^\circ, 6^\circ]$ optimized meshes. . . . .	170
8-6	The Mach number and normalized SA working variable for the three-element MDA airfoil case at $\alpha = 8^\circ$ and $\alpha = 24^\circ$ . . . . .	172
8-7	Parameter expansion mode strengths of the first two modes for select solution fields of the three-element MDA airfoil case. . . . .	173
8-8	Select optimized meshes for the three-element MDA airfoil case. ( $p = 2$ , $\text{dof} = 90,000$ ) . . . . .	175
8-9	The lift curve, drag polar, drag error, and drag error indicator for the three-element MDA case. . . . .	176
8-10	The Mach number and normalized SA working variable for the $\alpha = 24^\circ$ flow computed on the $\alpha = 8^\circ$ optimized mesh. . . . .	177
B-1	The 7136-element NACA 0012 mesh used for the fixed mesh tests. . . . .	196
B-2	The artificial viscosity, $\epsilon$ , for the Euler problem solved at $M_\infty = \sqrt{2}$ . The two meshes are identical, except that one of them is tilted by $45^\circ$ . . . . .	197
B-3	The Mach number distribution on the same non-tilted meshes for the $M_\infty = \sqrt{2}$ flow. The contour lines are in 0.1 increments. . . . .	197
B-4	The adapted meshes obtained using the vector- and tensor-based element size specifications. Each mesh contains approximately 7000 elements. . . .	198
C-1	Comparison of raw and regularized $c_p$ and $c_f$ distributions for transonic RANS-SA flow over an RAE 2822 airfoil. The $p = 1$ and $p = 2$ discretizations achieve the drag error of $ c_d - c_d^{\text{ref}}  \approx 7 \times 10^{-6}$ and $\approx 4 \times 10^{-7}$ , respectively. . . .	201
E-1	The convergence of the true output error. . . . .	245
E-2	The local and global effectivity of the error estimate 1. . . . .	246
E-3	The local and global effectivity of the error estimate 2. . . . .	247
E-4	The local and global effectivity of the error estimate 3. . . . .	248
G-1	LE and AI gradient descent from $\mathcal{M}_0 = I$ for $\eta/\eta_0 = (0.5, 1.0, 1.0)$ . . . . .	264
G-2	LE and AI gradient descent from $\mathcal{M}_0 = I$ for $\eta/\eta_0 = (0.5, 0.5, 1.0)$ . . . . .	264
G-3	LE and AI gradient descent from $\mathcal{R}(\mathcal{M}_0) = 5$ for $\eta/\eta_0 = (0.5, 1.0, 1.0)$ . . . .	265
G-4	LE and AI gradient descent from $\mathcal{R}(\mathcal{M}_0) = 5$ for $\eta/\eta_0 = (0.5, 0.5, 1.0)$ . . . .	265
G-5	LE and AI gradient descent from $\mathcal{R}(\mathcal{M}_0) = 50$ for $\eta/\eta_0 = (0.5, 1.0, 1.0)$ . . . .	266
G-6	LE and AI gradient descent from $\mathcal{R}(\mathcal{M}_0) = 50$ for $\eta/\eta_0 = (0.5, 0.5, 1.0)$ . . . .	266
G-7	Multi-step descent test from the identity tensor to an $\mathcal{R} = 2$ tensor. . . . .	268
G-8	Multi-step descent test from the identity tensor to an $\mathcal{R} = 20$ tensor. . . . .	269
G-9	Multi-step descent test from an $\mathcal{R} = 5$ tensor to an $\mathcal{R} = 20$ tensor. . . . .	270







# List of Tables

4.1	Summary of the optimized mesh parameters for the 2d boundary layer problem. . . . .	87
4.2	Summary of the optimized mesh parameters for the 3d boundary layer problem. . . . .	90
5.1	Set of parameters defining the three advection-diffusion problems. The volume output weight for the primal-only case is $g_{\Omega}^{\text{prim}} = \frac{1}{2\pi(0.0012)} \exp\left(-\frac{1}{2}\left[\frac{x_1^2}{0.02^2} + \frac{(x_2-0.25)^2}{0.06^2}\right]\right)$ . The solution identifications correspond to those in Figure 5-2. . . . .	93
6.1	Timing breakdown for a single adaptation cycle normalized by the primal solve time. . . . .	129
E.1	Summary of the local and global error estimate convergence. . . . .	243







# Chapter 1

## Introduction

### 1.1 Motivation

Advancement in numerical algorithms and computational hardware in recent decades has made numerical simulation an indispensable tool in assisting engineering decisions and scientific discoveries. In particular, partial differential equation (PDE) solvers are widely used to analyze various physical phenomena, ranging from fluid dynamics to solid mechanics to electromagnetics. Unlike physical experiments, numerical simulation does not require specialized testing facilities and handles virtually any physical conditions and design configurations. This accessibility and flexibility make simulation well-suited for meeting the primary goal of engineering analysis: the characterization of the relationship between input parameters — such as geometric configurations, material properties, and operating conditions — and output quantities — the performance variables of interests. However, despite their widespread use, the current PDE solvers lack in efficiency, reliability, and autonomy, making high-fidelity simulations unaffordable, producing unreliable prediction of outputs, and requiring frequent user intervention. These limitations prevent the current PDE solvers from realizing the full potential of simulation-based analysis in the engineering and scientific environments.

The lack of automated error control has not only limited the effectiveness of PDE-based simulations but also has led to catastrophic failures. One notable example is the Sleipner A platform accident in 1991, in which the 44,000-ton offshore platform sank due to a flawed design based on the finite element analysis that underestimated the shear stress in a supporting structure by 45% [84]. The cause of the underestimation was attributed



to the use of a coarse mesh with poorly shaped linear elements and the subsequent post-processing of the results. A more refined finite element analysis conducted as part of the failure investigation predicted the failure of the structure at the water depth of 62 meters, which was in agreement with the actual failure depth of 65 meters [49]. Thus, the reliance on human experience in grid generation caused the catastrophic failure that could have been prevented with automated error control.

While computational power has increased dramatically over the past two decades since the accident, the ability of PDE solvers to produce a reliable output prediction in an automated manner is still limited. For example, error assessment and grid generation procedures employed in the aerospace industry for computational fluid dynamics (CFD) simulations still rely heavily on the experience of the CFD users. The inadequacy of this practice, even for geometries frequently encountered in engineering practice, has been highlighted in a study by Mavriplis in 2007 [103] conducted as a part of the third AIAA Drag Prediction Workshop. In this study, two families of meshes were generated — one by NASA Langley and the other by Cessna Aircraft Co. — following each organization’s best practices for a typical transonic turbulent flow over a wing. Then, Mavriplis tested for the convergence of the drag output under uniform scaling of the elements using a second-order industrial-strength CFD solver. Even though 10 times more degrees of freedom were used than typically used in practice, the output values were not grid converged; the finest NASA and Cessna mesh predicted the drag coefficient of 194 counts and 201 counts, respectively. The difference of 7 drag counts is quite significant, as 1 drag count translates to four to eight passengers for a typical, long-range, passenger jet [56, 144]. Mavriplis concludes that the range of scales present in the turbulent flow cannot be adequately resolved using a second-order method with best-practice meshes.

To take advantage of ever-increasing computational power, providing an automated error control will become even more important for two reasons. First, faster computers enable simulation of increasingly complex phenomena, in which an engineer’s knowledge may be of little use in identifying the features relevant to accurate output evaluation. This is particularly true for multiscale problems, where small-scale features may have a large impact on the overall solution behavior. Second, and more importantly, the next generation of PDE solvers must support modern computational challenges — such as design optimization, uncertainty quantification, and inverse parameter estimation. A successful



execution of these tasks requires a PDE solver that can reliably quantify the input-output relationship over a wide range of input parameters in a fully-automated manner. This thesis presents work toward development of a fully-automated PDE solver, freeing engineers from the task of handling numerical issues, reducing the time to achieve desired accuracy, and improving reliability and robustness of complex simulations.

## 1.2 Thesis Objective

The objective of this work is to develop a versatile, adaptive, higher-order PDE solution framework for reliably predicting an engineering output of interest in a fully-automated manner and apply it to a wide range of engineering applications to assess its effectiveness. In particular, the critical areas of improvement are identified as follows:

- **Autonomy:** Autonomy is a measure of the solver to complete the entire PDE solution process with little to no user intervention. The solution process here refers to the procedure of starting with the geometry definition and the physical condition as inputs and predicting the output quantities of interest.
- **Efficiency:** Efficiency is a measure of the solver to produce an accurate output prediction for a given computational effort. An efficient solver is capable of producing low-fidelity solutions rapidly or producing a high-fidelity solution in a reasonable time.
- **Reliability:** Reliability is a measure of the PDE solver to produce an output prediction that the user can trust. A reliable solver provides not only an output value but also the degree of confidence the user should have in the prediction.
- **Robustness:** Robustness is a measure of the PDE solver to produce reliable solutions for a wide range of geometries and physical conditions. A robust solver produces results that the user can trust for radically different configurations with little *a priori* knowledge of the solution.
- **Versatility:** Versatility is a measure of the PDE solver to handle a variety of PDEs arising from different disciplines or applications. A versatile solver requires minimal development effort to solve problems in different disciplines (e.g. solid mechanics, fluid dynamics) or applications (e.g. unsteady problems, parameter space exploration).



### 1.2.1 Mathematical Description of the Objective

This section provides a mathematical description of the thesis objective and introduces notation used throughout this work. A common goal of engineering simulations is to estimate a certain output quantity or quantities — such as drag, heat transfer rate, or strain energy release rate — of a process modeled by a PDE. Thus, the automated PDE solver considered in this work focuses on measuring, controlling, and effectively minimizing the error in an output,  $J$ , given by

$$J = \mathcal{J}(u),$$

where  $\mathcal{J} : V \rightarrow \mathbb{R}$  is the output functional, and  $u$  is the field variable which is a member of an appropriate function space  $V$  on a  $d$ -dimensional domain  $\Omega$ . The field variable  $u \in V$  satisfies a system of conservation laws

$$\frac{\partial u}{\partial t} + \nabla \cdot \mathcal{F}^{\text{conv}}(u, x, t) - \nabla \cdot \mathcal{F}^{\text{diff}}(u, \nabla u, x, t) = \mathcal{S}(u, \nabla u, x, t), \quad \forall x \in \Omega, t \in [0, T],$$

where the convective flux  $\mathcal{F}^{\text{conv}}$ , the diffusive flux  $\mathcal{F}^{\text{diff}}$ , and the source  $\mathcal{S}$  specify the PDE. An approximation to the desired output is obtained by discretizing the conservation laws and evaluating the discrete output functional. In the finite element framework, this results in an approximate output  $J_{h,p}$  given by

$$J_{h,p} = \mathcal{J}_{h,p}(u_{h,p}) \quad \text{such that} \quad \mathcal{R}_{h,p}(u_{h,p}, v_{h,p}) = 0, \quad \forall v_{h,p} \in V_{h,p},$$

where  $V_{h,p}$  is the finite-dimensional approximation space,  $u_{h,p} \in V_{h,p}$  is the discrete solution,  $\mathcal{J}_{h,p} : V_{h,p} \rightarrow \mathbb{R}$  is the discrete functional, and  $\mathcal{R}_{h,p} : V_{h,p} \times V_{h,p} \rightarrow \mathbb{R}$  is the discrete semilinear form. The subscripts  $h$  and  $p$  signify the characteristic element size and the polynomial degree of the finite element space. The quality of the estimated output is measured by the output error

$$\mathcal{E} \equiv J - J_{h,p} = \mathcal{J}(u) - \mathcal{J}_{h,p}(u_{h,p}).$$

The computational cost associated with the approximation process is denoted by  $\mathcal{C}$ . The error  $\mathcal{E}$  and the cost  $\mathcal{C}$  are in general functions of discretization, approximation space,



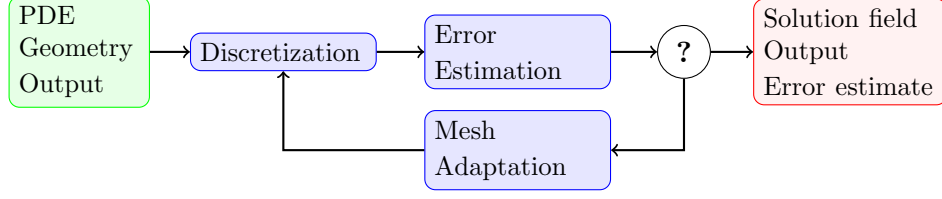


Figure 1-1: Illustration of the information flow in an automated PDE solver.

and linear and nonlinear solvers. Effectively controlling the output error  $\mathcal{E}$  using a given computational resource  $\mathcal{C}$  is the main objective of an automated PDE solver.

The ultimate goal is to develop a PDE solver that efficiently estimates the output quantity in a fully automated manner for a user defined problem characterized by 1) governing equations, 2) domain geometry, 3) boundary conditions, 4) output quantities, 5) output tolerances, and 6) available computational resources. Given a problem definition, the solver should provide an output prediction that meets the user prescribed tolerance with the least computational effort. This work develops technologies towards realizing such a fully automated general PDE solver.

### 1.3 Background

In the past decades, significant advancements have been made in improving the accuracy, autonomy, and reliability of PDE solvers. As identified in Figure 1-1, the technologies that constitute an automated PDE solver may be decomposed into three key pieces: discretization, error estimation, and mesh adaptation. Given a problem definition, the solver first discretize the problem and solve it on a (typically coarse) mesh, yielding the solution field and output predictions. Then, using an error estimation technique, the solver estimates the error in the output prediction. Finally, if the error is larger than the prescribed tolerance, the solver adapts the mesh to reduce the error. This work leverages recent developments in discretization, error estimation, and adaptation technologies. Specifically, this work builds on:

- **Discontinuous Galerkin (DG) methods:** DG methods can discretize a wide variety of 1st- and 2nd-order PDEs, provide stability for convection-dominated problems, maintain an element-wise compact stencil for high-order approximation, naturally treat boundary conditions, and support unstructured meshes.



- **Output-based error estimates:** Output-based error estimates provide a constant-free, *a posteriori* estimate of the error in the engineering output and an element-wise localization of the error to drive adaptation.
- **Anisotropic simplex adaptation mechanics:** Anisotropic simplex adaptation, combined with an output-based error estimate, provides an approximation space that minimizes the output error for a given cost by considering both the magnitude and directionality of the error distribution.

The following subsections review previous work on high-order discretizations, error estimation techniques, and adaptation strategies. The references provided are not intended to be an exhaustive survey, but rather consist of seminal works on the subject of interest.

### 1.3.1 High-Order Discretizations for General Geometries

In general, the output error is related to the characteristic mesh size,  $h$ , by

$$|\mathcal{E}| = |J - J_{h,p}| \leq Ch^r,$$

where the constant  $C$  and the convergence rate  $r$  are dependent on the discretization, (primal) solution, and output functional. For a sufficiently regular solution,  $r = cp + c_0$  for some constants  $c$  and  $c_0$ . High-order methods aim to accelerate the grid convergence of the numerical solution by achieving a higher convergence rate  $r$  through employing a higher  $p$ . There are two general classes of methods capable of providing high-order approximations of convection-dominated flows on unstructured meshes suited for general geometries: finite volume methods and stabilized finite element methods. Although this work focuses on finite element methods, it is worth noting that the high-order finite volume framework on unstructured meshes was pioneered by Barth [20] and subsequently extended by various researchers, e.g. [111, 151].

Development of high-order finite element methods starts with the work on the  $p$ -type finite element method by Babuska *et al.* [12]. In the  $p$ -type method, the solution resolution is improved by increasing the approximation order  $p$  instead of by decreasing the element diameter  $h$ . The  $p$ -type method combines the generality of the finite element methods and the exponential convergence property of the spectral methods for smooth problems. Later,



Patera introduced a variant of the  $p$ -type method called the spectral element method [90, 115].

While the finite element framework offers a conceptually simple path to achieving a higher solution order, it lacks stability for convection-dominated problems. To overcome this difficulty, Hughes and his collaborators introduced the streamline-upwind Petrov-Galerkin method (SUPG) [76, 83], which provides an additional stabilization for convection operators in the finite element framework. Through a series of papers, Hughes extended the method to support discontinuities [81, 82], system of equations [80], and time-dependent problems [78]. Hughes later generalized the concept of stabilization for finite element methods as the Galerkin Least-Squares (GLS) method [77]. Application of second-order GLS discretization to compressible flows is provided by Shakib and Hughes [131], and its higher-order extension is considered by Barth [21].

Another approach to providing stability for convection operators within the finite element framework is the discontinuous Galerkin (DG) method. The DG method stabilizes convection operators by incorporating Riemann solvers developed in the finite volume community. It also provides element-wise compact support of basis functions, which enables a straightforward implementation of  $hp$ -adaptation and efficient preconditioning. Reed and Hill introduced the original DG method to solve a scalar hyperbolic equation [123], and Johnson and Pitkäranta [85] and Richter [124] subsequently proved sharp *a priori* error estimates for the method. Chavent and Salzano extended the DG method to nonlinear hyperbolic equations by incorporating Godunov’s flux [41]. Cockburn and Shu introduced the RKDG method — which combined Runge-Kutta explicit time stepping with a slope limiter and DG spatial discretization — and extended the method to a multi-dimensional system of conservation laws in a series of papers [42, 44, 45, 47]. Independent from the aforementioned work, Allmaras and Giles presented a version of the DG method for Euler equations [6, 7]. Cockburn *et al.* provide an excellent summary of the early development of DG methods in their review paper [43].

To solve convection-diffusion systems, in particular the Navier-Stokes equations, Bassi and Rebay introduced a DG discretization of diffusive operators, which is now known as BR1 [24]. Cockburn and Shu generalized the method to yield the local discontinuous Galerkin (LDG) method [46]. Bassi and Rebay modified the BR1 method to maintain stability for purely elliptic problems and to recover the element-wise compact stencil,



yielding BR2 [25]. Similarly, Peraire and Person introduced the compact discontinuous Galerkin (CDG) method by modifying the LDG method to recover the compact stencil. Recent extensions of the discretization of diffusive operators within the DG framework include [48, 118, 143].

### 1.3.2 Error Estimation

A *posteriori* error estimation provides two distinct functions critical to driving an automated solution procedure. First is the estimation of the global error, which can be used to assess the quality of the finite element solution. Second is the localization of the error to elements, which is used to mark elements causing large errors for refinement or to mark those with small errors for coarsening.

Initial work on a *posteriori* error estimates focused on controlling the error measured in the energy norm of elliptic equations, such as those of structural elasticity. Development of energy-based error estimates began with the pioneering work of Babuška and Reinboldt [13]. Subsequently, a wide variety of error estimation and localization techniques were developed; a thorough review of energy-based error estimation techniques is provided by Ainsworth and Oden [1, 2] and references therein.

An increasing interest in development of solution-adaptive methods also led to development of simple error estimates based on estimating the interpolation error. The idea was originally proposed by Demkowicz *et al.* [50]. While not quantifying the solution error in a formal norm, interpolation-based error estimates — also called Hessian-based error estimates as majority of simulations were conducted using second-order methods — localized discretization error sufficiently for the purpose of adaptation. The use of the Hessian is motivated by the fact that the interpolation error for a linear interpolant is dictated by the solution Hessian. More importantly, an edge-based interpretation of the interpolation error provided an anisotropic description of the error. This allowed Peraire *et al.* [119] to develop an anisotropic adaptive scheme suitable for compressible flows with anisotropic features.

Both energy-based and interpolation-based error estimates assume that the local mesh refinement leads to reduction of the local solution error. This is true for elliptic equations, whose Green's functions decay rapidly away from the source. However, the estimates lose their effectiveness when applied to hyperbolic equations, whose Green's functions do not decay along the characteristics and can indefinitely transmit errors [138]. In other words,



the local solution error in a downstream region may be highly sensitive to upstream discretization errors, and this long-range error transmission may be hard to identify *a priori*.

Noting that engineering applications often require prediction of certain output quantities, output-based error estimation techniques have been developed more recently. These techniques incorporate an adjoint to estimate and localize the output error. An adjoint, or a dual solution, acts as a transfer function from the local residual perturbation to the output error, capturing the effect of error transmission. Due to the explicit inclusion of the adjoint, output-based error estimates work effectively for hyperbolic equations. The starting point for output-based error estimation is the error representation formula,

$$\mathcal{E} = J - J_{h,p} = \underbrace{\mathcal{R}_{h,p}(u_{h,p}, \psi - \psi_{h,p})}_{\text{(I)}} = \underbrace{\overline{\mathcal{R}}'_{h,p}[u, u_{h,p}](u - u_{h,p}, \psi - \psi_{h,p})}_{\text{(II)}}, \quad (1.1)$$

where  $J$  is the quantity of interest,  $u$  is the primal solution,  $\psi$  is the dual solution,  $\mathcal{R}_{h,p}(\cdot, \cdot)$  is the semilinear form for the governing PDEs, and  $\overline{\mathcal{R}}'_{h,p}[u, u_{h,p}](\cdot, \cdot)$  denotes the mean value linearization of the semilinear form about  $u$  and  $u_{h,p}$ . The expression (I) shows that the output error is equal to the primal residual weighted by the adjoint error. The expression (II) shows that the output error is the weighted product of the primal and dual errors, where the weight is specified by the bilinear form. Both expressions highlight the importance of controlling the behaviors of not only the primal solution but also the adjoint solution. Starting from the error representation formula, two classes of output-based error estimation techniques have been developed.

The first class of methods, developed by Patera, Peraire, and their collaborators, cast the output error estimation problem as a constrained minimization problem with a convex functional that naturally arises from coercive PDEs. The original method provides strict bounds of the output error for the fine “truth” mesh solution [101, 116]. Later, by incorporating the complementary variational principle, Budge and Peraire extended the method to provide strict bounds of the true output for the PDE solution [128, 129]. A different perspective on the same idea presented in Chapter 8 of Ainsworth and Oden [1] shows that the method is equivalent to applying the parallelogram identity to expression (II) of Eq. (1.1) and bounding the primal and dual errors using the equilibrated residual method. While this class of methods provide strict output bounds, i.e. a true certificate of the output estimate, further research is necessary to apply the method to PDEs that do not induce a natural



convex functional, e.g. the compressible Navier-Stokes equations with a semi-definite diffusion tensor. As the goal of this work is to develop an adaptation framework that works on a wide class of PDEs, including the Navier-Stokes equations, this class of bounding methods is not used in this work.

The second class of methods, developed by Becker and Rannacher, is called the dual-weighted residual (DWR) method [26, 27]. The method appeals directly to expression (I) of Eq. (1.1) to provide an error estimate, and the true adjoint  $\psi$  is approximated by the adjoint obtained on a finer discretization space. Within the DG framework, a common choice is to use  $\psi_{h,p+1} \in V_{h,p+1}$  obtained by increasing the polynomial degree. Unlike the method of Patera and Peraire, the DWR method does not provide a strict bound and only estimates the output error. However, it can in principle be applied to any PDE, including those that do not induce a natural convex functional. The applicability of the DWR framework to a wide class of engineering problems is demonstrated by, for example, Bangerth and Rannacher [16] and Giles and Süli [66]. All output-based adaptation cases considered in this work use the DWR error estimate to drive the adaptation.

### 1.3.3 Adaptation Mechanics

The goal of mesh adaptation is to control and effectively minimize the output error or, more precisely, minimize the output error estimate. The localized error estimate — which assigns a single scalar value that indicates the magnitude of the error contribution to each element — can readily drive isotropic adaptation, where elements with large error are refined and those with small error are coarsened. In terms of generality, the isotropic adaptation strategy in principle inherits the versatility of the underlying discretization and error estimate. The effectiveness of the isotropic adaptation strategies that incorporate the DWR error estimate have been demonstrated in numerous early studies [71, 146, 147]. Applicability of the approach to complex three-dimensional aerodynamic simulations has been validated by Nemec and Aftosmis [107] and Wintzer *et al.* [153].

While a simple isotropic adaptation strategy works well for a number of PDEs, the inability to provide anisotropic resolution is a major limitation for processes that exhibit strong directional features due to, e.g., a singularity or singular perturbation. By incorporating anisotropic adaptivity, the number of elements required to resolve the directional features can be significantly reduced. For example, high Reynolds number aerodynamic



flows exhibit features such as boundary layers, wakes, and shocks, where the resolution requirement in orthogonal directions differ by several orders of magnitude. Specifically, for a typical high Reynolds number flow ( $Re_c \geq 10^6$ ) over an aircraft wing, the mesh resolution requirement in the wall normal direction is about 1000 times higher than those in the streamwise or spanwise directions. Thus, appropriately shaped anisotropic elements can reduce the required number of degrees of freedom relative to isotropic elements by a factor of  $10^6$  in three dimensions. In other words, isotropic elements would require  $10^6$  times more degrees of freedom, rendering this typical three-dimensional problem intractable. To drive anisotropic adaptation, the adaptation process must control not only element sizes but also element stretchings and orientations. This section provides a brief summary of anisotropic mesh generation technologies and anisotropy detection methods currently available.

### **Curved Anisotropic Mesh Generation**

In order to support adaptivity for problems exhibiting anisotropic features using a higher-order discretization, the mesh generator must meet two requirements. First, the mesh generator must be capable of generating anisotropic elements whose anisotropy matches that of the local solution. Second, the mesh must capture higher-order geometry information of curved surfaces; the degradation in the solution quality resulting from the use of linear meshes (i.e. straight-edged meshes) for higher-order discretizations is documented by Bassi and Rebay [23]. The selection of adaptation strategies is largely dictated by the available anisotropic mesh generation technologies. Currently, there are two meshing strategies that can produce anisotropic elements in adaptive setting: simplex-based remeshing and quadrilateral-based hierarchical subdivision.

#### ***Simplex-Based Remeshing***

The first class of adaptive meshing strategy employs simplex elements to tessellate a domain. In terms of the initial mesh generation (i.e. geometry definition to computational mesh), simplices currently offer greater flexibility in meshing complex geometries than quadrilaterals. However, in the context of anisotropic adaptation, simplices cannot be locally subdivided anisotropically while maintaining element quality. Thus, to produce high-quality anisotropic elements, local remeshing that modifies multiple elements or fully global remeshing must be performed. This remeshing step can compromise the robustness of the adaptive



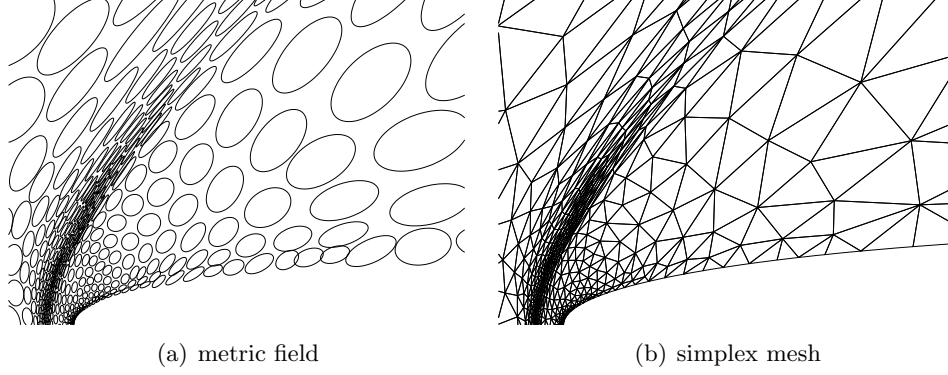


Figure 1-2: An example of a metric-mesh pair.

procedure, as meshing of complex curved geometries remains a challenging problem. However, remeshing offers an opportunity to produce arbitrarily oriented anisotropy, which can be critical to efficiently resolving features whose direction is not known *a priori*, e.g. oblique shocks and wakes in compressible flows.

In typical anisotropic simplex mesh generation, the desired characteristics of the next mesh are encoded as a Riemannian metric field, whose precise definition is provided in Section 2.3. The Riemannian metric field  $\{\mathcal{M}(x)\}_{x \in \Omega}$ , or a field of spatially varying symmetric positive definite (SPD) matrices, provides anisotropic specification of the desired element properties, i.e. size, stretching, and orientation. Specifically, given a metric field, the objective of mesh generation is to tessellate a domain using elements with unit edge length with respect to the Riemannian metric field [63, 119]. While a mesh that conforms to a given metric field is not unique, a family of metric-conforming meshes have similar approximation properties [97, 98]. An example of a metric-mesh pair is shown in Figure 1-2, where the field of SPD tensors is illustrated by ellipses.

The key enabling technology for simplex-based remeshing is a reliable metric-conforming simplex mesh generator. In two dimensions, the metric-mapped Delaunay triangulation algorithm has been successfully implemented in the Bidimensional Anisotropic Mesh Generator (BAMG) [30, 73]. Combined with an elasticity-based mesh curving algorithm that recovers high-order geometry information [110, 122], BAMG generates simplex meshes suitable for high-order discretizations. In three dimensions, the introduction of anisotropic adaptation strategies based on local remeshing in the past decade has significantly improved ability to generate anisotropic linear meshes in a reliable manner. Some of the seminal work on this subject include [31, 94, 113, 140]. So far, few work has considered gen-



eration of tetrahedral meshes with curved anisotropic elements in the context of adaptation, with notable exceptions of an elasticity-based approach by Michal and Krakos [104] and a “cut-cell”-based approach of Fidkowski [57]. The absence of a robust higher-order simplex mesh generator currently limits the capability of fully-automated simplex-based adaptation in three dimensions.

### ***Quadrilateral-Based Hierarchical Subdivision***

The second class of adaptive meshing strategy uses quadrilateral elements<sup>1</sup> with hanging nodes to tessellate a domain. Automated generation of an initial mesh of a general geometry using quadrilateral elements is considerably more difficult than using simplices. However, for aerospace applications, quadrilateral meshes for typical geometries are often available due to the prevalence of structured mesh solvers in the community. However, the reliance on these initial meshes is far from ideal for a general purpose PDE solver intended to explore radically different designs. The initial quadrilateral mesh generation is a problem that must be overcome to use the strategy in a fully automated adaptive PDE solver.

Once the initial mesh is in place, the quadrilateral meshes can be anisotropically refined through elemental subdivisions. The elemental subdivision process is robust and preserves element quality. Furthermore, the child elements naturally inherit the high-order geometry information of the parent element. An example of a mesh generated through anisotropic subdivisions of elements is shown in Figure 1-3. Note that the anisotropy is constrained by the initial mesh topology. The effect of limited anisotropy direction is expected to be minor for features that naturally align with geometry, such as boundary layers. However, the efficiency loss may be significant for arbitrarily oriented features, such as wakes and oblique shocks.

### **Anisotropic Adaptation: Interpolation-Based**

While the focus of this work is output-based adaptation, let us first provide a brief review of interpolation-based adaptation strategies, which played a key role in the development of simplex-based anisotropic adaptation techniques. As noted in Section 1.3.2, Peraire *et al.* pioneered the use of the Hessian of a select solution field to control the desired anisotropic

---

<sup>1</sup>The term “quadrilateral” should be understood as hexahedrons in three-dimensions throughout this section.



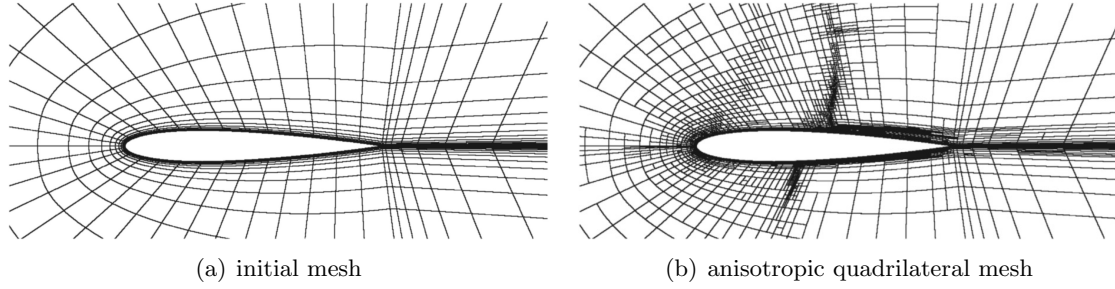


Figure 1-3: An example of anisotropic quadrilateral mesh generated by hierarchical subdivisions (reproduced with permission from [40]).

mesh configuration [119]. The objective of the interpolation-based anisotropic mesh adaptation is to equidistribute the Hessian-based interpolation error estimate computed on each edge by modifying the edge lengths and orientations.

Subsequently, interpolation-based anisotropic adaptation has been implemented with various discretizations, Hessian-recovery techniques, and mesh generation strategies, improving the robustness and applicability of the method to complex industrial applications. For example, Castro-Diaz *et al.* combined the Hessian of multiple solution fields to remove the arbitrary selection of a solution field [39]. More recently, the advancement of mesh generation technology based on local remeshing has enabled application of the interpolation-based adaptive method to complex three-dimensional problems, including: subsonic Euler flows [140], time-dependent flow problems [113], phase change problems [28], supersonic Euler flows [99], and time-dependent shock-propagation problems [4].

### **Anisotropic Adaptation: Output-Based**

The DWR error estimate, which assigns a single scalar value to each element, is insufficient for making anisotropy decisions. This section reviews anisotropy detection strategies developed in the context of output-based adaptation.

#### ***Anisotropy Detection by Solution Hessian***

In order to overcome the lack of information necessary to make anisotropy decisions, several researchers have proposed strategies that combine the DWR error estimate with Hessian-based anisotropy detection. Venditti and Darmofal combined the DWR technique with an anisotropy detection based on the Hessian of the Mach number to drive output-based adaptation for second-order finite volume discretization of the compressible Navier-Stokes



equations [148]. The work demonstrated the effectiveness and improved reliability of the output-based anisotropic adaptation compared to the interpolation-based anisotropic adaptation. Subsequently, Jones *et al.* validated the effectiveness of the approach using three-dimensional industrial aerospace applications [88]. Fidkowski and Darmofal later generalized the method to higher-order discretizations by using the higher derivative of the Mach number to guide their anisotropy decision [58]. Namely, for the degree- $p$  polynomial space, the  $p + 1$  derivative of Mach number is recovered by  $H^1$ -patch reconstruction and is used to make the anisotropy decision. Oliver [110] and Barter [18] improved the quality of  $p + 1$  derivative reconstruction for curved elements by using Jacobi smoothing instead of the patch reconstruction. The Hessian (or higher derivative) recovery technique can in principle be combined with either the simplex-based remeshing or quadrilateral-based hierarchical subdivision techniques. However, all work mentioned above used simplex remeshing.

Leicht and Hartmann introduced a variant of this strategy specifically for DG discretizations on quadrilateral meshes [92]. The anisotropy is detected by comparing the jump in Mach number across the element interfaces in two orthogonal directions. If the ratio of the jumps exceeds a prescribed threshold, then the quadrilateral element is subdivided anisotropically in the direction of larger jump. This strategy relies on tensor-product elements and must be used with the quadrilateral-based hierarchical subdivision technique. The strategy has been successfully applied to turbulent flow over a wing-body configuration, realizing the first output-based adaptive DG simulation of three-dimensional turbulent flow [72].

The Hessian- or jump-based anisotropy detection is unsuited for general PDE solver for a number of reasons. First, it assumes that a single scalar quantity, e.g. the Mach number, captures anisotropic behavior of the solution; the existence of such a scalar field is not guaranteed, and, even if it existed, the *a priori* knowledge of the solution behavior would be required to identify the variable. Second, in the context of output-based adaptation, it does not account for the anisotropic behavior of the adjoint solution, which is just as important as primal solution in controlling the output error as shown in Eq. (1.1). Third, by using the  $p + 1$  derivative, it assumes that the flow feature is resolved and the approximation is in the asymptotic range; this is often not the case in the early stage of adaptation and results in a lack of robustness particularly for higher-order discretizations.



### ***Anisotropy Adaptation using a Hessian-Based Error Model***

Another approach to performing anisotropic adaptation is to incorporate the solution Hessian information into the DWR error representation. Formaggia *et al.* derived a local error estimate that explicitly includes the Hessian of the dual solution for linear finite element discretization of the advection-diffusion equation [62] and the Stokes equations [61]. The element shape is then chosen to minimize the local error expression. Loseille *et al.* introduced a variant of this approach for the second-order finite volume discretization, in which the primal Hessian is weighted by the dual solution [96].

While the approaches devised by Formaggia *et al.* and Loseille *et al.* require reconstruction of solution Hessians, their adaptation principle is fundamentally different from the Hessian-based anisotropy detection discussed previously. The approaches considered by Formaggia *et al.* and Loseille *et al.* first construct an error model that captures the anisotropic behavior of the error, and then choose the anisotropic configuration that minimizes the error estimated by the model. Thus, anisotropy realized is not a result of fitting to a scalar quantity chosen *a priori*, but rather as a consequence of minimizing the anisotropic error estimate. The approaches cast the anisotropic adaptation problem as an error minimization problem in which the sizing and anisotropy decisions are treated in a unified manner.

Loseille and Alauzet in particular formalize this error minimization problem by introducing the concept of continuous mesh [97, 98]. The duality between a discrete mesh and continuous Riemannian metric field is established by studying the approximation properties of piecewise linear polynomials on a mesh conforming to the metric field. By incorporating the continuous representation of the approximation error to the Hessian-based anisotropic error model, Loseille *et al.* cast the output-error minimization problem as an optimization problem of the continuous metric field [96]. In Chapter 3, we will adopt this interpretation of anisotropic simplex adaptation as a continuous optimization problem.

In order to construct their error models, both Loseille *et al.* and Formaggia *et al.* take advantage of relatively simple output error expressions for second-order discretizations. The error expression becomes increasingly complex for higher-order discretizations as we will see in Section 2.3.4, rendering a direct extension of their approach difficult. The sensitivity of higher-order discretizations to irregular features can also compromise the robustness of the error model based on higher derivatives. The application of their approaches to arbitrary-



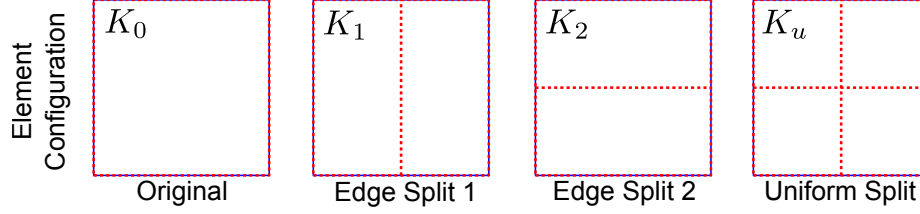


Figure 1-4: The original, anisotropic split, and isotropic split configurations for quadrilateral-based hierarchical subdivision strategy.

order discretizations is yet to be demonstrated.

### *Anisotropic Competitive Subdivision by Local Sampling*

The anisotropic competitive subdivision strategy directly controls the output error by considering anisotropic discrete refinement options on quadrilateral elements. This strategy works with the quadrilateral-based hierarchical subdivision meshing strategy. The algorithm starts by first marking a fraction of elements with largest error estimates (typically the top 10-20%). Then, for each element marked for refinement, the algorithm performs element-wise local solves on configurations obtained by locally splitting the edges; examples of such configurations are shown in Figure 1-4. A local solve of the governing PDE is performed by freezing the states on neighbor elements. Once the local solutions are obtained, the local error estimate is recomputed and recorded. The algorithm then chooses the local configuration that is most competitive in terms of the ratio of the error and the number of degrees of freedom. This strategy was first proposed by Houston and his collaborators [64, 65, 74]. A variant of strategy has been successfully applied to aerodynamic flows by Ceze and Fidkowski [40].

The anisotropic competitive subdivision strategy eliminates many of the shortcomings of Hessian-based anisotropy detection. The strategy solves a discrete output-error minimization problem in a greedy manner by directly monitoring the behavior of the local error estimate. The error estimate naturally incorporates the information about both the primal and adjoint solutions as well as behavior of all components of state for a system of equations. Furthermore, the strategy does not assume the solution is in the asymptotic range, making more robust decisions for higher-order discretizations. Our adaptation strategy developed in Chapter 3 inherits many of the advantages of this local sampling based strategy.

One major drawback of the competitive subdivision strategy is that it works only with



quadrilateral-based meshes, in which the tensor-product type elements allows natural directional sampling of the local error behavior. Thus, the strategy inherits the limitations of the quadrilateral-based hierarchical subdivision meshing. For example, only the features that align with the initial mesh topology can be effectively detected and refined, and the degree of coarsening is also limited by the initial mesh. Park [114] and Sun [139] attempted a similar competitive refinement strategy on simplex meshes by using edge split operations. However, the method performed poorly due to the negative feedback of irregular meshes generated in anisotropic hierarchical subdivision of simplex elements and noise in error estimates on such irregular meshes.

### ***Anisotropic Error Estimate***

Yet another approach to anisotropy detection is to split the DWR error estimate into contributions from different coordinate directions. Richter constructed a directional error estimate by performing  $p+1$  solution recovery in each direction of tensor-product type element [125]. The configurations considered for recovery are the same as the edge split configurations in Figure 1-4. A variant of the strategy was successfully adopted to a DG discretization by Leicht and Hartmann [93] and applied to compressible Navier-Stokes flows. These strategies are computationally more efficient than the competitive subdivision strategy as it does not require explicit local solves. The approach captures behavior of both the primal and adjoint solutions. However, similar to the competitive subdivision strategy, the anisotropic error estimate only works with quadrilateral-based meshes.

## **1.4 Thesis Overview**

This thesis presents work toward development of a versatile, adaptive PDE solver. In particular, the thesis presents a versatile, anisotropic adaptation algorithm, Mesh Optimization via Error Sampling and Synthesis (MOESS), which works particularly well with the DG discretization and the DWR error estimate. The specific contributions of the thesis are as follows.

- Development of an anisotropic adaptation algorithm that works with any localizable error estimate, handles any discretization order, permits arbitrarily oriented anisotropic simplex elements, robustly treats irregular features, and inherits the ver-



satility of the underlying discretization and error estimate.

- Presentation of a general parametrized anisotropic error model based on an affine-invariant interpolation of metric tensors and a scale-invariant interpolation of errors, and incorporation of a regression-based procedure to identify the parameter through local anisotropic metric-error sampling.
- Analysis of the optimal  $L^2$ -error anisotropic element size distribution for problems with canonical singularities and singular perturbations, and verification of MOESS to produce optimal meshes.
- Quantification of the importance of mesh adaptation for high-order discretizations of aerodynamic flows, and demonstration of the robustness and effectiveness of MOESS for this class of highly nonlinear systems exhibiting a wide range of scales.
- Realization of fully-unstructured space-time adaptivity using MOESS, and demonstration of the competitiveness of the formulation for linear and nonlinear wave propagation problems through the use of space-time anisotropy.
- Realization of spatial error control for Galerkin- (polynomial chaos) and collocation-based (reduced basis) parameter-space discretizations of parameterized PDEs using MOESS, and generation of universal optimal meshes suitable for a wide range of parameters.

This thesis is organized as follows. Chapter 2 reviews the discontinuous Galerkin discretization and the *a posteriori* error estimation method used in this work. The chapter further presents *a priori* error analysis of  $L^2$  projection error and output error for an arbitrary-order DG discretization of a system of conservation laws. Chapter 3 develops the MOESS anisotropic adaptation strategy. The chapter details the exploitation of the continuous mesh framework in formulating a mesh optimization problem and the incorporation of a novel tensor interpolation framework in describing the anisotropic error behavior. Then, Chapter 4 develops an optimal anisotropic element size distribution for approximating canonical singularities and singular perturbations frequently encountered in PDE solutions and verifies the MOESS's ability to generate optimal meshes. Chapter 5 studies the behavior of the anisotropic adaptation algorithm for a scalar advection-diffusion equation with manufactured primal and dual solutions. Comparison of MOESS with adaptive algorithms based



on isotropic elements and primal-based anisotropy detection highlights the effectiveness of anisotropic refinement and the consequence of neglecting the dual-solution behavior in determining the anisotropy. Chapter 6 first highlights the importance of mesh adaptation for high-order discretizations of aerodynamic flows. Then, the improved efficiency and robustness of MOESS compared to an adaptation strategy based on primal-based anisotropy detection is demonstrated through a number of complex aerodynamic flows governed by the Euler, Navier-Stokes, and Reynolds-averaged Navier-Stokes equations. Taking advantage of its versatility, Chapter 7 applies MOESS to a fully-unstructured space-time formulation of the wave and Euler equations, realizing fully-unstructured anisotropic space-time adaptivity for wave propagation problems. Chapter 8 applies MOESS to PDEs describing the system behavior over a wide range of parameters, initiating work towards reliable, fully-automated parameter-space exploration. Finally, Chapter 9 summarizes the conclusions of this work and suggests areas of future work.



## Chapter 2

# Discretization, *A Posteriori* Output Error Estimation, and Continuous Mesh Framework

This chapter reviews the discretization and the error estimation method used in this work. As discussed in Chapter 1, we employ a combination of the discontinuous Galerkin (DG) finite element method and the dual-weighted residual (DWR) error estimation method. The combination provides the versatility necessary for a fully-automated PDE solver of general conservation laws. We also present key results from the *a priori* output error analysis.

### 2.1 Discretization

Let  $\Omega \subset \mathbb{R}^d$  be the  $d$ -dimensional spatial domain and  $I \subset \mathbb{R}$  be the time interval of interest. A general system of conservation laws is of the form

$$\frac{\partial u}{\partial t} + \nabla \cdot \mathcal{F}^{\text{conv}}(u, x, t) - \nabla \cdot \mathcal{F}^{\text{diff}}(u, \nabla u, x, t) = \mathcal{S}(u, \nabla u, x, t) \quad \forall x \in \Omega, t \in I, \quad (2.1)$$

with the boundary conditions

$$\mathcal{B}(u, \hat{n} \cdot \mathcal{F}^{\text{diff}}(u, \nabla u, x, t), x, t; \text{BC}) = 0, \quad \forall x \in \partial\Omega, t \in I,$$



where  $u(x, t) \in \mathbb{R}^m$  is the state variable with  $m$  components,  $\mathcal{F}^{\text{conv}}$  is the convective flux,  $\mathcal{F}^{\text{diff}}$  is the diffusive flux,  $\mathcal{S}$  is the source, and  $\mathcal{B}$  imposes the boundary condition. The true output is given by

$$J = \mathcal{J}(u),$$

where  $\mathcal{J}$  is the output functional of interest, which can often be expressed as an integral quantity on surfaces or in the domain.

An approximation to the desired output is obtained by discretizing the conservation law and evaluating the discrete output functional. In particular, we seek a solution in a finite-dimensional approximation space  $V_{h,p}$  defined on a triangulation  $\mathcal{T}_h$  of the domain  $\Omega$  into non-overlapping elements  $\kappa$  of characteristic size  $h$ , i.e.

$$V_{h,p} = \{v_{h,p} \in [L^2(\Omega)]^m : v_{h,p} \circ f_\kappa^q \in [\mathcal{P}^p(\kappa_{\text{ref}})]^m, \forall \kappa \in \mathcal{T}_h\},$$

where  $\mathcal{P}^p$  denotes the space of complete polynomials of order  $p$ , and  $f_\kappa^q$  is the  $q$ -th degree polynomial parametric mapping from the reference element  $\kappa_{\text{ref}}$  to the physical element  $\kappa$ . The DG finite element method yields the weak form: Find  $u_{h,p} \in V_{h,p}$  such that

$$\mathcal{R}_{h,p}(u_{h,p}, v_{h,p}) = 0, \quad \forall v_{h,p} \in V_{h,p},$$

where  $\mathcal{R}_{h,p} : V_{h,p} \times V_{h,p} \rightarrow \mathbb{R}$  is the semilinear form corresponding to the conservation law. Throughout this work, we use Roe's approximate Riemann solver [126] for the convective numerical flux, the second discretization of Bassi and Rebay (BR2) [25] for the viscous flux, and a mixed form of Bassi *et al.* [22] for the source function with  $\nabla u$  dependence, which is asymptotically dual-consistent [110].

Upon obtaining the DG solution  $u_{h,p} \in V_{h,p}$ , the desired output is estimated by

$$J_{h,p} = \mathcal{J}_{h,p}(u_{h,p}),$$

where  $\mathcal{J}_{h,p} : V_{h,p} \rightarrow \mathbb{R}$  is the discrete functional that maintains dual consistency [68, 100, 110]. The details of the discretization, the discrete solution strategy, and the output evaluation procedure are provided in Appendix A.



## 2.2 Dual-Weighted Residual Method

### 2.2.1 Error Estimation

The objective of the functional error estimation is to approximate the true error in the output,

$$\mathcal{E}_{\text{true}} \equiv J - J_{h,p} = \mathcal{J}(u) - \mathcal{J}_{h,p}(u_{h,p}).$$

This work relies on the dual-weighted residual (DWR) method of Becker and Rannacher [26, 27] to estimate the output error. In DWR, the output error is expressed as

$$\mathcal{E}_{\text{true}} = \mathcal{J}(u) - \mathcal{J}_{h,p}(u_{h,p}) = -\mathcal{R}_{h,p}(u_{h,p}, \psi), \quad (2.2)$$

where  $\psi \in W \equiv V + V_{h,p}$  is the adjoint satisfying

$$\overline{\mathcal{R}}'_{h,p}[u, u_{h,p}](w, \psi) = \overline{\mathcal{J}}'_{h,p}[u, u_{h,p}](w), \quad \forall w \in W. \quad (2.3)$$

Here,  $\overline{\mathcal{R}}'_{h,p}[u, u_{h,p}] : W \times W \rightarrow \mathbb{R}$  and  $\overline{\mathcal{J}}'_{h,p}[u, u_{h,p}] : W \rightarrow \mathbb{R}$  are the mean-value linearized semilinear form and output functional, respectively, given by

$$\begin{aligned} \overline{\mathcal{R}}'_{h,p}[u, u_{h,p}](w, v) &= \int_0^1 \mathcal{R}'_{h,p}[(1-\theta)u + \theta u_{h,p}](w, v) d\theta \\ \overline{\mathcal{J}}'_{h,p}[u, u_{h,p}](w) &= \int_0^1 \mathcal{J}'_{h,p}[(1-\theta)u + \theta u_{h,p}](w) d\theta, \end{aligned}$$

where  $\mathcal{R}'_{h,p}[z](\cdot, \cdot)$  and  $\mathcal{J}'_{h,p}[z](\cdot)$  denote the Fréchet derivative of  $\mathcal{R}_{h,p}(\cdot, \cdot)$  and  $\mathcal{J}_{h,p}(\cdot)$  with respect to the first argument evaluated about  $z$ . Note that, by Galerkin orthogonality, Eq. (2.2) may be expressed as

$$\mathcal{E}_{\text{true}} = -\mathcal{R}_{h,p}(u_{h,p}, \psi - v_{h,p}), \quad \forall v_{h,p} \in V_{h,p},$$

or, by the definition of mean-value linearization,

$$\mathcal{E}_{\text{true}} = -\overline{\mathcal{R}}'_{h,p}[u, u_{h,p}](u - u_{h,p}, \psi - v_{h,p}), \quad \forall v_{h,p} \in V_{h,p}.$$



The expression signifies that the true error is a function of not only the error in the primal solution,  $u - u_{h,p}$ , but also the error of approximating the adjoint  $\psi$  in  $V_{h,p}$ . Thus, effective control of the output error requires an approximation space  $V_{h,p}$  that is suited for controlling both the primal and adjoint errors.

Note that the true adjoint,  $\psi \in W$  is not computable in general, as  $W$  is infinite dimensional and Eq. (2.3) requires the true primal solution  $u$ . For the purpose of error estimation, the true adjoint is replaced by an approximate adjoint  $\psi_{h,\hat{p}} \in V_{h,\hat{p}}$  that satisfies

$$\mathcal{R}'_{h,\hat{p}}[u_{h,p}](v_{h,\hat{p}}, \psi_{h,\hat{p}}) = \mathcal{J}'_{h,\hat{p}}[u_{h,p}](v_{h,\hat{p}}), \quad \forall v_{h,\hat{p}} \in V_{h,\hat{p}}, \quad (2.4)$$

where  $V_{h,\hat{p}} \supset V_{h,p}$  with  $\hat{p} = p + 1$  is the enriched space. The DWR error estimate of the output is given by substituting this approximate adjoint to Eq. (2.2), i.e.

$$\mathcal{E}_{\text{true}} \approx -\mathcal{R}_{h,p}(u_{h,p}, \psi_{h,\hat{p}}).$$

The quality of the error estimate depends on the linearization error arising from replacing the mean-value linearized functionals with  $u_{h,p}$ -linearized functionals and the error of approximating the adjoint in  $V_{h,\hat{p}} \subset W$ . The linearization and the adjoint-approximation errors may be significant, especially on coarse meshes; however, in practice, the error estimate is sufficiently accurate for the purpose of mesh adaptation. A detailed analysis of the linearization error is provided in [27].

### 2.2.2 Error Localization

To perform mesh adaptation, the error estimate must be localized to identify regions with large and small contributions to the error. To this end, the error estimate is localized through element-wise restriction of the adjoint weight. The local error estimate,  $\eta_\kappa$ , associated with element  $\kappa$  is defined by

$$\eta_\kappa \equiv |\mathcal{R}_{h,p}(u_{h,p}, \psi_{h,\hat{p}}|_\kappa)|. \quad (2.5)$$



The agglomeration of the locally positive error estimate results in a conservative error estimate,

$$\mathcal{E} \equiv \sum_{\kappa \in \mathcal{T}_h} \eta_\kappa.$$

Note that in Eq. (2.5), the residual is computed about  $p$  instead of  $\hat{p}$  such that the resulting estimate is both globally and locally convergent, as presented in Appendix E.

Due to the element-wise Galerkin orthogonality of the DG discretization, the localized error estimate can be expressed as

$$\begin{aligned} \eta_\kappa &= \inf_{v_{h,p} \in V_{h,p}} |\mathcal{R}_{h,p}(u_{h,p}, (\psi_{h,\hat{p}} - v_{h,p})|_\kappa)| \\ &= \inf_{v_{h,p} \in V_{h,p}} |\overline{\mathcal{R}}'_{h,p}[u, u_{h,p}](u - u_{h,p}, (\psi_{h,\hat{p}} - v_{h,p})|_\kappa)|, \end{aligned}$$

Again, the expression signifies that the localized error is a weighted product of the local primal error and the local adjoint error, and effective control of the output error requires  $V_{h,p}$  that accounts for the behaviors of both the primal and adjoint solutions.

## 2.3 Continuous Mesh Framework

The success of metric-based adaptation algorithms relies on the fact that the metric field controls the ability of a metric-conforming tessellation to approximate functions. In the first two subsections (Section 2.3.1 and 2.3.2), we provide a definition of metric-conforming meshes, introducing a geometric relationship between a Riemannian metric field and the corresponding anisotropic mesh. The following two subsections (Section 2.3.3 and 2.3.4) establish that both approximation and output errors incurred on a metric-conforming mesh can be approximated in terms of the Riemannian metric field. The result justifies a continuous relaxation of the mesh optimization problem (which is inherently discrete), the approach we will pursue in designing our adaptation algorithm in Chapter 3. Using the term coined by Loseille and Alauzet [97, 98], we will refer to this framework that enables continuous interpretation of a discrete mesh as the *continuous mesh framework*.



### 2.3.1 Metric-Conforming Meshes

Let us review the concept of Riemannian metric field used to encode an anisotropic description of element sizes in this work. Notation used in here — and throughout the rest of the chapter — closely follows that of Loseille and Alauzet [97, 98].

**Definition 2.1.** *A Riemannian metric field  $\{\mathcal{M}(x)\}_{x \in \Omega}$  is a smoothly varying field of symmetric positive definite (SPD) matrices on  $\Omega \subset \mathbb{R}^d$ . The length of a segment  $\vec{ab}$  from point  $a \in \Omega$  to  $b \in \Omega$  under the metric is given by*

$$\ell_{\mathcal{M}}(\vec{ab}) = \int_0^1 \sqrt{\vec{ab}^T \mathcal{M}(a + \vec{ab}s) \vec{ab}} ds.$$

**Definition 2.2** (metric conforming triangulation). *A metric-conforming triangulation is a triangulation such that all edges are close to unit length under the metric Riemannian field,  $\{\mathcal{M}(x)\}_{x \in \Omega}$ . Specifically, the mesh satisfies the edge-length condition*

$$\frac{1}{\sqrt{2}} \leq \ell_{\mathcal{M}}(e) \leq \sqrt{2}, \quad \forall e \in \text{Edges}(\mathcal{T}_h),$$

where  $\ell_{\mathcal{M}}(\cdot)$  is the length measure defined in Definition 2.1, and the element-quality condition,

$$Q_{\mathcal{M}}(\kappa) = \frac{(\int_{\kappa} \sqrt{\det(\mathcal{M}(x))} dx)^{2/d}}{\sum_{e \in \text{Edges}(\kappa)} \ell_{\mathcal{M}}^2(e)} \in [\alpha, 1] \quad \text{with } \alpha > 0,$$

where  $Q_{\mathcal{M}}$  is the element quality measure.

For a given  $\{\mathcal{M}(x)\}_{x \in \Omega}$ , the metric-conforming triangulation is not unique; however, the edge-length condition and the element-quality condition ensure that a family of metric-conforming triangulations have similar geometric configurations. This work uses Bidimensional Anisotropic Mesh Generator (BAMG) [73] developed by INRIA to generate all two-dimensional metric-conforming meshes and Edge Primitive Insertion and Collapse (EPIC) [104] developed by The Boeing Company to generate three-dimensional meshes. The initial, non-metric-conforming meshes are generated using Triangle developed by Shewchuk [132] in two dimensions and TetGen developed by Si [134] in three dimensions. For problems with curved geometries, the linear mesh is globally curved using linear elasticity to capture the higher-order geometry information [110, 122].



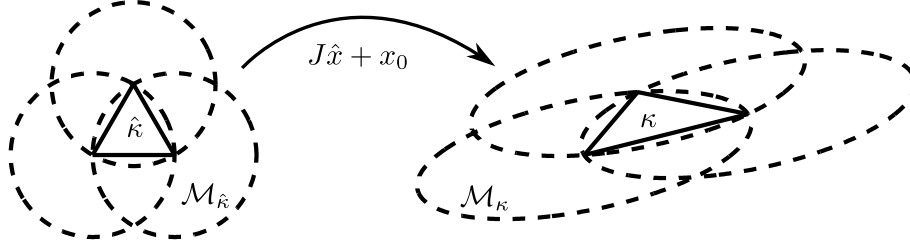


Figure 2-1: Illustration of the transformation of the reference element  $\hat{\kappa}$  into a physical element  $\kappa$ . The metric tensor associated with each element is shown in dashed lines.

### 2.3.2 Mesh-Conforming Metric Fields

Given a tessellation, it is also possible to find a metric field that conforms to the mesh. To perform the task, we first define the element-implied metric tensor as follows.

**Definition 2.3** (element-implied metric). *The element implied metric of a simplex element  $\kappa$ ,  $\mathcal{M}_{\kappa}$ , is a unique metric under which all edges of the elements are unit length, i.e.  $\mathcal{M}_{\kappa} \in \text{Sym}_d^+$  such that*

$$\sqrt{e^T \mathcal{M}_{\kappa} e} = 1, \quad \forall e \in \text{Edges}(\kappa), \quad (2.6)$$

where  $\text{Edges}(\kappa)$  is the set of  $d(d+1)/2$  edges of the simplex.

The uniqueness follows from the fact that a  $d$ -dimensional simplex has  $d(d+1)/2$  edges, which reduces satisfying Eq. (2.6) to solving a  $(d(d+1)/2)$ -by- $(d(d+1)/2)$  linear system for the coefficients of  $\mathcal{M}_{\kappa}$ . The linear system is non-singular as long as the simplex is non-degenerate. The relationship between an element and the associated metric tensor is illustrated in Figure 2-1. Assuming the element  $\kappa$  is part of a metric-conforming triangulation, the implied metric associated with  $\kappa$  is representative of the metric field over the region covered by  $\kappa$ .

A collection of element-implied metrics,  $\{\mathcal{M}_{\kappa}\}_{\kappa \in \mathcal{T}_h}$ , encodes the anisotropic element size information as a discontinuous field. Let us present our method to reconstruct a continuous metric field  $\{\mathcal{M}(x)\}_{x \in \Omega}$ , represented by the metrics associated with the vertices of the triangulation,  $\{\mathcal{M}_{\nu}\}_{\nu \in \mathcal{V}}$ , where  $\mathcal{V}$  is the set of vertices of the triangulation. To reconstruct vertex-based metrics  $\{\mathcal{M}_{\nu}\}_{\nu \in \mathcal{V}}$  from elemental metrics  $\{\mathcal{M}_{\kappa}\}_{\kappa \in \mathcal{T}_h}$ , we take the



affine-invariant mean of the elemental metrics of the elements surrounding the vertex, i.e.

$$\mathcal{M}_\nu = \text{mean}^{\text{affinv}}(\{\mathcal{M}_\kappa\}_{\kappa \in \omega(\nu)}).$$

Here  $\omega(\nu)$  is the set of elements surrounding the vertex  $\nu$ . The mean of the set of metrics is defined as the minimizer of the sum of the squared distance in the affine invariant sense, as defined by Pennec *et al.* [117], i.e.

$$\text{mean}^{\text{affinv}}(\{\mathcal{M}_\kappa\}_{\kappa \in \omega(\nu)}) = \arg \min_{\mathcal{M}} \sum_{\kappa \in \omega(\nu)} \|\log(\mathcal{M}_\kappa^{-1/2} \mathcal{M} \mathcal{M}_\kappa^{-1/2})\|_F^2.$$

The mean value is computed iteratively using the intrinsic gradient descent algorithm described in [117]. Then, we define a continuous metric field over element  $\kappa$  as a weighted mean of vertex matrices,

$$\mathcal{M}(x) = \arg \min_{\mathcal{M}} \sum_{\nu \in \mathcal{V}(\kappa)} w_\nu(x) \|\log(\mathcal{M}_\nu^{-1/2} \mathcal{M} \mathcal{M}_\nu^{-1/2})\|_F^2, \quad x \in \kappa,$$

where  $w_\nu(x)$  is the barycentric coordinate corresponding to the vertex  $\nu$ . Note that this element-wise continuous reconstruction results in a globally continuous metric field. Using these steps, we can generate either elemental (discontinuous) or vertex-based (continuous) representation of the metric field associated with a triangulation. The combination of a metric-conforming mesh generator, such as those described in Section 2.3.1, and the mesh-to-metric recovery algorithm described here completes the geometric duality between an anisotropic mesh and a Riemannian metric field.

### 2.3.3 Metric-based Representation of Polynomial Approximation Errors

Let us now introduce a key result that shows that the function approximation error incurred on a metric-conforming mesh is a function of the Riemannian metric field from which the mesh is generated. The result relies on the geometric duality of the discrete mesh and the Riemannian metric field and anisotropic polynomial approximation theory. Several variants of anisotropic polynomial approximation theory for piecewise-linear polynomials have been developed by, for example, Formaggia *et al.* [62], Shewchuk [133], Cao [35], and Loseille and Alauzet [97, 98]. An extension to higher-degree polynomials have been proposed by Houston



*et al.* [74], Pagnutti and Ollivier-Gooch [112], and Cao [36–38]. The formulation used in this work closely follows that of Houston *et al*; the details are omitted here for brevity but are provided in Appendix D.1. The main result is stated in the following theorem.

**Proposition 2.4.** *Let  $\mathcal{T}_h$  be a tessellation conforming to a metric field  $\{\mathcal{M}(x)\}_{x \in \Omega}$ . The  $H^n$  error that results from the  $L^2$  projection of a function  $v \in H^{k_v}(\Omega)$  to the piecewise degree- $p$  polynomial space defined on  $\mathcal{T}_h$ ,  $V_{h,p}$ , is approximated by,*

$$|v - \Pi_{h,p} v|_{H^n(\Omega)}^2 \lesssim C_{p,d} \int_{\Omega} (\lambda_{\max}(\mathcal{M}(x)))^{-n/2} E_{\mathcal{M}}^s(\mathcal{M}(x); v(x)) dx, \quad n = 0, 1,$$

where  $s = \min(p + 1, k_v)$ ,  $C_{p,d}$  is a constant that only depends on the polynomial degree  $p$  and the dimension  $d$ ,  $E_{\mathcal{M}}^s$  is the metric-based error kernel given by

$$E_{\mathcal{M}}^s(\mathcal{M}; v) = \sum_{i_1=1}^d \cdots \sum_{i_s=1}^d \left( \frac{\partial^s v}{\partial x_{j_1} \cdots \partial x_{j_s}} \mathcal{M}_{j_1 i_1}^{-1/2} \cdots \mathcal{M}_{j_s i_s}^{-1/2} \right)^2,$$

and  $\mathcal{M}^{-1/2}$  is the metric square root of  $\mathcal{M}$ . Summation on the repeated indices  $j_1, \dots, j_s$  is implied.

The proposition states that the continuous metric field  $\{\mathcal{M}(x)\}_{x \in \Omega}$  provides a convenient means of encoding the ability of the triangulation  $\mathcal{T}_h$  to approximate a function. In fact, the proposition is an extension of the continuous mesh interpolation error model established by Loseille and Alauzet [97, 98] for linear polynomials to arbitrary-degree polynomials. The duality not only justifies the metric-based continuous optimization framework for mesh adaptation, but also enables development of analytical expressions for optimal anisotropic element size distributions in Chapter 4.

### 2.3.4 *A Priori* Metric-based Representation of the Output Error

Let us now analyze the output error for a system of equations. For simplicity, we consider linear equations with constant coefficients and Dirichlet boundary conditions, i.e.

$$\begin{aligned} \nabla \cdot (\mathcal{A}u) - \nabla \cdot (\mathcal{K} \nabla u) + \mathcal{C}u &= 0, \quad \text{in } \Omega \\ u &= g, \quad \text{on } \partial\Omega, \end{aligned} \tag{2.7}$$



where  $\mathcal{A}_i \in \mathbb{R}^{m \times m}$ ,  $i = 1, \dots, d$ , is the flux Jacobian,  $\mathcal{K}_{ij} \in \mathbb{R}^{m \times m}$ ,  $i, j = 1, \dots, d$ , constitute the viscosity tensor, and  $\mathcal{C} \in \mathbb{R}^{m \times m}$  is the reaction matrix. Furthermore, we assume that the DG solution is optimal in  $H^1$  and  $L^2$  volume and face errors, as defined precisely by Assumption D.7. Then, the output error incurred by a metric-conforming triangulation can be expressed in terms of the governing Riemannian metric field as follows. (Details are provided in Appendix D.2.)

**Proposition 2.5.** *Let  $\mathcal{T}_h$  be a tessellation conforming to a metric field  $\{\mathcal{M}(x)\}_{x \in \Omega}$ . Furthermore, let the primal and adjoint solutions to the advection-diffusion-reaction system Eq. (2.7) be  $u \in H^{k_u}(\Omega)$  and  $\psi \in H^{k_\psi}(\Omega)$ , respectively. Assuming the DG approximation  $u_{h,p} \in V_{h,p}$  satisfies the optimality condition, Assumption D.7, the output error using the degree- $p$  DG discretization on  $\mathcal{T}_h$  is approximated by*

$$\begin{aligned} \mathcal{E} \lesssim & C \sum_{\kappa \in \mathcal{T}_h} \left[ \sum_{k=1}^m \sum_{i=1}^d \frac{|\lambda_k^{\mathcal{A}_i}|}{h_{\min, \kappa}} \left( \int_{\kappa} E_{\mathcal{M}}^{s_u}(\mathcal{M}(x); (r_k^{\mathcal{A}_i})^T u(x)) dx \right)^{1/2} \left( \int_{\kappa} E_{\mathcal{M}}^{s_\psi}(\mathcal{M}(x); (l_k^{\mathcal{A}_i})^T \psi(x)) dx \right)^{1/2} \right. \\ & + \sum_{f \in F(\kappa)} \sum_{k=1}^m \frac{|\lambda_k^{\mathcal{A}_{\hat{n}}^-}|}{h_{\min, \kappa}} \left( \int_{\kappa} E_{\mathcal{M}}^{s_u}(\mathcal{M}(x); (r_k^{\mathcal{A}_{\hat{n}}^-})^T u(x)) dx \right)^{1/2} \left( \int_{\kappa} E_{\mathcal{M}}^{s_\psi}(\mathcal{M}(x); (l_k^{\mathcal{A}_{\hat{n}}^-})^T \psi(x)) dx \right)^{1/2} \\ & + \sum_{k=1}^m \sum_{i,j=1}^d \frac{|\lambda_k^{\mathcal{K}_{ij}}|}{h_{\min, \kappa}^2} \left( \int_{\kappa} E_{\mathcal{M}}^{s_u}(\mathcal{M}(x); (r_k^{\mathcal{K}_{ij}})^T u(x)) dx \right)^{1/2} \left( \int_{\kappa} E_{\mathcal{M}}^{s_\psi}(\mathcal{M}(x); (l_k^{\mathcal{K}_{ij}})^T \psi(x)) dx \right)^{1/2} \\ & \left. + \sum_{k=1}^m |\lambda_k^{\mathcal{C}}| \left( \int_{\kappa} E_{\mathcal{M}}^{s_u}(\mathcal{M}(x); (r_k^{\mathcal{C}})^T u(x)) dx \right)^{1/2} \left( \int_{\kappa} E_{\mathcal{M}}^{s_\psi}(\mathcal{M}(x); (l_k^{\mathcal{C}})^T \psi(x)) dx \right)^{1/2} \right] \end{aligned}$$

where  $s_u = \min(p+1, k_u)$ ,  $s_\psi = \min(p+1, k_\psi)$ ,  $h_{\min, \kappa} = (\sup_{x \in \kappa} \lambda_{\max}(\mathcal{M}(x)))^{-1/2}$  is the minimum implied element length, and  $C$  depends only on the dimension  $d$  and the polynomial degree  $p$ . For an arbitrary matrix  $B$ ,  $\lambda_k^B$ ,  $r_k^B$ , and  $l_k^B$  denote the  $k$ -th eigenvalue, right eigenvector, and left eigenvector, respectively, i.e.  $B = \sum_{k=1}^m \lambda_k^B r_k^B (l_k^B)^T$ . The matrices  $\mathcal{A}_i$ ,  $\mathcal{K}_{ij}$ , and  $\mathcal{C}$  are those specifying the advection-diffusion-reaction system Eq. (2.7).

The proposition shows that, assuming the discretization is stable, the error associated with the DG approximation of an output is a function of the metric field,  $\{\mathcal{M}(x)\}_{x \in \Omega}$ . The behavior of the error bound is characterized by the higher derivatives of all components of the primal and dual solutions. In particular, the error is dictated by the  $p+1$  derivative of the solution in the smooth regions, whereas the lower-order derivatives comes into play in low-regularity regions. The manner in which these derivatives enter the output error is dependent on the modal decompositions of the flux Jacobian, viscosity tensor, and the source coefficient matrix. Thus, the relationship between the metric field,  $\{\mathcal{M}(x)\}_{x \in \Omega}$ , and



the *a priori* output error,  $\mathcal{E}$ , is complex and depends on many variables that are hard to approximate or not available, e.g. solution regularity and higher derivatives. The design of our adaptation scheme presented in Chapter 3 is motivated by the need to overcome these complexities.







## Chapter 3

# Mesh Optimization via Error Sampling and Synthesis

This chapter presents the anisotropic adaptation algorithm, Mesh Optimization via Error Sampling and Synthesis (MOESS).

### 3.1 Output Error Minimization Problem

#### 3.1.1 Problem Definition and Continuous Relaxation

In Section 2.2, we introduced the means of expressing discretization errors as a function of the approximation space,  $V_{h,p}$ . The objective of our adaptation is to find the space  $V_{h,p}^*$  that minimizes the error for a given dimension of  $V_{h,p}$ , i.e.

$$V_{h,p}^* = \arg \inf_{V_{h,p}} \mathcal{E}(V_{h,p}) \quad \text{s.t.} \quad \dim(V_{h,p}) \leq N,$$

where  $N$  is the maximum permissible dimension of  $V_{h,p}$ . In particular, if  $V_{h,p}$  consists of elements with a constant polynomial degree  $p$ , then  $V_{h,p}$  is described by the triangulation  $\mathcal{T}_h$  and the scalar  $p$ , i.e.  $V_{h,p} = V_{h,p}(\mathcal{T}_h, p)$ . Thus, for a fixed  $p \in \mathbb{R}^+$ , the optimization problem simplifies to that of finding the optimal triangulation  $\mathcal{T}_h^*$  such that

$$\mathcal{T}_h^* = \arg \inf_{\mathcal{T}_h} \mathcal{E}(V_{h,p}(\mathcal{T}_h, p)) \quad \text{s.t.} \quad \dim(V_{h,p}(\mathcal{T}_h, p)) \leq N. \quad (3.1)$$



This is a discrete-continuous optimization problem, as the triangulation  $\mathcal{T}_h$  is defined by the node locations and the connectivity of the nodes. In general, the problem is intractable.

In order to find an approximate solution to the problem, we consider a continuous relaxation of the discrete problem, following the approach pursued by Loseille *et al.* [96, 98]. In particular, we appeal to the fact that the Riemannian metric field  $\mathcal{M} = \{\mathcal{M}(x)\}_{x \in \Omega}$  controls the discretization error associated with a metric-conforming triangulation  $\mathcal{T}_h$  according to Propositions 2.4 and 2.5. Thus, we can cast a continuous relaxation of the discrete problem, Eq. (3.1), as

$$\mathcal{M}^* = \arg \inf_{\mathcal{M}} \mathcal{E}(V_{h,p}(\mathcal{M}, p)) \quad \text{s.t.} \quad \dim(V_{h,p}(\mathcal{M}, p)) \leq N.$$

For brevity, we write the optimization problem as

$$\mathcal{M}^* = \arg \inf_{\mathcal{M}} \mathcal{E}(\mathcal{M}) \quad \text{s.t.} \quad \mathcal{C}(\mathcal{M}) \leq N, \quad (3.2)$$

where  $\mathcal{E}$  and  $\mathcal{C}$  are the error and cost functionals that map the metric tensor field to the error and cost, respectively. The expression assumes that the polynomial degree,  $p$ , is constant and fixed. The extension of the continuous optimization framework to  $hp$ -adaptation would require introduction of the solution order field  $\{p(x)\}_{x \in \Omega}$ ; the extension is not considered in this work.

### 3.1.2 Error and Cost Functionals

In order to solve the optimization problem Eq. (3.2), we need a means of approximating the behavior of the error and cost functionals. If we use the number of degrees of freedom as the measure of cost, then the cost functional takes the form

$$\mathcal{C}(\mathcal{M}) = \int_{\Omega} c(\mathcal{M}(x), x) dx,$$

where  $c(\cdot, \cdot) : \text{Sym}_d^+ \times \mathbb{R}^d \rightarrow \mathbb{R}^+$  is the local cost function. In the view of the continuous mesh framework, the local cost function for the discontinuous piecewise polynomial space is given by

$$c(\mathcal{M}(x), x) = c_p \sqrt{\det(\mathcal{M}(x))}, \quad (3.3)$$



where  $c_p$  is the degrees of freedom associated with a reference element normalized by the size of the reference element,  $\hat{\kappa}$ . In particular, the coefficients associated with triangular and tetrahedral elements are

$$c_p^{\text{tri}} = \frac{2}{\sqrt{3}}(p+1)(p+2) \quad \text{and} \quad c_p^{\text{tet}} = \sqrt{2}(p+1)(p+2)(p+3),$$

respectively. These choices allow us to recover the correct elemental cost,  $\rho_\kappa$ , when the local cost function is integrated over an element, i.e.

$$\rho_\kappa = \int_\kappa c(\mathcal{M}(x), x) dx \approx \int_\kappa c_p \sqrt{\det(\mathcal{M}_\kappa)} dx = c_p |\hat{\kappa}| = \text{dof}(\hat{\kappa}).$$

To estimate the behavior of the error functional, we make an assumption that the functional results from a sum of the local contributions, i.e.

$$\mathcal{E}(\mathcal{M}) = \int_\Omega e(\mathcal{M}(x), x) dx, \tag{3.4}$$

where  $e(\cdot, \cdot) : \text{Sym}_d^+ \times \mathbb{R}^d \rightarrow \mathbb{R}^+$  is the local error function that maps the configuration described by  $\mathcal{M}(x)$  to the local contribution to the output error. This locality assumption is formally only applicable to errors that only depend on local properties, e.g.  $L^2$  projection errors. However, we have found that the algorithm developed based on the assumption works well in practice for output-based error estimates for DG discretizations. Under the locality assumption, we can write the elemental error contribution as

$$\eta_\kappa = \int_\kappa e(\mathcal{M}(x), x) dx \approx \int_\kappa e(\mathcal{M}_\kappa, x) dx \quad \Rightarrow \quad \eta_\kappa = \eta_\kappa(\mathcal{M}_\kappa).$$

That is, the elemental error — or the error associated with the region covered by  $\kappa$  — is a function of the metric  $\mathcal{M}_\kappa$  that encodes the approximation properties of the region covered by  $\kappa$ . The expression is consistent with the continuous error expressions for  $L^2$  projection and output errors based on *a priori* error analysis in Proposition 2.4 and 2.5, respectively. Thus, the form of our continuous error model is compatible with the expected local error behavior and is capable of representing the error behavior. Our primary task is to approximate the dependency of  $\eta_\kappa$  on  $\mathcal{M}_\kappa$ , to model the error functional, and then to minimize the functional.



### 3.1.3 Design Criteria and Approach

Our main goal is to design a versatile mesh optimization algorithm that works with a wide range of discretizations and error estimates. Let us use the results of the *a priori* error analysis for  $L^2$  projection error and the output error using the DG discretization presented in Section 2.3.3 and 2.3.4 to motivate our approach to designing the algorithm.

Let us first consider the case of  $L^2$  projection error. Proposition 2.4 states the error is dictated by  $s$ -th derivative of the solution, where  $s$  is dependent on the degree of the approximating polynomial and the regularity of the solution. Thus, direct use of the *a priori* error analysis result, Proposition 2.4, as the error kernel of Eq. (3.4) requires a means of estimating the regularity of the solution and evaluating the appropriate higher derivatives of the solution. In the context of  $L^2$  approximation error control, the true function is assumed to be accessible. Thus, these quantities could be estimated directly and the *a priori* error expression could serve as the error kernel.

In the case of output error, construction of the error model by the direct evaluation of the *a priori* error bound in Proposition 2.5 is more complicated, requiring a means of: estimating the regularity of all components of the primal and dual solutions; approximating the appropriate higher derivatives of all components of primal and dual solutions; and estimating the mean-value linearized flux Jacobian, viscosity tensor, and source matrix. While the first two tasks may appear the same as that for the  $L^2$  approximation error control, the tasks are complicated by the fact that the true primal and dual solutions are unknown. Some of the irregular features may be induced by nonlinearity, which makes the *a priori* estimation of the regularity impossible. While there are procedures for estimating the regularity (e.g. [75]) and reconstructing the higher derivatives (e.g. [58]), these quantities are hard to approximate and results may be unreliable, especially for higher-order discretizations. Thus, the direct evaluation and minimization of the *a priori* error bound — as done for second-order discretizations in [61] and [96] — is likely not a viable strategy for constructing the error kernel of Eq. (3.4) for a high-order discretization of a system of equations with irregular features.

A key observation is that, even for a high-order discretization whose output error depends on a large amount of data, the degrees of freedom that we can control — the  $d(d+1)/2$ -dimensional metric tensor — is the same as a lower-order discretization. Noting that the



degrees of freedom that we can control is significantly smaller than the amount of data governing the error behavior, our approach to capturing the anisotropic error behavior is not to estimate the complicated underlying dependencies, but rather to characterize the error behavior by directly monitoring the change in the error under the change in the metric tensor. We will accomplish this by solving local problems and re-evaluating the error estimates, adopting the idea developed for quadrilateral elements in [74] to simplices.

Using the direct error sampling strategy in the context of the continuous optimization framework described in Section 3.1.1 requires a means of constructing a continuous error-to-metric map using the samples collected. This is accomplished by an error model that builds on the tensor interpolation framework described in Section 3.2.1. Finally, optimization is performed using the surrogate error model to iterate toward a mesh that minimizes the error for a given degrees of freedom.

## 3.2 Optimization Algorithm: MOESS

### 3.2.1 Metric Manipulation Framework

Let us first introduce a framework for manipulating metric tensors (i.e. SPD matrices) based on the work by Pennec *et al.* [117]. The most intuitive method of manipulating a tensor may be to simply treat the tensor as an array of numbers and to directly modify the entries of the matrix in the standard Euclidean sense, i.e.

$$\mathcal{M} = \mathcal{M}_0 + \delta\mathcal{M},$$

where  $\mathcal{M}_0$  is the original matrix,  $\delta\mathcal{M}$  is the modification to the matrix, and  $\mathcal{M}$  is the new matrix. However, this method is unsuited for our purpose, as the update,  $\delta\mathcal{M}$ , must be chosen carefully to maintain the positive definiteness of the tensor. Furthermore, the entries of the update  $\delta\mathcal{M}$  are not strongly related to the change in the approximation property of the space. The approximability of the space equipped with a metric  $\mathcal{M}$  can be described using the directional length,  $h(\hat{e})$ , defined by

$$h(\hat{e}; \mathcal{M}) \equiv (\hat{e}^T \mathcal{M} \hat{e})^{-1/2},$$



where  $\hat{e}$  is a unit vector specifying the direction of interest. The change in the approximability in a given direction, or the ratio of the directional lengths between the configurations induced by  $\mathcal{M}$  and  $\mathcal{M}_0$ , is

$$\frac{h(e; \mathcal{M})}{h(e; \mathcal{M}_0)} = \left( \frac{e^T \mathcal{M}_0^{1/2} e}{e^T \mathcal{M}^{1/2} e} \right)^{1/2}.$$

With the entry-wise direct manipulation of  $\mathcal{M}_0$ , the change in this ratio of directional lengths is not strongly related to the magnitude of the entries of  $\delta\mathcal{M}$ .

Instead, we consider the tensor manipulation framework that results from endowing the tensor space with an affine-invariant Riemannian metric introduced by Pennec *et al.* [117]. The affine-invariant metric produces a manifold structure where matrices with zero and infinite eigenvalues are infinite distance from any SPD matrix and a geodesic joining any two tensors is unique. On the Riemannian manifold induced by the affine-invariant metric, the exponential map of a tangent vector  $S \in \text{Sym}_d$  in the tangent space about  $\mathcal{M}_0$  to the manifold is given by

$$\mathcal{M}(S) \equiv \mathcal{M}_0^{1/2} \exp(S) \mathcal{M}_0^{1/2}, \quad (3.5)$$

where  $\exp(\cdot)$  is the matrix exponential. Conversely, the logarithmic map of a tensor  $\mathcal{M}$  to the tangent space about  $\mathcal{M}_0$  is given by

$$S \equiv \log(\mathcal{M}_0^{-1/2} \mathcal{M} \mathcal{M}_0^{-1/2}),$$

where  $\log(\cdot)$  is the matrix logarithm. The distance between two tensors  $\mathcal{M}$  and  $\mathcal{M}_0$  is equal to  $\| \log(\mathcal{M}_0^{-1/2} \mathcal{M} \mathcal{M}_0^{-1/2}) \|_F$ , where  $\| \cdot \|_F$  denotes the Frobenius norm of the matrix [117]. As the tangent vector  $S$  specifies the change in the metric field,  $S$  is referred to as the *step matrix* from hereon. With this choice of  $S$ , the fractional change in the directional length is bounded by

$$\exp\left(-\frac{1}{2}\|S\|_F\right) \leq \exp\left(-\frac{1}{2}\lambda_{\max}(S)\right) \leq \frac{h(e; \mathcal{M}(S))}{h(e; \mathcal{M}_0)} \leq \exp\left(-\frac{1}{2}\lambda_{\min}(S)\right) \leq \exp\left(\frac{1}{2}\|S\|_F\right). \quad (3.6)$$

In other words, we can control the change in the directional approximability by controlling



the magnitude of  $S$ . The proof of the relationship is provided in Appendix F.1.

By decomposing the step matrix,  $S$ , into the isotropic and the tracefree parts, we can gain a better insight into the manipulation of the tensors in the tangent space. Let us denote the decomposition by

$$S = sI + \tilde{S},$$

where  $I$  is the identity matrix, and  $s = \text{tr}(S)/d$  such that  $\text{tr}(\tilde{S}) = 0$ . The exponential map of the decomposed step tensor yields

$$\mathcal{M}(sI + \tilde{S}) = \mathcal{M}_0^{1/2} \exp(sI + \tilde{S}) \mathcal{M}_0^{1/2} = \exp(s) \mathcal{M}_0^{1/2} \exp(\tilde{S}) \mathcal{M}_0^{1/2}.$$

The expression shows that the isotropic part,  $sI$ , simply scales the resulting tensor while preserving the shape. In contrary, the change induced by the tracefree part,  $\tilde{S}$ , modifies the shape while preserving the volume, or the determinant, i.e.

$$\begin{aligned} \det(\mathcal{M}(\tilde{S})) &= \det(\mathcal{M}_0^{1/2} \exp(\tilde{S}) \mathcal{M}_0^{1/2}) = \det(\mathcal{M}_0) \det(\exp(\tilde{S})) \\ &= \det(\mathcal{M}_0) \exp(\text{tr}(\tilde{S})) = \det(\mathcal{M}_0). \end{aligned}$$

The decomposition yields a convenient means of manipulating the size and the shape separately, which we exploit in designing the optimization algorithm.

### 3.2.2 Local Error Sampling

The goal of the local error sampling step is to probe the behavior of the local elemental error,  $\eta_\kappa$ , as a function of the local metric,  $\mathcal{M}_\kappa$ . Here, we probe the functional dependency by directly monitoring the behavior of the elemental error or *a posteriori* error estimate for several different local configurations. Let us first describe the procedure in context of  $L^2$  error control.

We consider  $n_{\text{config}}$  configurations obtained by locally splitting the edges, as shown in Figures 3-1 and 3-2 for two- and three-dimensional cases, respectively. We will denote the configuration obtained by the  $i$ -th local modification by  $\kappa_i$ . By convention,  $\kappa_0$  is the original configuration. For configuration  $\kappa_i$ , we solve the  $L^2$  projection problem to obtain



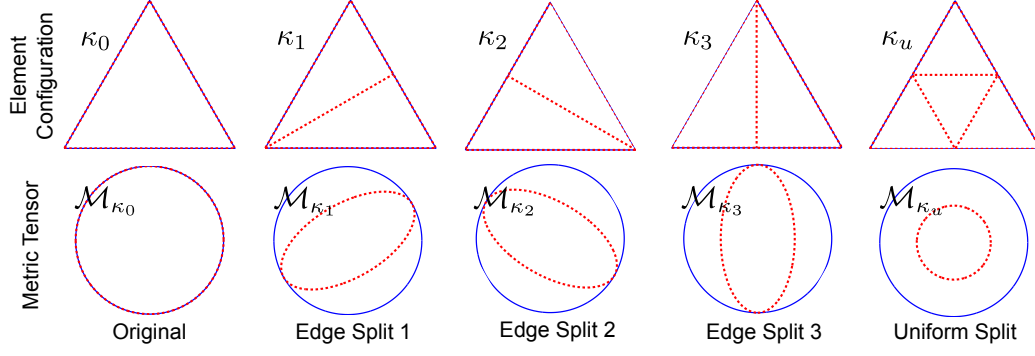


Figure 3-1: The original, edge split, and uniformly split configurations used to sample the local error behavior in two dimensions. The metrics implied by the sampled configurations are shown in dashed lines.

the associated solution  $u_{h,p}^{\kappa_i}$ , i.e.

$$u_{h,p}^{\kappa_i} = \arg \inf_{v_{h,p}^{\kappa_i} \in V_{h,p}(\kappa_i)} \|u - v_{h,p}^{\kappa_i}\|_{L^2(\kappa)}^2,$$

where  $V_{h,p}(\kappa_i)$  is the piecewise polynomial space associated with  $\kappa_i$ . Once we obtain the solution, we can compute the error associated with the configuration,  $\eta_{\kappa_i}$ , i.e.

$$\eta_{\kappa_i} = \|u - u_{h,p}^{\kappa_i}\|_{L^2(\kappa)}^2.$$

We expect the error  $\eta_{\kappa_i}$  to be lower than that of the original configuration,  $\eta_{\kappa_0}$ , because  $V_{h,p}(\kappa_i) \supset V_{h,p}(\kappa_0)$ ,  $i = 1, \dots, n_{\text{config}}$ . Different configurations yield different reduction in the error, depending on how the approximability of the space is modified by the edge split operation with respect to the function  $u$ . In particular, we encode the approximability of configuration  $\kappa_i$  into the associated metric  $\mathcal{M}_{\kappa_i}$ , the affine invariant mean of the elemental metric tensors of the split configuration, i.e.

$$\mathcal{M}_{\kappa_i} = \text{mean}^{\text{affinv}}(\{\mathcal{M}_{\kappa_i}^j\}_{j=1}^{n_{\text{elem}}^{\text{split}}}),$$

where  $n_{\text{elem}}^{\text{split}} = 2$  for edge split, and  $n_{\text{elem}}^{\text{split}} = 4$  for uniform split in two dimensions. Repeating the procedure for all  $n_{\text{config}}$  configurations, we construct metric-error pairs

$$\{\mathcal{M}_{\kappa_i}, \eta_{\kappa_i}\}_{i=1}^{n_{\text{config}}}.$$



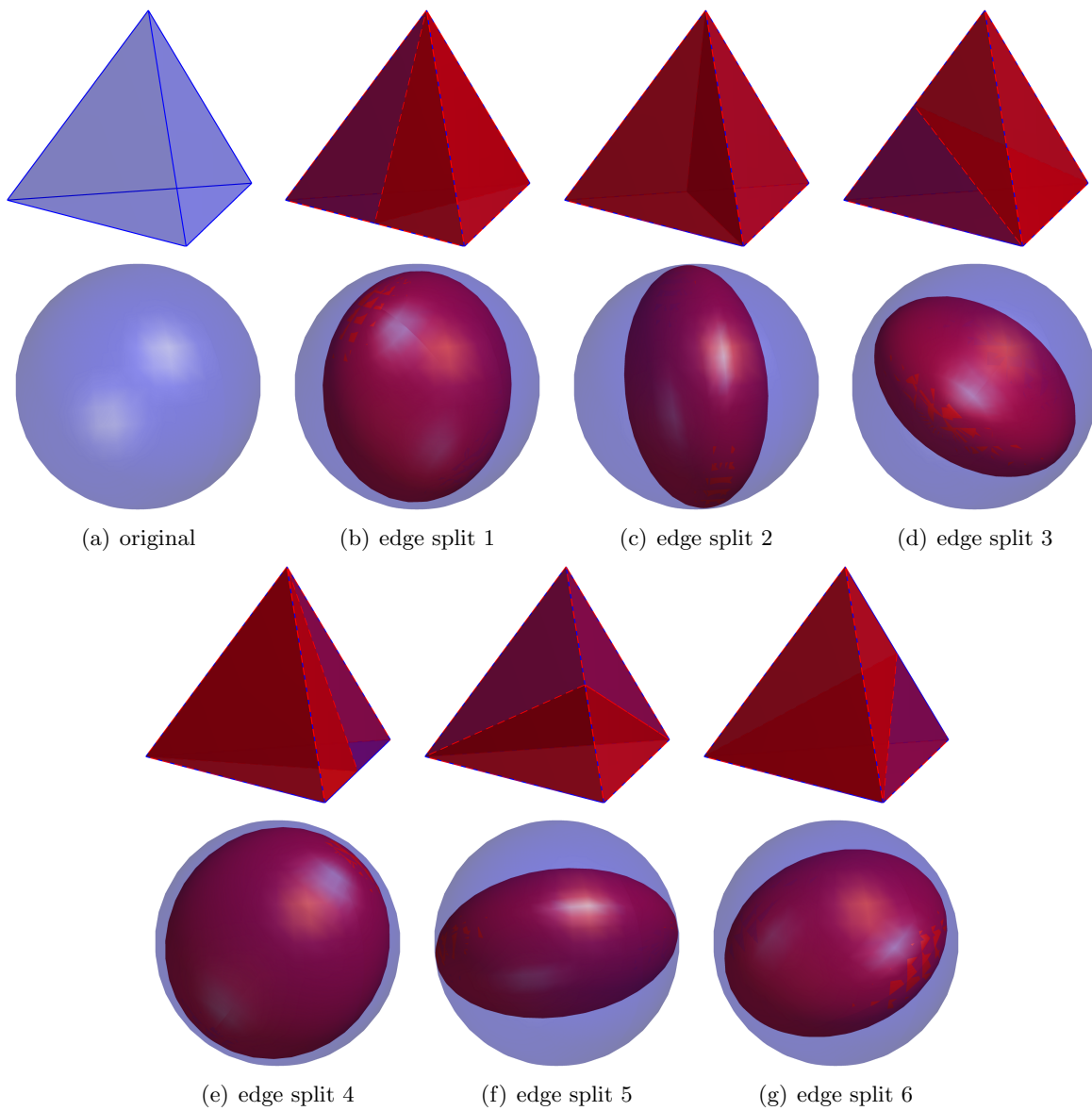


Figure 3-2: The original and edge split configurations used to sample the local error behavior in three dimensions.



Along with the isotropic behavior, these pairs capture the anisotropic behavior of the local error because anisotropic edge split configurations are included in the  $n_{\text{config}}$  configurations.

The construction of metric-error pairs for the DG discretization with the DWR error estimate follows a similar procedure. First, we solve an element-wise local problem associated with  $\kappa_i$ . The local solution,  $u_{h,p}^{\kappa_i} \in V_{h,p}(\kappa_i)$ , is a function defined on the subdivided mesh,  $\kappa_i$ , that satisfies

$$\mathcal{R}_{h,p}^{\kappa_i}(u_{h,p}^{\kappa_i}, v_{h,p}^{\kappa_i}) = 0, \quad \forall v_{h,p}^{\kappa_i} \in V_{h,p}(\kappa_i),$$

where the local semilinear form,  $\mathcal{R}_{h,p}^{\kappa_i}(\cdot, \cdot)$ , sets the boundary fluxes on  $\kappa_i$  assuming the solution on the neighbor elements does not change. Here, we take advantage of the element-wise discontinuous nature of the DG solution. Then, we recompute the localized DWR error estimate corresponding to the subdivided mesh as

$$\eta_{\kappa_i} \equiv |\mathcal{R}_{h,p}(u_{h,p}^{\kappa_i}, \psi_{h,\hat{p}}|_{\kappa_0})|,$$

where  $\hat{p} = p + 1$  as used for the global error estimate in Section 2.2. Due to the local Galerkin orthogonality of the DG scheme, we can rewrite the local error as

$$\eta_{\kappa_i} = |\mathcal{R}_{h,p}(u_{h,p}^{\kappa_i}, (\psi_{h,\hat{p}} - \psi_{h,p}^{\kappa_i})|_{\kappa_0})|.$$

The equality signifies that the local sampling procedure automatically accounts for the improvement in the adjoint approximability resulting from the local refinement even though the local adjoint problem is not explicitly solved.<sup>1</sup> Thus, the local sampling technique based on the *a posteriori* error estimate automatically captures the behaviors of both primal and dual solutions. Finally, we compute the local metric associated with  $\kappa_i$ ,  $\mathcal{M}_{\kappa_i}$ , to construct metric-error pairs  $\{\mathcal{M}_{\kappa_i}, \eta_{\kappa_i}\}_{i=1}^{n_{\text{config}}}$ .

### 3.2.3 Local Error Model Synthesis

The goal of the model synthesis step is to construct a continuous metric-error function  $\eta_{\kappa}(\cdot) : \text{Sym}_d^+ \rightarrow \mathbb{R}^+$  from the pairs  $\{\mathcal{M}_{\kappa_i}, \eta_{\kappa_i}\}_{i=1}^{n_{\text{config}}}$  collected in the sampling stage. Our

---

<sup>1</sup>We have also experimented with solving the  $\hat{p} = p + 1$  local dual problems as done in [65] for quadrilateral elements, but numerically observed no quantifiable difference in the quality of the error estimate and hence the adaptation efficiency.



error model builds on Pennec’s affine invariant framework for tensor manipulation [117] briefly reviewed in Section 3.2.1.

First, we recall that the logarithmic map of a metric about the original configuration  $\mathcal{M}_{\kappa_0}$  provides a convenient means of characterizing the change in the anisotropic approximability of the region, as discussed in Section 3.2.1. Thus, we will measure the changes in the configuration as

$$S_{\kappa_i} = \log \left( \mathcal{M}_{\kappa_0}^{-1/2} \mathcal{M}_{\kappa_i} \mathcal{M}_{\kappa_0}^{-1/2} \right), \quad i = 0, \dots, n_{\text{config}}.$$

Note that, by construction, the original configuration,  $\mathcal{M}_{\kappa_0}$ , maps to the origin, i.e.  $S_{\kappa_0} = 0$ . Similarly, we measure the associated changes in the errors as

$$f_{\kappa_i} = \log (\eta_{\kappa_i} / \eta_{\kappa_0}), \quad i = 0, \dots, n_{\text{config}}.$$

Again, the original error,  $\eta_{\kappa_0}$ , maps to zero by construction.

In practice, we enforce that the logarithm of the relative error,  $f_{\kappa_i}$ , to be strictly negative, i.e.

$$f_{\kappa_i} = -|\log(\eta_{\kappa_i} / \eta_{\kappa_0})|, \quad i = 0, \dots, n_{\text{config}}.$$

This modification is not necessary if the error estimate monotonically decreased with the increase in the local resolution by the edge splits, which is the case for the  $L^2$  error or output error on sufficiently refined meshes. Unfortunately, on a very coarse mesh, the DWR error estimate could increase with the local refinement; this is because the error is underestimated on the original  $\kappa_0$  configuration. Thus, the increase in the error suggests severe underresolution and inaccurate error estimate. In order to ensure that the error minimization algorithm refines these elements, we make the modification, which results in the error model that measures the “impact,” rather than the “decrease,” of the metric configuration on the error.

Once we have the pairs  $\{S_{\kappa_i}, f_{\kappa_i}\}_{i=1}^{n_{\text{config}}}$  that characterizes the change in the error as a function of the change in the configuration, our objective is to construct a continuous



function  $f_\kappa(\cdot) : \text{Sym}_d \rightarrow \mathbb{R}$ . We choose to construct a linear function in the entries of  $S_\kappa$ ,

$$f_\kappa(S_\kappa) = \text{tr}(R_\kappa S_\kappa), \quad (3.7)$$

where  $R_\kappa$  is a  $d \times d$  matrix that governs the behavior of the linear function. Since  $S_\kappa \in \text{Sym}_d$ , we take  $R_\kappa \in \text{Sym}_d$  without loss of generality. To find an appropriate  $R_\kappa$ , we perform the least-squares regression of the known data, i.e.

$$R_\kappa = \arg \min_{Q \in \text{Sym}_d} \sum_{i=1}^{n_{\text{config}}} (f_{\kappa_i} - \text{tr}(Q S_{\kappa_i}))^2.$$

Note that, if  $n_{\text{config}}$  is equal to the degrees of freedom of the symmetric matrix  $R_\kappa$  (e.g. three in two dimensions and six in three dimensions), the regression becomes an interpolation, and the resulting linear function matches exactly at the data points. In two dimensions, we use four configurations (i.e. three anisotropic edge splits and one uniform refinement), so the linear function is not an interpolant.

Rearranging Eq. (3.7), the local error model is given as

$$\eta_\kappa(S_\kappa) = \eta_{\kappa_0} \exp(\text{tr}(R_\kappa S_\kappa)).$$

We can gain a better insight into the error function behavior by decomposing the rate tensor  $R_\kappa$  into the isotropic and the tracefree parts, i.e.

$$R_\kappa = r_\kappa I + \tilde{R}_\kappa,$$

where  $r_\kappa = \text{tr}(R_\kappa)/d$  such that  $\text{tr}(\tilde{R}_\kappa) = 0$ . Combined with the decomposition of the step tensor  $S_\kappa$  into  $S_\kappa = s_\kappa I + \tilde{S}_\kappa$ , the local error model simplifies to

$$\begin{aligned} \eta_\kappa(s_\kappa I + \tilde{S}_\kappa) &= \eta_{\kappa_0} \exp\left(\text{tr}\left((r_\kappa I + \tilde{R}_\kappa)(s_\kappa I + \tilde{S}_\kappa)\right)\right) \\ &= \eta_{\kappa_0} \exp\left(r_\kappa s_\kappa d + \text{tr}\left(\tilde{R}_\kappa \tilde{S}_\kappa\right)\right) \\ &= \eta_{\kappa_0} \exp(r_\kappa s_\kappa d) \exp\left(\text{tr}\left(\tilde{R}_\kappa \tilde{S}_\kappa\right)\right), \end{aligned}$$

where the cross terms vanish because  $\text{tr}(\tilde{R}_\kappa I) = 0$  and  $\text{tr}(\tilde{S}_\kappa I) = 0$ . The decomposition shows that  $r_\kappa$  (i.e. the trace of  $R_\kappa$ ) controls the change in the error under isotropic scaling,



and  $\tilde{R}_\kappa$  (i.e. the tracefree part of  $R_\kappa$ ) controls the change in the error under shape modification. Thus, the rate matrix  $R_\kappa$  can be thought of as a generalization of the convergence rate for isotropic scaling to anisotropic manipulation. A precise relationship between our anisotropic error model and the standard isotropic error model,

$$\eta_\kappa^{\text{iso}}(h) = \eta_{\kappa_0} \left( \frac{h}{h_0} \right)^{r_\kappa^{\text{iso}}},$$

where  $r_\kappa^{\text{iso}}$  is the isotropic convergence rate, is derived in Appendix F.2.

One of the important properties of the proposed error reconstruction scheme is that the quality of the reconstruction is not affected by the current configuration,  $\mathcal{M}_{\kappa_0}$ . In other words, the quality of the model — and subsequent adaptation decisions — is preserved even on high aspect ratio elements encountered in anisotropic adaptation. This property is proved in Appendix F.3. The importance of this property is highlighted in a comparison of the proposed error model and another model based on the log-Euclidean tensor interpolation [11] — an error model that also includes the isotropic error model but whose reconstruction quality is dependent on the current configuration — in Appendix G.

### 3.2.4 Local Cost Model

The element-wise cost function model,  $\rho_\kappa$ , is obtained by using the metric-manipulation relation Eq. (3.5) and directly integrating the continuous local cost function over an element, i.e.

$$\begin{aligned} \rho_\kappa(S_\kappa) &= \int_\kappa c(\mathcal{M}(x), x) dx = \int_\kappa c_p \sqrt{\det \mathcal{M}(x)} dx = \int_\kappa c_p \sqrt{\det(\mathcal{M}_{\kappa_0}^{1/2} \exp(S_\kappa) \mathcal{M}_{\kappa_0}^{1/2})} dx \\ &= \int_\kappa c_p \sqrt{\det(\mathcal{M}_{\kappa_0}) \det(\exp(S_\kappa))} dx = \int_\kappa c_p \sqrt{\det(\mathcal{M}_{\kappa_0})} \sqrt{\exp(\text{tr}(S_\kappa))} dx \\ &= \rho_{\kappa_0} \exp\left(\frac{1}{2} \text{tr}(S_\kappa)\right) = \rho_{\kappa_0} \exp\left(\frac{d}{2} s_\kappa\right). \end{aligned}$$

Note that the cost is only a function of  $s_\kappa$ , which controls the scaling of the tensor, and not  $\tilde{S}_\kappa$ , which controls the shape.



### 3.2.5 Optimization of the Surrogate Model

The final step of the adaptation algorithm is to optimize the Riemannian metric field  $\{\mathcal{M}\}_{x \in \Omega}$ , described by vertex values  $\{\mathcal{M}_\nu\}_{\nu \in \mathcal{V}}$ . The vertex-based metric can then be used to generate a metric-conforming mesh using an anisotropic mesh generator. To manipulate the metric tensors at vertices, we describe the changes in the tangent space about the original configuration,  $\mathcal{M}_{0,\nu}$ , and use the exponential map, i.e.

$$\mathcal{M}_\nu(S_\nu) = \mathcal{M}_{0,\nu}^{1/2} \exp(S_\nu) \mathcal{M}_{0,\nu}^{1/2}. \quad (3.8)$$

Here,  $S_\nu \in \text{Sym}_d$  describes the change in the metric at vertex  $\nu$ . Thus, given  $\{\mathcal{M}_{0,\nu}\}_{\nu \in \mathcal{V}}$ , our objective is to choose the step matrices  $\{S_\nu\}_{\nu \in \mathcal{V}}$  to reduce the error.

To solve the optimization problem, we first need to write the objective function  $\mathcal{E}$  and the cost constraint  $\mathcal{C}$  in terms of the optimization variables  $\{S_\nu\}_{\nu \in \mathcal{V}}$ . Substitution of the local error model into the error functional yields

$$\mathcal{E}(\mathcal{M}) = \int_{\Omega} e(x, \mathcal{M}(x)) dx \approx \sum_{\kappa \in \mathcal{T}_h} \eta_\kappa(S_\kappa). \quad (3.9)$$

In other words, we have approximated the behavior of the error functional in terms of the changes in the configuration in each region covered by  $\kappa$ ,  $S_\kappa$ . We assign the change in the configuration over an element  $S_\kappa$  as the simple arithmetic mean of the changes at its vertices. That is, denoting the vertices of  $\kappa$  by  $\mathcal{V}(\kappa)$ , we have

$$S_\kappa = \overline{\{S_\nu\}_{\nu \in \mathcal{V}(\kappa)}} \equiv \frac{1}{|\mathcal{V}(\kappa)|} \sum_{\nu \in \mathcal{V}(\kappa)} S_\nu.$$

Substitution of the expression into the error model Eq. (3.9) yields our objective function,

$$\mathcal{E}(\{S_\nu\}_{\nu \in \mathcal{V}}) = \sum_{\kappa \in \mathcal{T}_h} \eta_\kappa \left( \overline{\{S_\nu\}_{\nu \in \mathcal{V}(\kappa)}} \right).$$

Similarly, we can write our cost constraint in terms of our  $\{S_\nu\}_{\nu \in \mathcal{V}}$  as

$$\mathcal{C}(\{S_\nu\}_{\nu \in \mathcal{V}}) = \sum_{\kappa \in \mathcal{T}_h} \rho_\kappa \left( \overline{\{S_\nu\}_{\nu \in \mathcal{V}(\kappa)}} \right).$$

We note that the error model is a good approximation of the error behavior only in



vicinity of the original configuration, because the model is built from the local samples of the configuration. Thus, we need to limit the change in the metric field in each step. This is accomplished by limiting the entries of  $S_\nu$ ,  $\nu \in \mathcal{V}$ , i.e.

$$|(S_\nu)_{ij}| \leq \alpha, \quad i, j = 1, \dots, d, \quad \forall \nu \in \mathcal{V},$$

where the constant  $\alpha$  specifies the region over which the metric-error map is considered reliable. For this work, we use  $\alpha = 2 \log(2)$ , which limits the change in the approximability to 2 in any direction — the range over which the sampling is performed and the error model is assumed reliable.

By introducing the surrogate error and cost functions, we have turned our infinite dimensional optimization problem of the metric tensor field (with an unknown error function) into a finite dimensional optimization of vertex step matrices. The surrogate optimization problem for the optimal  $\{S_\nu\}_{\nu \in \mathcal{V}}$  is

$$\{S_\nu^*\}_{\nu \in \mathcal{V}} = \arg \inf_{\{S_\nu\}_{\nu \in \mathcal{V}}} \mathcal{E}(\{S_\nu\}_{\nu \in \mathcal{V}}) \quad (3.10)$$

$$\text{s.t. } \mathcal{C}(\{S_\nu\}_{\nu \in \mathcal{V}}) = N \quad (3.11)$$

$$|(S_\nu)_{ij}| \leq \alpha, \quad i, j = 1, \dots, d, \quad \forall \nu \in \mathcal{V}. \quad (3.12)$$

We emphasize that we do not intend to solve the problem exactly, because our error model, based on local sampling and surrogate model, is only an approximation to the true problem. Thus, investing a large computational effort into solving the surrogate optimization problem would be counterproductive.

We will now present gradient expressions for the error and cost models, which we will use to develop the first order optimality conditions and our optimization algorithm. The gradient of the error and cost functions with respect to the vertex step matrix is

$$\begin{aligned} \frac{\partial \mathcal{E}}{\partial S_\nu} &= \sum_{\kappa \in \omega(\nu)} \left[ \eta_\kappa \left( \overline{\{S_\nu\}_{\nu \in \mathcal{V}(\kappa)}} \right) \frac{1}{|\mathcal{V}(\kappa)|} R_\kappa \right] \\ \frac{\partial \mathcal{C}}{\partial S_\nu} &= \sum_{\kappa \in \omega(\nu)} \left[ \rho_\kappa \left( \overline{\{S_\nu\}_{\nu \in \mathcal{V}(\kappa)}} \right) \frac{1}{2|\mathcal{V}(\kappa)|} I \right], \end{aligned}$$

where  $\omega(\nu)$  is the set of elements that have  $\nu$  as one of their vertices. Because the cost



function is only dependent on the trace of  $S_\nu$ , it is more convenient to measure the sensitivity with respect to the trace and the trace-free part separately. In particular, let us decompose  $S_\nu$  as

$$S_\nu = s_\nu I + \tilde{S}_\nu,$$

where  $s_\nu = \text{tr}(S_\nu)/d$  and  $\tilde{S}_\nu$  is the trace-free part of  $S_\nu$ . The sensitivities of the error with respect to  $s_\nu$  and  $\tilde{S}_\nu$  are

$$\begin{aligned}\frac{\partial \mathcal{E}}{\partial s_\nu} &= \sum_{\kappa \in \omega(\nu)} \left[ \eta_\kappa \left( \overline{\{S_\nu\}_{\nu \in \mathcal{V}(\kappa)}} \right) \frac{1}{|\mathcal{V}(\kappa)|} \text{tr}(R_\kappa) \right] \\ \frac{\partial \mathcal{E}}{\partial \tilde{S}_\nu} &= \sum_{\kappa \in \omega(\nu)} \left[ \eta_\kappa \left( \overline{\{S_\nu\}_{\nu \in \mathcal{V}(\kappa)}} \right) \frac{1}{|\mathcal{V}(\kappa)|} \tilde{R}_\kappa \right].\end{aligned}$$

Similarly, the sensitivities of the cost are

$$\begin{aligned}\frac{\partial \mathcal{C}}{\partial s_\nu} &= \sum_{\kappa \in \omega(\nu)} \left[ \rho_\kappa \left( \overline{\{S_\nu\}_{\nu \in \mathcal{V}(\kappa)}} \right) \frac{d}{2|\mathcal{V}(\kappa)|} \right] \\ \frac{\partial \mathcal{C}}{\partial \tilde{S}_\nu} &= 0.\end{aligned}$$

Assuming that the current configuration is sufficiently close to the optimal configuration such that the constraints Eq. (3.12) are inactive, the first order optimality condition for the optimization problem Eq. (3.10)-(3.11) is given by

$$\frac{\partial \mathcal{E}}{\partial s_\nu} + \lambda \frac{\partial \mathcal{C}}{\partial s_\nu} = 0, \tag{3.13}$$

$$\frac{\partial \mathcal{E}}{\partial \tilde{S}_\nu} = 0, \quad \forall \nu \in \mathcal{V}, \tag{3.14}$$

for some Lagrange multiplier  $\lambda \in \mathbb{R}$ . The first condition, Eq. (3.13), is a global condition for the size distribution. In particular, if we define the “local” Lagrange multiplier as

$$\lambda_\nu \equiv \frac{\partial \mathcal{E}}{\partial s_\nu} \bigg/ \frac{\partial \mathcal{C}}{\partial s_\nu},$$

then for optimality we must have  $\lambda_\nu = \lambda, \forall \nu \in \mathcal{V}$ . The global coupling is provided by the Lagrange multiplier,  $\lambda$ . The “local” Lagrange multiplier,  $\lambda_\nu$ , is interpreted as the marginal improvement in the local error for a given investment in the local cost, which is the degrees of



freedom in the context of mesh adaptation. The global condition states that, at optimality, the investment to any element results in the same marginal improvement in the error.

The second condition, Eq. (3.14), is a local condition that states that the error is stationary with respect to the shape change. Note that this second optimality condition is satisfied if

$$\tilde{R}_\kappa = 0, \quad \forall \kappa \in \mathcal{T}_h. \quad (3.15)$$

The shape change, induced by  $\tilde{S}_\nu$ , does not affect the cost. Thus, if  $\tilde{R}_\kappa \neq 0$ , then we can reduce the error by choosing a  $\tilde{S}_\nu$  such that  $\text{tr} \left( \tilde{R}_\kappa \overline{\{S_\nu\}_{\nu \in \mathcal{V}(\kappa)}} \right) < 0$  without affecting the cost. Thus, the stationarity with respect to the shape change is required at optimality.

If the current configuration is far from the optimal configuration, then some of the constraints Eq. (3.12) become active and the equalities in the two optimality conditions Eq. (3.13) and (3.14) are replaced by inequalities on those constrained variables.

Let us now propose a gradient-based algorithm to solve the surrogate optimization problem Eq. (3.10)-(3.12). We again emphasize that our objective is to only approximately solve the problem. Our algorithm for solving the optimization problem is:

0. Evaluate/reconstruct  $\rho_{\kappa_0}$ ,  $\eta_{\kappa_0}$ , and  $R_\kappa$  that define the local cost and error models
1. Set  $\delta s = \alpha/n_{\text{step}}$ , which controls the incremental change in the metric such that the maximum change over  $n_{\text{step}}$  steps is limited to  $\alpha$ . This enforces Eq. (3.12) and prevents large changes that would render our error model inaccurate.  $S_\nu^0 = 0, \forall \nu \in \mathcal{V}$ . Set the iteration index to  $n = 0$ .
2. Compute vertex derivatives,  $\partial \mathcal{E}/\partial s_\nu$ ,  $\partial \mathcal{E}/\partial \tilde{S}_\nu$ , and  $\partial \mathcal{C}/\partial s_\nu$  and the local Lagrange multiplier  $\lambda_\nu \equiv (\partial \mathcal{E}/\partial s_\nu)/(\partial \mathcal{C}/\partial s_\nu)$  about  $\{S_\nu^n\}_{\nu \in \mathcal{V}}$ .
3. Work toward equidistributing the local Lagrange multiplier and satisfying the global optimality condition, Eq. (3.13), by updating the isotropic part of  $S_\nu$  according to:
  - Refine top 30%<sup>2</sup> of the vertices  $\nu$  with the largest  $\lambda_\nu$  by setting  $S_\nu^{n+1/3} = S_\nu^n + \delta s I$
  - Coarsen top 30% of the vertices  $\nu$  with the smallest  $\lambda_\nu$  by setting  $S_\nu^{n+1/3} = S_\nu^n - \delta s I$

---

<sup>2</sup>Because the refinement and coarsening fractions are used inside the  $n_{\text{step}}$  steps of the optimization loop, the algorithm is not very sensitive to the particular choice of the fractions. This is unlike “fixed-fraction” adaptation strategies for which the choice is important for efficient mesh generation.



This fixed-fraction type refinement results in a more robust mesh adaptation than a simple steepest descent, which can behave poorly when the error is dominated by few elements.

4. Work toward satisfying the local shape optimality condition, Eq. (3.14), by updating the anisotropic part of  $S_\nu$  according to  $S_\nu^{n+2/3} = S_\nu^{n+1/3} - \delta s(\partial \mathcal{E} / \partial \tilde{S}_\nu) / (\partial \mathcal{E} / \partial s_\nu)$ .
5. Rescale  $S_\nu^{n+2/3}$  to obtain a metric field with desired degrees of freedom. That is,  $S_\nu^n = S_\nu^{n+2/3} + \beta I$ , where  $\beta$  is selected to satisfy Eq. (3.11).
6. Set  $n = n + 1$ . If  $n < n_{\text{step}}$  go back to 2.

After obtaining the desired field of vertex step matrices  $\{S_\nu\}_{\nu \in \mathcal{V}}$ , we modify the vertex metrics using the exponential map, Eq. (3.8), obtaining  $\{\mathcal{M}_\nu\}_{\nu \in \mathcal{V}}$ . Finally, the resulting metric field, described by the vertex values, is fed to a metric-conforming mesh generator to generate a new mesh.

The proposed adaptation algorithm is independent of the particular coordinate representation of the tensors. This property implies that the same physical problem represented in two different coordinate systems produces an identical sequences of tensor fields with respect to the physical problem. The property is proved in Appendix F.4.

### 3.3 Properties of MOESS

Let us now summarize properties of MOESS that are particularly important for practical output-based mesh adaptation.

- The method handles any discretization order
- The method uses the simplex remeshing strategy, which allow for arbitrarily-oriented anisotropic elements.
- The method does not make any *a priori* assumption about the convergence behavior of the error. Because no *a priori* assumptions are utilized on the convergence rate, the method is more robust when features are under-resolved in the presence of a singularity or singular perturbation.



- Of the three steps of the adaptation algorithm (local error sampling, error model synthesis, and surrogate model optimization), the local error sampling constitutes majority (over 90%) of the computational cost. The local solves are perfectly scalable and are particularly suited for multi-core processors.
- Both the sizing and the anisotropy decisions are driven directly by the *a posteriori* error estimates, which automatically captures the behaviors of both the primal and dual solutions as well as all components of the states. Improved efficiency is expected for problems in which primal and adjoint solutions exhibit different directional features.
- The method inherits the versatility of the adjoint-based error estimate, which exclusively governs adaptation decisions. For example, the framework straightforwardly extends to different governing equations (e.g. Navier-Stokes, structural elasticity, Maxwell's).

### 3.4 Practical Considerations and Data Reported

Let us illustrate how MOESS works in practice and clarify the data reported in all the subsequent chapters. We will use the  $L^2$  error control problem for a two-dimensional boundary layer that will be considered in Section 4.4 as an example. The problem is solved using a  $p = 3$ , dof = 1000 discretization.<sup>3</sup>

A sequence of meshes obtained for this problem is shown in Figure 3-3. The initial mesh consists of 32  $p = 3$  elements. As the target degrees of freedom is set to 1000, all subsequent meshes contain approximately 100  $p = 3$  elements. The dof history shown in Figure 3-4(a) confirms the stationarity of the number of degrees of freedom. The error convergence history, shown in Figure 3-4(b), indicates that the adaptation leads to an error reduction of over three orders of magnitude. MOESS achieves this by redistributing element sizes and employing highly anisotropic elements. Figure 3-4(b) also shows that, after 5 adaptation cycles, the mesh is optimized for this problem, and the  $L^2$  error fluctuates around  $8 \times 10^{-6}$ . Note that the number of adaptation cycles required to achieve optimality is dependent on the quality of the initial mesh; a typical case requires fewer adaptation cycles than this case, because the initial mesh typically comes from an optimized mesh at lower degrees of

---

<sup>3</sup>The choice of the number of degrees of freedom at which adaptation is performed is currently left to the user. Future work to solve the minimum-dof error-constrained problem is discussed in Section 9.2.



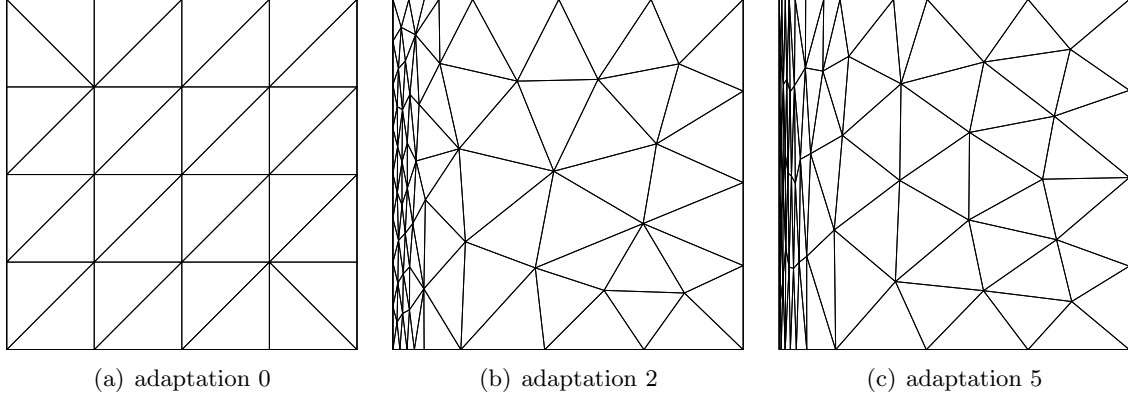


Figure 3-3: Sequence of adapted meshes for the 2d boundary layer  $L^2$  error control problem.

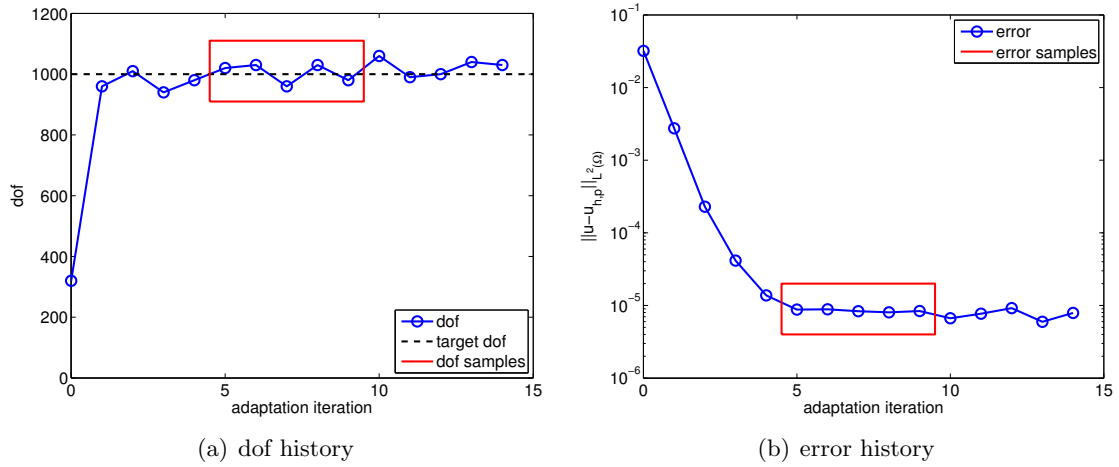


Figure 3-4: Variation in the degrees of freedom and error with the adaptation iterations. The samples used for assessment are marked in red boxes.

freedom or from a case with small changes in parameter.

The error convergence history also shows that the optimization algorithm generates a family of optimal meshes that have similar error levels, in this case generated after the 5th adaptation iterations. All of these meshes have similar metric fields but slightly different triangulations, which arise from the non-uniqueness of the meshes that realize a given metric field. To account for this fluctuation in the error, we average the errors obtained on five meshes belonging to a given family and report that averaged error. (Note, we report the average of the errors, not the error of the average.) This method is used to compute the error for all cases considered in the subsequent chapters. For output errors, which are not normed quantities, the method also reduces the chance of reporting falsely low error due to cancellation.



## Chapter 4

# $L^2$ Projection and Error Control

### 4.1 Introduction

In this chapter, we study the behavior of the  $L^2$  approximation error for problems with a canonical singularity and a singular perturbation often encountered in solving PDEs. Specifically, the objective of this chapter is twofold. First is to analytically develop the optimal anisotropic element size distributions for these select problems using the anisotropic polynomial approximation result stated in Proposition 2.4 and calculus of variations. Second is to verify the ability of our adaptation algorithm, Mesh Optimization via Error Sampling and Synthesis (MOESS), to produce optimal meshes in the  $L^2$  error control setting. To this end, the  $L^2$  projector is used as the “solver,” and the  $L^2$  error is used as the quantity of interest. This solver-error pair eliminates the issues associated with the stability of discretization and allows us to focus on the ability of the space to approximate a given function. Thus,  $L^2$  error control is well-suited for initial verification of our adaptation algorithm.

The  $L^2$  projection “solver” finds the solution  $u_{h,p} \in V_{h,p}$  that minimizes the square of the  $L^2$  projection error, i.e.

$$u_{h,p} = \arg \inf_{v_{h,p} \in V_{h,p}} \mathcal{E}(v_{h,p}),$$

where

$$\mathcal{E}(v_{h,p}) \equiv \|u - v_{h,p}\|_{L^2(\Omega)}^2 = \int_{\Omega} (u - v_{h,p})^2 dx.$$



Because the approximation space  $V_{h,p}$  is discontinuous across element interfaces, the  $L^2$  projection problem entails element-by-element inversion of the mass matrix. A straightforward localization of the error functional to an element yields the local error

$$\eta_\kappa \equiv \|u - u_{h,p}\|_{L^2(\kappa)}^2 = \int_\kappa (u - u_{h,p})^2 dx,$$

and the local errors satisfy  $\mathcal{E} = \sum_{\kappa \in \mathcal{T}_h} \eta_\kappa$ .

## 4.2 Conditions for the Optimal Approximant

This section develops general conditions for the optimal approximant, the approximant that minimizes the  $L^2$  error for a given number of degrees of freedom. The expression can be developed in terms of the metric tensor field  $\{\mathcal{M}(x)\}_{x \in \Omega}$  or its singular value decomposition pairs,  $\{U(x), \sigma(x)\}_{x \in \Omega}$ ; let us use the decomposition pairs for convenience. (Details of the connection between error representation based on  $\{\mathcal{M}(x)\}_{x \in \Omega}$  and its decomposition  $\{U(x), \sigma(x)\}_{x \in \Omega}$  are presented in Appendix D.1.) Based on Proposition 2.4, the  $L^2$  projection error is approximated in terms of  $\{U(x), \sigma(x)\}_{x \in \Omega}$  by

$$\mathcal{E}(\sigma, U; u) = C \int_\Omega E_\Sigma^{\tilde{p}}(U(x), \sigma(x); u(x)) dx, \quad (4.1)$$

where the error kernel is given by

$$E_\Sigma^{\tilde{p}}(U, \sigma; u) \equiv \sum_{i_1=1}^d \cdots \sum_{i_{\tilde{p}}=1}^d \left( \frac{\partial^{\tilde{p}} u}{\partial x_{j_1} \cdots \partial x_{j_{\tilde{p}}}} U_{j_1 i_1} \cdots U_{j_{\tilde{p}} i_{\tilde{p}}} \sigma_{i_1} \cdots \sigma_{i_{\tilde{p}}} \right)^2, \quad (4.2)$$

where  $\tilde{p} = p + 1$  and, as before, the summation on repeated indices  $j_1, \dots, j_{\tilde{p}}$  is implied. The cost functional is

$$\mathcal{C}(\sigma) = c \int_\Omega \prod_{j=1}^d \sigma_j(x)^{-1} dx. \quad (4.3)$$

Note that the cost functional is not a function of  $U$ , which induces a volume-preserving transformation.

Forming the Lagrangian  $\mathcal{L}(U, \sigma) = \mathcal{E}(U, \sigma) + \lambda \mathcal{C}(\sigma)$  and finding the first-order variation with respect to  $U$  and  $\sigma$  yields the first-order optimality conditions in the differential form



as stated in the following theorem.

**Theorem 4.1** (Optimality conditions for an  $L^2$  approximant). *The  $L^2$  approximant that minimizes the  $L^2$  error for a given number of degrees of freedom satisfies*

$$\begin{aligned} \frac{\partial E_{\Sigma}^{p+1}}{\partial \sigma_i} - \hat{\lambda} \sigma_i^{-1} \prod_{j=1}^d \sigma_j^{-1} &= 0, \quad i = 1, \dots, d \\ \frac{\partial E_{\Sigma}^{p+1}}{\partial U} \delta U &= 0, \quad \forall \delta U \text{ permissible}, \end{aligned}$$

where  $\hat{\lambda} = \lambda c / C \in \mathbb{R}$  is the scaled Lagrange multiplier,  $E_{\Sigma}^{p+1}(U, \sigma)$  is the error kernel Eq. (4.2), and the permissible variation  $\delta U$  satisfies

$$U^T \delta U + \delta U^T U = 0.$$

*Proof.* The optimality conditions follow from direct differentiation of the Lagrangian. The permissibility condition on  $\delta U$  arises from differentiating the orthogonality condition,  $U^T U = I$ , where  $I$  is the identity matrix.  $\square$

In general, obtaining a closed-form expression for the field of optimal pairs  $\{U(x), \sigma(x)\}_{x \in \Omega}$  for an arbitrary function  $u$  is impossible. The following sections develop the optimality conditions for special cases for which the optimal element size distribution can be found in a closed form.

### 4.3 $r^\alpha$ -Type Corner Singularity

We consider a function with a  $r^\alpha$ -type corner singularity in two dimensions, where  $r$  is the distance from the singular corner and  $\alpha > 0$  is a constant determining the strength of the singularity. This class of singularity appears at geometric corners of the solution to elliptic equations. The general form of the singularity, located at the origin, is given by

$$u(r, \theta) = r^\alpha \sin[\alpha(\theta + \theta_0)],$$

where  $r^2 = x_1^2 + x_2^2$ ,  $\tan(\theta) = x_2/x_1$ ,  $\alpha \notin \mathbb{Z}$  specifies the singularity strength, and  $\theta_0$  is the offset angle.



### 4.3.1 Analytical Solution

Let us obtain the optimal anisotropic element size distribution using the optimal approximant conditions stated in Theorem 4.1. First, we develop a lemma regarding the higher derivatives of the function.

**Lemma 4.2.** *The  $\tilde{p}$ -th derivative of the function  $u = r^\alpha \sin[\alpha(\theta + \theta_0)]$  is given by*

$$\nabla^{\tilde{p}} u = \left[ \prod_{j=0}^{\tilde{p}-1} (\alpha - j) \right] r^{\alpha-\tilde{p}} \Im \left[ e^{i\beta} (\hat{r} + i\hat{\theta})^{\tilde{p}} \right], \quad (4.4)$$

where  $\beta = \alpha(\theta + \theta_0)$ ,  $\hat{r}$  and  $\hat{\theta}$  are the unit vectors defining the locally orthogonal polar coordinates,  $i$  is the imaginary unit, and  $\Im(\cdot) : \mathbb{C} \rightarrow \mathbb{R}$  returns the imaginary part of the argument.

*Proof.* The proof follows by induction. The function of interest can be compactly written as

$$u = r^\alpha \Im(e^{i\beta}),$$

where  $\beta = \alpha(\theta + \theta_0)$ . Note that  $\partial\beta/\partial\theta = \alpha$ . The first derivative ( $\tilde{p} = 1$ ), expressed in terms of  $\hat{r}$  and  $\hat{\theta}$ , is given by

$$\nabla u = \hat{r} \frac{\partial u}{\partial r} + \hat{\theta} \frac{1}{r} \frac{\partial u}{\partial \theta} = \hat{r} \left[ \alpha r^{\alpha-1} \Im(e^{i\beta}) \right] + \hat{\theta} \frac{1}{r} \left[ r^\alpha \Im(i\alpha e^{i\beta}) \right] = \alpha r^{\alpha-1} \Im \left[ e^{i\beta} (\hat{r} + i\hat{\theta}) \right],$$

which verifies Eq. (4.4) for  $\tilde{p} = 1$ .

Assuming Eq. (4.4) is true for the  $\tilde{p}$ -th derivative, some arithmetic operations yield

$$\begin{aligned} \frac{\partial}{\partial r} \nabla^{\tilde{p}+1} u &= \left[ \prod_{j=0}^{\tilde{p}} (\alpha - j) \right] r^{\alpha-\tilde{p}-1} \Im \left[ e^{i\beta} (\hat{r} + i\hat{\theta})^{\tilde{p}} \right] \\ \frac{\partial}{\partial \theta} \nabla^{\tilde{p}+1} u &= \left[ \prod_{j=0}^{\tilde{p}-1} (\alpha - j) \right] r^{\alpha-\tilde{p}} \Im \left[ i\alpha e^{i\beta} (\hat{r} + i\hat{\theta})^{\tilde{p}} + e^{i\beta} \tilde{p} (\hat{r} + i\hat{\theta})^{\tilde{p}-1} (\hat{\theta} - i\hat{r}) \right] \\ &= \left[ \prod_{j=0}^{\tilde{p}} (\alpha - j) \right] r^{\alpha-\tilde{p}} \Im \left[ e^{i\beta} (\hat{r} + i\hat{\theta})^{\tilde{p}} i \right]. \end{aligned}$$



Using the expressions,  $\nabla^{\tilde{p}+1}u$  can be expressed as

$$\nabla^{\tilde{p}+1}u = \nabla(\nabla^{\tilde{p}}u) = \hat{r} \frac{\partial}{\partial r} \nabla^{\tilde{p}}u + \hat{\theta} \frac{1}{r} \frac{\partial}{\partial \theta} \nabla^{\tilde{p}}u = \left[ \prod_{j=0}^{\tilde{p}} (\alpha - j) \right] r^{\alpha - (\tilde{p}+1)} \Im \left[ e^{i\beta} (\hat{r} + i\hat{\theta})^{\tilde{p}+1} \right].$$

The results verifies Eq. (4.4) for the  $\tilde{p} + 1$  derivative under the induction hypothesis, and this concludes the proof.  $\square$

**Corollary 4.3.** *The components of  $\nabla^{\tilde{p}}u$  in the coordinate  $(\hat{r}, \hat{\theta})$  can be expressed as*

$$\frac{\partial^{\tilde{p}}u}{\partial r_{j_1} \dots \partial r_{j_{\tilde{p}}}} = g(r) \Im (v_{j_1} \dots v_{j_{\tilde{p}}}),$$

where  $r_1 = r$ ,  $r_2 = \theta$ , and

$$g(r) = \left[ \prod_{j=0}^{\tilde{p}-1} (\alpha - j) \right] r^{\alpha - \tilde{p}} \quad \text{and} \quad v_{j_k} = \begin{cases} e^{i\frac{\beta}{\tilde{p}}}, & j_k = 1 \\ ie^{i\frac{\beta}{\tilde{p}}}, & j_k = 2 \end{cases}, \quad k = 1, \dots, \tilde{p}.$$

The following theorem provides the optimal element size distribution for the  $r^\alpha$ -type singularity.

**Theorem 4.4.** *The optimal mesh for degree- $p$  polynomial approximation of the function  $u = r^\alpha \sin[\alpha(\theta + \theta_0)]$  consists of isotropic elements with diameter*

$$h(r) = Cr^k$$

where  $C$  is a constant independent of  $r$ , and  $k$  is the optimal grading constant

$$k = 1 - \frac{\alpha + 1}{p + 2}.$$

In other words, the optimal metric distribution is given by  $\mathcal{M} = \tilde{C}r^{-2k}I$ , where  $\tilde{C}$  is a constant independent of  $r$ .

*Proof.* Recall that any unitary matrix in  $\mathbb{R}^2$  can be expressed as

$$U = \begin{pmatrix} \cos(\phi) & \sin(\phi) \\ -\sin(\phi) & \cos(\phi) \end{pmatrix}$$



for some  $\phi \in \mathbb{R}$ . For convenience, let us denote the rank- $\tilde{p}$  tensor appearing in the error kernel  $E_{\Sigma}^{\tilde{p}}$  defined in Eq. (4.2) by  $F$ , i.e.

$$F_{i_1 \dots i_{\tilde{p}}}^{\tilde{p}} = \frac{\partial^{\tilde{p}} u}{\partial x_{j_1} \dots \partial x_{j_{\tilde{p}}}} U_{j_1 i_1} \dots U_{j_{\tilde{p}} i_{\tilde{p}}} \sigma_{i_1} \dots \sigma_{i_{\tilde{p}}} \quad \text{and} \quad E_{\Sigma}^{\tilde{p}} = \sum_{i_1=1}^d \dots \sum_{i_{\tilde{p}}=1}^d (F_{i_1 \dots i_{\tilde{p}}})^2,$$

with  $d = 2$ . Appealing to Corollary 4.3, the tensor  $F$  can be expressed as

$$\begin{aligned} F_{i_1 \dots i_{\tilde{p}}} &= g(r) \Im(v_{j_1} \dots v_{j_{\tilde{p}}}) U_{j_1 i_1} \dots U_{j_{\tilde{p}} i_{\tilde{p}}} \sigma_{i_1} \dots \sigma_{i_{\tilde{p}}} = g(r) \Im(v_{j_1} U_{j_1 i_1} \sigma_{i_1} \dots v_{j_{\tilde{p}}} U_{j_{\tilde{p}} i_{\tilde{p}}} \sigma_{i_{\tilde{p}}}) \\ &= g(r) \Im(f_{i_1} \dots f_{i_{\tilde{p}}}), \end{aligned}$$

where

$$f_{i_k} = \begin{cases} e^{i(\frac{\beta}{\tilde{p}} - \phi)} \sigma_1, & i_k = 1 \\ i e^{i(\frac{\beta}{\tilde{p}} - \phi)} \sigma_2, & i_k = 2 \end{cases}, \quad k = 1, \dots, \tilde{p}.$$

Thus, the tensor  $F$  simplifies to

$$\begin{aligned} |F_{i_1 \dots i_{\tilde{p}}}| &= |g(r) \Im(i^{i_1 + \dots + i_{\tilde{p}} - \tilde{p}} e^{i(\beta - \tilde{p}\phi)} \sigma_{i_1} \dots \sigma_{i_{\tilde{p}}})| \\ &= \begin{cases} |g(r) \cos(\beta - \tilde{p}\phi)| \sigma_{i_1} \dots \sigma_{i_{\tilde{p}}}, & i_1 + \dots + i_k - \tilde{p} \in \text{Odd} \\ |g(r) \sin(\beta - \tilde{p}\phi)| \sigma_{i_1} \dots \sigma_{i_{\tilde{p}}}, & i_1 + \dots + i_k - \tilde{p} \in \text{Even} \end{cases}, \end{aligned}$$

where Odd and Even denote the set of odd and even numbers, respectively. Thus, the local error kernel,  $E_{\Sigma}^{\tilde{p}}(U, \sigma; v)$ , becomes

$$\begin{aligned} E_{\Sigma}^{\tilde{p}}(U, \sigma; v) &= \sum_{i_1=1}^d \dots \sum_{i_{\tilde{p}}=1}^d |F_{i_1 \dots i_{\tilde{p}}}|^2 = g^2(r) \sin^2(\beta - \tilde{p}\phi) \sum_{\substack{0 \leq k \leq \tilde{p} \\ k \in \text{Even}}} \binom{\tilde{p}}{k} \sigma_1^{2(\tilde{p}-k)} \sigma_2^{2k} \\ &\quad + g^2(r) \cos^2(\beta - \tilde{p}\phi) \sum_{\substack{0 \leq k \leq \tilde{p} \\ k \in \text{Odd}}} \binom{\tilde{p}}{k} \sigma_1^{2(\tilde{p}-k)} \sigma_2^{2k}, \end{aligned}$$

with  $d = 2$ .

Differentiating the error function with respect to  $\sigma_1$  and  $\sigma_2$  and evaluating them about



$\sigma_1 = \sigma_2 = h$  yields

$$\begin{aligned}\left. \frac{\partial E_{\Sigma}^{\tilde{p}}}{\partial \sigma_1} \right|_{\sigma_1 = \sigma_2 = h} &= 2g^2(r)h^{2\tilde{p}-1} \left\{ \sin^2(\beta - \tilde{p}\phi) \sum_{\substack{0 \leq k \leq \tilde{p} \\ k \in \text{Even}}} \binom{\tilde{p}}{k} (\tilde{p} - k) + \cos^2(\beta - \tilde{p}\phi) \sum_{\substack{0 \leq k \leq \tilde{p} \\ k \in \text{Odd}}} \binom{\tilde{p}}{k} (\tilde{p} - k) \right\} \\ \left. \frac{\partial E_{\Sigma}^{\tilde{p}}}{\partial \sigma_2} \right|_{\sigma_1 = \sigma_2 = h} &= 2g^2(r)h^{2\tilde{p}-1} \left\{ \sin^2(\beta - \tilde{p}\phi) \sum_{\substack{0 \leq k \leq \tilde{p} \\ k \in \text{Even}}} \binom{\tilde{p}}{k} (k) + \cos^2(\beta - \tilde{p}\phi) \sum_{\substack{0 \leq k \leq \tilde{p} \\ k \in \text{Odd}}} \binom{\tilde{p}}{k} (k) \right\}.\end{aligned}$$

Since

$$\sum_{\substack{0 \leq k \leq \tilde{p} \\ k \in \text{Even}}} \binom{\tilde{p}}{k} (\tilde{p} - k) = \sum_{\substack{0 \leq k \leq \tilde{p} \\ k \in \text{Even}}} \binom{\tilde{p}}{k} (k) = \sum_{\substack{0 \leq k \leq \tilde{p} \\ k \in \text{Odd}}} \binom{\tilde{p}}{k} (\tilde{p} - k) = \sum_{\substack{0 \leq k \leq \tilde{p} \\ k \in \text{Odd}}} \binom{\tilde{p}}{k} (k) = \frac{1}{2} \sum_{k=0}^{\tilde{p}} \binom{\tilde{p}}{k} k = 2^{\tilde{p}-2} \tilde{p},$$

the derivatives simplify to

$$\left. \frac{\partial E_{\Sigma}^{\tilde{p}}}{\partial \sigma_1} \right|_{\sigma_1 = \sigma_2 = h} = \left. \frac{\partial E_{\Sigma}^{\tilde{p}}}{\partial \sigma_2} \right|_{\sigma_1 = \sigma_2 = h} = 2^{\tilde{p}-1} \tilde{p} g^2(r) h^{2\tilde{p}-1},$$

for any  $\phi \in \mathbb{R}$  (i.e. for any transformation  $U$ ). Substitution of the expressions to the first of the  $L^2$  optimality conditions in Theorem 4.1 yields

$$\left[ \frac{\partial E_{\Sigma}^{p+1}}{\partial \sigma_i} - \lambda \sigma_i^{-1} \prod_{j=1}^d \sigma_j^{-1} \right]_{\sigma_1 = \sigma_2 = h} = 2^{\tilde{p}-1} \tilde{p} g^2(r) h^{2\tilde{p}-1} - \lambda h^{-3} = 0, \quad i = 1, 2.$$

Simple arithmetic manipulation together with the definition of  $g(r)$  in Corollary 4.3 yields the  $h$  grading

$$h = \left[ \frac{\lambda}{2^p(p+1)} \prod_{j=1}^p (\alpha - j)^{-2} \right]^{\frac{1}{2p+4}} r^{1 - \frac{\alpha+1}{p+2}}.$$

To verify the second of the  $L^2$  optimality conditions in Theorem 4.1 is satisfied, we differentiate the error kernel with respect to  $\phi$  and evaluate it at  $\sigma_1 = \sigma_2 = h$ , i.e.

$$\left. \frac{\partial E_{\Sigma}^{\tilde{p}}}{\partial \phi} \right|_{\sigma_1 = \sigma_2 = h} = 2g^2(r) \tilde{p} \sin(\beta - \tilde{p}\phi) \cos(\beta - \tilde{p}\phi) h^{2\tilde{p}} \left[ - \sum_{0 \leq k \leq \tilde{p}} (-1)^k \binom{\tilde{p}}{k} \right] = 0, \quad \forall \phi \in \mathbb{R},$$

where we have used the fact that the sum of the odd terms and even terms of the binomial expansion cancel. Having shown that both  $L^2$  optimality conditions in Theorem 4.1 are



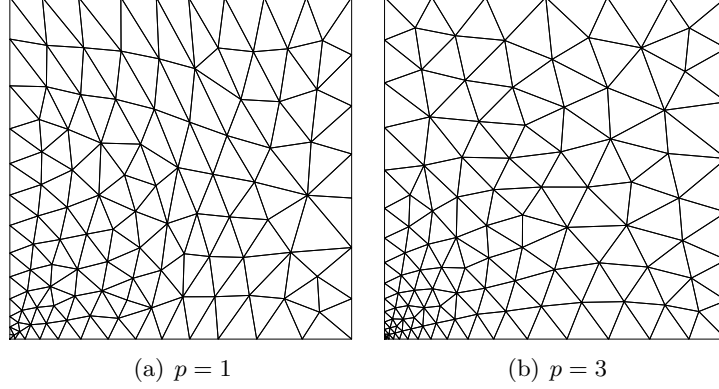


Figure 4-1: The optimized meshes for the corner singularity problem. Each mesh contains approximately 200 elements.

satisfied, this concludes the proof.

□

The theorem states that the grading becomes stronger as  $\alpha$  decreases or  $p$  increases for the corner singularity.

### 4.3.2 Numerical Results

We apply MOESS to a  $r^\alpha$ -type corner singularity problem with  $\alpha = 2/3$ . Examples of optimized meshes obtained for the problem using  $p = 1$  and 3 approximation are shown in Figure 4-1. Each mesh contains approximately 200 elements. MOESS correctly deduces that the optimal mesh for this problem is isotropic. Moreover, the stronger grading toward the singularity located at the bottom left corner for the  $p = 3$  mesh is evident from the figure.

A more quantitative assessment of MOESS-generated meshes is obtained by studying the distribution of the element size,  $h$ , against the distance from the singularity,  $r$ , and comparing the distribution with the analytical result. Figure 4-2 shows the distribution of  $h$  against  $r$  for the optimized meshes. The element size  $h$  is computed based on the volume, i.e.  $h = \det(\mathcal{M}_\kappa)^{-1/4}$  where  $\mathcal{M}_\kappa$  is the elemental implied metric. The distance  $r$  is measured from the singularity to the centroid of the element. The optimization is performed for  $p = 1$  and  $p = 3$  at the degrees of freedom count of 1000 and 4000. The optimal grading coefficient calculated analytically using Theorem 4.4 for  $p = 1$  and 3 are  $k^{\text{analytic}} = 0.44$  and 0.67, respectively. Knowing the optimal values of  $h$  and  $r$  varies linearly in log-log space,



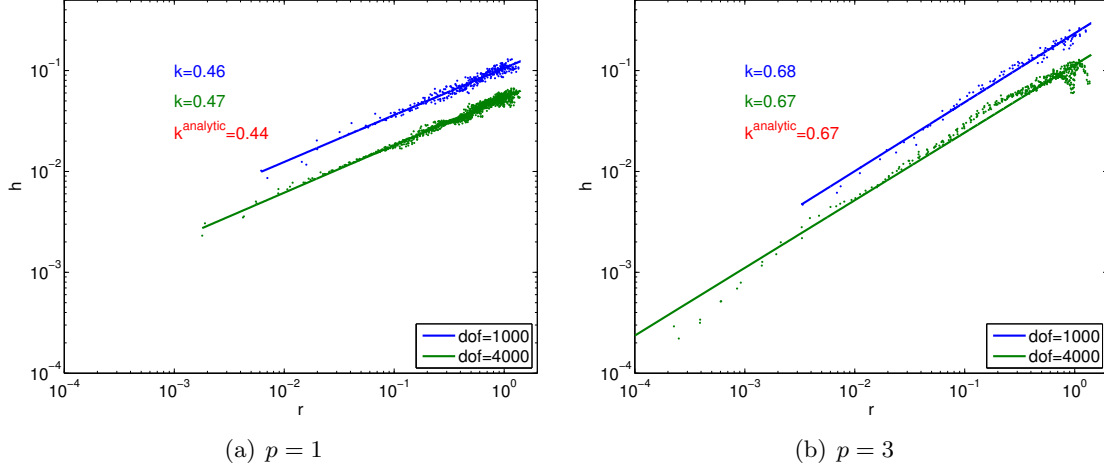


Figure 4-2: The element size  $h$  vs. the distance of the element centroid from the corner  $r$  for the optimized meshes for the corner singularity problem with  $\alpha = 2/3$ . The lines and coefficients shown result from least-squares fit in  $\log(h)$  vs.  $\log(r)$ .

we also plot the least-squares fit to  $\log(h)$  vs.  $\log(r)$ . MOESS produces meshes with the grading factor of  $k = 0.47$  and  $0.67$  for  $p = 1$  and  $p = 3$ , respectively, at 4000 degrees of freedom. Thus, the adaptive algorithm obtains the optimal grading automatically for each  $p$  without any *a priori* knowledge of the solution behavior for the two  $p$ 's.

## 4.4 2d Boundary Layer

We consider a boundary layer solution resulting from a singular perturbation. The solution is essentially one-dimensional, but we regularize the solution by adding a constant  $p + 1$  derivative in the parallel direction, i.e.

$$u(x_1, x_2) = \exp\left(-\frac{x_1}{\epsilon}\right) + \frac{\beta}{(p+1)!} x_2^{p+1}, \quad (4.5)$$

where  $\epsilon$  is the characteristic length of the singular perturbation,  $\beta$  is the regularization constant, and  $x_1$  and  $x_2$  are the coordinates perpendicular and parallel, respectively, to the boundary. As we will see shortly, the particular form of the regularization results in a simple exponential variation in the optimal aspect ratio distribution, which facilitates the verification of MOESS-generated meshes.



#### 4.4.1 Optimality Conditions for Functions with No Mixed Partial

Let us first develop a general optimal anisotropic element size distributions for functions with vanishing mixed partial derivatives, i.e.

$$\frac{\partial^{\tilde{p}} u}{\partial x_{j_1} \cdots \partial x_{j_{\tilde{p}}}} = \begin{cases} u_{x_{j_1}}^{(\tilde{p})}, & j_1 = \cdots = j_{\tilde{p}} \\ 0, & \text{otherwise,} \end{cases}$$

where  $u_{x_i}^{(\tilde{p})}$ ,  $i = 1, \dots, d$  denotes the  $\tilde{p}$  derivative of  $u$  with respect to the  $i$ -th coordinate. Note that the 2d boundary layer described by Eq. (4.5) fits in this form. The key optimality condition that enables an explicit expression of the element size distribution is stated in the following lemma.

**Lemma 4.5.** *For a function with vanishing mixed partial derivatives, the second optimality condition of Theorem 4.1 is satisfied for  $U = I$  for any  $\sigma$ , where  $I$  is the identity matrix.*

*Proof.* The proof follows from a direct evaluation of the second optimality condition of Theorem 4.1. The first variation of the error kernel in the direction of  $\delta U$  is given by

$$\begin{aligned} \frac{\partial E_{\Sigma}^{\tilde{p}}}{\partial U} \delta U &= C \sum_{i_1=1}^d \cdots \sum_{i_{\tilde{p}}=1}^d \left[ \left\{ \frac{\partial^{\tilde{p}} u}{\partial x_{j_1} \cdots \partial x_{j_{\tilde{p}}}} U_{j_1 i_1} \cdots U_{j_{\tilde{p}} i_{\tilde{p}}} \sigma_{i_1} \cdots \sigma_{i_{\tilde{p}}} v \right\} \right. \\ &\quad \cdot \left. \left\{ \sum_{s=1}^{\tilde{p}} \frac{\partial^{\tilde{p}} u}{\partial x_{j_1} \cdots \partial x_{j_{\tilde{p}}}} U_{j_1 i_1} \cdots U_{j_{s-1} i_{s-1}} U_{j_{s+1} i_{s+1}} \cdots U_{j_{\tilde{p}} i_{\tilde{p}}} (\delta U)_{j_s i_s} \sigma_{i_1} \cdots \sigma_{i_{\tilde{p}}} \right\} \right]. \end{aligned}$$

For a function with no mixed partial derivatives, the expression simplifies to

$$\begin{aligned} \frac{\partial E_{\Sigma}^{\tilde{p}}}{\partial U} \delta U &= C \sum_{i_1=1}^d \cdots \sum_{i_{\tilde{p}}=1}^d \left[ \left\{ \sum_{j=1}^d u_{x_j}^{(\tilde{p})} U_{j i_1} \cdots U_{j i_{\tilde{p}}} \sigma_{i_1} \cdots \sigma_{i_{\tilde{p}}} \right\} \right. \\ &\quad \cdot \left. \left\{ \sum_{s=1}^{\tilde{p}} \sum_{j=1}^d u_{x_j}^{(\tilde{p})} U_{j i_1} \cdots U_{j i_{s-1}} U_{j i_{s+1}} \cdots U_{j i_{\tilde{p}}} (\delta U)_{j i_s} \sigma_{i_1} \cdots \sigma_{i_{\tilde{p}}} \right\} \right]. \end{aligned}$$

For  $U = I$ , the term in the first curly bracket vanishes unless  $i_1 = \cdots = i_{\tilde{p}}$ . Thus, the



expression simplifies to

$$\begin{aligned}
\left. \frac{\partial E_{\Sigma}^{\tilde{p}}}{\partial U} \right|_{U=I} \delta U &= C \sum_{i=1}^d \left[ \left\{ \sum_{j=1}^d u_{x_j}^{(\tilde{p})} \delta_{ji}^{\tilde{p}} \sigma_i^{\tilde{p}} \right\} \cdot \left\{ \sum_{s=1}^{\tilde{p}} \sum_{j=1}^d u_{x_j}^{(\tilde{p})} \delta_{ji}^{\tilde{p}-1} (\delta U)_{ji} \sigma_i^{\tilde{p}} \right\} \right] \\
&= C \sum_{i=1}^d \left[ \left\{ u_{x_i}^{(\tilde{p})} \sigma_i^{\tilde{p}} \right\} \cdot \left\{ \tilde{p} u_{x_i}^{(\tilde{p})} (\delta U)_{ii} \sigma_i^{\tilde{p}} \right\} \right] \\
&= C \tilde{p} \sum_{i=1}^d (u_{x_i}^{(\tilde{p})})^2 \sigma_i^{2\tilde{p}} (\delta U)_{ii}
\end{aligned}$$

Furthermore, the permissibility condition on  $\delta U$  about  $U = I$  simplifies to

$$\delta U + \delta U^T = 0.$$

In other words,  $\delta U$  must be skew-symmetric. In particular, the diagonal entries of  $\delta U$  are zero, i.e.  $(\delta U)_{ii} = 0$ ,  $i = 1, \dots, d$ . Thus, we have

$$\left. \frac{\partial E_{\Sigma}^{\tilde{p}}}{\partial U} \right|_{U=I} \delta U = C \int_{\Omega} 2\tilde{p} \sum_{i=1}^d (u_{x_i}^{(\tilde{p})})^2 \sigma_i^{2\tilde{p}} (\delta U)_{ii} dx = 0, \quad \forall \delta U \text{ permissible}, \forall \sigma.$$

Thus, the first-order optimality condition is satisfied by choosing  $U = I$  for any choice of  $\sigma$ .  $\square$

Having shown that the second of the optimality conditions in Theorem 4.1 is satisfied for  $U = I$  for any  $\sigma$ , our task is to obtain  $\sigma$  that satisfy the first of the optimality conditions in Theorem 4.1. The main result is stated in the following theorem.

**Theorem 4.6** (Optimal element size distribution for functions with vanishing mixed partials). *For a function with vanishing mixed partial derivatives, the optimal aspect ratios  $\gamma_i \equiv \sigma_i/\sigma_1$ ,  $i = 2, \dots, d$ , are given by*

$$\gamma_i = \frac{\sigma_i}{\sigma_1} = \left( \frac{u_{x_1}^{(p+1)}}{u_{x_i}^{(p+1)}} \right)^{\frac{1}{p+1}}, \quad i = 2, \dots, d, \quad (4.6)$$

and the optimal  $\sigma_1$  spacing is given by

$$\sigma_1 = \left( \frac{\lambda c}{2Cp+1} \right)^{\frac{1}{2p+1+d}} \left( u_{x_1}^{(p+1)} \right)^{-\frac{2p+d+1}{(p+1)(2p+d+2)}} \prod_{i=2}^d \left( u_{x_i}^{(p+1)} \right)^{\frac{1}{(p+1)(2p+d+2)}}, \quad (4.7)$$



where  $u_{x_i}^{(p+1)}$ ,  $i = 1, \dots, d$ , denote the  $p+1$  derivative with respect to the  $i$ -th coordinate.

*Proof.* Substitution of the vanishing-mixed-partial condition and evaluation of  $\partial E_{\Sigma}^{p+1}/\partial \sigma_i$  about  $U = I$  yields

$$\left. \frac{\partial E_{\Sigma}^{\tilde{p}}}{\partial \sigma_i} \right|_{U=I} = 2\tilde{p}(u_{x_i}^{(\tilde{p})})^2 \sigma_i^{2\tilde{p}-1}, \quad i = 1, \dots, d,$$

where  $\tilde{p} \equiv p+1$ . The first of the optimality conditions thus becomes

$$2\tilde{p}(u_{x_i}^{(\tilde{p})})^2 \sigma_i^{2\tilde{p}-1} - \lambda c \sigma_i^{-1} \prod_{j=1}^d \sigma_j^{-1} = 0, \quad i = 1, \dots, d.$$

Let us denote the aspect ratios by  $\gamma_i \equiv \sigma_i/\sigma_1$ ,  $i = 2, \dots, d$ . Subtracting the  $i$ -th equation,  $i \neq 1$ , from the first equation ( $i = 1$ ) and rearranging the expression yield

$$\gamma_i = \frac{\sigma_i}{\sigma_1} = \left( \frac{u_{x_1}^{(\tilde{p})}}{u_{x_i}^{(\tilde{p})}} \right)^{\frac{1}{\tilde{p}}}, \quad i = 2, \dots, d.$$

Substitution of the optimal aspect ratio conditions to the first equation yields

$$\sigma_1 = \left( \frac{\lambda c}{2C\tilde{p}} \right)^{\frac{1}{2\tilde{p}+d}} \left( u_{x_1}^{(\tilde{p})} \right)^{-\frac{2\tilde{p}+d-1}{\tilde{p}(2\tilde{p}+d)}} \prod_{i=2}^d \left( u_{x_i}^{(\tilde{p})} \right)^{\frac{1}{\tilde{p}(2\tilde{p}+d)}}.$$

Substituting  $\tilde{p} = p+1$  proves the desired result.  $\square$

#### 4.4.2 Analytical Solution to the 2d Boundary Layer Problem

The optimal mesh size distribution for the boundary layer function, Eq. (4.5), is obtained via direct evaluation of Theorem 4.6. The  $p+1$  derivatives of the solution are

$$|u_{x_1}^{(p+1)}| = \epsilon^{-(p+1)} \exp(-x_1/\epsilon) \quad \text{and} \quad |u_{x_2}^{(p+1)}| = \beta.$$

Substituting the derivatives into Eq. (4.7) and manipulating the expression, we obtain the optimal  $h_1 = \sigma_1$  grading

$$h_1 = C \exp(k_1 x_1)$$



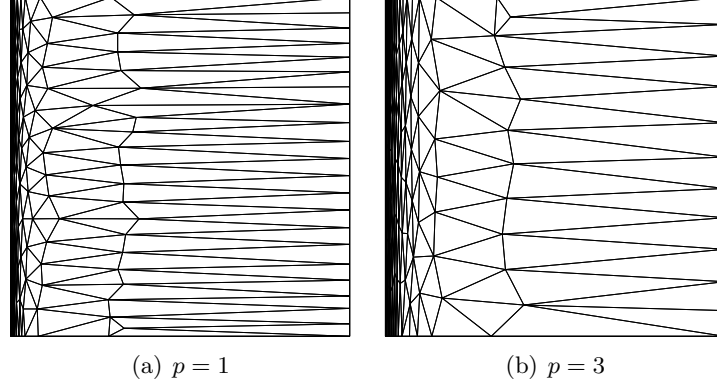


Figure 4-3: Examples of optimized boundary layer meshes for  $p = 1$  and  $p = 3$ . Each mesh contains approximately 200 elements.

with the optimal characteristic thickness

$$\delta = \frac{1}{k_1} = \epsilon \left( p + \frac{3}{2} \right) \left( 1 - \frac{1}{4p^2 + 12p + 9} \right).$$

We note that this optimal characteristic thickness is close to that of the one-dimensional boundary layer problem,  $\delta_{1d} = \epsilon(p + 3/2)$ . Unlike the corner singularity case, the optimal mesh grading decreases as  $p$  increases. Similarly, Eq. (4.6) yields the optimal aspect ratio distribution of

$$\mathcal{R} = \mathcal{R}_0 \exp(k_{\mathcal{R}}^{\text{analytic}} x_1)$$

with the aspect ratio at the root,  $\mathcal{R}_0$ , and the grading factor,  $k_{\mathcal{R}}$ , given by

$$\mathcal{R}_0 = \frac{1}{\beta^{\frac{1}{p+1}} \epsilon} \quad \text{and} \quad k_{\mathcal{R}} = -\frac{1}{\epsilon(p+1)}.$$

Note that the maximum aspect ratio is achieved on the boundary, and the ratio decreases exponentially away from the boundary.

#### 4.4.3 Numerical Results

We apply MOESS to the boundary layer problem with  $\epsilon = 1/100$  and  $\beta = 2^{p+1}$ , which results in  $\mathcal{R}_0 = 50$ . Figure 4-3 shows examples of  $p = 1$  and  $p = 3$  optimized meshes. Each mesh contains approximately 200 elements. Highly anisotropic elements are employed to resolve the boundary layer on the left boundary. Visually, the  $p = 3$  optimized mesh



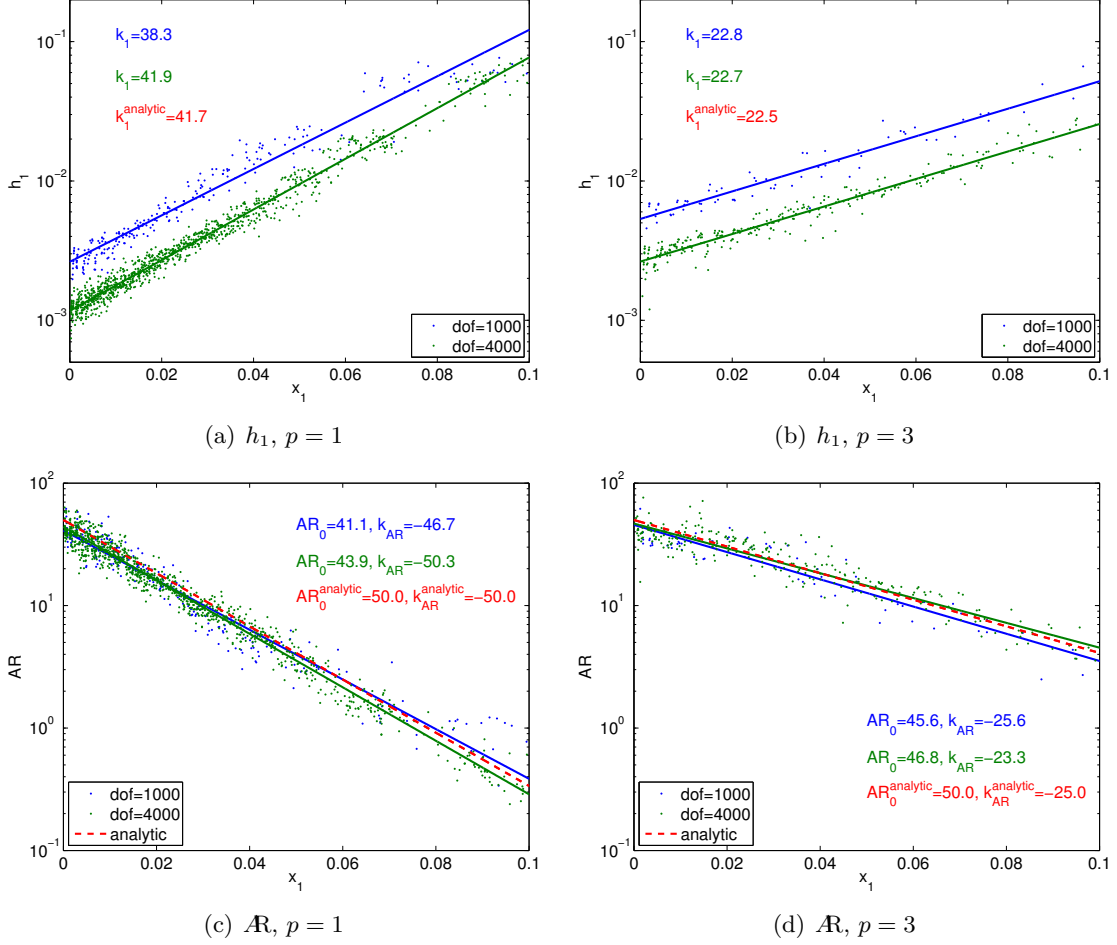


Figure 4-4: The element size in the perpendicular direction,  $h_1$ , and the aspect ratio distribution,  $\mathcal{R} = h_2/h_1$ , for the 2d boundary layer problem with  $\epsilon = 0.01$  and  $\beta = 2^{p+1}$ .

exhibits a weaker  $h_1$  grading toward the boundary layer, as predicted by the analytical result.

A more quantitative assessment of the optimized meshes is provided by Figure 4-4. The  $h_1$  and  $h_2$  value for each element is computed by first calculating the elemental implied metric  $\mathcal{M}_\kappa$ , and then taking  $h_1 = (\mathcal{M}_\kappa)_{11}^{-1/2}$  and  $h_2 = (\mathcal{M}_\kappa)_{22}^{-1/2}$ . Figure 4-4(a) shows the distribution of  $h_1$  against the distance from the boundary  $x_1$  in log-linear scale for the  $p = 1$  discretization with 1000 and 4000 degrees of freedom. The distribution is essentially linear in the  $\log(h_1)$ - $x_1$  space, and the least-squares fit in the space shows that the grading factor in the direction perpendicular to the boundary is  $k_1 = 41.9$  (for  $\text{dof} = 4000$ ), which agrees with the analytical optimal value of  $k_1^{\text{analytic}} = 41.7$ . Figure 4-4(c) shows the aspect ratio distribution, and the least-squares fit in the  $\log(\mathcal{R})$ - $x_1$  space. The aspect ratio at



	$k_1$	$\mathcal{R}_0$	$k_{\mathcal{R}}$
$p = 1$ numerical	41.9	43.9	-50.3
$p = 1$ analytical	41.7	50.0	-50.0
$p = 3$ numerical	22.7	46.8	-23.3
$p = 3$ analytical	22.5	50.0	-25.0

Table 4.1: Summary of the optimized mesh parameters for the 2d boundary layer problem.

the boundary obtained using the algorithm is  $\mathcal{R}_0 = 43.9$ , which is slightly lower than the analytical result of  $\mathcal{R}_0^{\text{analytic}} = 50.0$ ; however, the values are still in good agreement. The negative grading away from the boundary of  $k_{\mathcal{R}} = 50.3$  matches closely with that of analytical result,  $k_{\mathcal{R}}^{\text{analytic}} = 50.0$ . The comparison of the analytical and numerical mesh parameters is summarized in Table 4.1.

Figure 4-4(b) and 4-4(d) show the same  $\log(h_1)$ - $x_1$  and  $\log(\mathcal{R})$ - $x_1$  analysis for the  $p = 3$  discretization. The grading for  $h_1$  and  $\mathcal{R}$  are weaker for  $p = 3$  than for  $p = 1$ , which is consistent with the theory. All parameters of the optimized meshes match well with those of analytical results. Again, without relying on the *a priori* error convergence behavior or the solution Hessian (or a higher derivative equivalent), MOESS deduces the optimal anisotropic mesh distribution.

## 4.5 3d Boundary Layer

Let us now consider a boundary layer in three dimensions. The solution is essentially one-dimensional, but we regularize the solution by adding a constant  $p + 1$  derivative in the two parallel direction, i.e.

$$u(x_1, x_2, x_3) = \exp\left(-\frac{x_1}{\epsilon}\right) + \frac{\beta_2}{(p+1)!}x_2^{p+1} + \frac{\beta_3}{(p+1)!}x_3^{p+1}, \quad (4.8)$$

where  $\epsilon$  is the characteristic length of the singular perturbation,  $\beta_2$  and  $\beta_3$  are the regularization constants in the two parallel directions.



### 4.5.1 Analytical Solution

The analytical solution is found by evaluating the expressions in Theorem 4.6 using the derivatives of the solution Eq. (4.8). The  $p + 1$  derivatives of the solution are

$$|u_{x_1}^{(p+1)}| = \epsilon^{-(p+1)} \exp(-x_1/\epsilon) \quad \text{and} \quad |u_{x_i}^{(p+1)}| = \beta_i, \quad i = 2, 3.$$

Simple algebraic manipulation of Eq. (4.7) yields optimal  $h_1$  grading

$$h_1 = C \exp(k_1 x_1)$$

with the optimal characteristic thickness given by

$$\delta = \frac{1}{k_1} = \epsilon \left( p + \frac{3}{2} \right) \left( 1 - \frac{1}{2p^2 + 7p + 6} \right),$$

where the characteristic thickness is again expressed as a function of the one-dimensional boundary layer characteristic thickness,  $\delta_{1d} = \epsilon(p+3/2)$ . The optimal aspect ratios obtained from Eq. (4.6) are

$$\mathcal{R}_i = \mathcal{R}_{i,0} \exp(k_{i,\mathcal{R}} x_1), \quad i = 2, 3,$$

with the aspect ratios at the root,  $\mathcal{R}_{i,0}$ , and the grading factor,  $k_{i,\mathcal{R}}$ , given by

$$\mathcal{R}_{i,0} = \frac{1}{\beta_i^{\frac{1}{p+1}} \epsilon} \quad \text{and} \quad k_{i,\mathcal{R}} = -\frac{1}{\epsilon(p+1)}, \quad i = 2, 3.$$

### 4.5.2 Numerical Results

We apply MOESS to the boundary layer problem with  $\epsilon = 1/100$ ,  $\beta_2 = 2^{p+1}$ , and  $\beta_3 = 4^{p+1}$ , which result in the optimal root aspect ratios of  $\mathcal{R}_{2,0} = 50$  and  $\mathcal{R}_{3,0} = 25$ .

Figure 4-5 provides assessment of the  $p = 1$ , dof = 18000 and  $p = 2$ , dof = 36000 optimized meshes. As in the two-dimensional case, the  $h_1$ ,  $h_2$ , and  $h_3$  values for each element is computed from the diagonal entries of the elemental implied metric. The distribution shows that the spread in the  $h_1$ -spacing and the aspect ratios for a given  $x_1$  are in general larger than those for the two-dimensional case. A larger spread is due to the difficulty of constructing meshes that tightly conform to the metric requests in three dimensions. How-



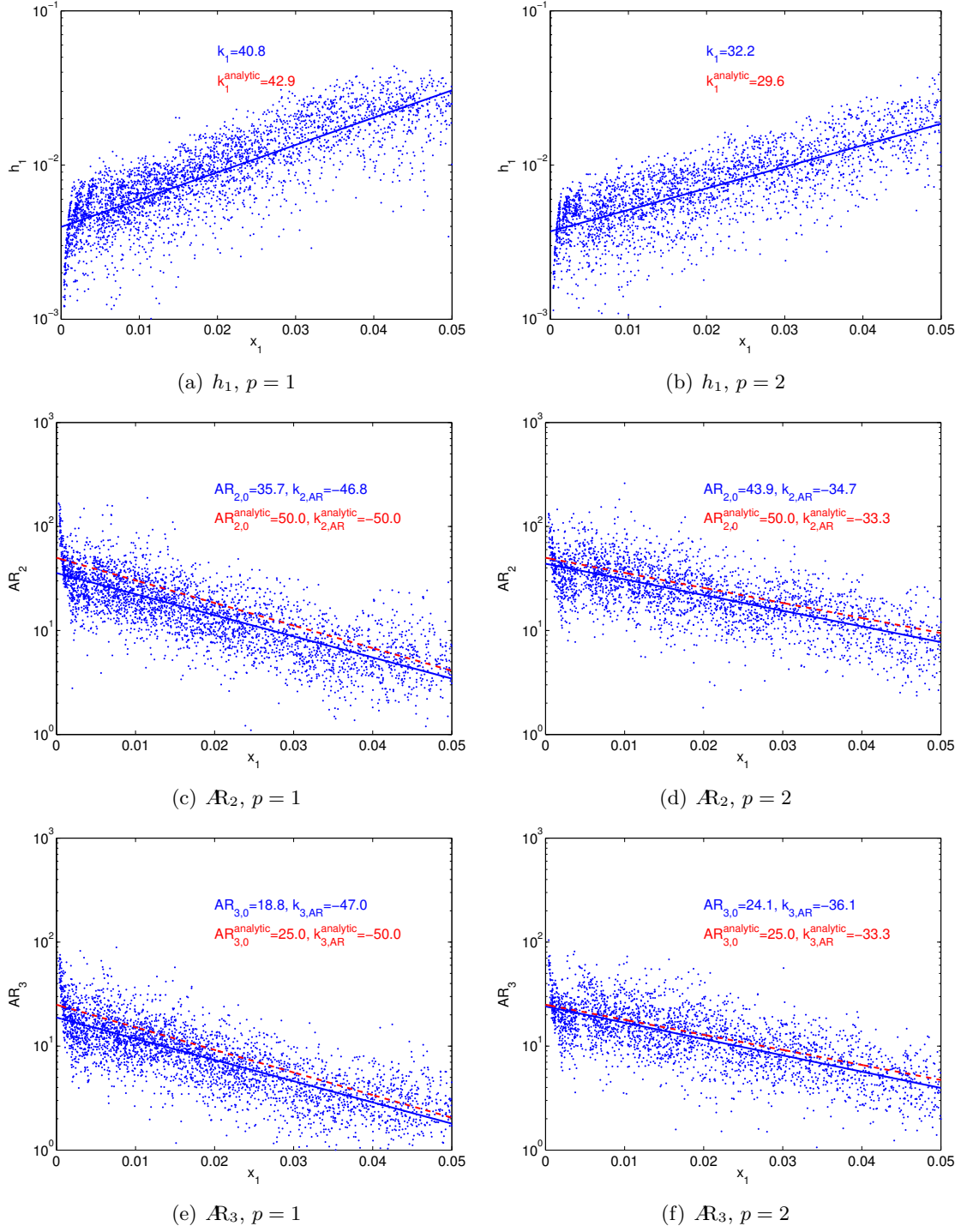


Figure 4-5: The element size in the perpendicular direction,  $h_1$ , and the aspect ratio distribution,  $AR_i = h_i/h_1$ , for the 3d boundary layer problem with  $\epsilon = 0.01$  and  $\beta_2 = 2^{p+1}$  and  $\beta_3 = 4^{p+1}$ .



	$k_1$	$\mathcal{R}_{2,0}$	$k_{2,\mathcal{R}}$	$\mathcal{R}_{3,0}$	$k_{3,\mathcal{R}}$
$p = 1$ numerical	40.8	35.7	-46.8	18.8	-47.0
$p = 1$ analytical	42.9	50.0	-50.0	25.0	-50.0
$p = 2$ numerical	22.7	43.9	-34.7	24.1	-36.1
$p = 2$ analytical	29.6	50.0	-33.3	25.0	-33.3

Table 4.2: Summary of the optimized mesh parameters for the 3d boundary layer problem.

ever, the regression coefficients are in general in good agreement with the analytical values, as summarized in Table 4.2. In particular, for both the  $p = 1$  and  $p = 2$  discretizations, the  $h_1$  gradings of the optimized meshes agree well with the analytical values. MOESS underestimate the optimal anisotropy by approximately 30% for the  $p = 1$  discretization. The matching is better for the  $p = 2$  discretization, with the optimized mesh underestimating  $\mathcal{R}_2$  and  $\mathcal{R}_3$  by only 12% and 4%, respectively.

## 4.6 Conclusion

We derived the optimal anisotropic element size distribution for a few canonical  $L^2$  approximation problems by using a continuous relaxation of the anisotropic approximation theory, Proposition 2.4, and calculus of variations. The MOESS algorithm produced  $p$ -specific optimal meshes consistent with analytical results without any *a priori*  $p$ -dependent function information.



## Chapter 5

# Advection-Diffusion Equation

This chapter considers an application of the proposed adaptation algorithm, Mesh Optimization via Error Sampling and Synthesis (MOESS), to the scalar advection-diffusion equation. The objective is to study the behavior of the adaptation algorithm in the context of output error control for problems with simple and carefully chosen primal and dual solution behaviors. In particular, we will compare the performance of MOESS to two output-based adaptation strategies: isotropic adaptation and anisotropic adaptation based on the higher derivatives of the primal solution.

### 5.1 Governing Equation and Problem Setup

We consider the advection-diffusion equation in a rectangular domain  $\Omega \equiv [-1.5, 1.5] \times [0, 1]$  shown in Figure 5-1. The governing equation is given by

$$\nabla \cdot (\beta u) - \nabla \cdot (\epsilon \nabla u) = f \quad \text{in } \Omega,$$

where  $\beta \in \mathbb{R}^2$  defines the advection field,  $\epsilon \in \mathbb{R}^+$  is the viscosity, and  $f$  is the source function. For all the problems considered, we set  $\beta = [1, 0]$  and  $\epsilon = 10^{-3}$ , so that the Peclet



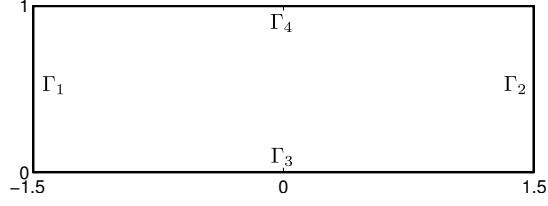


Figure 5-1: The domain for the advection-diffusion problems.

number is  $10^3$ . The boundary conditions are given by

$$\begin{aligned} -(\beta \cdot n)u + \epsilon \frac{\partial u}{\partial n} &= 0, & \text{on } \Gamma_1 \\ \epsilon \frac{\partial u}{\partial n} &= 0, & \text{on } \Gamma_2 \\ u &= u_{\Gamma_3}, & \text{on } \Gamma_3 \\ u &= 0, & \text{on } \Gamma_4, \end{aligned}$$

where the boundaries  $\Gamma_i$ ,  $i = 1, \dots, 4$ , are as specified in Figure 5-1, and  $u_{\Gamma_3}$  specifies the solution value on  $\Gamma_3$ . The general form of the output functional considered is

$$\mathcal{J}(u) = \int_{\Omega} g_{\Omega} u ds + \int_{\Gamma_3} g_{\Gamma_3} \epsilon \frac{\partial u}{\partial n} ds,$$

where  $g_{\Omega}$  and  $g_{\Gamma_3}$  are the two parameters that characterize the output. For the specified form of the output and the boundary conditions, the dual problem is given by

$$-\beta \cdot \nabla \psi - \nabla \cdot (\epsilon \nabla u) = g_{\Omega} \quad \text{in } \Omega$$

with the boundary conditions

$$\begin{aligned} \epsilon \frac{\partial \psi}{\partial n} &= 0, & \text{on } \Gamma_1 \\ (\beta \cdot n)\psi + \epsilon \frac{\partial \psi}{\partial n} &= 0, & \text{on } \Gamma_2 \\ \psi &= g_{\Gamma_3}, & \text{on } \Gamma_3 \\ \psi &= 0, & \text{on } \Gamma_4. \end{aligned}$$

We will consider three different combinations of the source function  $f$ , the boundary value  $u_{\Gamma_3}$ , and the output functional parameters  $g_{\Omega}$  and  $g_{\Gamma_3}$  to produce primal and dual



	Primal-Dual	Dual-Only	Primal-Only
$f$	0	$-\sin\left(\frac{10\pi}{3}x_1\right)\sin(\pi x_2)$	0
$u_{\Gamma_3}$	1	0	1
$g_\Omega$	0	0	$g_\Omega^{\text{prim}}$
$g_{\Gamma_3}$	1	1	0
Primal Solution	$P_1$	$P_2$	$P_1$
Dual Solution	$D_1$	$D_1$	$D_2$

Table 5.1: Set of parameters defining the three advection-diffusion problems. The volume output weight for the primal-only case is  $g_\Omega^{\text{prim}} = \frac{1}{2\pi(0.0012)} \exp\left(-\frac{1}{2}\left[\frac{x_1^2}{0.02^2} + \frac{(x_2-0.25)^2}{0.06^2}\right]\right)$ . The solution identifications correspond to those in Figure 5-2.

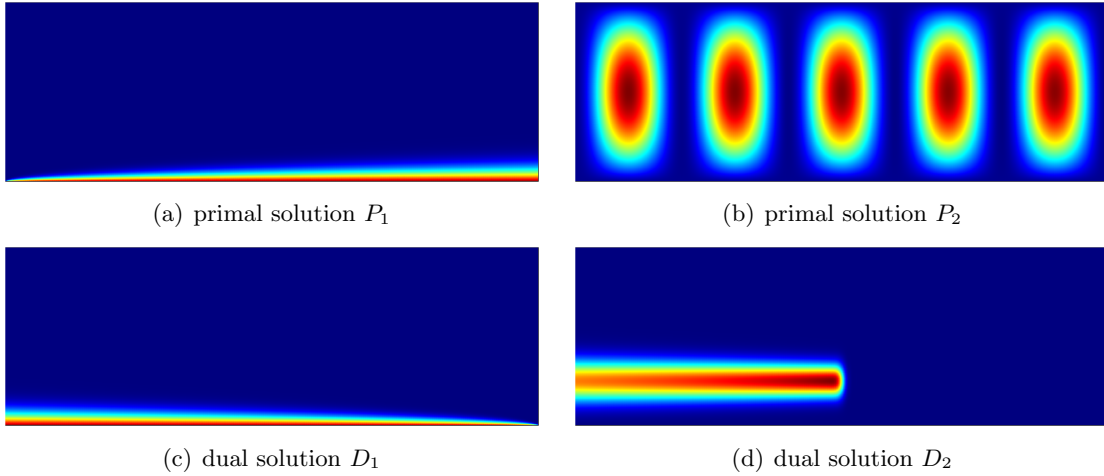


Figure 5-2: Solutions to the boundary layer problems.

solutions suitable for assessing MOESS. The choice of problem parameters and the corresponding primal and dual solutions are summarized in Table 5.1. A pair of primal solutions,  $P_1$  and  $P_2$ , and a pair of dual solutions,  $D_1$  and  $D_2$ , are shown in Figure 5-2. The first problem is called “primal-dual,” as the choice of parameters induces boundary layers in both the primal and dual solutions ( $P_1$  and  $D_1$ ). The second problem is called “dual-only,” as it exhibits a boundary layer in the dual solution ( $D_1$ ) but not in the primal solution ( $P_2$ ). Similarly, the third problem is called “primal-only,” as a boundary layer appears only in the primal solution ( $P_1$ ) and not in the dual solution ( $D_2$ ).

Note that because the governing PDE is a scalar equation with constant coefficients, the *a priori* error bound in Proposition 2.5 simplifies to

$$\mathcal{E} \lesssim C \sum_{\kappa \in \mathcal{T}_h} \left[ \left( \frac{\|\beta\|_{\ell^\infty}}{h_{\min}} + \frac{|\kappa|}{h_{\min}^2} \right) \left( \int_{\kappa} E_{\mathcal{M}}^{s_u}(\mathcal{M}_\kappa; u) dx \right) \left( \int_{\kappa} E_{\mathcal{M}}^{s_\psi}(\mathcal{M}_\kappa; \psi) dx \right) \right]$$



where  $s_u = \min(p+1, k_u)$  and  $s_\psi = \min(p+1, k_\psi)$  for  $u \in H^{k_u}(\Omega)$  and  $\psi \in H^{k_\psi}(\Omega)$ . In other words, the higher derivatives of the primal and dual solution dictate the output error, and we can qualitatively assess the adapted meshes by comparing the meshes to the solutions shown in Figure 5-2.

## 5.2 Results

### 5.2.1 Assessment Procedure

In order to assess the effectiveness of MOESS applied to the advection-diffusion equation, we compare the approach with two different adaptation strategies. First is the isotropic refinement based on the DWR error estimate. Second is the anisotropic refinement that uses the DWR error estimate for the element area decision and the primal solution for the shape decision. Specifically, the method solves the  $p+1$  discretization of the primal problem, takes the first principal direction in the direction of the maximum  $p+1$  derivative, selects the ratio of the first and second principal lengths to equidistribute the interpolation error in the two principal directions, and scales the principal lengths to achieve the desired area. The detailed implementation of the algorithm is presented in [156]. We emphasize that all adaptations strategies in this chapter use the adjoint-based error estimate; the primary difference in the methods lies in the anisotropy decision process.

For each of the advection-diffusion problems, the finite element solutions are obtained using the  $p = 1$  and  $p = 2$  discretizations at 250, 500, 1000, and 2000 degrees of freedom. The reference solutions are obtained using the  $p = 3$  discretization at 40,000 degrees of freedom.

### 5.2.2 Primal-Dual Boundary Layer

The primal-dual boundary layer problem exhibits boundary layers in both the primal ( $P_1$ ) and dual ( $D_1$ ) solution, as shown in Figure 5-2. Figure 5-3 shows the output error convergence for the three adaptation schemes using the  $p = 1$  and  $p = 2$  discretizations. Compared to isotropic adaptation, MOESS reduces the error by approximately two orders of magnitude for  $p = 1$  and three orders of magnitude for  $p = 2$  at a given number of degrees of freedom. These reductions are achieved for the moderate Peclet number of  $10^3$ ; the advantage of the anisotropic boundary layer resolution further increases for higher Peclet number



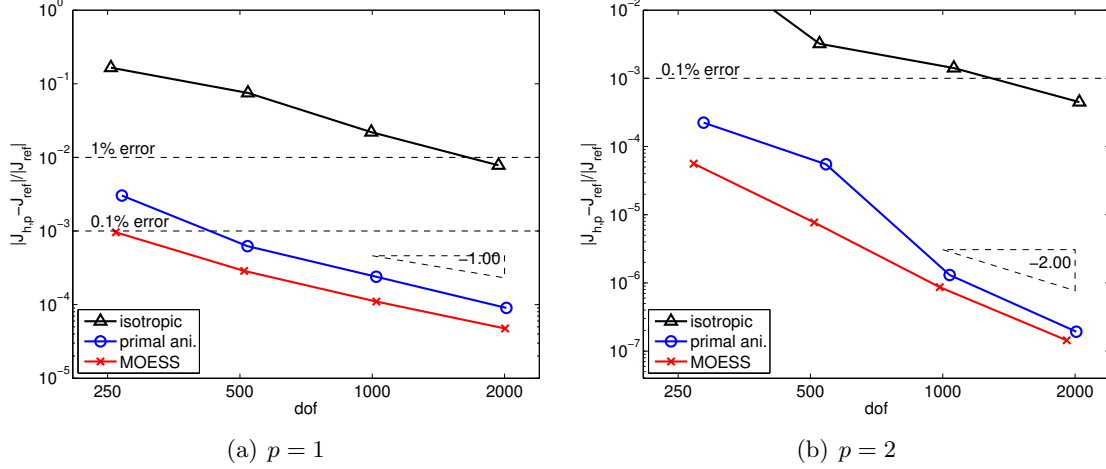


Figure 5-3: Output error convergence for the primal-dual boundary layer problem using the  $p = 1$  and  $p = 2$  discretizations.

cases. Note that the  $p = 2$  discretization outperforms the  $p = 1$  discretization for the entire range of the degrees of freedom considered, and the adaptive methods achieve the asymptotic output convergence rate of  $\mathcal{E} \sim h^{2p} \sim (\text{dof})^p$  even on coarse meshes.

For this problem, the primal-based anisotropy detection is expected to perform well because the solution anisotropy of the primal and dual solutions match each other. Thus, targeting the primal anisotropic feature coincidentally results in resolving the dual anisotropic feature. Nevertheless, Figure 5-3 shows that MOESS outperforms the primal-based anisotropy detection for both  $p = 1$  and  $p = 2$ .

Figure 5-4 shows the  $p = 1$  meshes with 1000 degrees of freedom and  $p = 2$  meshes with 2000 degrees of freedom. Because the primal and dual solutions are mirror image of each other about  $x_1 = 0$ , the isotropic adaptation produces a mesh whose size functions are symmetric about  $x_1 = 0$ , as shown in Figures 5-4(a) and 5-4(b). Recalling that the output error is a (weighted) product of the primal and dual errors, the symmetry of the mesh (and hence the equal level of the resolution of primal and dual solutions) agrees with intuition. On the other hand, the primal-based anisotropy results in a scheme that is biased toward resolving the directional features in the primal solution, as shown in Figures 5-4(c) and 5-4(d). The biased-treatment of the primal solution suggests that the element anisotropy is not optimal. Nevertheless, for example on the optimized  $p = 2$  mesh, the primal-based anisotropy detection results in over 60% of the elements having aspect ratio over 10 and 20% having the aspect ratio over 30, contributing to the efficient resolution of the boundary layer



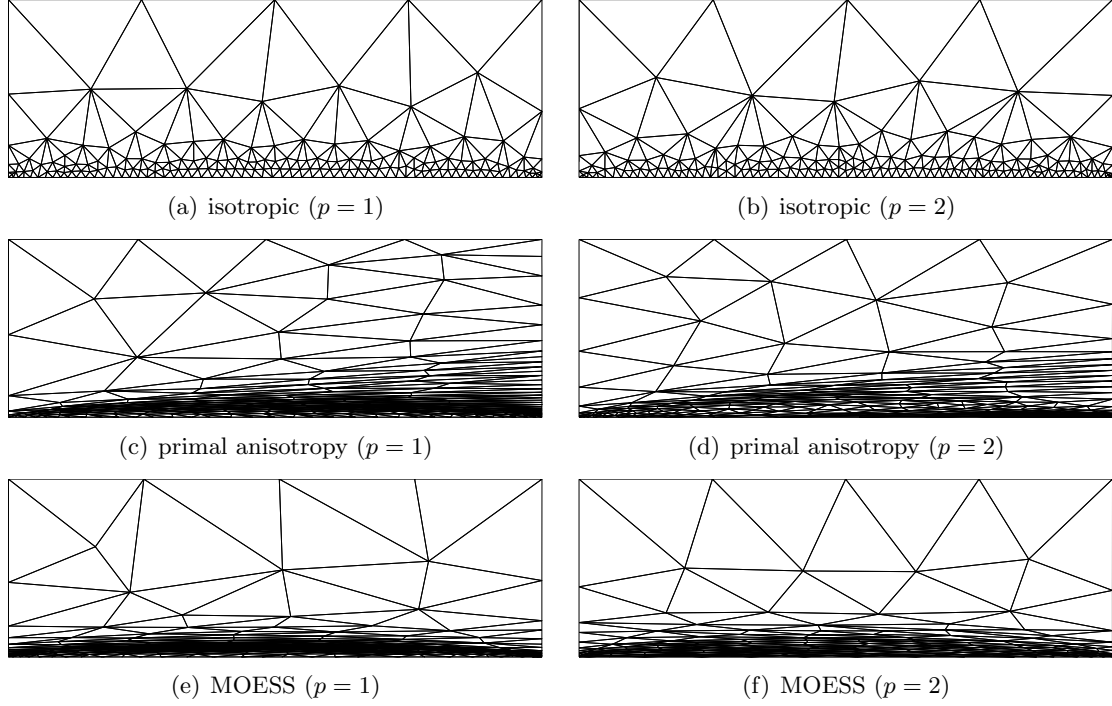


Figure 5-4: Adapted meshes for the primal-dual boundary layer problem. All  $p = 1$  and  $p = 2$  meshes have  $\text{dof} = 1000$  and  $\text{dof} = 2000$ , respectively.

and outperforming the isotropic adaptation. Figure 5-4(e) and 5-4(f) show that MOESS produces meshes whose size and anisotropy distributions are symmetric about  $x_1 = 0$ . This is not surprising, as the method is driven completely by the behavior of the *a posteriori* error estimate and automatically balances the influences of the primal and dual solutions for this case. On the  $p = 2$  optimized mesh, over 80% of the elements have aspect ratio of over 10 and 20% have the aspect ratio of over 30.

### 5.2.3 Dual-Only Boundary Layer

The dual-only boundary layer problem produces a boundary layer in the dual solution ( $D_1$ ) but not in the primal solution ( $P_2$ ), as shown in Figure 5-2. Figure 5-5 shows the output error convergence for the three adaptation schemes using the  $p = 1$  and  $p = 2$  discretizations. For the  $p = 1$  discretization, MOESS requires approximately half the degrees of freedom of the isotropic adaptation to achieve a given error tolerance. The dof-saving is slightly lower for the  $p = 2$  discretization, but MOESS nevertheless improves efficiency. For this problem, the primal-based anisotropy detection performs worse than the isotropic adaptation for both  $p = 1$  and  $p = 2$ , requiring about twice as many degrees of freedom to achieve a given error



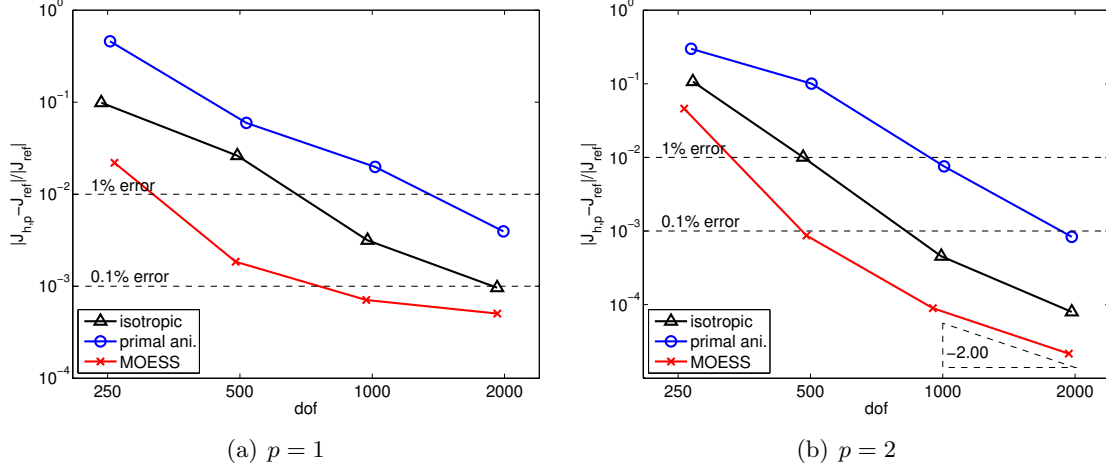


Figure 5-5: Output error convergence for the dual-only boundary layer problem using the  $p = 1$  and  $p = 2$  discretizations.

tolerance. The loss of efficiency is due to the use of inappropriate anisotropy, as we will see shortly.

Figure 5-6 shows  $p = 1$ ,  $\text{dof} = 1000$  meshes and  $p = 2$ ,  $\text{dof} = 2000$  meshes obtained using the three adaptation methods. The isotropic adaptation targets the boundary layer in the dual solution as shown in Figures 5-6(a) and 5-6(b), however, its efficiency is limited due to the use of isotropic elements. Figures 5-6(c) and 5-6(d) show that primal-based anisotropy detection produces elements that are aligned with the primal sine source function. This  $x_2$ -aligned anisotropy is inappropriate for resolving the dual boundary layer, resulting in the method performing worse than the isotropic adaptation. MOESS targets the dual boundary layer using anisotropic elements, as shown in Figures 5-6(e) and 5-6(f). However, because the primal solution is not anisotropic near the bottom wall, the elements are not as anisotropic as those in the primal-dual boundary layer problem. For example, for the  $p = 2$  discretization, the fraction of elements with the aspect ratio of over 30 are only 6% for this case, compared to over 20% for the primal-dual boundary layer case. The result again demonstrates that MOESS automatically balances the resolution of the primal and dual solution to minimize the output error.

#### 5.2.4 Primal-Only Boundary Layer

The primal-only boundary layer problem considers a regularized line output, and produces a boundary layer in the primal solution ( $P_1$ ) but not in the dual solution ( $D_2$ ), as shown



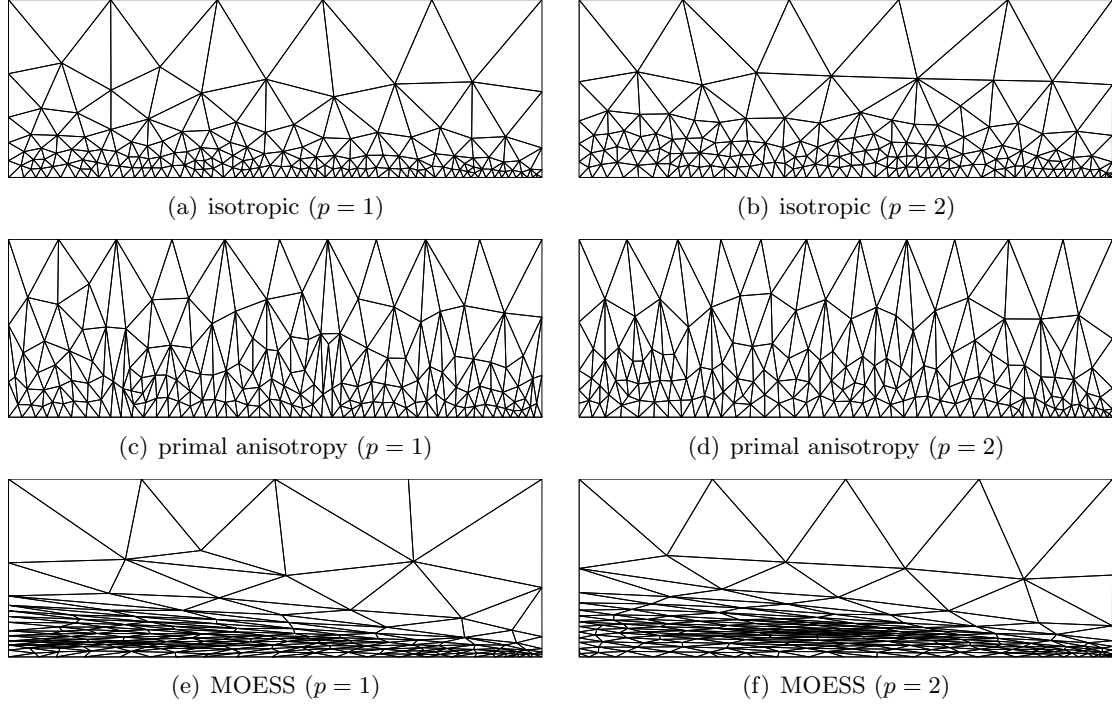


Figure 5-6: Adapted meshes for the dual-only boundary layer problem. All  $p = 1$  and  $p = 2$  meshes have  $\text{dof} = 1000$  and  $\text{dof} = 2000$ , respectively.

in Figure 5-2. The output error convergence for the primal-only boundary layer problem is shown in Figure 5-7. For the  $p = 1$  discretization, both anisotropic adaptation methods are significantly more efficient than the isotropic adaptation, reducing the number of degrees of freedom required to meet a given error by almost an order of magnitude. On the other hand, for the  $p = 2$  discretization, the primal-based anisotropy detection performs worse than the isotropic adaptation for  $\text{dof} \geq 1000$ . In fact, the primal-based anisotropy renders the  $p = 2$  discretization less efficient than the  $p = 1$  discretization. MOESS applied to the  $p = 2$  discretization converges rapidly to the true solution as the number of degrees of freedom increases.

Figure 5-8 shows  $p = 1$ ,  $\text{dof} = 1000$  meshes and  $p = 2$ ,  $\text{dof} = 2000$  meshes obtained using the three adaptation strategies. Figures 5-8(a) and 5-8(b) show that the isotropic adaptation targets the region of overlap between the primal boundary layer and the dual shear layer; however, similar to the previous two cases, its efficiency is limited due to the use of isotropic elements.

Figure 5-8(c) and 5-8(c) show that, for both  $p = 1$  and  $p = 2$ , the primal-based anisotropy detection employs anisotropic elements suitable for resolving the primal bound-



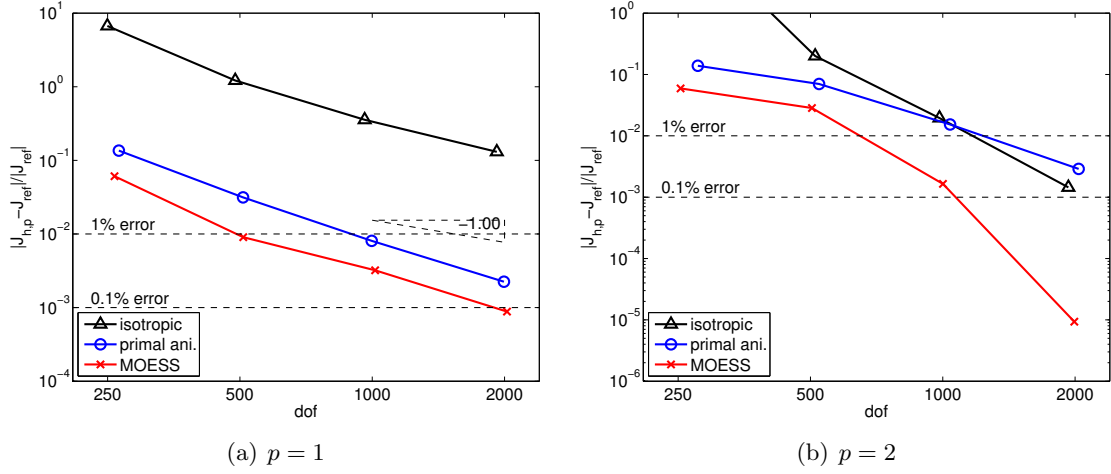


Figure 5-7: Output error convergence for the primal-only boundary layer problem using the  $p = 1$  and  $p = 2$  discretizations.

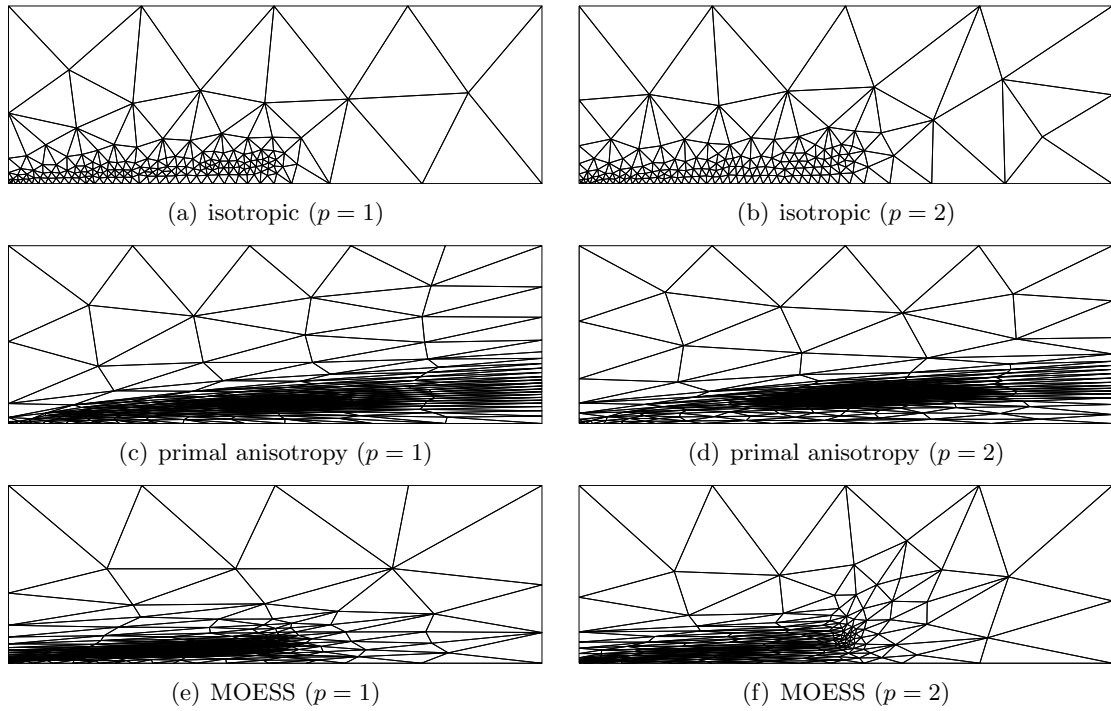


Figure 5-8: Adapted meshes for the primal-only boundary layer problem. All  $p = 1$  and  $p = 2$  meshes have  $\text{dof} = 1000$  and  $\text{dof} = 2000$ , respectively.



ary layer. Because its sizing decision is based on DWR, smaller elements are employed in vicinity of the dual source term and elements downstream of the source location are large. We note that the apparent refinement in the region downstream the source is due to the use of highly anisotropic elements. However, these anisotropic elements aligned with the primal boundary layer are unsuited for resolving the dual solution that exhibit a strong variation in the  $x_1$ -direction just downstream of the source. The poor  $p = 2$  performance of the primal-based anisotropy detection shows that anisotropy only suited for resolving the primal solution is in fact worse than no anisotropy at all even for the problem that exhibits a strong anisotropic feature in the primal solution. Furthermore, the degradation in the performance is dependent on the discretization order.

MOESS balances the anisotropy requirements for resolving the primal and dual solutions, as shown in Figures 5-8(e) and 5-8(f). In the region upstream of the dual source, both the primal and dual solution exhibit a strong variation in the  $x_2$ -direction, and the algorithm resolves these features using highly anisotropic elements. However, just downstream of the Gaussian source, the dual solution experiences a strong variation in the  $x_1$  direction. For the  $p = 1$  discretization at this error level, the result suggests that the primal solution requires more resolution than the dual solution in the region, resulting in elements that provide more resolution in the  $x_2$  direction. On the other hand, for the  $p = 2$  discretization, the resolution requirement for primal and dual solutions are balanced, and the algorithm employs isotropic elements. The difference in the anisotropy requirement for  $p = 1$  and  $p = 2$  explains the degradation in the performance observed for the  $p = 2$  discretization with the primal-based anisotropy detection.

### 5.3 Conclusions

Using three advection-diffusion problems with carefully chosen primal and dual solutions, we studied the behavior of MOESS and compared its performance with the isotropic adaptation and the anisotropic adaptation with primal-based anisotropy detection. For all three problems considered, MOESS outperformed the other two strategies for both  $p = 1$  and  $p = 2$ . The results also highlight that the primal-based anisotropy can perform worse than isotropic refinement even if the primal solution exhibits strong directional features. Furthermore, the optimal anisotropy is highly dependent on the discretization order.



## Chapter 6

# Compressible Navier-Stokes Equations

This chapter applies our adaptation algorithm, Mesh Optimization via Error Sampling and Synthesis (MOESS), to aerodynamic flows governed by the Euler equations (inviscid), the compressible Navier-Stokes equations (laminar), and the Reynolds-averaged Navier-Stokes equations (turbulent). These equations exhibit a number of challenges common to solving many PDEs in engineering, including: convection dominance, a wide range of scales, anisotropic solution features, high nonlinearity, and discontinuities. The presence of geometry- and nonlinearity-induced singularities poses challenges for high-order discretizations, both in terms of robustness and efficiency. Thus, aerodynamic simulations serve as an excellent testbed for autonomous PDE solver technologies. An overview of challenges associated with the development of reliable computational fluid dynamics (CFD) capabilities is provided by Allmaras *et al.* [9]. Recent reviews of error estimation and adaptation technologies for aerospace CFD applications are detailed in Hartmann and Houston [72] and Fidkowski and Darmofal [55].



## 6.1 Governing Equations

### 6.1.1 Euler and Navier-Stokes Equations

The compressible Navier-Stokes equations consists of  $m = d+2$  equations. The conservative state consists of mass, momentum, and energy per unit volume and is given by

$$u = \begin{pmatrix} \rho \\ \rho v_j \\ \rho E \end{pmatrix},$$

where  $\rho$  is the density,  $v_j$  is the velocity in the  $j$ -th coordinate direction, and  $E$  is the total internal energy per unit mass. The convective (or inviscid) flux in the  $i$ -th coordinate direction is given by

$$\mathcal{F}_i^{\text{conv}} = \begin{pmatrix} \rho v_i \\ \rho v_j v_i + \delta_{ij} p \\ \rho H v_i \end{pmatrix},$$

where the pressure,  $p$ , and the total enthalpy,  $H$ , are given by

$$p = (\gamma - 1) \left( \rho E - \frac{1}{2} \rho v_i v_i \right)$$

$$H = E + \frac{p}{\rho},$$

and  $\gamma$  is the ratio of specific heats.

The diffusive (or viscous) flux in the  $i$ -th coordinate direction is given by

$$\mathcal{F}_i^{\text{diff}} = \begin{pmatrix} 0 \\ \tau_{ij} \\ \tau_{ij} v_j + \kappa_T \frac{\partial T}{\partial x_i} \end{pmatrix},$$

where  $\tau$  is the shear stress,  $\kappa_T$  is the thermal conductivity, and  $T = p/(\rho R)$  is the temperature, and  $R$  is the gas constant. The shear stress is given by

$$\tau_{ij} = \mu \left( \frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right) + \delta_{ij} \lambda \frac{\partial v_k}{\partial x_k},$$



where  $\mu$  is the dynamic viscosity, and  $\lambda = -2/3\mu$  is the bulk viscosity coefficient. The dynamic viscosity is modeled using Sutherland's law,

$$\mu = \mu_{\text{ref}} \left( \frac{T}{T_{\text{ref}}} \right)^{1.5} \frac{T_{\text{ref}} + T_s}{T + T_s},$$

unless specified otherwise. The thermal conductivity,  $\kappa_T$ , is related to the dynamic viscosity by the Prandtl number,  $Pr$ , according to

$$\kappa_T = c_p \frac{\mu}{Pr},$$

where  $c_p = \gamma R/(\gamma - 1)$  is the specific heat at constant pressure. The implementation of various boundary conditions follows those detailed in [110].

### 6.1.2 Reynolds-Averaged Navier-Stokes Equations

The Reynolds-averaged Navier-Stokes (RANS) equations are obtained by temporally averaging the Navier-Stokes equations using the Favre averaging procedure. In this work, the closure of the RANS system is accomplished by the Spalart-Allmaras (SA) turbulence model [137]. The particular implementation of the SA equation used in this work incorporates two modifications by Oliver and Allmaras to the original SA model [110]. First is a generalization of the original model for incompressible flow to compressible flow. Second is a set of modifications intended to improve the robustness of the RANS-SA simulations for higher-order discretizations.

The conservative variable for the RANS-SA equations, corresponding to the mean state, is given by

$$u = \begin{pmatrix} \rho \\ \rho v_j \\ \rho E \\ \rho \tilde{\nu} \end{pmatrix},$$

where  $\tilde{\nu}$  is the working variable for the turbulence model, which is algebraically related to



the eddy viscosity,  $\mu_t$ , by

$$\mu_t = \begin{cases} \rho \tilde{\nu} f_{v1}, & \tilde{\nu} \geq 0 \\ 0, & \tilde{\nu} < 0 \end{cases},$$

where

$$f_{v1} = \frac{\chi^3}{\chi^3 + c_{v1}^3}, \quad \chi = \frac{\tilde{\nu}}{\nu},$$

and  $\nu = \mu/\rho$  is the kinematic viscosity. The convective and diffusive fluxes of the RANS-SA equations in the  $i$ -th coordinate direction are given by

$$\mathcal{F}_i^{\text{conv}} = \begin{pmatrix} \rho v_i \\ \rho v_j v_i + \delta_{ij} p \\ \rho H v_i \\ \rho \tilde{\nu} \end{pmatrix} \quad \text{and} \quad \mathcal{F}_i^{\text{diff}} = \begin{pmatrix} 0 \\ \tau_{ij}^{\text{RANS}} \\ \tau_{ij}^{\text{RANS}} v_j + \kappa_T^{\text{RANS}} \frac{\partial T}{\partial x_i} \\ \frac{1}{\sigma} \eta \frac{\partial \tilde{\nu}}{\partial x_i} \end{pmatrix},$$

where the effective shear stress,  $\tau^{\text{RANS}}$ , and the thermal conductivity,  $\kappa_T^{\text{RANS}}$ , incorporate the effect of the eddy viscosity, i.e.

$$\begin{aligned} \tau^{\text{RANS}} &= (\mu + \mu_t) \left[ \left( \frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right) + \delta_{ij} \lambda \frac{\partial v_k}{\partial x_k} \right] \\ \kappa_T^{\text{RANS}} &= c_p \left( \frac{\mu}{Pr} + \frac{\mu_t}{Pr_t} \right), \end{aligned}$$

and the diffusion coefficient for the SA equation,  $\eta$ , is

$$\eta = \begin{cases} \mu(1 + \chi), & \chi \geq 0 \\ \mu(1 + \chi + \frac{1}{2}\chi^2), & \chi < 0 \end{cases}.$$

The source term of the RANS-SA system is given by

$$\mathcal{S} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ P - D + c_{b2} \rho \frac{\partial \tilde{\nu}}{\partial x_k} \frac{\partial \tilde{\nu}}{\partial x_k} \end{pmatrix}.$$



The production term,  $P$ , of the SA equation is

$$P = \begin{cases} c_{b1} \tilde{S} \rho \tilde{\nu}, & \chi \geq 0 \\ c_{b1} S \rho \tilde{\nu} g_n, & \chi < 0 \end{cases},$$

where  $g_n = 1 - f_{g_n} \chi^2 / (1 + \chi^2)$ ,  $f_{g_n} = 10^5$ , and

$$\tilde{S} = \begin{cases} S + \bar{S}, & \bar{S} \geq -c_{v2} S \\ S + \frac{S(c_{v2}^2 S + c_{v3} \bar{S})}{(c_{v3} - 2c_{v2})S - \bar{S}}, & \bar{S} < -c_{v2} S \end{cases}.$$

Here,  $S = \sqrt{2\Omega_{ij}\Omega_{ij}}$  is the magnitude of the vorticity,  $\Omega_{ij} = \frac{1}{2}(\frac{\partial u_i}{\partial x_j} - \frac{\partial u_j}{\partial x_i})$ , and the near wall correction term is given by

$$\bar{S} = \frac{\tilde{\nu} f_{v2}}{\kappa^2 d^2} \quad \text{with} \quad f_{v2} = 1 - \frac{\chi}{1 + \chi f_{v1}},$$

where  $d$  is the distance to the nearest wall. The destruction term,  $D$ , is given by

$$D = \begin{cases} c_{w1} f_w \frac{\rho \tilde{\nu}^2}{d^2}, & \chi \geq 0 \\ -c_{w1} \frac{\rho \tilde{\nu}^2}{d^2}, & \chi < 0 \end{cases},$$

where

$$f_w = g \left( \frac{1 + c_{w3}^6}{g^6 + c_{w3}^6} \right)^{1/6}, \quad g = r + c_{w2}(r^6 - r), \quad \text{and} \quad r = \frac{\tilde{\nu}}{\tilde{S} \kappa^2 d^2}.$$

The constants of the turbulence model are set to:  $c_{b1} = 0.1355$ ,  $\sigma = 2/3$ ,  $c_{b2} = 0.622$ ,  $\kappa = 0.41$ ,  $c_{w1} = c_{b1}/\kappa^2 + (1 + c_{b2})/\sigma$ ,  $c_{w2} = 0.3$ ,  $c_{w3} = 2$ ,  $c_{v1} = 7.1$ ,  $c_{v2} = 0.7$ ,  $c_{v3} = 0.9$ , and  $Pr_t = 0.9$ .

## 6.2 The Importance of Mesh Adaptation for Higher-Order Discretizations of Aerodynamic Flows

Let us study the importance of adaptation for higher-order discretizations using two simple flows over isolated airfoils. The first problem considered is Euler flow with a single dominant geometry-induced singularity, which is similar to the  $r^\alpha$  singularity studied in the context



$L^2$  error control in Section 4.3. The second problem is subsonic RANS flow exhibiting various features with a wide range of scales. For second-order methods, careful studies quantifying the effect of adaptation for aerodynamic flows exhibiting multiple scales have been conducted by Dervieux *et al.* [51] and Loseille *et al.* [99]. This section quantifies the effect of adaptation for higher-order discretizations of aerodynamic flows.

### 6.2.1 NACA 0012 Subsonic Euler

The first problem considered is  $M_\infty = 0.5$  Euler flow over a NACA 0012 airfoil at  $\alpha = 2^\circ$ . The farfield boundary is set at 10,000 chord (i.e.  $10000c$ ) away from the airfoil to minimize the finite-boundary effect on the lift and drag. To quantify the effect of adaptation, adaptive refinement is first performed using MOESS at 5,000 degrees of freedom. Then, for uniform refinement, each element of the optimized mesh is divided into four elements and the solution is obtained on the refined mesh having 20,000 degrees of freedom. For adaptive refinement, the results are obtained using the MOESS algorithm at 20,000 degrees of freedom. The results for the uniformly and adaptively refined  $\text{dof} = 20,000$  meshes for  $p = 1$ ,  $p = 2$ , and  $p = 3$  discretizations are compared to assess the impact of adaptation for different discretization orders.

The key feature that dictates the drag error of this problem is the geometry-induced singularity at the trailing edge of the airfoil. A simplified analysis based on the potential flow theory reveals that the singularity is of the  $r^\alpha$ -type studied in Section 4.3, where the strength  $\alpha$  is dependent on the trailing edge angle. Thus, given a sequence of properly graded meshes, we expect the optimal output error convergence rate of  $\mathcal{E} \sim \bar{h}^{2p+1} \sim (\text{dof})^{-(p+1/2)}$  for this hyperbolic problem (c.f. [67]).

Figure 6-1 shows the behavior of the drag error. With uniform refinement, the  $p = 1$  discretization converges at a rate close to the optimal rate (against the  $\text{dof}$ ) of  $-1.5$ . On the other hand, uniform refinement limits the convergence rate of both the  $p = 2$  and  $p = 3$  discretizations to approximately  $-1.5$ , which is significantly lower than the optimal rates of  $-2.5$  and  $-3.5$ , respectively. The suboptimal convergence rates are due to presence of the corner singularity at the trailing edge; the result is consistent with the theory.

With adaptive refinement, the  $p = 2$  and  $p = 3$  discretizations achieve the convergence rates of  $-2.8$  and  $-4.0$ , respectively, which are slightly higher than the optimal error rate based on the *a priori* error analysis. The output convergence rate of the  $p = 1$  discretization



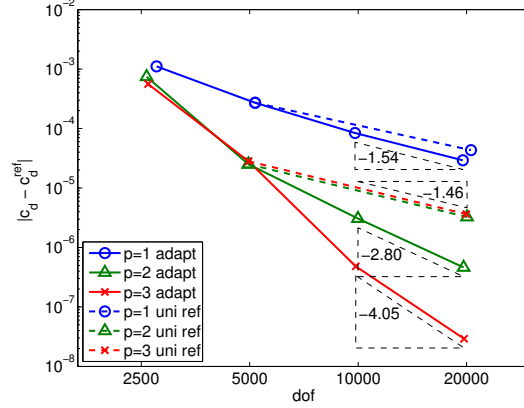


Figure 6-1: Comparison of the error convergence for uniform and adaptive refinements for the subsonic NACA 0012 Euler flow.

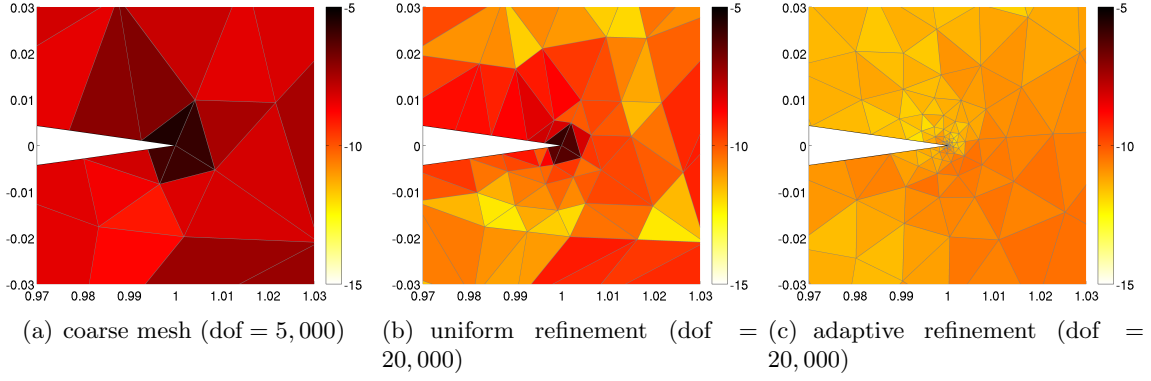


Figure 6-2: Comparison of the trailing edge mesh grading and error indicator distribution of the  $p = 3$ , dof = 20,000 meshes obtained from uniform and adaptive refinements of the  $p = 3$ , dof = 5,000 optimized mesh for the subsonic NACA 0012 Euler flow. The color scale is in  $\log_{10}(\eta_\kappa)$ .

does not significantly improve from that obtained using uniform refinement as the rate was near-optimal. The result does not imply that adaptation is not important for the  $p = 1$  discretization. The efficiency of the  $p = 1$  discretization is still dependent on the element size distribution; however, once a good initial element size distribution is obtained, then uniform refinement is sufficient to maintain a near-optimal performance. This is contrary to the  $p = 2$  and  $p = 3$  discretizations, whose performances significantly degrade with uniform refinement even if the initial element size distribution is optimal.

To understand the differences in the error convergence behaviors of the uniform and adaptive refinements, the element size and local error distribution near the trailing edge singularity are analyzed. Figure 6-2 shows the error indicator distribution at the trailing edge for the original  $p = 3$ , dof = 5,000 mesh, the  $p = 3$ , dof = 20,000 mesh that results



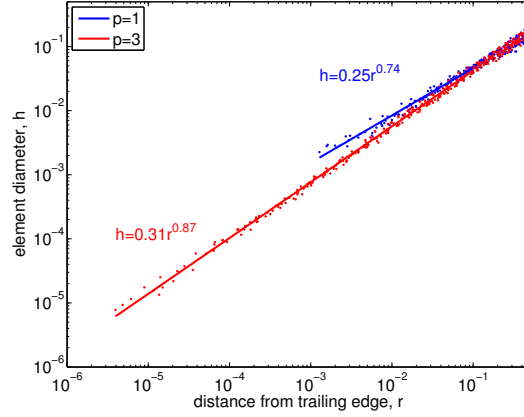


Figure 6-3: Element size distributions in the vicinity of the trailing edge of the  $p = 1$  and  $p = 3$  optimized meshes for the subsonic NACA 0012 Euler flow.

from a step of uniform refinement, and a  $p = 3$ ,  $\text{dof} = 20,000$  mesh obtained after adaptive refinements. On the original mesh with 5,000 degrees of freedom shown in Figure 6-2(a), the adaptation algorithm nearly equidistributes the error indicator. (Note that completely equidistributing the error on this coarse mesh with only 500 elements is difficult.) Figure 6-2(b) shows that a step of uniform refinement significantly reduces the error contribution from the elements not on the trailing edge but only marginally improves the error contribution from the trailing edge elements. As a result, the error contribution of the trailing edge elements is several orders of magnitude greater than that of other elements, indicating the mesh is suboptimal due to the inefficient element size distribution. Figure 6-2(c) shows that the adaptive refinement targets the corner elements dominating the error and produces a strongly graded mesh that nearly equidistributes the error. The diameter of the trailing edge element is approximately  $1 \times 10^{-5}c$  for the adapted mesh, whereas that for the uniformly refined mesh is approximately  $5 \times 10^{-3}c$ . In other words, in increasing the number of degrees of freedom by a factor of 4, the  $p = 3$  adaptive refinement decreases the trailing edge element diameter by a factor of 1,000, instead of a factor of 2 obtained by a step of uniform refinement.

The difference in the mesh grading required to achieve optimality for the  $p = 1$  and  $p = 3$  discretizations is also important to understand. Figure 6-3 shows the variation in the element diameter,  $h$ , as a function of the distance from the trailing edge,  $r$ , for  $p = 1$  and  $p = 3$  optimized meshes. Each mesh has approximately 2,000 elements. Recalling from Section 4.3 that the optimal element size distribution for the  $r^\alpha$ -type singularity is of



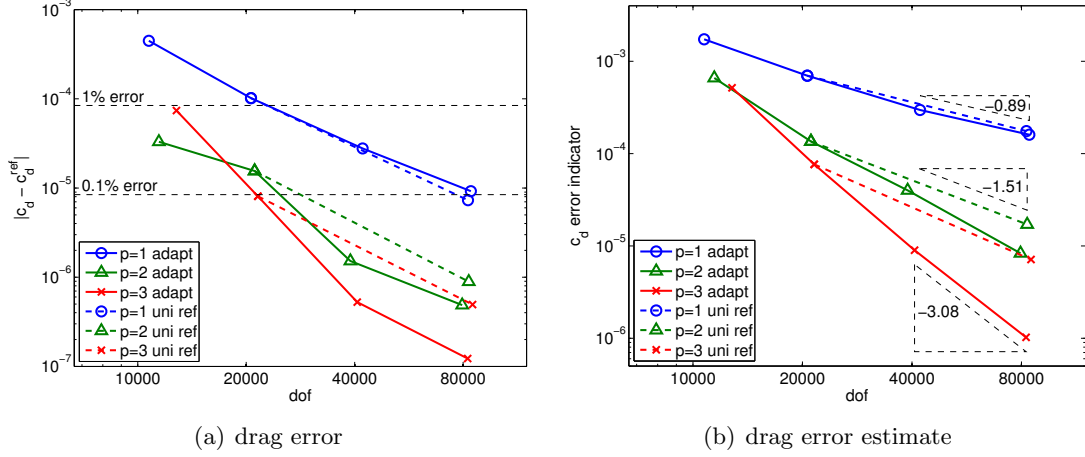


Figure 6-4: Comparison of the error convergence for uniform and adaptive refinements for the subsonic RAE 2822 RANS-SA flow.

the form  $h \sim r^k$  where  $k$  is the grading factor, the solid regression lines are produced by performing least-squares fit in the  $\log(h)$ - $\log(r)$  space for the elements in  $r < 0.1c$ . The  $p = 3$  optimal mesh has a grading factor of 0.87 and employs elements of diameter  $1 \times 10^{-5}c$  on the trailing edge. In comparison, the  $p = 1$  optimal mesh has a weaker grading factor of 0.74, and its trailing edge elements are of diameter  $3 \times 10^{-3}c$ . The higher-order discretization requires a stronger grading toward the corner singularity to equidistribute the error; this result is consistent with the analytical result for the  $r^\alpha$ -type singularity in Section 4.3. In addition, the higher-order discretization is more sensitive to suboptimal  $h$  distribution as the error scales with a higher power of  $h$ . Thus,  $h$ -adaptation is indispensable to achieve the full benefit of higher-order discretizations for flows with low regularity.

### 6.2.2 RAE 2822 Subsonic RANS-SA

The second problem we consider to quantify the importance of mesh adaptation for aerodynamic flows is a  $M_\infty = 0.3$ ,  $Re_c = 6.5 \times 10^6$  turbulent flow over an RAE 2822 airfoil at  $\alpha = 2.31^\circ$ . Following the procedure for the Euler NACA 0012 case, adaptation is first performed at 20,000 degrees of freedom to generate optimized meshes, and then uniform and adaptive refinements are started from those meshes.

The true drag error and the drag error estimate are shown in Figure 6-4. Similar to the Euler case, the adaptive refinement makes little difference in the convergence rate of the  $p = 1$  discretization compared to uniform refinement when the original mesh is optimized at a lower degrees of freedom. Again, the result does not imply adaptation is not important



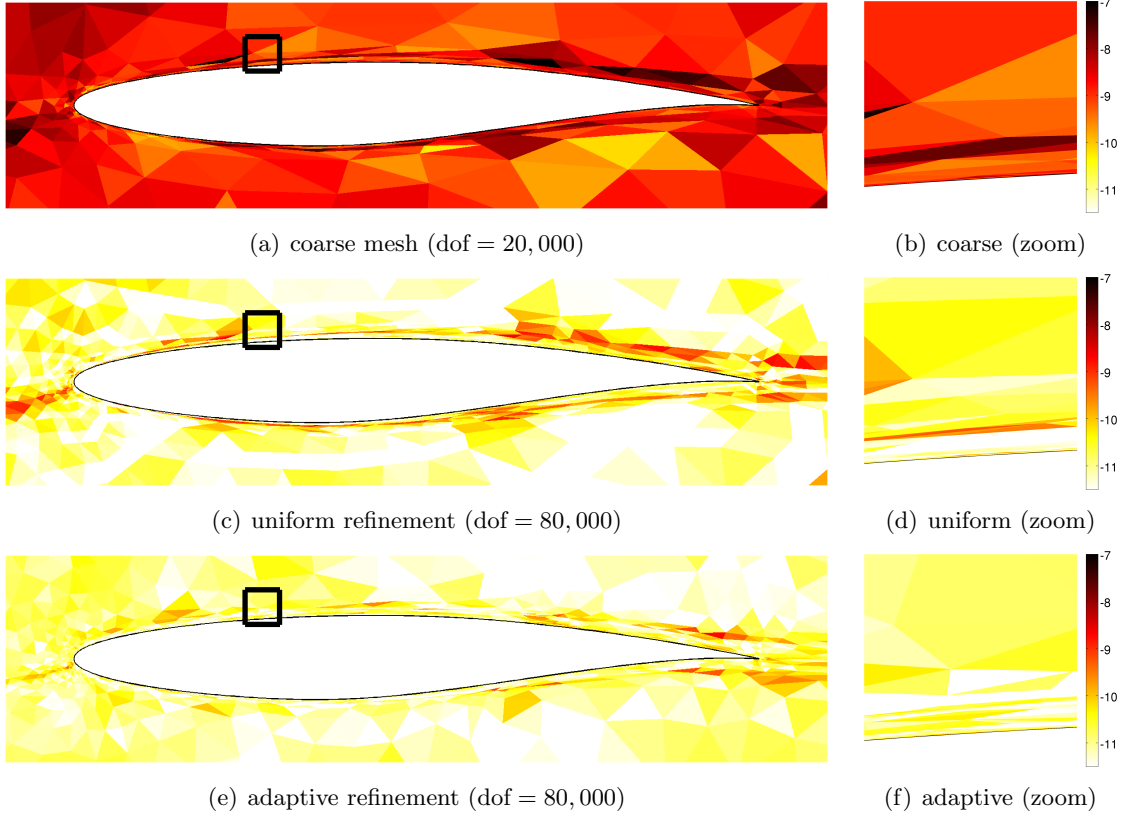


Figure 6-5: Comparison of the error indicator distributions of  $p = 3$ ,  $\text{dof} = 80,000$  meshes obtained from uniform and adaptive refinements of the  $p = 3$ ,  $\text{dof} = 20,000$  optimized mesh for the subsonic RAE 2822 RANS-SA flow. The color scale is in  $\log_{10}(\eta_\kappa)$ .

for  $p = 1$ ; it merely means that uniform refinement is sufficient to maintain a near-optimal performance given the refinement is applied on an optimized mesh. The convergence rate for the  $p = 2$  and  $p = 3$  discretizations are limited by the solution regularity when the mesh is uniformly refined; however, with the adaptive refinement, the optimal output error indicator convergence rate of  $\mathcal{E} \sim (\text{dof})^{2p/d}$  is recovered (cf. [67]). The optimized  $p = 3$  mesh achieves a drag error of approximately  $10^{-5}$  using 20,000 degrees of freedom (2,000 elements), whereas the optimal  $p = 1$  mesh requires 80,000 degrees of freedom (27,000 elements) to achieve the same fidelity. Thus, for a high-fidelity simulation, the  $p = 3$  discretization is significantly more efficient than the  $p = 1$  discretization. The result also shows that, at the level of  $c_d$  accuracy required in a practical engineering setting — which is about  $10^{-5}$  at minimum — the  $p = 3$  discretization is only marginally more efficient than the  $p = 2$  discretization.

In order to understand the region limiting the performance of uniform refinement, the



error indicator distribution obtained after a step of uniform refinement from the  $\text{dof} = 20,000$  optimized mesh is shown in Figure 6-5. The elements at the edge of the boundary layer have high error indicators, likely due to the singularity in the SA equation in the region [109]. The adaptive refinement correctly identifies the region and makes necessary adjustments to better control the error due to these high-error elements. The boundary layer edge singularity is an example of a flow feature that is hard to locate *a priori*. However, this subtle flow feature limits the convergence rate of higher-order discretizations and exemplifies the need for adaptation driven by an *a posteriori* error estimate for higher-order methods.

## 6.3 Assessment of MOESS Applied to Aerodynamic Flows

### 6.3.1 Assessment Procedure

We present numerical examples of applying MOESS to aerodynamic problems. Some of the results have been presented in [155]. As a comparison, we also provide the results obtained using the method based on fixed-fraction marking and the Mach number-based anisotropy detection [156], a modification of the algorithm developed by Fidkowski [57]. The fixed-fraction marking, which controls the size of the elements, is based on the DWR error indicator described in Section 2.2. The anisotropy request is driven by the  $p + 1$  derivative of the Mach number estimated by approximately solving the flow problem in the  $p$ -enriched space,  $V_{h,p+1}$ . Note that while the sizing decision accounts for the influence of the adjoint, the anisotropy decision is driven by a single scalar characterization of the primal solution. This approach will be referred to as the fixed-fraction Mach-anisotropy method, or FFMA, from here on.

Throughout this section, we will assess the performance of the adaptive procedures by measuring the true output error rather than the error estimate. As the analytical solutions to the problems are not available, we approximate the true output by computing the solution in a space that is much richer than the solutions being compared by increasing the number of degrees of freedom and, in some cases, the polynomial order. To assess the quality of the reference solution, we first adaptively solve the problem of interest in the enriched space using both the MOESS and the FFMA approach. If the error computed with respect to the two reference solutions is indistinguishable, then the reference solution is deemed accurate enough for the purpose of the assessment.



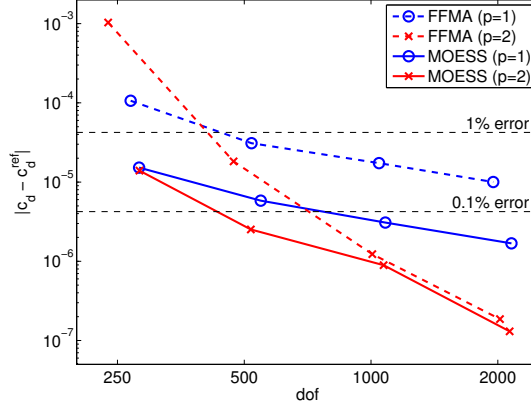


Figure 6-6: Drag error convergence for the laminar flat plate problem.

### 6.3.2 Laminar Flat Plate

We first consider laminar flow over a flat plate. The problem is solved on a rectangular domain of size  $[-0.5, 1.0] \times [0, 0.5]$  with the plate spanning from  $x = 0.0$  to  $1.0$ . The inflow Mach number is  $M_\infty = 0.2$ , the Reynolds number is  $Re_L = 10^5$ , and the adiabatic no-slip condition is specified along the plate. The output of interest is the drag on the plate. This canonical problem tests ability of MOESS to produce anisotropic elements in the boundary layer and to control the effect of the leading edge singularity.

Figure 6-6 shows the convergence of the drag error for the  $p = 1$  and  $p = 2$  discretizations adapted using MOESS and FFMA. The reference solution is obtained on an adapted  $p = 3$ ,  $dof = 20,000$  mesh. The convergence history shows that, for the  $p = 1$  discretization, the MOESS produces four to five times smaller drag error than FFMA for a given problem size. Another interpretation is that MOESS using 500 dof achieves a similar level of error as FFMA using 2,000 dof for  $p = 1$ . For the  $p = 2$  discretization, MOESS performs significantly better than FFMA on coarse meshes (e.g.  $dof = 250$ ); the improvement means that MOESS achieves the 0.5% error range ( $\approx 2 \times 10^{-5} c_d$ ) using approximately half the degrees of freedom of FFMA. In the asymptotic range, the  $p = 2$  performances of the two adaptive schemes are similar.

The difference in the output accuracy for the  $p = 1$  case is due to the difference in the anisotropy of the elements used to resolve the boundary layer, as shown in Figure 6-7. When the Mach-based anisotropy detection is employed, the anisotropy of elements on the wall is limited, as the Mach profile has an inflection point at the wall. Having a vanishing second derivative, the Mach-anisotropy detection employs elements with relatively small aspect



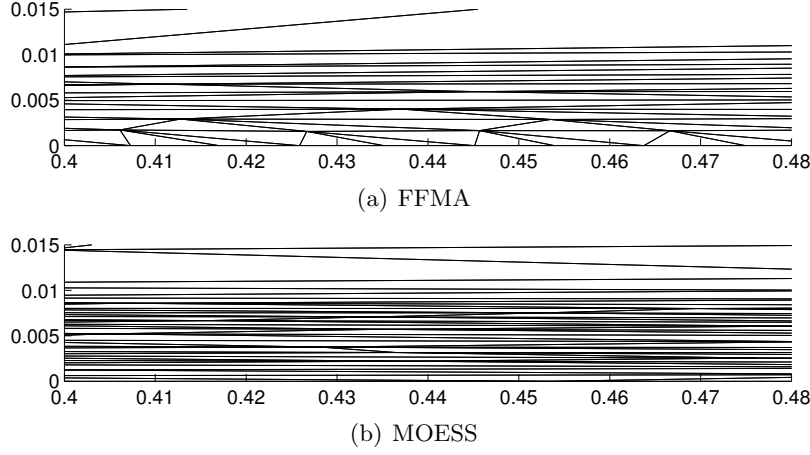


Figure 6-7: Close views of the meshes for the laminar flat plate problem. ( $p = 1$ ,  $\text{dof} = 2,000$ )

ratios on the wall. MOESS employs elements with much higher aspect ratios, resulting in a smaller error for  $p = 1$ . The difference between the adaptation strategies is smaller for  $p = 2$ , as the third derivative of the Mach profile is large near the wall and FFMA employs highly anisotropic elements on the wall.

This simple case demonstrates a problem of using *a priori* knowledge of the solution behavior to control anisotropy. While the Mach number has been found to be a good indicator for making the anisotropy decision in previous works [57, 72, 148], there are instances where the indicator fails to capture the anisotropic behavior of the flow. The example also demonstrates that the ability of the Mach-anisotropy to produce the required anisotropy is dependent on the discretization order. In particular, while the inappropriate aspect ratio that results from the presence of inflection points in the Mach number is a known problem for second-order discretizations [39], there could be instances where vanishing higher-order derivatives can lead to inappropriate aspect ratio for higher-order discretizations. In contrast, MOESS driven by the *a posteriori* error estimates from the local solves automatically considers the behaviors of all state variables, providing robust anisotropy decisions for arbitrary-order discretization of the system of equations.

### 6.3.3 RAE 2822 Transonic RANS-SA

We consider turbulent transonic flow over an RAE 2822 airfoil. The freestream Mach number is  $M_\infty = 0.734$ , the Reynolds number is  $Re_c = 6.5 \times 10^6$ , and the angle of attack



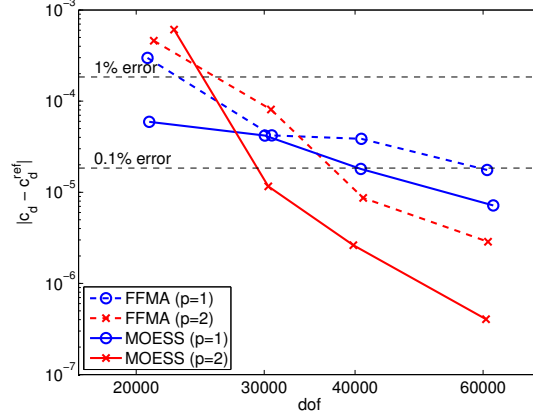


Figure 6-8: Drag error convergence for the RAE 2822 transonic RANS-SA problem.

is  $\alpha = 2.79^\circ$ . Each mesh consists of cubic ( $q = 3$ ) elements representing the geometry, and the farfield is 10,000 chord lengths away. The output of interest is the drag on the airfoil. This standard RANS test case requires accurate computation of the shock-boundary layer interaction and also exhibits multiple singular and singularly perturbed features.

Figure 6-8 shows the drag output convergence history. The reference solution is obtained using the adaptive  $p = 3$ ,  $\text{dof} = 250,000$  discretization. For the  $p = 1$  discretization, MOESS outperforms FFMA for all numbers of degrees of freedom considered. In particular, MOESS requires less than 20,000 degrees of freedom to achieve the drag error of less than 1 count. For the  $p = 2$  discretization, MOESS outperforms FFMA for  $\text{dof} \geq 30,000$ . At  $\text{dof} = 20,000$ , neither FFMA nor MOESS is capable of producing a well-resolved  $p = 2$  solution; however, once all relevant solution features are sufficiently resolved, the  $p = 2$  discretization becomes very effective. For either adaptation strategy, the error level at which the  $p = 2$  discretization becomes more efficient than the  $p = 1$  discretization is approximately 0.5 counts. For a higher-fidelity simulation requiring a tighter error tolerance, the  $p = 2$  discretization is clearly more effective.

The difference in the drag error convergence between MOESS and FFMA can be understood by comparing the meshes generated by the two adaptation strategies; the adapted meshes for the  $p = 2$ ,  $\text{dof} = 60,000$  discretizations are shown in Figure 6-9. In particular, recalling that the output error is a product of the primal residual and the dual error, we can compare the primal and the dual features targeted by the strategies. Both strategies target the boundary layers using highly anisotropic elements that have the aspect ratio approaching  $10^3$ . Similarly, the shock is resolved using anisotropic elements. The key differ-



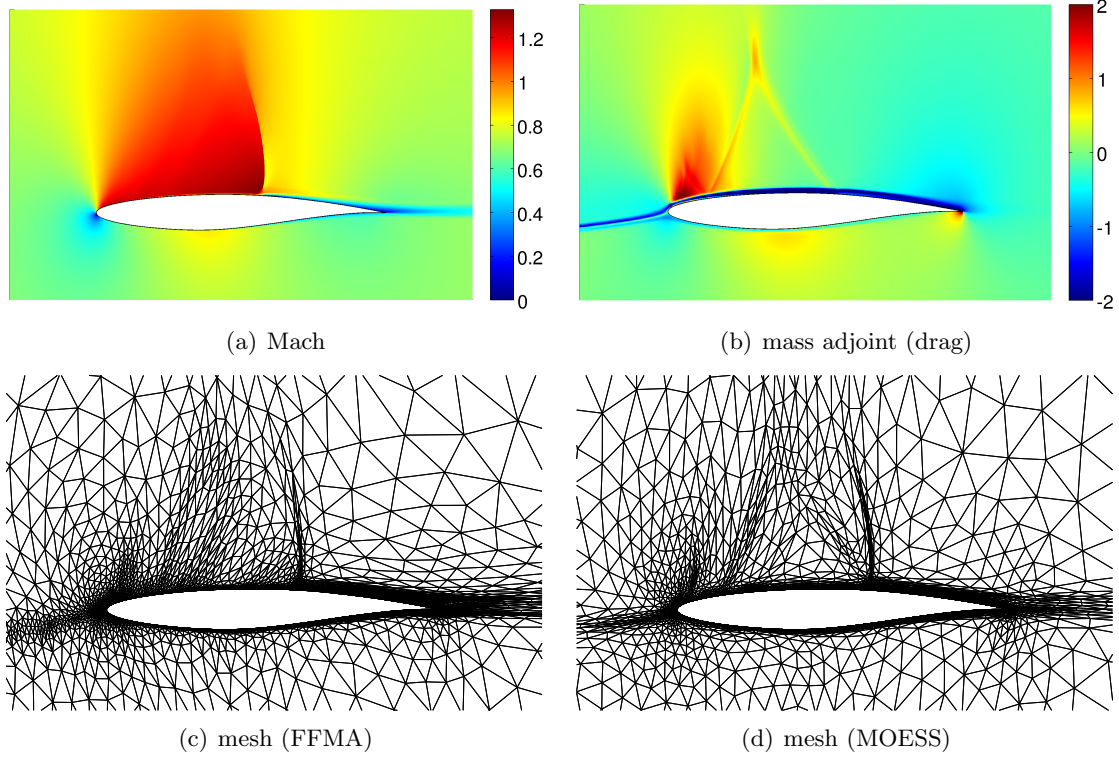


Figure 6-9: The Mach number, the mass adjoint, and the meshes for the RAE 2822 transonic RANS-SA problem. ( $p = 2$ ,  $\text{dof} = 60,000$ )

ence between the methods is the choice of elements used to resolve the stagnation streamline and adjoint features in the sonic pocket. Because the primal solution, and the Mach number in particular, does not exhibit anisotropic behavior along the stagnation streamline, Mach-anisotropy detection chooses isotropic elements along the stagnation streamline. However, the adjoint solution exhibits a wake-like feature along the stagnation streamline (of the primal solution), as shown in Figure 6-9(b). MOESS employs anisotropic elements to resolve this feature, as the local *a posteriori* error estimates automatically accounts for both the primal and adjoint solution behaviors.

The regularized  $c_p$  and  $c_f$  distributions computed using the  $p = 1$  and  $p = 2$  discretizations on  $\text{dof} = 30,000$ ,  $40,000$ , and  $60,000$  meshes obtained using MOESS are shown in Figure 6-10. The regularization of the surface quantity distributions is performed using the procedure described in Appendix C. The  $c_p$  distribution is essentially grid converged at  $\text{dof} = 30,000$  for both  $p = 1$  and  $p = 2$ , and the all curves essentially lie on top of each other. The regularized  $c_f$  distributions computed on the  $p = 1$  meshes exhibit some fluctuations, even at  $60,000$  degrees of freedom. The  $p = 2$ ,  $\text{dof} = 30,000$  result slightly



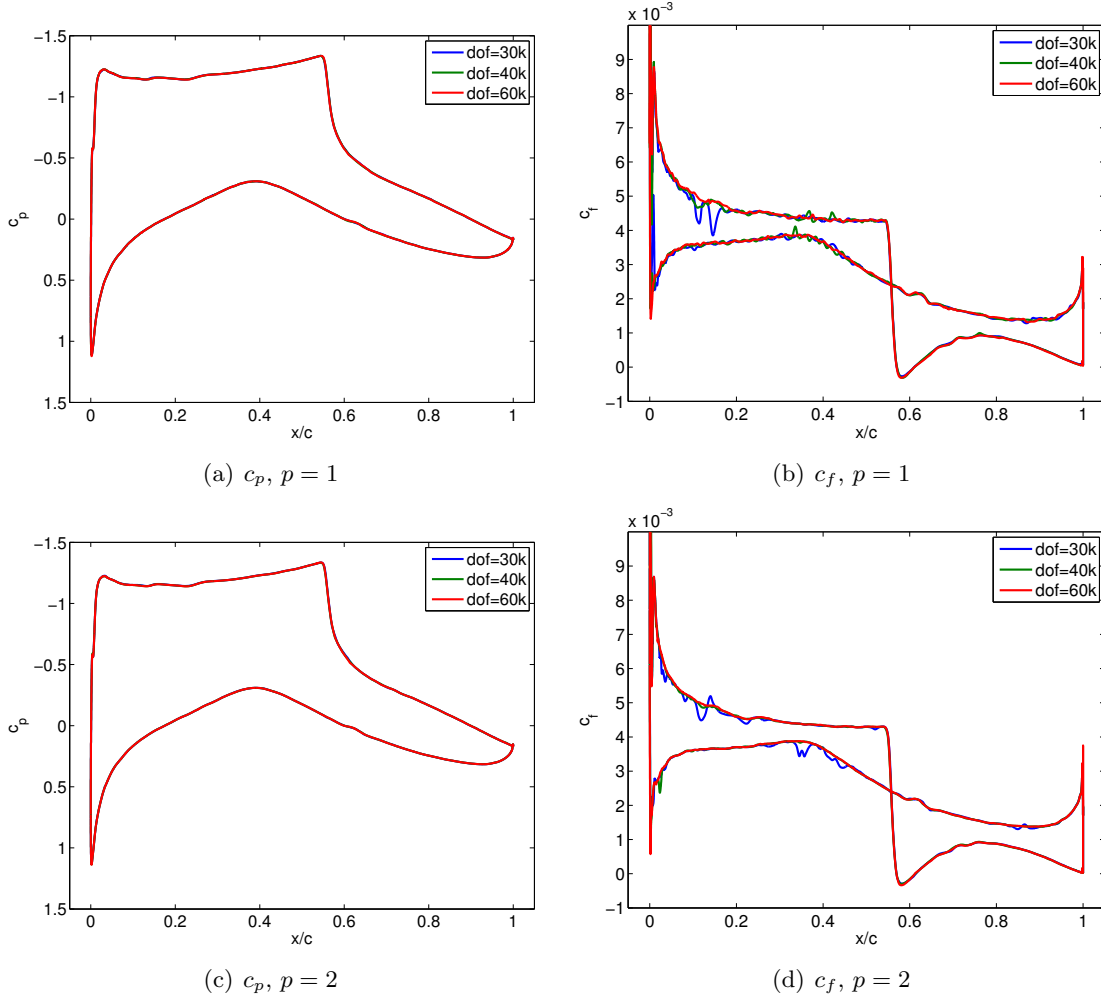


Figure 6-10: The regularized  $c_p$  and  $c_f$  distributions for the transonic RAE 2822 RANS-SA problem computed on  $p = 1$  and  $p = 2$  adapted meshes obtained using MOESS.

deviates from those obtained on the higher-dof meshes, especially in the leading edge region. The distribution computed on the  $p = 2$ , dof = 40,000 and dof = 60,000 meshes are indistinguishable for practical purposes.

### 6.3.4 NACA 0006 Euler Supersonic Shock Propagation

We consider a problem of predicting the sonic boom generated by supersonic flow over a NACA 0006 airfoil. The freestream Mach number is  $M_\infty = 2.0$ , and the airfoil is at  $0^\circ$  angle of attack. The Euler equations are solved using a  $p = 2$  DG discretization, and the mesh is adapted for the pressure output 50 chord lengths below the airfoil. In particular,



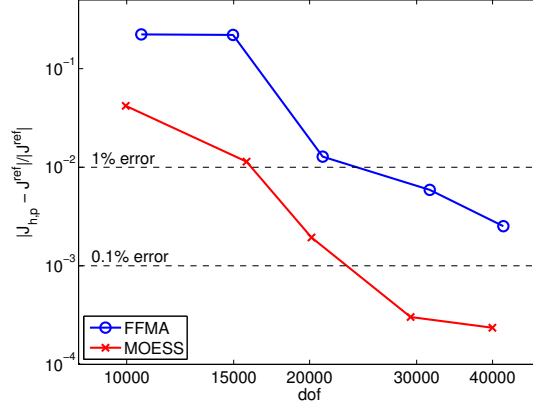


Figure 6-11: Pressure line output error convergence for the NACA 0006 Euler shock propagation problem ( $p = 2$ ).

the output functional is given by

$$\mathcal{J}(u) = \int_{\Gamma_{\text{line}}} (p(u) - p_{\infty})^2 ds,$$

where  $p(u)$  is the pressure,  $p_{\infty}$  is the freestream pressure, and  $\Gamma_{\text{line}}$  is the line along which the pressure perturbation is measured. Meshes consist of cubic elements, and the farfield is 200 chord lengths away. This problem tests the ability of the adaptive schemes to propagate singular features over a long distance.

Figure 6-11 shows the convergence of the pressure line integral error. The reference solution is computed on an adaptive  $p = 2$  discretization with 120,000 dof. MOESS shows approximately an order of magnitude improvement in the pressure line error compared to FFMA for the entire range of degrees of freedom considered.

To understand the difference in the pressure line errors, we compare the meshes obtained by MOESS and FFMA, shown in Figure 6-12. As expected, both meshes employ highly anisotropic elements to resolve the shock formed in front of the airfoil and the shock emanating from the trailing edge.

The FFMA method uses anisotropic elements in the flow direction behind the trailing shock, which does not seem to be appropriate for this flow. These elements are generated due to negative interaction between the solver and the adaptation algorithm. First, the numerical solution through the shock experiences  $\mathcal{O}(h)$  noise, producing an artificial variation in the flow quantities along the shock direction [19]. Second, this variation is convected downstream with little dissipation due to the use of the high-order method, creating stream-



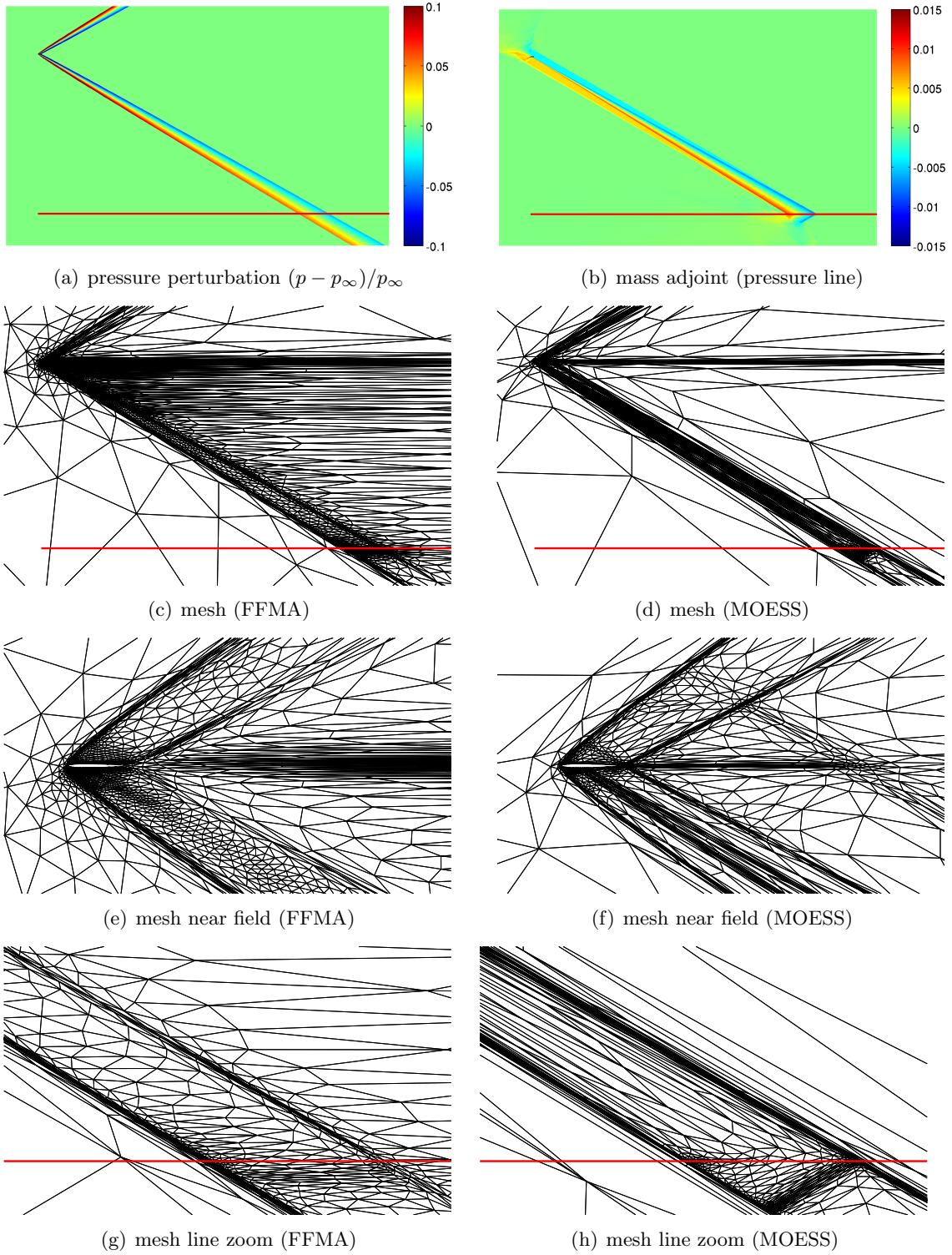


Figure 6-12: The pressure, the mass adjoint, and the meshes for the NACA 0006 Euler supersonic shock propagation problem. The pressure line is depicted in a red line. ( $p = 2$ ,  $\text{dof} = 40,000$ )



wise streaks. Third, the anisotropy detection based on the higher-order derivative of the Mach number captures these artificial streaks, requesting elements that are stretched in the stream-wise direction. Finally, the process worsens in the next adaptation iteration, as the stream-wise refinement of the elements results in generation of even smaller streaks. This case highlights the shortcomings of the anisotropy detection algorithm based on *a priori* convergence behavior of the solution, especially when a high-order discretization is applied to aerodynamic flows with low regularity.

Contrary to the FFMA method, MOESS produces large, low aspect ratio elements downstream of the second shock. The method clearly wastes no degrees of freedom in this region. Driven by the anisotropy in the adjoint solution, the method employs highly anisotropic elements aligned with the shock direction in the region between the leading and trailing shocks. The close up of the mesh near the airfoil shows that the method resolves complex adjoint features using anisotropic elements. Unlike the Mach-based anisotropy detection, the *a posteriori* error estimate based on local solves automatically captures the influence of the solution regularity to the local error. This in turn results in a more robust assessment of required anisotropies and generation of more efficient meshes, when a high-order discretization is applied to flows with limited regularity.

### 6.3.5 Multi-Element Supercritical 8 Transonic RANS-SA

We consider transonic turbulent flow over a multi-element supercritical airfoil (MSC8). The original geometry with sharp trailing edges, provided by Drela [53], is modified to have blunt trailing edges to facilitate adaptive meshing [105]. The freestream Mach number is  $M_\infty = 0.775$ , the Reynolds number is  $Re_c = 2 \times 10^7$ , and the angle of attack is  $\alpha = -0.7^\circ$ . The farfield is 200 chord lengths away. The output of interest is the combined drag on the two elements. The solution to the problem is shown in Figure 6-13. This flow exhibits complex interactions between the main element and the flap; accurately capturing the two shocks and their interaction with the wake and the stagnation streamline present challenges in this case.

#### The Initial Transition

Making the initial transition from an isotropic mesh, shown in Figure 6-14, to a mesh suitable for RANS calculation is particularly challenging for this problem. Note that the



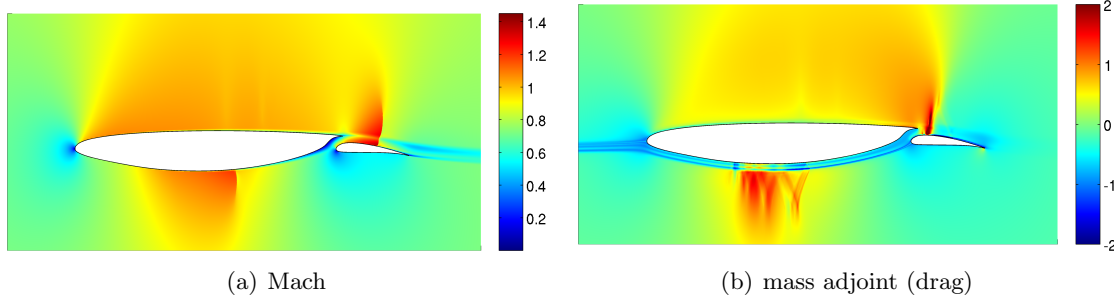


Figure 6-13: The Mach number distribution and the mass adjoint for the MSC8 transonic RANS-SA problem.

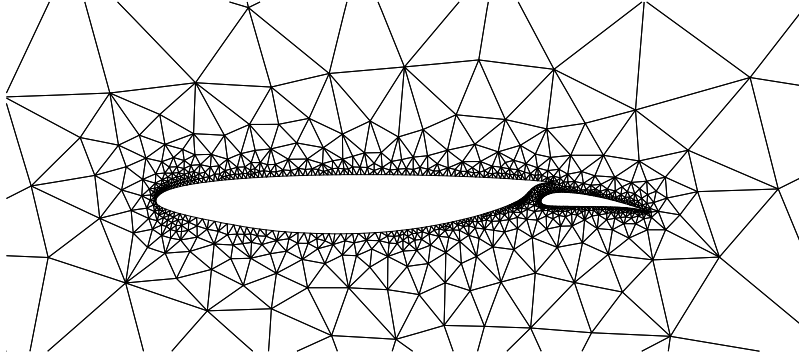


Figure 6-14: The initial mesh for the MSC8 transonic RANS-SA problem.

initial mesh has the first boundary spacing of  $y^+ \approx 10^4$ , making it unsuitable for RANS simulation. To illustrate the challenge, we consider the transition for the  $p = 1$  discretization using 40,000 degrees of freedom.

The drag convergence histories for the transition stage are shown in Figure 6-15. The reference solution is computed on an adapted  $p = 2$ ,  $\text{dof} = 250,000$  mesh. The figure shows that the drag computed using FFMA does not approach the reference value, even though MOESS shows that the  $p = 1$ ,  $\text{dof} = 40,000$  discretization is in fact sufficient to approximate the drag to within 2 counts. To understand the cause of the failed adaptation, let us study the fifth mesh generated by the FFMA algorithm, shown in Figure 6-16, which is representative of the other meshes generated by the adaptive scheme. Because the flow is supersonic over the upper surface of the airfoil, any small non-smooth perturbation from the underresolved boundary layer can induce a shock over the upper surface. Note also that the boundary layer for this high Reynolds number flow is completely underresolved at this early stage of adaptation, and the presence of the boundary layer cannot be detected through the variation in the Mach number. As the artificial shocks are observable while the



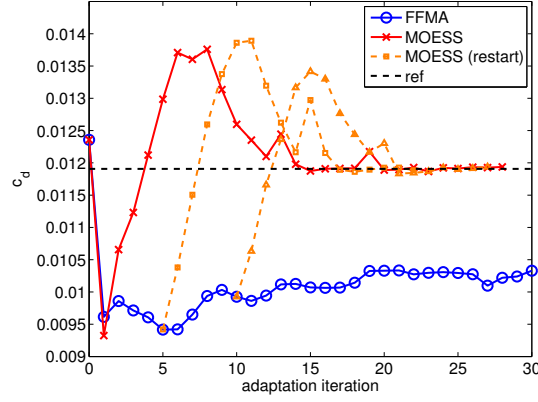


Figure 6-15: Drag adaptation histories for the  $p = 1$ ,  $\text{dof} = 40,000$  isotropic-to-RANS mesh transition test.

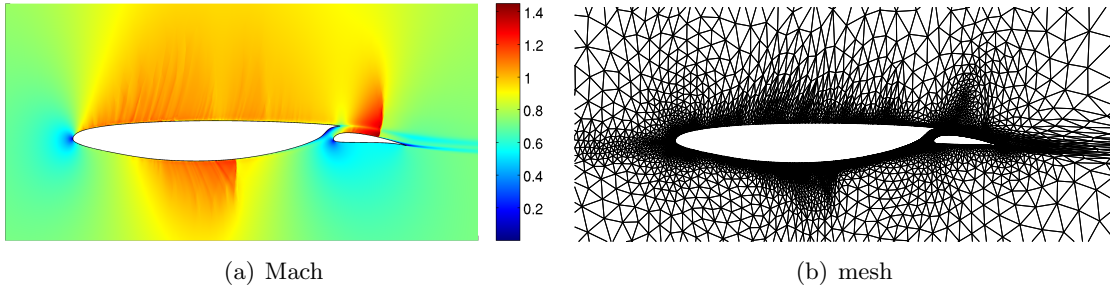


Figure 6-16: The Mach number distribution and the mesh for the fifth adaptation iteration starting from the isotropic mesh in Figure 6-14 using FFMA ( $p = 1$ ,  $\text{dof} = 40,000$ ).

boundary layer is not, the FFMA scheme attempts to resolve the features above the upper surface using anisotropic elements aligned with the artificial shocks — elements aligned in the direction perpendicular to the boundary layer. Due to the use of inappropriate anisotropy, FFMA is unable to detect and resolve the boundary layer, and the transition to a RANS mesh fails even after 30 adaptation iterations.

Figure 6-15 shows that MOESS makes a successful transition from the isotropic mesh to a RANS mesh, converging to the reference solution in about 15 adaptation iterations. The mesh obtained after five adaptation iterations is shown in Figure 6-17(a). Similar to the fifth FFMA-adapted mesh, MOESS also suffers from the presence of the artificial shocks on the suction side and uses shock-aligned anisotropic elements away from the boundary. However, right on the boundary, the method employs boundary-aligned anisotropic elements. With five more adaptation iterations, MOESS generates a RANS mesh shown in Figure 6-17(b). The boundary layer is resolved using highly anisotropic elements, the artificial shocks disappear, and the drag output rapidly approaches the reference value. To



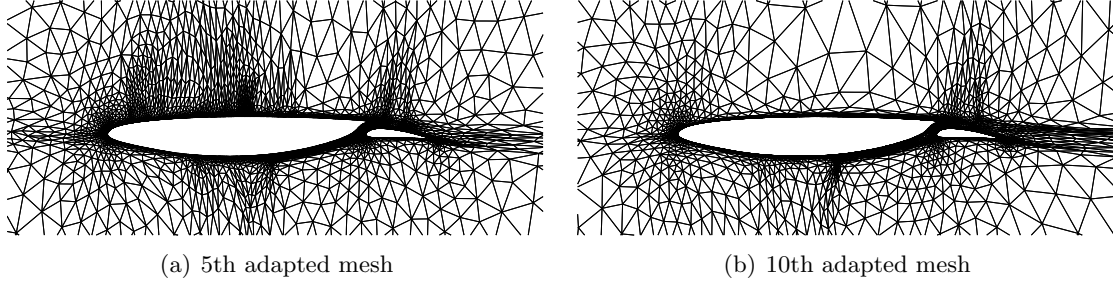


Figure 6-17: The adapted meshes starting from the isotropic mesh in Figure 6-14 using MOESS.

illustrate the reliability of this transition, the drag convergence histories starting from the fifth and tenth FFMA-adapted meshes are also shown in Figure 6-15. For both cases, the drag value approaches the reference value in about 15 adaptation iterations.

Our experience suggests that MOESS can infer the presence of an anisotropic feature through DWR-based local sampling even if the feature is significantly underresolved. In other words, even on a coarse mesh on which the  $p + 1$  solution reconstruction — and the subsequent  $p + 1$ -derivative-based anisotropy detection — is unreliable, the sampling-based anisotropy detection appears to behave correctly. Thus, MOESS is more robust than FFMA in the presence of underresolved features.

### Drag Error Convergence Results

Figure 6-18 shows the convergence of the drag error for FFMA and MOESS. As FFMA is incapable of making an isotropic-to-RANS mesh transition, the initial RANS mesh for FFMA is constructed by performing several FFMA adaptation iterations starting from a RANS mesh prepared using MOESS. For both the  $p = 1$  and  $p = 2$  discretizations, MOESS in general achieves lower error than FFMA for a given number of degrees of freedom. In particular, significant improvement is observed for the low-dof  $p = 2$  discretizations, making the  $p = 2$  discretization competitive against the  $p = 1$  discretization for simulations using as few as 40,000 degrees of freedom. Thus, MOESS is not only more robust in isotropic-to-RANS mesh transition but also more efficient than FFMA for this complex, multi-element, multi-shock problem.

Figure 6-19(a) shows the  $p = 2$ , dof = 120,000 meshes generated by the FFMA method. The mesh features highly anisotropic elements in the boundary layer regions and also in the shocks. The stagnation streamlines emanating from the main element and the flap elements



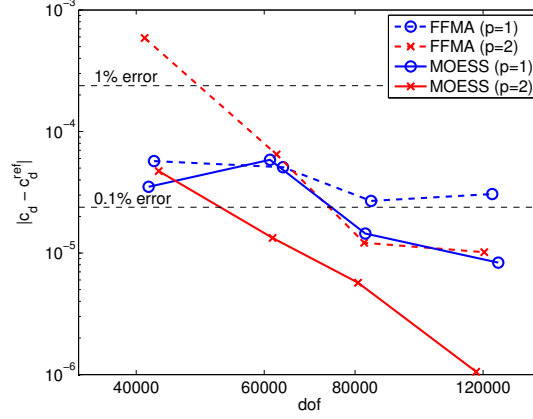


Figure 6-18: Drag error convergence for the MSC8 Transonic RANS-SA problem.

are both resolved using isotropic elements, resulting in a large number of elements employed to resolve these features.

Figure 6-19(b) shows the  $p = 2$ ,  $\text{dof} = 120,000$  mesh generated by MOESS. In addition to the boundary layers and the shocks, anisotropic elements are employed to resolve the stagnation streamlines. In particular, the effect of the shock on the lower surface of the main element is effectively propagated downstream toward the flap element. The efficient resolution of adjoint features appears to have a larger impact on the output quality for this complex multi-element airfoil case than in the isolated RAE 2822 case.

The regularized  $c_p$  and  $c_f$  distributions produced by the adaptive  $p = 1$  and  $p = 2$  discretizations using MOESS are shown in Figure 6-20. As in the transonic RAE 2822 case, the regularization of the surface quantity distributions is performed using the procedure described in Appendix C. The distributions show rapid variations in the force coefficients across the shocks. Similar to the force distributions for the RAE 2822 case, the  $c_p$  distribution converges quicker than the  $c_f$  distribution. All  $c_p$  distributions shown are essentially grid converged. On the other hand, the  $c_f$  distribution for the  $p = 1$  discretization is noisy even on the  $\text{dof} = 120,000$  mesh, which achieves the drag error of less than 0.1 counts. The  $p = 2$ ,  $\text{dof} = 60,000$  mesh, which achieves a similar  $c_d$  error, produces a much smoother  $c_f$  distribution, providing sufficient resolution for qualitative assessment of the surface quantity distribution for practical engineering purposes.



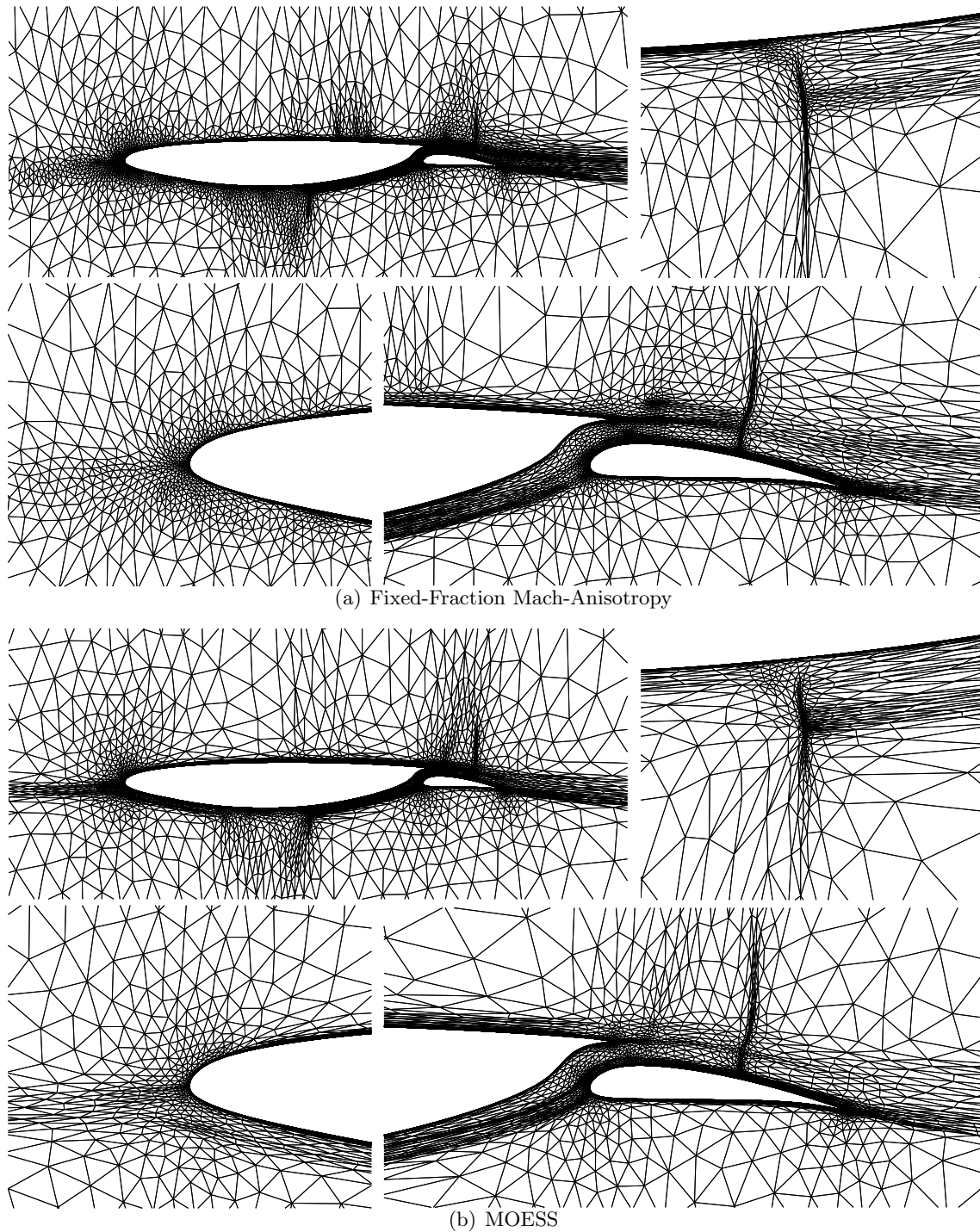


Figure 6-19: Drag-adapted meshes for the transonic MSC8 RANS-SA problem. For each subfigure: overview (top left); main-element shock (top right); main-element leading edge (bottom left); and flap-element (bottom right). ( $p = 2$ ,  $\text{dof} = 120,000$ )



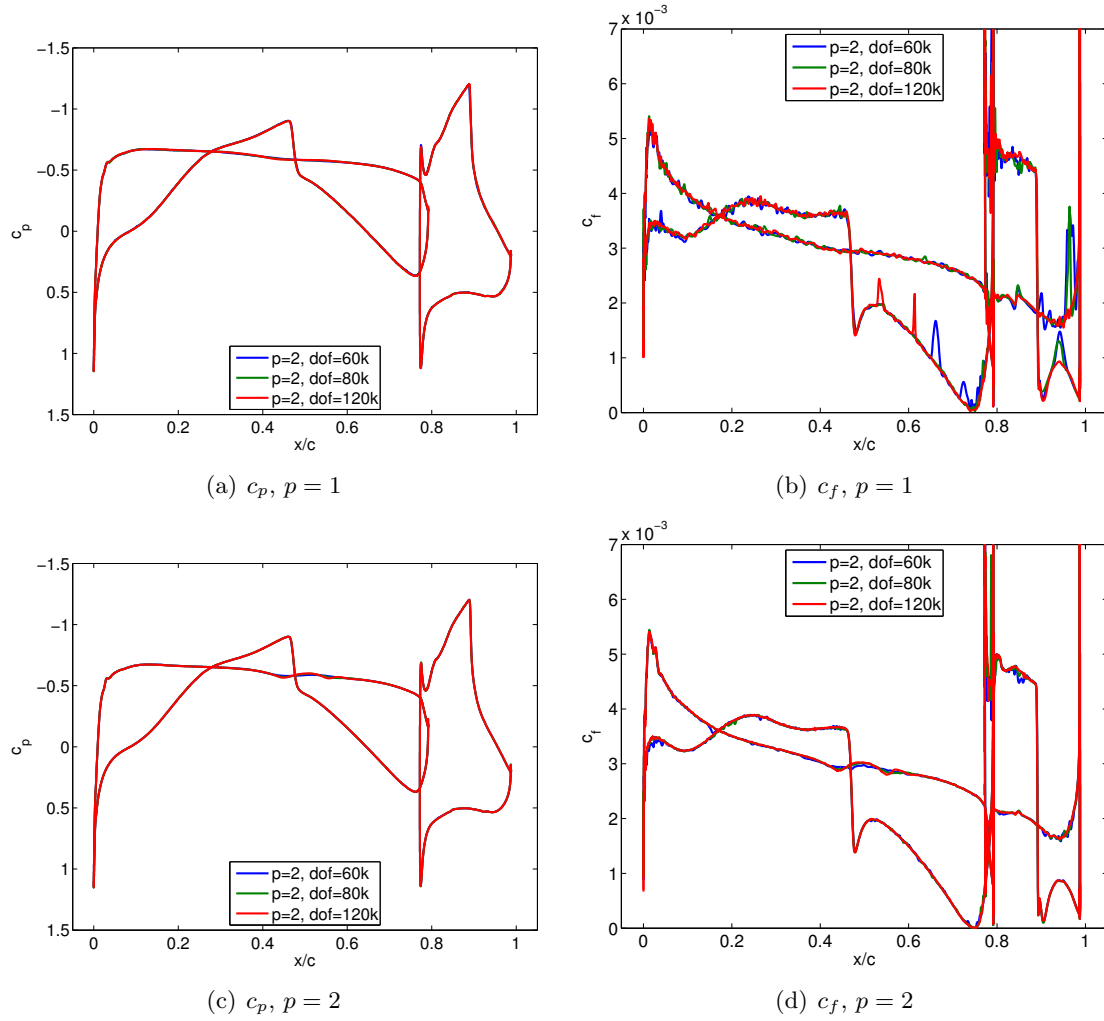


Figure 6-20: The  $c_p$  and  $c_f$  distributions for the transonic MSC8 RANS-SA problem computed on adapted meshes obtained using MOESS.



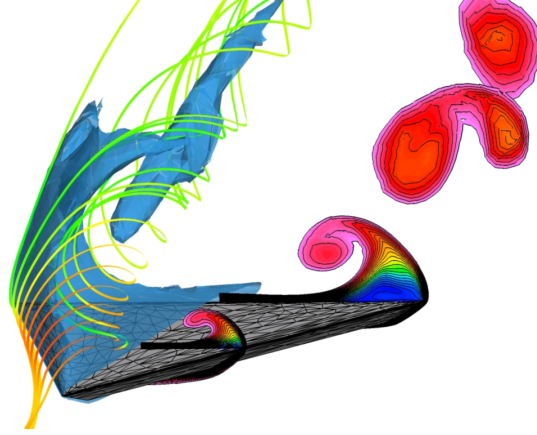


Figure 6-21: The Mach number isosurface, Mach number slices, and the streamlines for the delta wing case.

### 6.3.6 Laminar Flow over a Delta Wing

As the final case, we consider laminar flow over a delta wing at a high angle of attack, the case originally considered by Leicht and Hartmann [93] as a part of the ADIGMA project. The delta wing has a sharp leading edge and a blunt trailing edge. The freestream Mach number is  $M_\infty = 0.3$ , the angle of attack is  $\alpha = 12.5^\circ$ , and the Reynolds number based on the root chord is  $Re_{c_r} = 4000$ . The viscosity is assumed to be constant and the Prandtl number is set to 0.72. Isothermal no-slip boundary condition with the wall temperature equal to the freestream condition is imposed on the wing.

The Mach number distribution and streamlines of the flow around the delta wing are shown in Figure 6-21. The flow rolls up over the sharp leading edge and creates large vortices on the upper surface of the wing. Both the singularity along the leading edge and the smooth vortices on the upper surface must be captured to accurately compute the lift and drag on the wing. The reference values of the drag and lift coefficients computed for the ADIGMA project are  $C_D = 0.1658$  and  $C_L = 0.347$  [93].

Figure 6-22 shows the convergence of the drag coefficient using several different methods. The “HOW mesh” results are obtained on a series of “best-practice” meshes prepared by NLR for the 1st International Workshop on High-Order CFD Methods [152]. The drag values are taken from those reported by the University of Michigan group. The “L&H” corresponds to the results reported by Leicht and Hartmann in [93] using their hexahedron-based, anisotropic hierarchical subdivision adaptation. The “MOESS” results are generated by applying the MOESS algorithm starting from an initial mesh consists of only 26 elements.



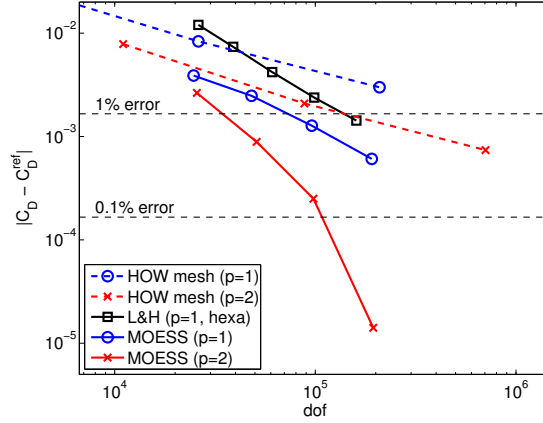


Figure 6-22: Drag error convergence for the laminar delta wing case. “HOW mesh” are the high-order workshop meshes prepared by NLR for the High-Order Workshop [152]. “L&H” is the result reported by Leicht and Hartmann using their hexahedron-based hierarchical subdivision strategy [93].

In fact, as shown in Figure 6-23(a), the initial mesh uses a single face of a tetrahedron to cover the entire upper surface of the delta wing.

Due to the presence of multiple geometry-induced singularities, both the  $p = 1$  and  $p = 2$  discretizations achieve the same low convergence on *a priori* generated HOW meshes. In particular, the benefit of higher-order discretization is not realized on this family of meshes. MOESS significantly improves the quality of the drag prediction. For the  $p = 1$  discretization, MOESS produces a family of meshes that are more efficient than the meshes generated *a priori* or those generated through hexahedron-based, anisotropic hierarchical subdivision. Furthermore, MOESS significantly improves the performance of the  $p = 2$  discretization, reducing the number of degrees of freedom required to achieve 10 drag counts of error by over an order of magnitude. For a higher-fidelity simulation requiring a tighter error tolerance, the adaptation achieves more drastic improvement in the error-to-dof efficiency. The higher-efficiency is achieved through aggressive mesh refinement toward the geometric singularities, as evident from the meshes shown in Figure 6-23(b).

### 6.3.7 Computational Cost

Let us make a few remarks regarding the computational cost of the adaptation process. To analyze the computational cost, we decompose the time for a single adaptation cycle into:

**Primal solve:** the time for solving the primal equation (i.e. the flow equation)



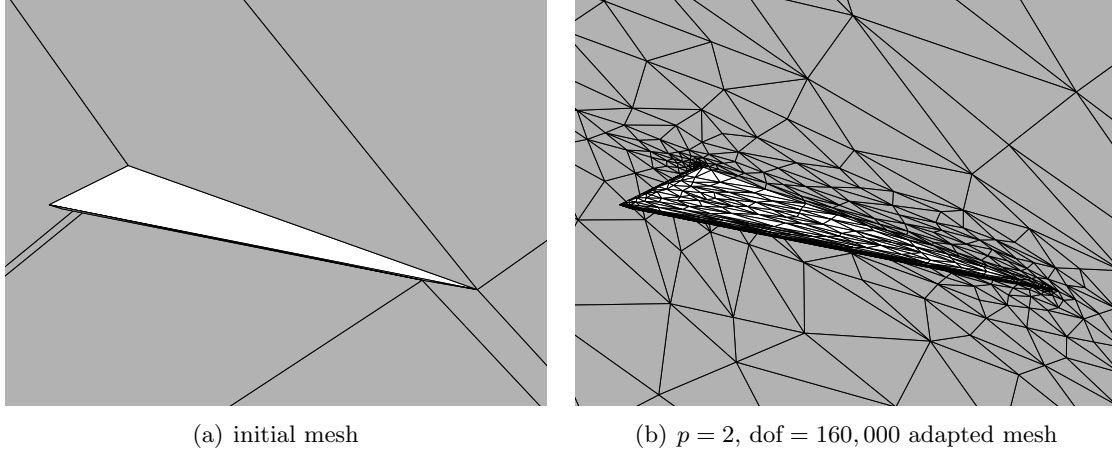


Figure 6-23: The 26-element initial mesh and the  $p = 2$ , dof = 160,000 adapted mesh. The symmetry plane is shown in gray.

**Dual solve:** the time to obtain the  $p + 1$  degree surrogate solution to the dual problem

**Adapt (FFMA):** the time to perform 10 Newton iterations of  $p + 1$  degree primal solve to construct an approximate  $p + 1$  derivative

**Adapt (MOESS):** the time to sample local errors, synthesize the errors, and optimize the surrogate error model

In the context of output error control, “primal solve” is the cost of computing the output, “dual solve” is the cost of endowing the output with an error estimate, and “adapt” is the cost of controlling and improving the output error in the next solve. An effective adaptation algorithm must keep the cost of error estimation and control a fraction of the flow solve.

The first row of Table 6.1 shows a timing breakdown for the NACA 0006 Euler shock propagation problem considered in Section 6.3.4. Both FFMA and MOESS use the same  $p = 2$ , dof = 20,000 mesh. For this case with a small number of degrees of freedom, computing the dual surrogate solution and constructing an error estimate requires 17% of the flow solve time. For FFMA, the additional cost of constructing a  $p + 1$  derivative approximation is 50% of the flow solve. For MOESS, the additional adaptation cost is 43% of the flow solve, the majority of which stems from the local solves. For both FFMA and MOESS, the additional cost for error estimation and control is a fraction of flow solve, even for this relatively small case. Moreover, MOESS is not only more efficient than FFMA in terms of error-per-dof (as shown in Section 6.3.4) but also slightly faster in terms of timing-per-dof.



Case	Primal ( $p$ )	Dual ( $p + 1$ )	FFMA Adapt	MOESS Adapt
NACA 0006 Euler shock ( $p = 2$ , dof = 20000)	1.000	0.174	0.495	0.431
RAE 2822 transonic RANS ( $p = 2$ , dof = 60000)	1.000	0.092	0.157	0.072

Table 6.1: Timing breakdown for a single adaptation cycle normalized by the primal solve time.

As the second example, we consider a more complex flow: the RAE 2822 transonic RANS case considered in Section 6.3.3. The second row of Table 6.1 shows the timing breakdown on a  $p = 2$ , dof = 60,000 mesh. Due to the increased complexity of the problem, the nonlinear primal problem is harder to converge, and the relative cost of solving the dual problem, which is inherently linear, decreases to about 9% of the flow solve. Moreover, the local sampling cost for MOESS decreases to 7% of the flow solve cost. This decrease is attributed to two factors. First, even though RANS equations are highly nonlinear, each element-wise localized problem can still be solved in a few Newton iterations for most of the cases. Second, the time for the local solves scales linearly with the number of elements, whereas the cost of the global linear solve scales superlinearly. As a result, the relative cost of the adaptation stage decreases with the problem complexity. We also note that the anisotropy detection by  $p + 1$  Mach derivative reconstruction requires 16% of the flow solve time, compared to the 7% of MOESS. Again, MOESS is not only more accurate (as shown in Section 6.3.3) but also faster than FFMA for a given number of degrees of freedom.

## 6.4 Conclusions

This chapter first quantified the importance of mesh adaptation for higher-order discretizations of aerodynamic flows. The numerical experiments demonstrated that uniform refinement is insufficient to attain the benefits of higher-order discretizations even starting from optimized coarse meshes. Strong mesh grading toward singular features are required to control the effect of the singularities, some of which are hard to identify *a priori* for complex aerodynamic flows.

The second half of the chapter compared the performance of MOESS against state-of-the-art adaptation algorithms. For a wide range of aerodynamic flows considered, MOESS was at least as competitive as the method based on fixed-fraction marking and Mach-based



anisotropy detection, and in some cases produced over an order of magnitude improvement in the output error for a given number of degrees of freedom. In particular, as the method stems from the first principle of output error minimization and is guided by the *a posteriori* error behavior, it does not suffer from degradation of the performance when the flow includes features that violate *a priori* assumptions of the error behavior, e.g. the approximation error is not governed by the  $p + 1$  derivatives of the solution because the solution is underresolved or singular. Moreover, numerical results demonstrated that, with proper mesh selection, higher-order methods are more efficient than lower-order methods for high-fidelity simulations. In terms of computational cost, the time spent on error estimation and adaptation is a fraction of the flow solve time, and the relative cost decreases for complex problems requiring a larger number of degrees of freedom.



## Chapter 7

# Fully-Unstructured Space-Time Adaptivity for Wave Propagation Problems

### 7.1 Introduction

This chapter considers a unified space-time formulation of conservation laws and application of our anisotropic adaptation algorithm, Mesh Optimization via Error Sampling and Synthesis (MOESS), to space-time adaptivity. In particular, we consider fully-unstructured space-time adaptivity for wave propagation problems governed by the wave equation and the time-dependent Euler equations using the combination of the discontinuous Galerkin (DG) method, the dual-weighted residual (DWR) error estimation, and MOESS. We then assess the competitiveness of the space-time formulation through numerical experiments.

The idea of using a variational discretization in both spatial- and temporal-space has been explored previously. In particular, Johnson [86] proposed the use of the discontinuous Galerkin (DG) method for temporal integration, resulting in a class of variational integrators that is suited for stiff ODEs and facilitates *a posteriori* error estimation and control. The DG temporal integrator has been combined with the continuous Galerkin (CG) spatial discretization for parabolic problems [54] and second-order hyperbolic problems [79, 87]. By casting the space-time problem within the variational framework, this so-called CG-DG formulation (spatially-continuous, temporally discontinuous) allows rigorous *a priori* and *a*



*posteriori* error analysis using general variational techniques, offers flexibility of using unstructured space-time meshes, and permits discretization of complex or time-varying spatial domains. Later, Bangerth and Rannacher combined the second-order accurate CG-DG finite element discretization of the wave equation with the dual-weighted residual (DWR) error estimate and performed adaptation on tensor-product space-time meshes [15]. A recent review of the CG-DG approach for the wave equation is provided in [14].

Due to the success of the DG discretization for steady-state conservation laws, a number of researchers have considered the use of the DG discretization in space-time [17, 59, 142]. In particular, Hartmann combined the space-time DG formulation with the DWR error estimate to perform output-based space-time adaptation on tensor-product space-time meshes for the one-dimensional Burgers [69] and the Euler [70] equations; Süli and Houston also considered the tensor-product space-time formulation for the one-dimensional wave equation [138]. Again, the rigor and flexibility of the variational formulation are thought to warrant the higher computational cost of the DG time integrator compared to, say, an implicit Runge-Kutta integrator.

It is also worth noting that (spatially) anisotropic adaptation has been successfully carried out for unsteady problems on complex three-dimensional domains using simplex meshes. In particular, Pain *et al.* [113] have considered anisotropic adaptivity for unsteady incompressible fluid flows, and Alauzet *et al.* have combined their time-slab based anisotropic mesh adaptation scheme with interpolation-based [4] and output-based [5] error estimate to simulate shock wave propagation over complex domains.

In this work, we consider fully-unstructured space-time adaptivity for wave propagation problems. In particular, we abandon the idea of “time-slab” used almost universally to solve unsteady problems using time-marching techniques. Instead, we recast a time-dependent problem in the  $d$ -dimensional space as a “steady-state” hyperbolic problem in the  $d + 1$  dimensional space-time space. Specifically, the wave equation is recast as a system of  $d + 1$  first-order hyperbolic equations, and the Euler equation is recast in the  $d + 1$  dimensional space with the “temporal flux” in the first dimension. Such a unified space-time formulation permits space-time meshes with arbitrarily-oriented anisotropic space-time simplex elements, which in principle can efficiently track the characteristic waves in the space-time plane. We realize this fully-unstructured space-time adaptivity by applying the MOESS algorithm to wave propagation problems recast as a “steady-state” problem over the space-



time domain. While the fully-unstructured space-time formulation requires solution of a fully-coupled space-time system, numerical results demonstrate that an effective use of space-time anisotropy can effectively reduce the dimensionality of the problem, warranting the use of such a unified space-time formulation.

## 7.2 The Wave Equation and Discretization

Consider the wave equation

$$\rho \frac{\partial^2 \phi}{\partial t^2} - \nabla_x \cdot (\kappa \nabla_x \phi) = 0 \quad \text{in } \Omega \times I,$$

with the homogeneous Neumann boundary condition

$$\hat{n} \cdot \kappa \nabla_x \phi = 0 \quad \text{on } \partial\Omega \times I$$

and initial conditions

$$\begin{aligned} \phi &= \phi_0^1 \\ \frac{\partial \phi}{\partial t} &= \phi_0^2 \quad \text{on } \Omega \times [0]. \end{aligned}$$

Here,  $\Omega \subset \mathbb{R}^d$  is the spatial domain,  $\partial\Omega$  is the spatial boundary, and  $I \in (0, T]$  is the time interval. The equation is characterized by a density  $\rho(x) \in \mathbb{R}$  and a symmetric positive-definite stiffness matrix  $\kappa(x) \in \mathbb{R}^{d \times d}$ . The initial condition is specified by  $\phi_0^1$  and  $\phi_0^2$ . The subscript  $x$  on  $\nabla_x$  signifies that the derivative is taken with respect to the spatial dimension.

To apply a DG discretization to the wave equation, we reformulate the equation as a system of hyperbolic conservation laws, i.e.

$$\begin{aligned} \frac{\partial}{\partial t}(\rho w) - \nabla_x \cdot (\kappa q) &= 0 \\ \frac{\partial q}{\partial t} - \nabla_x w &= 0 \quad \text{in } \Omega \times I, \end{aligned}$$

where  $w(x) \in \mathbb{R}$ , and  $q(x) \in \mathbb{R}^d$  is an auxiliary variable. The homogeneous Neumann



boundary condition becomes

$$\hat{n} \cdot \kappa q = 0 \quad \text{on} \quad \partial\Omega \times I,$$

and the initial conditions are

$$\begin{aligned} w &= w_0 \\ q &= q_0 \quad \text{on} \quad \Omega \times [0]. \end{aligned}$$

We will denote our state by  $u$ , i.e.  $u = (w, q^T)^T$ . This hyperbolic form arises naturally in, for example, acoustics, in which  $w$  is the pressure and  $q$  is the particle velocity, or in shallow water modeling, in which  $w$  is the perturbed height and  $q$  is the fluid velocity.

In order to realize fully-unstructured space-time adaptivity in a straightforward manner, we will reinterpret the time-dependent hyperbolic conservation law as a “steady-state” conservation law by treating the temporal dimension as the 0-th dimension. The conservation law can be concisely written as

$$\sum_{i=0}^d \frac{\partial}{\partial x_i} \mathcal{F}_i^{\text{conv}}(u, x) = 0 \quad \text{in} \quad \Omega \times I,$$

where

$$\mathcal{F}_0^{\text{conv}}(u, x) = \begin{pmatrix} \rho w \\ q \end{pmatrix} \quad \text{and} \quad \mathcal{F}_i^{\text{conv}}(u, x) = - \begin{pmatrix} \kappa_{ij} q_j \\ \hat{e}_i w \end{pmatrix} \quad i = 1, \dots, d,$$

and  $\hat{e}_i \in \mathbb{R}^d$  is a unit vector with 1 in the  $i$ -th entry and 0 elsewhere. Appropriate “boundary” conditions, which now includes the initial conditions, are imposed on the space-time boundaries.

### 7.3 Energy Error Estimate

This section develops an energy error estimate that is suitable for capturing the entire wave behavior as oppose to a specific output quantity. The formulation closely follows that recently used by Fidkowski and Roe [60] to develop an “entropy” error estimate in the context of compressible flows.



Because the wave equation written in the hyperbolic form is symmetrizable, it possesses an energy pair. Let us define the energy as

$$\mathfrak{U} = \frac{1}{2}\rho w^2 + \frac{1}{2}q^T \kappa q,$$

with the associated energy flux

$$\mathfrak{F}_i = -\kappa_{ij} w q_j.$$

The energy pair satisfies

$$\frac{\partial \mathfrak{U}}{\partial t} + \nabla_x \cdot \mathfrak{F} = 0.$$

Integrating the energy equation over  $\Omega \times (0, t]$  for some  $t \in I$ , we observe

$$\int_0^t \int_{\Omega} \left[ \frac{\partial \mathfrak{U}}{\partial t} + \nabla_x \cdot \mathfrak{F} \right] dx dt = \left[ \int_{\Omega} \mathfrak{U} dx \right]_{t=0}^t + \int_0^t \int_{\partial\Omega} \hat{n} \cdot \mathfrak{F} dx dt \stackrel{0}{=} 0,$$

where the boundary flux vanishes as  $\hat{n} \cdot \mathfrak{F} = -n_i \kappa_{ij} w q_j = 0$  on  $\partial\Omega$  due to the homogeneous Neumann boundary condition. In other words, the energy is conserved and

$$\int_{\Omega} \mathfrak{U} dx \Big|_{t=t'} = \int_{\Omega} \mathfrak{U} dx \Big|_{t=0}, \quad \forall t' \in I.$$

By defining the output of interest as

$$J^E = \mathcal{J}^E(u) = \int_{\Omega} \mathfrak{U}(u) dx \Big|_{t=T} = \int_{\Omega} \left( \frac{1}{2}\rho w^2 + \frac{1}{2}q^T \kappa q \right) dx \Big|_{t=T},$$

we can effectively identify the regions of spurious energy generation or dissipation.

Moreover, note that the adjoint solution  $(\psi^w, \psi^q)$  to this output is governed by

$$\begin{aligned} \rho \frac{\partial \psi^w}{\partial t} - \nabla_x \cdot \psi^q &= 0 \\ \frac{\partial \psi_i^q}{\partial t} - \kappa^T \nabla_x \psi^w &= 0 \quad \text{in } \Omega \times I \end{aligned}$$



with the boundary condition  $\hat{n} \cdot \psi^q = 0$  and the terminal conditions

$$\psi^w = w \quad \text{and} \quad \psi^q = \kappa q \quad \text{on} \quad \Omega \times [T].$$

In particular, the variables  $(\psi^w, \tilde{\psi}^q)$  with  $\tilde{\psi}^q = \kappa^{-1}\psi^q$  is identical to  $(w, q)$  as they are governed by the identical set of equations. Thus, the adjoint  $(\psi^w, \psi^q)$  can be computed from the primal variables using simple algebraic relations,  $\psi^w = w$  and  $\psi^q = \kappa q$ . This error indicator, which targets the regions of spurious energy generation, does not require a backward adjoint solve, which is required for a general output. In the context of mesh adaptation, the error indicator is useful for obtaining a “well-rounded” mesh that evenly resolves all solution features.

## 7.4 Results: The Wave Equation

### 7.4.1 Assessment Procedure

To assess the performance of space-time adaptive schemes for the wave equation, we consider permutations of two error estimators (energy or output) and two adaptation mechanics (isotropic or anisotropic). While all combinations of error estimates and adaptation mechanics are implemented using the unified space-time formulation, each combination is aimed to represent performance of different adaptation strategies used in practice. Namely:

- **Uniform Refinement:** Corresponds to solving the wave equation using a fixed mesh with a fixed time stepping.
- **Isotropic Energy-Based Adaptation:** Corresponds to solving the wave equation using an adaptive Rothe method — a traditional time-slab based time-marching solver that incorporates different spatial mesh in each time slab — with an energy-based error indicator. The main advantage of this method is that it does not require an adjoint calculation. Thus, adaptation could be performed in a single-pass time-dependent solve (assuming the target local error level is fixed *a priori*). In the space-time plane, this method produce isotropic elements as it does not permit element faces that cuts through space-time.
- **Isotropic Output-Based Adaptation:** Corresponds to solving the wave equation



using an adaptive Rothe method with an adjoint-based error indicator. The strategy requires multiple time-dependent primal and adjoint solves. An efficient implementation for a large-scale (nonlinear) problem requires a well-designed checkpointing scheme. As in the energy-based Rothe method, isotropic elements are produced in the space-time plane as the element faces must align with the axes. Bangerth *et al.* have recently performed a comparison of energy-based and output-based error estimates for an adaptive Rothe method [14].

- **Anisotropic Energy-Based Adaptation:** To our knowledge, this strategy has not been used in practice. The method requires a fully-unstructured  $(d+1)$ -dimensional space-time mesh. It is useful for generating a “well-rounded” mesh for resolving all features of the wave.
- **Anisotropic Output-Based Adaptation:** This strategy also has not been used in practice. The method requires a fully-unstructured  $(d+1)$ -dimensional space-time mesh. The additional burden of performing the adjoint solve in the fully-unstructured context is considerably smaller than in a time-marching scheme, as the space-time anisotropic algorithms do not take advantage of hyperbolicity of the equation in the temporal dimension to start with.

Let us now consider two wave propagation problems. The first one is a verification case in one spatial dimension. The second problem is a demonstration case in two spatial dimensions. For each problem, we measure the energy and output errors against the total number of space-time degrees of freedom. Note that the total space-time degrees of freedom may not be representative of the computational cost, as solving a fully-unstructured (i.e. fully-coupled) space-time problem with  $N$  degrees of freedom is arguably more expensive than solving the problem using a time-marching formulation with  $n_{\text{step}}$  time steps, each step containing  $N/n_{\text{step}}$  degrees of freedom. This is particularly true in the absence of an efficient preconditioner for the fully-unstructured space-time formulation. Thus, the following error-to-dof results should not be interpreted as an absolute comparison of computational efficiencies, but merely as one way of comparing the methods.



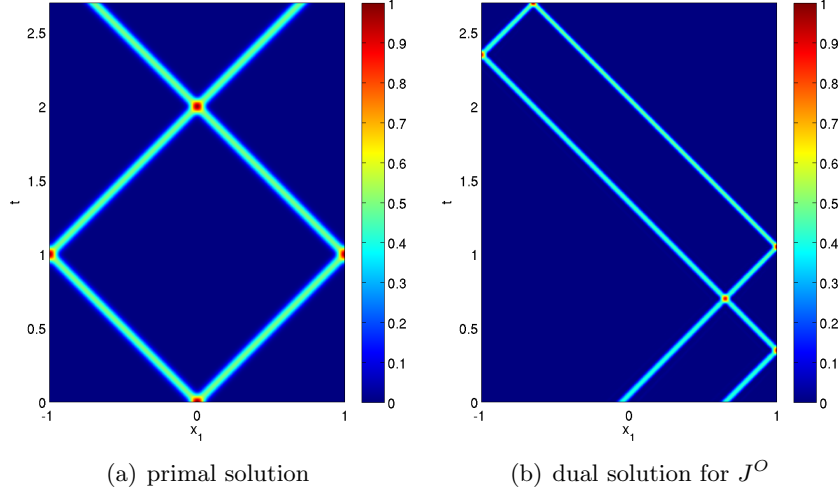


Figure 7-1: The first component of the primal and dual solutions to the 1+1d wave problem.

### 7.4.2 1+1d Wave Propagation

We consider a 1+1d wave propagation problem similar to the one considered by Bangerth and Rannacher [15]. The computational domain is  $\Omega \times I = [-1, 1] \times (0, 2.7]$ , and the initial condition is given by

$$w(x, 0) = \exp\left(-\left(\frac{x}{s}\right)^2\right) \quad \text{and} \quad q(x, 0) = 0, \quad (7.1)$$

with a characteristic length  $s = 0.05$ . Homogeneous Neumann boundary condition is imposed everywhere. The output of interest is a local solution value at the final time, represented as a Gaussian weighted integral, i.e.

$$J^O = \mathcal{J}^O(u) = \int_{\Omega} g(x)w(x, T)dx \quad \text{with} \quad g(x) = \exp\left(-\left(\frac{x + 0.65}{0.025}\right)^2\right).$$

The output captures a part of one of the branches of the wave. The primal and dual solutions to the problem are shown in Figure 7-1.

Figure 7-2 shows the convergence of the energy error and the output error using the  $p = 2$  DG discretization with five adaptive strategies discussed in Section 7.4.1. The reference energy and output values are obtained by solving the wave equation using the method of characteristics and by evaluating simple one-dimensional integrals at the final time.

Figure 7-2(a) shows that the energy-based adaptation performs significantly better than



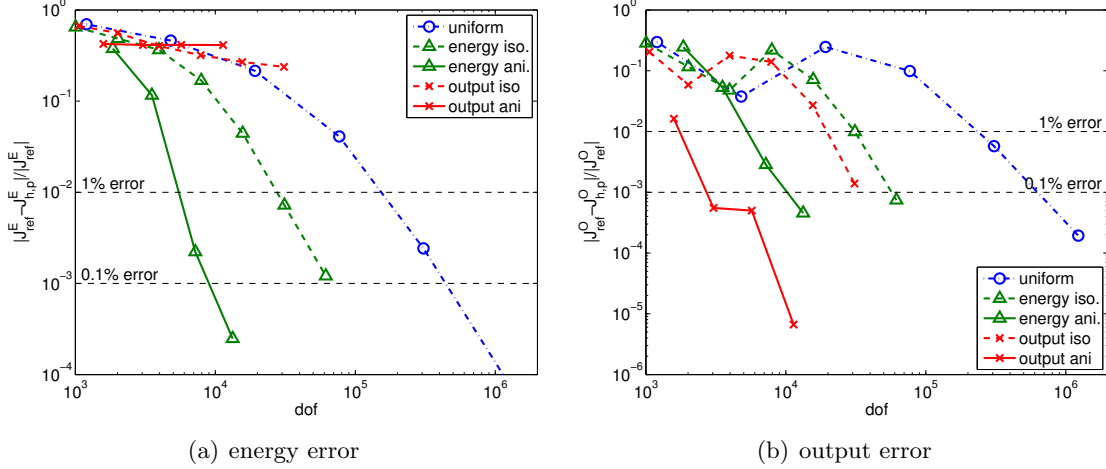


Figure 7-2: Energy and output error convergence for the 1+1d wave problem. ( $p = 2$ )

uniform refinement in controlling the energy error. Using the energy-based isotropic refinement reduces the degrees of freedom required to achieve the fractional error level of  $10^{-3}$  by approximately a factor of eight compared to uniform refinement. Allowing fully-unstructured space-time anisotropy further reduces the degrees of freedom required to achieve the error level by another factor of eight compared to the isotropic adaptation. The meshes in Figure 7-3 shows that energy-based adaptation targets both branches of the wave in space-time. With fully-unstructured space-time anisotropy, the algorithm convects the waves efficiently using a very few space-time elements that align with the direction of a constant phase (for example from  $t = 0.1$  to  $t = 0.9$  or from  $t = 1.1$  to  $t = 1.9$ ). Thus, the anisotropic adaptation effectively reduces the dimensionality of the problem from two to one. When the two waves with a different wave phase interfere with each other (including the boundary reflection), the isotropic elements are employed because there is no dominant characteristic direction.

Figure 7-2(b) shows the convergence of the output error. Again, the adaptive strategies require significantly fewer degrees of freedom than uniform refinement. Moreover, both anisotropic adaptation strategies outperform the isotropic strategies. This, from the interpretation provided in Section 7.4.1, means that fully-unstructured space-time mesh is significantly more effective for this class of problem than adaptive Rothe methods. The output-adapted meshes shown in Figure 7-3 targets only one branch of the wave that is relevant to evaluating the output. The combination of the output-based error estimate and anisotropic adaptation reduces the degrees of freedom required to achieve the error level of



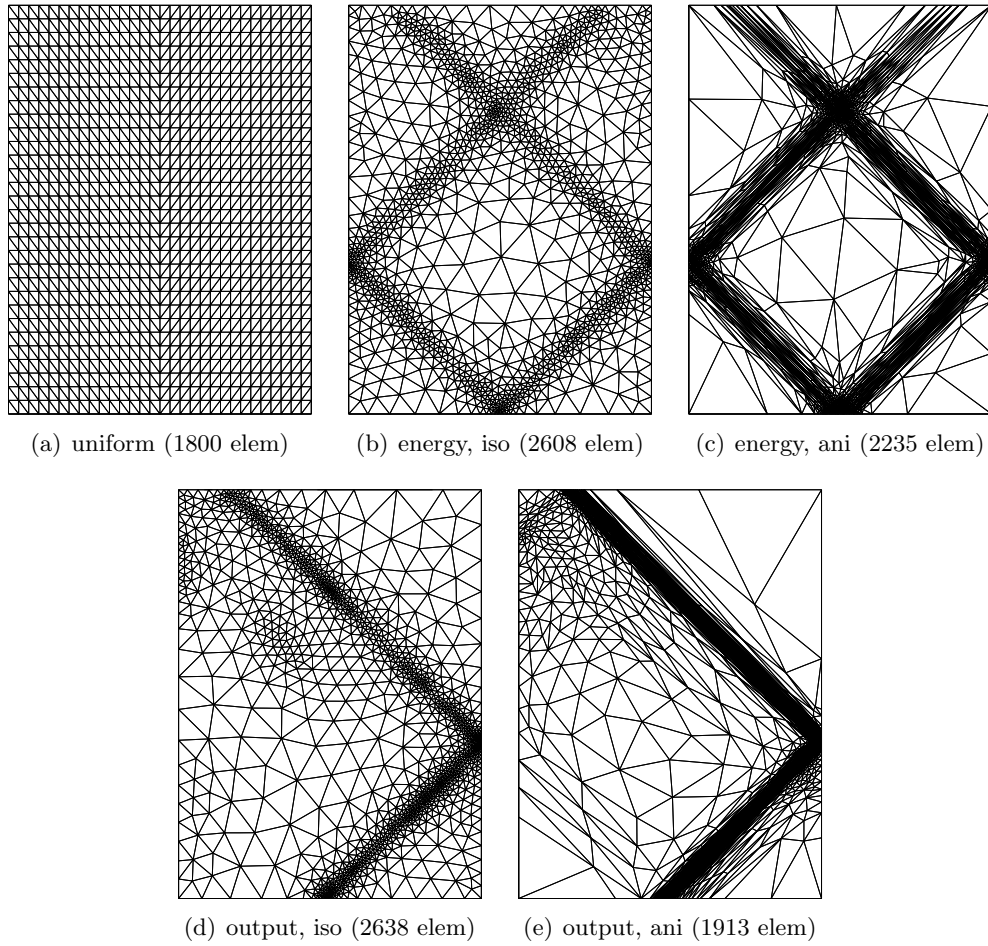


Figure 7-3: Adapted meshes for the  $1+1d$  wave problem. ( $p = 2$ )



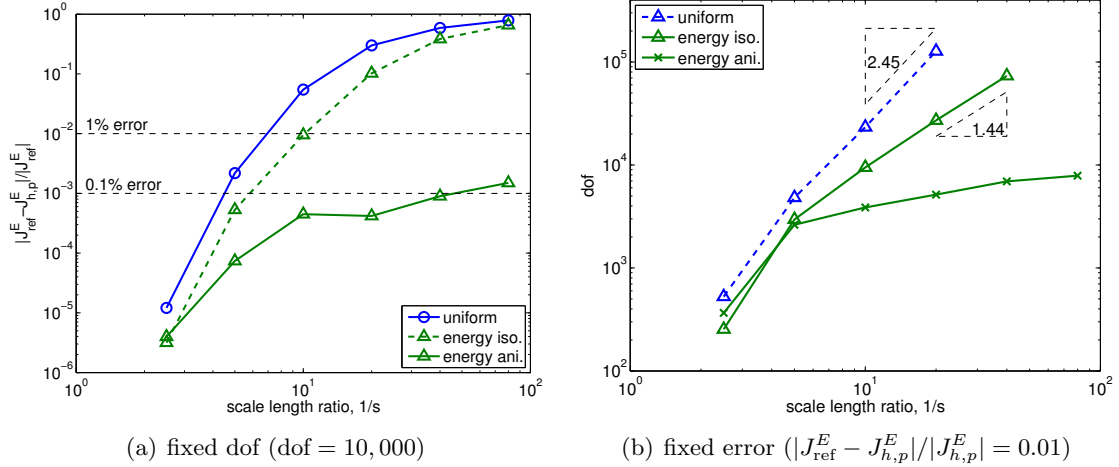


Figure 7-4: Scaling of the energy-error-to-dof efficiency with the characteristic length ratio,  $1/s$ , for the 1+1d wave problem. ( $p = 2$ )

$10^{-3}$  by over two orders of magnitude compared to uniform refinement. In fact, in order to achieve the  $10^{-3}$  relative error level, a uniform mesh with a fixed time stepping requires approximately 230 *spatial* elements (or 700 *spatial* degrees of freedom) and 230 third-order accurate temporal time steppings. On the other hand, output-based, anisotropic space-time adaptation requires 3000 *space-time* degrees of freedom to achieve the same error level. That is, the space-time anisotropic mesh requires the same order of space-time degrees of freedom as the spatial degrees of freedom for a uniform mesh, suggesting that the space-time anisotropy has effectively reduced the dimensionality of the problem by one.

It is important to note that the advantage of fully-unstructured anisotropic space-time adaptivity compared to isotropic adaptation or uniform refinement further increases as the ratio of the propagation length to the spatial characteristic length of the wave increases. For this problem, this ratio is controlled by the domain size ( $\mathcal{O}(1)$ ) and the characteristic length of the initial perturbation, the variable  $s$  in Eq. (7.1). The variation in the energy error against this scale length ratio for a fixed number of degrees of freedom is shown in Figure 7-4(a). Each discretization contain approximately 10,000 degrees of freedom. The energy error of both uniform refinement and isotropic adaptation increases rapidly with the scale length ratio. On the other hand, the energy error of the fully-unstructured space-time adaptation is much less sensitive to the increase in the range of scales. Figure 7-4(b) shows the variation in the number of degrees of freedom required to achieve a fixed error tolerance of approximately 1% fractional error. Again, the dimensionality reduction achieved by the



space-time anisotropy makes the anisotropic adaptation much less sensitive to the variation in the range of scales.

### 7.4.3 2+1d Wave Propagation

Let us consider a more practical problem in two spatial dimensions similar to the one considered in [15]. We consider wave propagation through a heterogeneous medium characterized by density and stiffness distributions

$$\begin{aligned}\rho(x) &= 1.0 \\ \kappa(x) &= 1.0 + 9.0 \left( \frac{1}{2} + \frac{1}{2} \tanh \left( \frac{x_1 - 0.2}{0.01} \right) \right).\end{aligned}$$

The coefficient  $\kappa$  changes rapidly (but smoothly) from 1.0 to 10.0 along  $x_1 = 0.2$ . The initial condition is given by

$$w(x, 0) = \exp \left( -\frac{x_1^2 + x_2^2}{s^2} \right) \quad \text{and} \quad q(x, 0) = 0,$$

with a characteristic length  $s = 0.05$ . Homogeneous Neumann boundary condition is imposed everywhere. The snapshots of the solution at several different times are shown in Figure 7-5.

For this problem, we are interested in the time history of the solution at  $(x_1, x_2) = (0.0, 0.75)$ , marked by a red circle in Figure 7-5. To target this point (or a line in space-time) using the output-based adaptation framework, we choose a regularized functional,

$$J^O = \mathcal{J}^O(u) = \int_I \int_{\Omega} g(x) w^2(x, t) dx dt \quad \text{where} \quad g(x) = \exp \left( -\frac{x_1^2 + (x_2 - 0.75)^2}{0.025^2} \right),$$

as our output of interest.

Figure 7-6 shows the convergence of the energy and output error using the  $p = 2$  DG discretization with the five different adaptation strategies discussed in Section 7.4.1. The reference energy and output function values are obtained from the energy- and output-adapted solutions, respectively, at 500,000 degrees of freedom.

Figure 7-6(a) shows that, as in the 1+1d-case, the anisotropic, energy-based adaptation outperforms other methods by a wide margin in preserving the total wave energy. In fact, using just 240,000 space-time degrees of freedom, the method achieves the 1% rela-



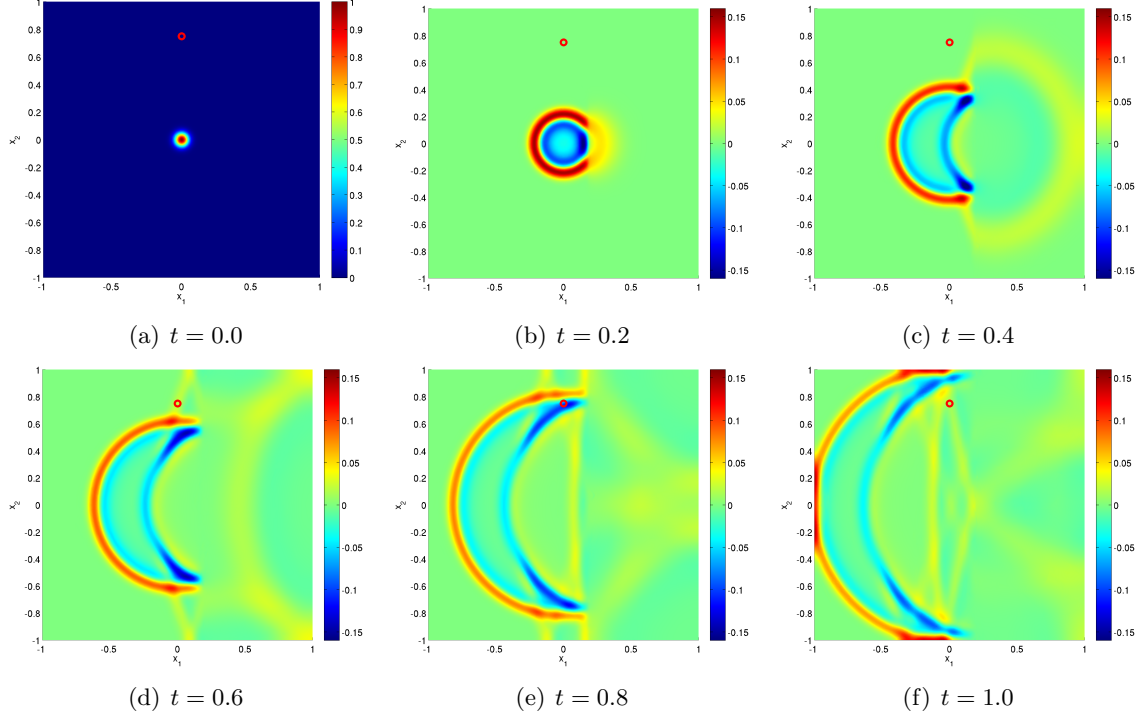


Figure 7-5: Time slices of the solution to the 2+1d wave problem. The output evaluation point is marked by a red circle. Note that the color scale for  $t = 0.0$  is different from that for all the others.

tive error level in energy. Comparing the energy-based anisotropic and isotropic adaptive results, anisotropy appears to make an even larger difference for the 2+1d problem. Again, this should be interpreted as a difference between an adaptive Rothe method and fully-unstructured space-time adaptation. In 2+1d, anisotropic adaptation effectively resolves anisotropic features in not only the space-time dimension but also within the spatial dimension. Note that it is difficult to realize arbitrary spatial anisotropy in a Rothe method as the use of different unstructured meshes in each time slab necessitates solving a complicated interfacing matching (i.e. arbitrary “hanging node”) problem across each time slab. While an approximate solution to this interface-matching problem may suffice for a low-order time integration, an accurate solution to the problem is necessary for a higher-order time integration. Thus, we expect the isotropic adaptation results to be representative of the performance one might get from an adaptive Rothe scheme in 2+1d.

Figure 7-6(b) shows that the combination of output-based error estimate and anisotropic adaptation is very effective at evaluating the (regularized) point output. Comparing the energy-based anisotropic adaptation and output-based isotropic adaptation, the addition



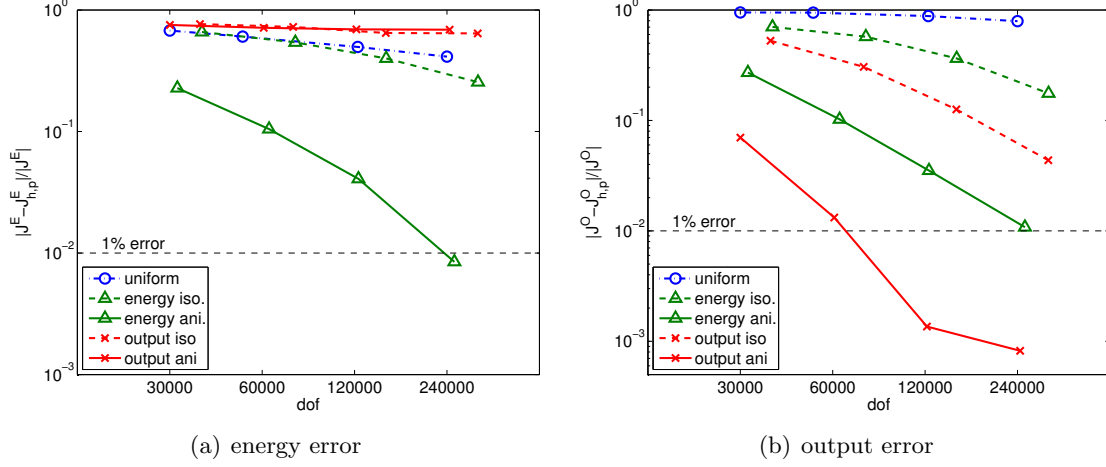


Figure 7-6: Energy and output error convergence for the 2+1d wave propagation problem. ( $p = 2$ )

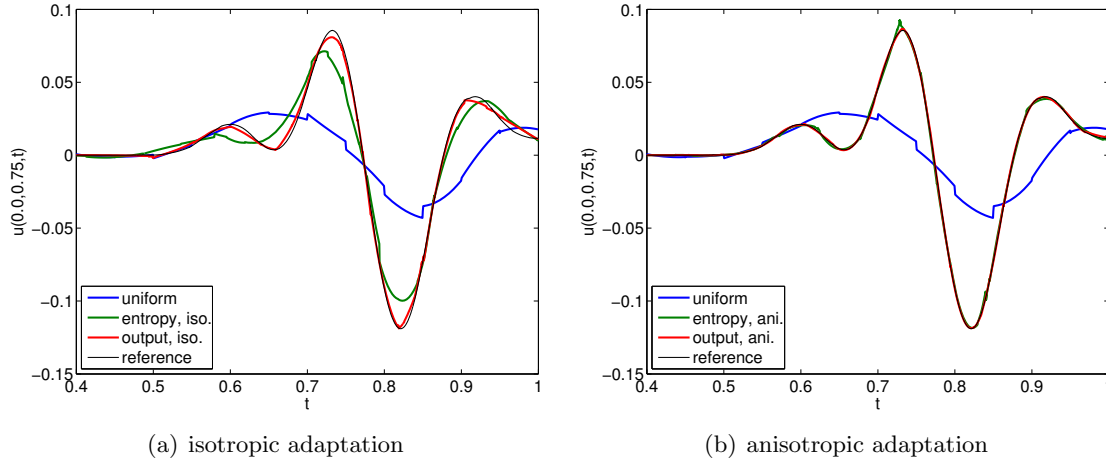


Figure 7-7: Solution history at  $x_1 = 0.0$ ,  $x_2 = 0.75$  for the 2+1d wave propagation problem. ( $p = 2$ ,  $\text{dof} \approx 240000$ )

of space-time anisotropy is more important than the output-based error estimate for this problem.

The solution history at  $(x_1, x_2) = (0, 0.75)$  is shown in Figure 7-7. At 240,000 degrees of freedom, the uniform mesh is clearly insufficient for capturing the solution history. Of the two isotropic adaptation strategies, the output-based adaptation performs considerably better than energy-based adaptation. The trend agrees with the result reported in [15]. Consistent with the output error convergence result, the two anisotropic adaptation strategies perform significantly better than the isotropic counterparts. In particular, the point output from the output-based anisotropic adaptation appears to be grid converged for the



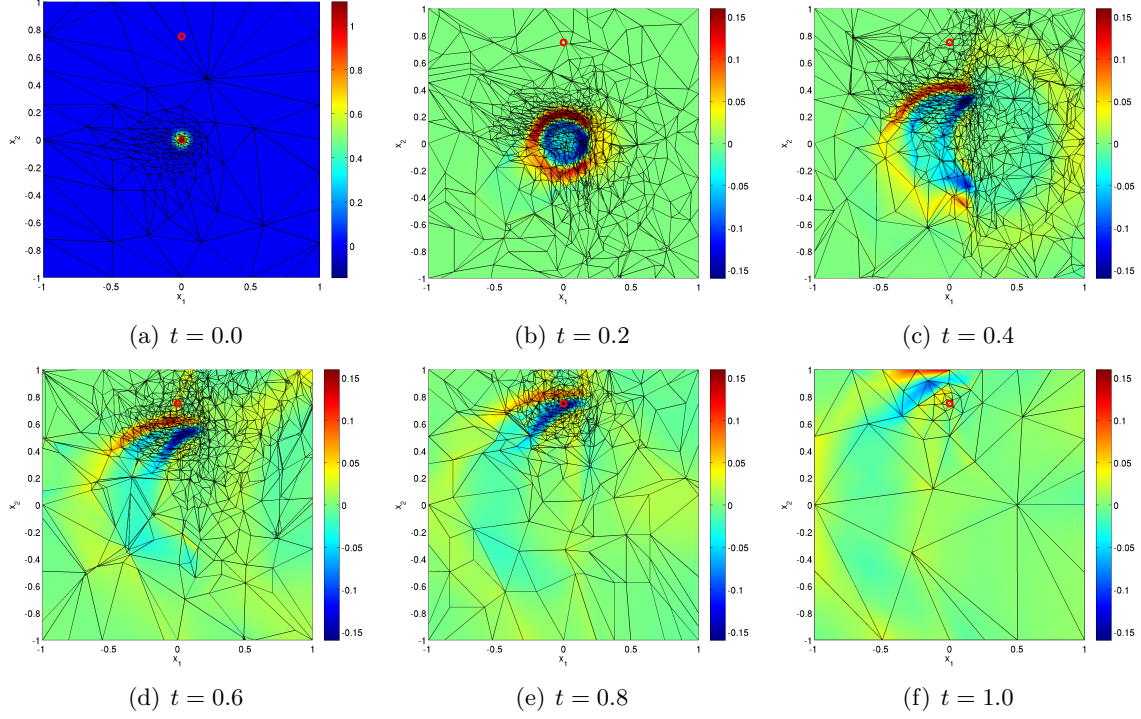


Figure 7-8: Time slices of the solution to the 2+1d wave problem obtained on the  $p = 2$ , dof = 240,000 output-adapted mesh. The output evaluation point is marked by a red circle. (c.f. the reference solution in Figure 7-5)

purpose of plotting at this degrees of freedom.

Slices of the mesh at select time instances obtained using the  $p = 2$ , dof = 240,000 output-based adaptation are shown in Figure 7-8. The output-based adaptation tracks only the wave features relevant to accurately evaluating the solution at  $(x_1, x_2) = (0, 0.75)$  (c.f. the reference solution in Figure 7-5). The slices of the solutions and crinkle cuts of the meshes obtained using the two anisotropic adaptation strategies are shown in Figures 7-9. The anisotropy in space-time planes is evident in the meshes.

## 7.5 Nonlinear Waves: Space-Time Euler Equations

Having shown the effectiveness of anisotropic space-time adaptation for linear wave propagation problems, this section applies the anisotropic adaptation algorithm to nonlinear wave propagation problems governed by the Euler equations. The Euler equations, reviewed in Chapter 6, are recast as a system of “steady-state” conservation laws in  $d + 1$  dimensions, where the conserved states constitutes the flux in the 0-th dimension. The Riemann solver



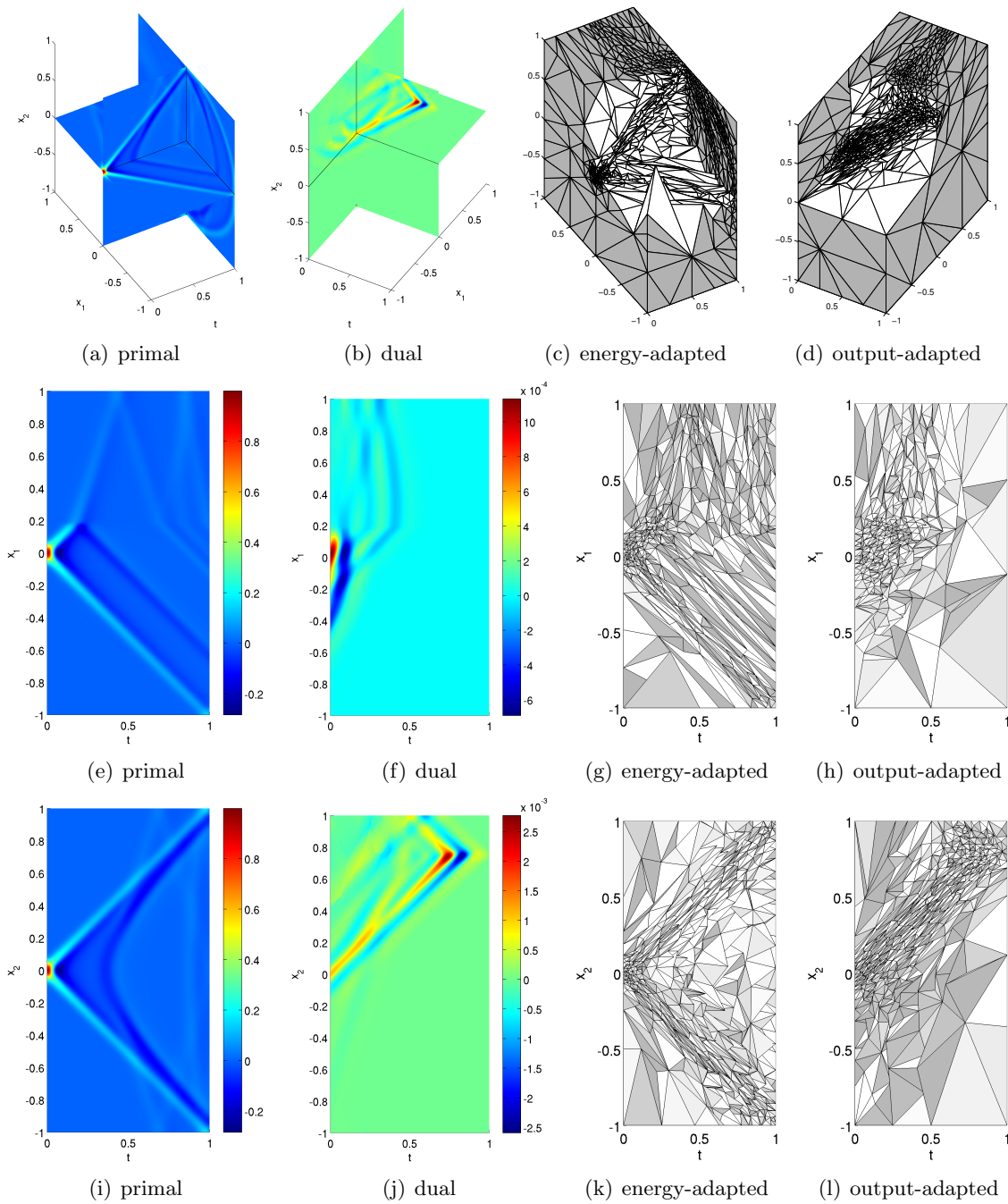


Figure 7-9: The primal solution, the dual solution, and  $p = 2$ ,  $\text{dof} = 240,000$  adapted meshes.  $2+1d$  view (top row); the  $x_2 = 0$  plane (middle row); and the  $x_1 = 0$  plane (bottom row).



is also modified accordingly to support arbitrarily-oriented space-time faces encountered in the fully-unstructured space-time formulation.

The discontinuity regularization technique described in Section A.1.1 is used to regularize the shocks. Unlike the steady state cases considered in Chapter 6, the unsteady Euler equations exhibit both shocks and contact discontinuities. Distinguishing the two types of discontinuities is important for accurate and robust simulations. While shocks are nonlinear features that require nonlinear stabilization, contact discontinuities are linear and the standard DG method results in a stable discretization. In fact, unlike in a shock, the waves do not coalesce in a contact discontinuity; thus, the dissipation must be minimized in order to preserve the sharp contact discontinuity. The physical viscosity model automatically distinguishes shocks and contact discontinuities, as the viscous flux vanishes across a contact discontinuity regardless of the value of the viscosity. However, we further use the jump in the pressure as the shock switch kernel so that the viscosity itself is not added across a contact discontinuity.

### 7.5.1 2+1d Vortex Convection

First, we consider convection of an isentropic vortex in two dimensions, which is similar to the problem considered by Wang and Mavriplis [150]. The freestream condition is given by  $\rho_\infty = 1$ ,  $u_\infty = 0.5$ ,  $v_\infty = 0$ , and  $T_\infty = 1$ . The convecting vortex centered at the origin is produced by perturbing the freestream condition by

$$\begin{aligned}\delta u &= -\frac{\alpha}{2\pi}x_2 \exp(1 - r^2) \\ \delta v &= \frac{\alpha}{2\pi}x_1 \exp(1 - r^2) \\ \delta T &= -\frac{\alpha^2(\gamma - 1)}{16\gamma\pi^2} \exp(2(1 - r^2)),\end{aligned}$$

where  $\alpha = 4$ ,  $r^2 = x_1^2 + x_2^2$ , and  $\gamma = 1.4$  is the ratio of specific heats. The isentropic condition specifies the density to be  $\rho = (T_\infty + \delta T)^{1/(\gamma-1)}$ . The vortex convects at the speed of  $u_\infty$  over the time interval  $[0, T]$  with  $T = 20$ . Figure 7-10 shows the variation in the density field over time.



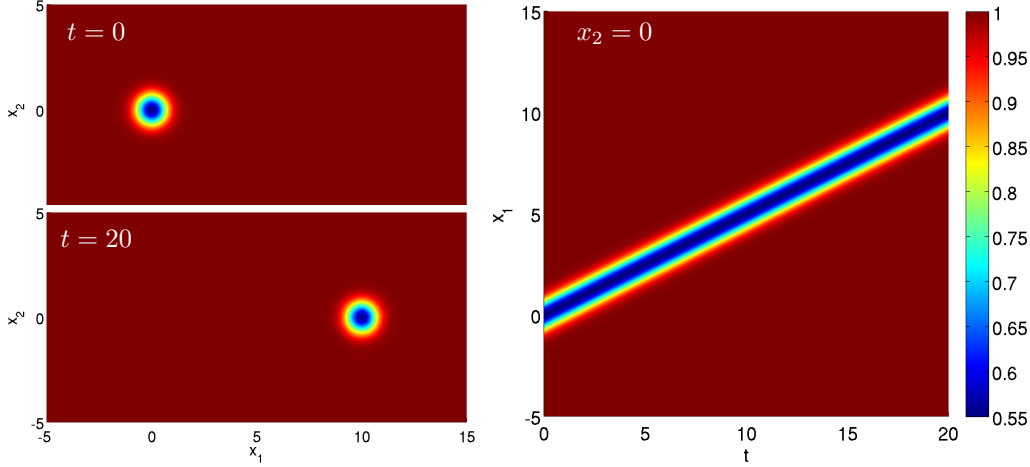


Figure 7-10: The density field of the isentropic vortex convection problem. The solution at  $t = 0$  and  $t = 20$  (left), and the space-time cut along  $x_2 = 0$  (right).

We consider the momentum perturbation at the final time as the output of interest, i.e.

$$J = \mathcal{J}(u) = \int_{\Omega} [(\rho(x, T)u(x, T) - \rho_{\infty}u_{\infty})^2 + (\rho(x, T)v(x, T) - \rho_{\infty}v_{\infty})^2] dx.$$

The mass adjoint corresponding to the output is shown in Figure 7-11. The complexity of the adjoint solution suggests that the relatively simple primal solution in fact results from complex nonlinear interactions of multiple waves.

Figure 7-12 shows the convergence of the momentum-perturbation output for the  $p = 1$ ,  $p = 2$ , and  $p = 3$  discretizations. Similar to the wave equation cases, the use of space-time anisotropy significantly improves the quality of the output prediction for a given number of space-time degrees of freedom. For this smooth problem, the output superconverges as  $\mathcal{E} \sim h^{2p} \sim (\text{dof})^{2p/(d+1)} = (\text{dof})^{2p/3}$  with uniform refinement, where  $d$  is the spatial dimension. The result is consistent with the theory. On the other hand, for the range of errors considered, the anisotropic refinement results in the output error converging as  $\mathcal{E} \sim (\text{dof})^{2p/d} = (\text{dof})^p$ . In other words, the error-to-dof scaling is similar to that expected for a  $2d$  problem rather than a  $2+1d$  problem. The result suggests that the use of space-time anisotropy effectively reduces the dimensionality of the problem. The efficiency gain for the fully-unstructured space-time formulation is expected to further increase with the ratio of the convection length to the vortex core size, as observed for the wave equation in Section 7.4.2.



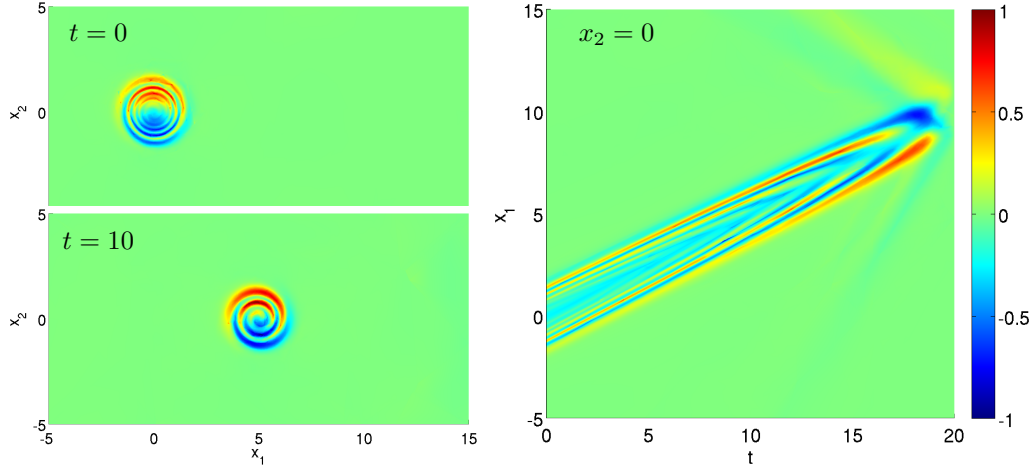


Figure 7-11: The mass adjoint for the momentum perturbation output of the isentropic vortex convection problem. The solution at  $t = 0$  and  $t = 10$  (left), and the space-time cut along  $x_2 = 0$  (right).

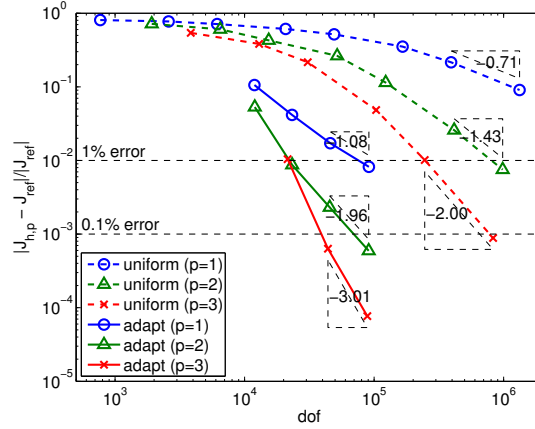


Figure 7-12: Convergence of the momentum-perturbation output for the isentropic vortex convection problem.



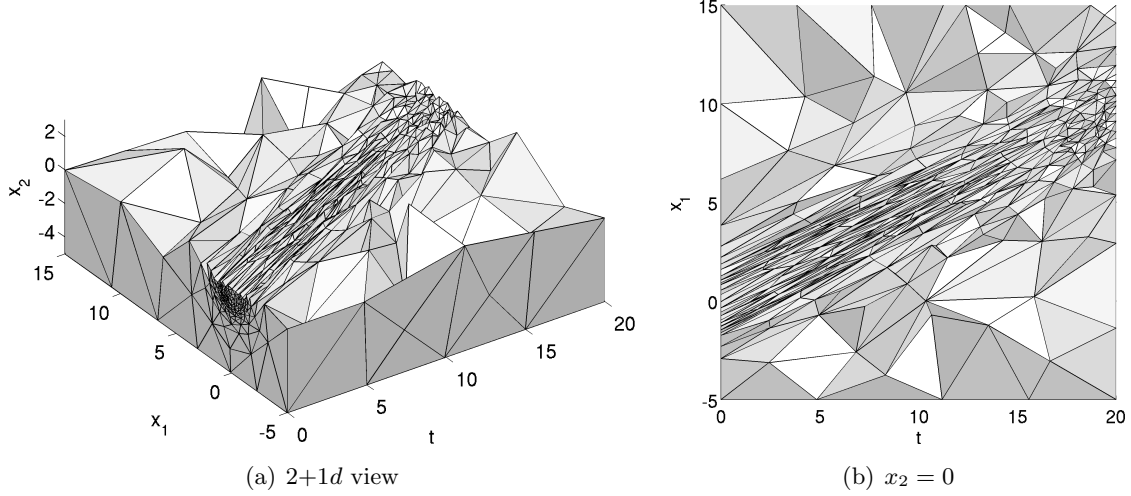


Figure 7-13: Space-time adapted mesh for the isentropic vortex convection problem. ( $p = 2$ , dof = 80,000)

Figure 7-13 shows a typical space-time adapted mesh. As expected, the mesh is refined only along the path traveled by the vortex. Note the use of highly anisotropic space-time elements, particularly for  $t < 15$ .

### 7.5.2 1+1d Riemann Problem

Let us consider a simple Riemann problem in one spatial dimension, which is a slight modification of Sod's classical shock tube problem [136]. The problem is solved on a space-time domain  $\Omega \times I = [-0.5, 0.5] \times [0, 0.75]$ . The air is initially at rest with a pressure ratio  $p_R/p_L = 2.5$  and a temperature ratio  $T_R/T_L = 1$ . The space-time fields of the density and the pressure are shown in Figure 7-14. The density field shows the shock, contact discontinuity, and rarefaction waves emanating from the initial discontinuity. At  $t \approx 0.6$ , the reflected shock interacts with the contact discontinuity, creating two shocks and one contact discontinuity. Our goal is to accurately capture the propagation and the interaction of the waves using the space-time anisotropic adaptation.

For this problem, we consider two different outputs: the squared pressure and density perturbations at the final time  $T$ , i.e.

$$J^\rho = \mathcal{J}^\rho(u) = \int_{\Omega} \rho^2(u(x, T)) dx \quad \text{and} \quad J^p = \mathcal{J}^p(u) = \int_{\Omega} p^2(u(x, T)) dx.$$

Accurate prediction of the density output requires resolution of both the shocks and contact



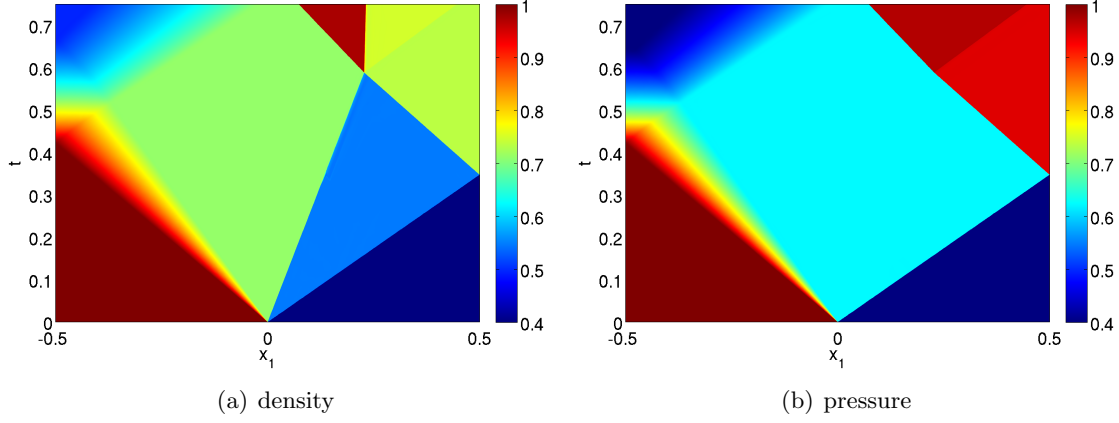


Figure 7-14: Solution to the shock tube problem.

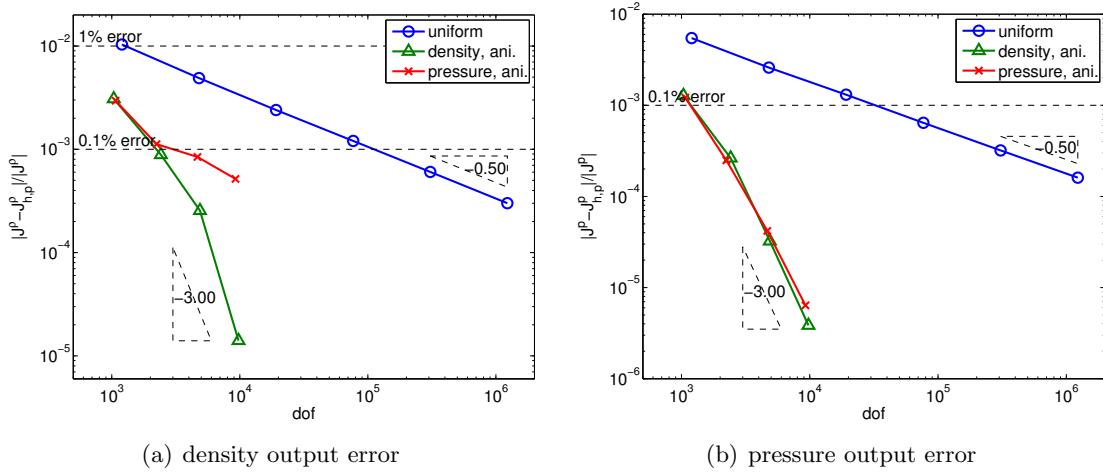


Figure 7-15: Convergence of the two outputs of the shock tube problem. ( $p = 2$ )

discontinuities, but the pressure output does not require resolution of the contact discontinuities.

Figure 7-15 shows the convergence of the density and pressure outputs with the number of degrees of freedom for the  $p = 2$  discretization. The reference solution was obtained using the adaptive  $p = 2$ ,  $\text{dof} = 32,000$  discretization. When uniform refinement is employed, both outputs converge at the rate of  $\mathcal{E} \sim h^1 \sim (\text{dof})^{-1/2}$ ; a higher-order convergence is not observed due to the presence of the discontinuities. Note that this convergence behavior is different from that for the smooth problems in Section 7.4.2, in which the higher-order convergence is eventually obtained in the asymptotic range.

The density-output-based anisotropic refinement produces meshes that target both the shocks and contact discontinuities, as shown in Figure 7-16(a). The adaptation does not



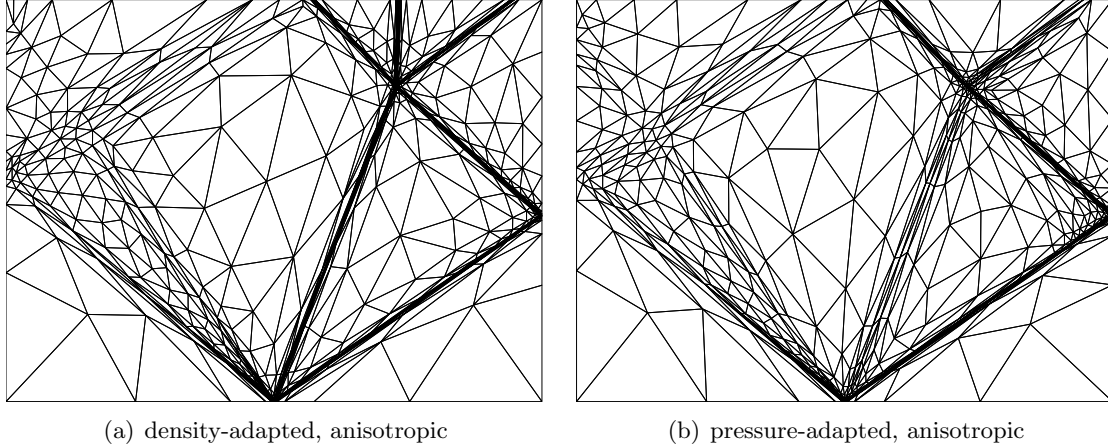


Figure 7-16: Adapted meshes for the shock tube problem. ( $p = 2$ ,  $\text{dof} = 10,000$ )

target the rarefaction waves because the waves are effectively propagated using the higher-order discretization. The anisotropic resolution of the discontinuities significantly improves the convergence of the output quantities. Figure 7-15 shows that the density-based refinement results in the error convergence of approximately  $\mathcal{E} \sim h^6 \sim (\text{dof})^{-3}$  for both the density and pressure outputs. The observed convergence rate exceeds the theoretical isotropic convergence rate using the  $p = 2$  discretization for a smooth problem,  $h^4$ . This is likely because the discontinuities in this problem are lower-dimensional features, and the use of anisotropy effectively reduced the dimensionality of the problem. The anisotropic refinement reduces the degrees of freedom required to achieve the fractional pressure output error of  $10^{-3}$  by a factor of 30. At a lower error level of  $10^{-5}$ , the anisotropic refinement reduces the degrees of freedom requirement by a factor of approximately  $4 \times 10^4$ . In fact, the 7,000 space-time degrees of freedom used by the anisotropic adaptation is smaller than the 17,000 spatial-only degrees of freedom that is expected to be required for the uniform refinement. In other words, as observed in Section 7.4.2, the space-time anisotropy effectively reduces the dimensionality of the problem by one.

As shown in Figure 7-16(a), the pressure-output-based anisotropic refinement produces meshes that only targets the shocks. As the contact discontinuities are not targeted by the adaptation, the density output does not converge rapidly, as shown in Figure 7-15. On the other hand, the pressure output converges at the rate of  $(\text{dof})^{-3}$ .

Figure 7-17 shows the density and pressure distributions at  $t = 0.25$  and  $t = 0.75$ . The uniform refinement result is obtained on a  $40 \times 40$ ,  $p = 2$  mesh, which corresponds



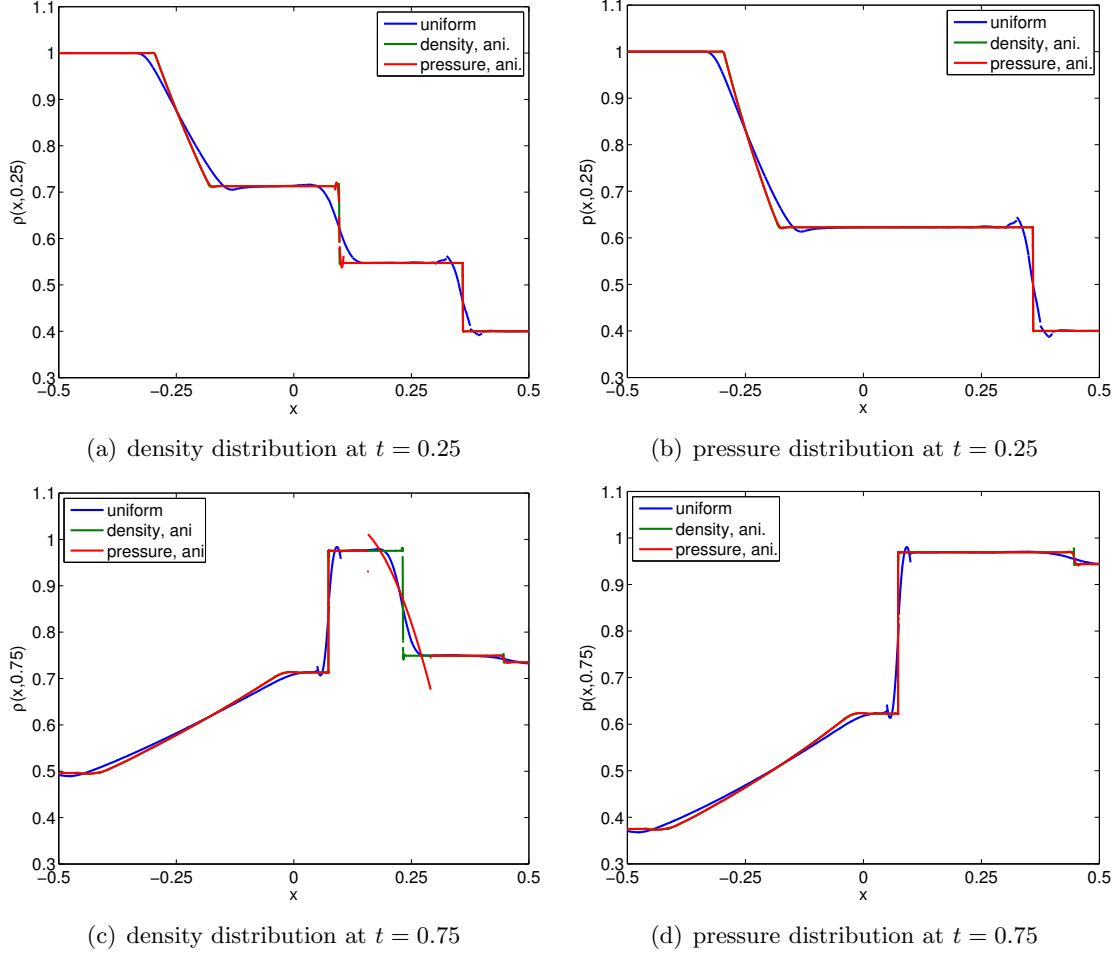


Figure 7-17: The density and pressure distributions of the shock tube problem at two different time instances. The uniform mesh contains approximately 20,000 degrees of freedom whereas the adapted meshes contain approximately 10,000 degrees of freedom.

to approximately 20,000 space-time degrees of freedom. At  $t = 0.25$ , the shock, contact discontinuity, and rarefaction waves are all smeared due to the lack of resolution. At  $t = 0.75$ , the contact discontinuity has further smeared due to the lack of the coalescing effect. The density-adapted mesh produces sharp density and pressure profiles at both time instances considered. The pressure-adapted mesh produces the pressure profiles that are indistinguishable from those of the density-adapted mesh, but the density profile at  $t = 0.75$  is inaccurate across the contact discontinuity.



## 7.6 Conclusions

A unified treatment of the spatial and temporal dimensions leads to a straightforward implementation of a fully-unstructured space-time anisotropic adaptive solver for the wave equation and the Euler equations. While the addition of the temporal dimension may appear costly, the numerical examples have demonstrated that an effective use of space-time anisotropy could significantly reduce the computational cost. By aligning the element anisotropy in the constant-phase direction, it appears that we can effectively reduce the dimensionality of the problem by one. In particular, space-time anisotropy is beneficial for problems exhibiting a wide range of scales. In higher spatial dimensions, the method also exploits anisotropy within the spatial dimension.

In addition to the use of space-time anisotropy, the unified space-time formulation offers a number of benefits compared to the traditional time-marching formulation. First, in the context of output error control for nonlinear problems, the additional cost incurred in solving the adjoint problem is significantly smaller for the space-time formulation, as the primal solution over the entire time is available by construction. Thus, output-based adaptation can be easily added; an efficient implementation of an adjoint solver in a typical time-marching solver requires an effective checkpointing scheme. Second, at least conceptually, the space-time formulation enables straightforward treatment of problems with moving boundaries using high-order discretizations.

Our numerical results suggest that the space-time formulation may be competitive with other existing adaptive schemes. We have estimated what might be possible with an adaptive Rothe method, which does not permit space-time oriented faces, using isotropic adaptation. The inability to produce space-time anisotropy hinders the performance of the Rothe method for wave propagation problems, whether energy-based or output-based error estimate is used. We also note that it is difficult, particularly for a high-order time integrator, to take advantage of arbitrary spatial anisotropy in the Rothe method as non-embedded meshes would require interface matching problem across each time slab.

In order for the unified space-time formulation to become truly competitive for large-scale wave propagation problems, a few challenges must be overcome. First, a preconditioner that takes advantage of the hyperbolicity of the problem in the temporal dimension must be developed. Without an efficient space-time preconditioner, the error-to-dof results



presented in this chapter are not necessary representative of computational efficiency, as discussed in Section 7.4.1. Second, a generation of  $(3+1)d$  unstructured space-time meshes remain an open problem. In particular, high-order meshing of complex four-dimensional space — which is required for, for example, three-dimensional simulations with moving boundaries — poses a significant challenge, possibly limiting the applicability of the fully-unstructured space-time formulation in the near future. Third, a mesher should handle internal boundaries to effectively resolve material discontinuities, which facilitates high-order simulation of waves through inhomogeneous media. Once these problems are solved, with an effective anisotropic adaptation mechanics, the unified space-time formulation may be a viable strategy for multiscale wave propagation problems in seismology, acoustics, and electromagnetics.







## Chapter 8

# Adaptation for Parametrized Partial Differential Equations

### 8.1 Introduction

As the technology to perform single-design-point simulations matures, developing techniques to characterize the behavior of performance variables over a wide range of design parameters has become increasingly important. Rapid characterization of the input-output relationship is crucial to enable computationally demanding tasks that require a large number of queries, such as design optimization, uncertainty quantification, and inverse parameter inference. While an efficient finite-element-based PDE solver may be used directly for tasks requiring a moderate number of evaluations, a further acceleration is necessary for tasks requiring thousands or even millions of input-output evaluations. Two popular acceleration techniques that take advantage of low dimensionality of the parameter-induced solution manifold are polynomial chaos and reduced order modeling. This chapter focuses on the development of an efficient and reliable finite-element-based PDE solver that can serve as a backbone of these two acceleration techniques, working toward development of a multi-fidelity solver that enables rapid characterization of parametrized PDEs.



### 8.1.1 Mathematical Description of the Problem

We consider a system of steady-state, parametrized conservation laws of the form

$$\nabla \cdot \mathcal{F}^{\text{conv}}(u, x; \mu) - \nabla \cdot \mathcal{F}^{\text{diff}}(u, \nabla u, x; \mu) = \mathcal{S}(u, \nabla u, x; \mu), \quad \forall x \in \Omega, \mu \in \Omega_\mu, \quad (8.1)$$

with the boundary conditions

$$\mathcal{B}(u, \hat{n} \cdot \mathcal{F}^{\text{diff}}(u, \nabla u, x; \mu), x; \text{BC}(\mu)) = 0, \quad \forall x \in \partial\Omega, \mu \in \Omega_\mu.$$

Here,  $\mu \in \mathbb{R}^{m_\mu}$  is the input parameter,  $u(x; \mu) \in \mathbb{R}^m$  is the parametrized state variable,  $\Omega \subset \mathbb{R}^d$  is the spatial domain, and  $\Omega_\mu \subset \mathbb{R}^{m_\mu}$  is the  $m_\mu$ -dimensional parameter domain.

## 8.2 Space-Parameter Galerkin Method

One approach to solving the parametrized PDE is to use a polynomial expansion to approximate the solution dependence on the parameters. Then, the appropriate coefficients of the polynomial expansion may be found by using the Galerkin projection in the parameter space. In particular, the application of this technique to quantify the propagation of stochastic parameters through a system is called polynomial chaos (PC), which has recently gathered considerable interest in the uncertainty quantification community. A recent review of PC methods in computational fluid dynamics is provided by Najm [106]. As our goal is to simply model the parameter dependency — which may or may not be stochastic — we will simply refer to this approach as the space-parameter Galerkin formulation. In particular, our goal is to adaptively control the spatial discretization error of the formulation to facilitate the application of the technique to problems exhibiting a wide range of spatial scales.

### 8.2.1 Formulation

We seek an approximate, weak solution to the parametrized PDE, Eq. (8.1), in a finite dimensional space-parameter space. Specifically, we augment the spatial finite element



space consisting of piecewise  $p$ -degree (complete) polynomials, introduced in Section 2.1,

$$V_{h,p} = \{v_{h,p} \in [L^2(\Omega)]^m : v_{h,p} \circ f_\kappa \in [\mathcal{P}^p(\kappa_{\text{ref}})]^m, \forall \kappa \in \mathcal{T}_h\},$$

by an  $s$ -degree tensor-product polynomial parameter space,

$$V_s^\mu = [\mathcal{P}^s(\Omega_\mu)]^{m_\mu},$$

and seek the solution  $u_{h,p,s} \in V_{h,p} \times V_s^\mu$ . Note that the parameter approximation space  $V_s^\mu$  is  $m_\mu \cdot (s + 1)$  dimensional. Using the spatial DG discretization described in Section 2.1, the approximate solution for a given parameter,  $u(\cdot; \mu) \in V_{h,p}$ , satisfies

$$\mathcal{R}_{h,p}(u_{h,p}(\cdot; \mu), v_{h,p}; \mu) = 0, \quad \forall v_{h,p} \in V_{h,p},$$

where  $\mathcal{R}_{h,p}(\cdot, \cdot; \mu)$  is the parameter-dependent semilinear form for the conservation law, Eq. (8.1). Projection of the spatial-residual form onto the parameter space results in a weak form: Find  $u_{h,p,s} \in V_{h,p} \times V_s^\mu$  such that

$$\mathcal{R}_{h,p,s}(u_{h,p,s}, v_{h,p,s}) = 0, \quad \forall v_{h,p,s} \in V_{h,p} \times V_s^\mu, \quad (8.2)$$

where  $\mathcal{R}_{h,p,s}(\cdot, \cdot)$  is the space-parameter semilinear form given by

$$\mathcal{R}_{h,p,s}(w_{h,p,s}, v_{h,p,s}) \equiv \int_{\Omega_\mu} \mathcal{R}_{h,p}(w_{h,p,s}(\cdot; \mu), v_{h,p,s}(\cdot; \mu); \mu) d\mu.$$

Because the solution space results from the tensor product of the spatial and parameter spaces, the solution may be decomposed as

$$u_{h,p,s}(x; \mu) = \sum_{i=1}^{\dim(V_s^\mu)} u_{h,p}^{(i)}(x) \chi_s^{(i)}(\mu),$$

where  $\{\chi_s^{(i)}\}_{i=1}^{\dim(V_s^\mu)}$  is a set of basis functions that spans  $V_s^\mu$ , and the field  $u_{h,p}^{(i)} \in V_{h,p}$  is the parameter expansion mode strength associated with the  $i$ -th mode.



Once the solution  $u_{h,p,s}$  is obtained, a functional output can be evaluated by

$$J_{h,p,s} = \mathcal{J}(u_{h,p,s}) \equiv \int_{\Omega_\mu} g(\mathcal{J}_{h,p}(u_{h,p,s}(\cdot; \mu); \mu)) d\mu,$$

where  $\mathcal{J}_{h,p}(\cdot; \mu)$  is the parameter-dependent output functional, and function  $g$  defines the dependency of the functional on  $V_{h,p} \times V_s^\mu$  to the functional on  $V_{h,p}$ . For example, if the output of interest is the mean of the output over the parameter domain, then  $g$  is the identity map.

### 8.2.2 Stability of the Space-Parameter Formulation

The stability of the space-parameter formulation applied to a nonlinear hyperbolic system is summarized in the following theorem.

**Theorem 8.1** (Global entropy norm stability of nonlinear hyperbolic system). *Suppose, for each parameter  $\mu \in \Omega_\mu$ , the conservation law of interest possess an entropy pair  $\{\mathfrak{U}, \mathfrak{F}\}$  that satisfies*

$$\frac{\partial \mathfrak{U}}{\partial t} + \nabla \cdot \mathfrak{F} \leq 0,$$

where  $\mathfrak{U}(u) : \mathbb{R}^m \rightarrow \mathbb{R}^+$  is a nonnegative convex entropy function, and the DG discretization is equipped with the symmetric mean-value numerical flux function defined by Barth [21]. Then, the space-parameter discretization is globally entropy stable in the sense that

$$\int_{\Omega_\mu} \int_{\Omega} \mathfrak{U}(x, t^1; \mu) dx d\mu \leq \int_{\Omega_\mu} \int_{\Omega} \mathfrak{U}(x, t^0; \mu) dx d\mu, \quad \forall t^1 \geq t^0.$$

*Proof.* The global nonlinear stability of the space-parameter system is a direct consequence of the variational formulation and the entropy stability of the DG discretization [21] for each instance of the parameter. Namely, the DG discretization applied to a hyperbolic system with an entropy pair satisfies the global entropy balance, i.e. testing against the entropy variable  $v^T \equiv \frac{\partial \mathfrak{U}}{\partial u}$  yields

$$\begin{aligned} 0 &= \int_{t^0}^{t^1} \int_{\Omega} v^T(x, t; \mu) \frac{\partial u(x, t; \mu)}{\partial t} dx dt + \int_{t^0}^{t^1} \mathcal{R}^{\text{conv}}(v(\cdot, t; \mu), u(\cdot, t; \mu); \mu) dt \\ &= \int_{\Omega} \mathfrak{U}(x, t^1; \mu) dx - \int_{\Omega} \mathfrak{U}(x, t^0; \mu) dx + \Theta^2(\mu), \quad \forall \mu \in \Omega_\mu, \end{aligned}$$



where  $\Theta^2(\mu) \geq 0$  is the dissipation function, whose exact form is dependent on the numerical flux function. By construction of the space-parameter semilinear form,

$$\begin{aligned}
0 &= \int_{t^0}^{t^1} \int_{\Omega_\mu} \int_{\Omega} v^T(x, t; \mu) \frac{\partial u(x, t; \mu)}{\partial t} dx d\mu dt + \int_{t^0}^{t^1} \mathcal{R}^{\text{conv}}(v(\cdot, t; \cdot), u(\cdot, t; \cdot)) dt \\
&= \int_{\Omega_\mu} \left[ \int_{t^0}^{t^1} \int_{\Omega} v^T(x, t; \mu) \frac{\partial u(x, t; \mu)}{\partial t} dx dt + \int_{t^0}^{t^1} \mathcal{R}^{\text{conv}}(v(\cdot, t; \mu), u(\cdot, t; \mu); \mu) dt \right] d\mu \\
&= \int_{\Omega_\mu} \left[ \int_{\Omega} \mathfrak{U}(x, t^1; \mu) dx - \int_{\Omega} \mathfrak{U}(x, t^0; \mu) dx + \Theta^2(\mu) \right] d\mu.
\end{aligned}$$

Noting  $\int_{\Omega_\mu} \Theta^2(\mu) d\mu \geq 0$  proves the desired result.  $\square$

We emphasize that our space-parameter discretization is different from that obtained by applying the DG spatial discretization to a large, parameter-expanded system that results from projecting the flux onto a polynomial approximation of the parameter space. In other words, unlike the approach taken by Lin *et al.* [95] and Tryoen *et al.* [141], our discretization does not result from spatially discretizing

$$\frac{\partial}{\partial t} \mathbf{u}(x, t) + \nabla \cdot \mathcal{F}(\mathbf{u}(x, t)) = 0, \quad \forall x \in \Omega, t \in I, \quad (8.3)$$

where the parameter-expanded state,  $\mathbf{u}(x, t) \in m \cdot \dim(V_s^\mu)$ , and flux,  $\mathcal{F}(\mathbf{u}(x, t)) \in d \times m \cdot \dim(V_s^\mu)$  are given by

$$\begin{aligned}
\mathbf{u}^{(i)}(x, t) &= \int_{\Omega_\mu} \chi_s^{(i)}(\mu) u(x, t; \mu) d\mu \\
\mathcal{F}^{(i)}(\mathbf{u}(x, t)) &= \int_{\Omega_\mu} \chi_s^{(i)}(\mu) \mathcal{F}(u(x, t; \mu); \mu) d\mu,
\end{aligned}$$

and  $\chi_s^{(i)}$  is the  $i$ -th parameter basis. Hyperbolicity of this parameter-expanded system, Eq. (8.3), is not guaranteed in general [141]. Even if the system is hyperbolic, constructing a stable discretization requires an appropriate upwinded numerical flux for the parameter-expanded flux,  $\mathcal{F}$ . In general this requires a solution to an  $m \cdot \dim(V_s^\mu)$  dimensional eigenproblem. To circumvent the costly operation, Lin *et al.* [95] and Tryoen *et al.* [141] introduce approximate upwinding fluxes for their finite volume discretizations, which are not provably stable and are unsuited for implicit solvers. The space-parameter variational framework employed in this work is entropy stable and requires no modifications to the standard



single-parameter numerical flux function.

In practice, our implementation operates on the conservative variables instead of the entropy variables, and the symmetric mean-value numerical flux is replaced by Roe's approximate Riemann solver. While the resulting space-parameter discretization is not provably entropy stable [21], the same modifications have been made to the space-only discretization considered in Chapter 6 without any practical problems. Thus, these modifications are expected to have negligible impact in the context of space-parameter discretization of hyperbolic conservation laws.

### 8.2.3 Spatial Error Estimation and Control

In order to estimate the output error due to the lack of spatial resolution (and not the parameter resolution), the DWR error estimate is constructed by enriching only the spatial space. Namely, defining the spatial contribution to the output error as

$$\mathcal{E}_{\text{true},s} \equiv \lim_{h \rightarrow 0} [J_{h,p,s}] - J_{h,p,s},$$

we estimate the error by

$$\mathcal{E}_{\text{true},s} \approx -\mathcal{R}_{h,p,s}(u_{h,p,s}, \psi_{h,\hat{p},s}),$$

where  $\psi_{h,\hat{p},s} \in V_{h,\hat{p}} \times V_s^\mu$  is the approximate truth adjoint satisfying

$$\mathcal{R}'_{h,\hat{p},s}[u_{h,p,s}](v_{h,\hat{p},s}, \psi_{h,\hat{p},s}) = \mathcal{J}'_{h,\hat{p},s}[u_{h,p,s}](v_{h,\hat{p},s}), \quad \forall v_{h,\hat{p},s} \in V_{h,\hat{p}} \times V_s^\mu,$$

and  $\hat{p} = p + 1$ . The spatially local error contribution is estimated by

$$\eta_\kappa = |\mathcal{R}_{h,p,s}(u_{h,p,s}, \psi_{h,\hat{p},s}|_\kappa)|, \tag{8.4}$$

where  $\psi_{h,\hat{p},s}|_\kappa$  should be understood as the restriction of  $\psi_{h,\hat{p},s}$  to the space-parameter element  $\kappa \times \Omega_\mu$ .



### 8.2.4 Practical Considerations

While the formulation handles parameter space of arbitrary dimensions in principle, for simplicity, we consider only one-dimensional parameter domain in this work. Thus, we have  $m_\mu = 1$ , and the parameter space is given by  $V_s^\mu = \mathcal{P}^s(\Omega_\mu)$  where  $\Omega_\mu \subset \mathbb{R}$ . Without loss of generality, let us represent the parameter and solution variation using a spectral decomposition of the form

$$\mu(\theta) = \sum_{i=0}^s \mu^{(i)} \chi_s^{(i)}(\theta) \quad \text{and} \quad u_{h,p,s}(x; \mu(\theta)) = \sum_{i=0}^s u_{h,p}^{(i)}(x) \chi_s^{(i)}(\theta), \quad \forall x \in \Omega,$$

where  $\theta \in [0, 1]$ ,  $\chi^{(i)}$  is the  $i$ -th Legendre polynomial, and  $u_{h,p}^{(i)} \in V_{h,p}$  is the parameter expansion mode strength of the  $i$ -th mode. Throughout the rest of the chapter, the  $i$ -th parameter mode strength refers to the field of coefficients associated with the  $i$ -th spectral mode of this decomposition.

## 8.3 Space-Galerkin Parameter-Collocation Method

The second approach to solving the parametrized PDE is based on using collocation in the parameter space. Our goal in this case is to generate a single spatial approximation space,  $V_{h,p}$ , suited for the entire range of parameters, i.e. construction of an optimal universal mesh. Such a universal mesh can serve as an efficient “truth” finite-element mesh in reduced order model space generation, for example by proper orthogonal decomposition [29, 135] or greedy sampling [149]. In this context, the “truth” space must capture all features on the parameter-induced solution manifold relevant to evaluation of the output. On the other hand, the space should not be excessively large to enable efficient calculation of the snapshots and to facilitate more extensive search over the parameter domain. The universal mesh can also be used to construct simple interpolation for parameter variations directly for small parameter dimensions. Thus, our goal is to use our versatile adaptation algorithm to generate optimal universal meshes suited for the entire range of parameters.

### 8.3.1 Formulation

The space-Galerkin parameter-collocation method approximates the variation of the output  $J(\mu)$  with respect to the parameter  $\mu$  by simply finding the solution at select collocation



points. That is, our objective is to find a set of snapshots  $\{u_{h,p}(\cdot; \mu)\}_{\mu \in M}$ , where  $M$  is a set of parameter-evaluation points and  $u_{h,p}(\cdot; \mu) \in V_{h,p}$ . Here,  $M$  serves as a finite dimensional surrogate of the parameter space  $\Omega_\mu$ . The standard single-parameter discretization presented in Chapter 2 is used to construct each snapshot, i.e. Find  $u_{h,p}(\cdot; \mu) \in V_{h,p}$  for  $\mu \in M$  such that

$$\mathcal{R}_{h,p}(u_{h,p}(\cdot; \mu), v_{h,p}; \mu) = 0, \quad \forall v_{h,p} \in V_{h,p},$$

and evaluate the output

$$J(\mu) \equiv \mathcal{J}(u_{h,p}(\cdot; \mu); \mu).$$

### 8.3.2 Spatial Error Estimate and Error Control

We define the output error for the parameter-collocation method as

$$\mathcal{E}_{\text{true}, M} \equiv \sum_{\mu \in M} |J(\mu) - J_{h,p}(\mu)|.$$

Note that the error is defined as the sum of the errors at the prescribed collocation points,  $M$ . In particular, the error due to an insufficient distribution of  $M$  over the parameter space  $\Omega_\mu$  is not accounted for in this formulation; this is analogous to the omission of the error due to an insufficient parameter expansion in the space-parameter Galerkin formulation in Section 8.2. We assume that  $M$  is sufficiently large that the maximum error observed for  $\mu \in M$  is that encountered over  $\Omega_\mu$ . We estimate the error by

$$\mathcal{E}_{\text{true}, M} \approx \sum_{\mu \in M} |\mathcal{R}_{h,p}(u_{h,p}(\cdot; \mu), \psi_{h,\hat{p}}(\cdot; \mu); \mu)|,$$

where  $\psi_{h,\hat{p}}(\cdot; \mu) \in V_{h,\hat{p}}$  is the approximate adjoint satisfying

$$\mathcal{R}'_{h,\hat{p}}[u_{h,p}(\cdot; \mu)](v_{h,\hat{p}}, \psi_{h,\hat{p}}(\cdot; \mu); \mu) = \mathcal{J}'_{h,\hat{p}}[u_{h,p}(\cdot; \mu)](v_{h,\hat{p}}; \mu), \quad \forall v_{h,\hat{p}} \in V_{h,\hat{p}},$$



and  $\hat{p} = p + 1$ . We will define the elemental error contribution as the sum of the error contribution from each parameter collocation point, i.e.

$$\eta_\kappa \equiv \sum_{\mu \in M} |\mathcal{R}_{h,p}(u_{h,p}(\cdot; \mu), \psi_{h,\hat{p}}(\cdot; \mu)|_\kappa; \mu)|. \quad (8.5)$$

Interpreting the summation of the errors at the collocation points as an approximate integral, the elemental error captures the error contribution of the space-parameter element.

## 8.4 Numerical Results

### 8.4.1 RAE 2822 Subsonic RANS-SA

We first consider subsonic RANS-SA flow over an RAE 2822 airfoil with a freestream Mach number of  $M_\infty = 0.3$  and a Reynolds number of  $Re_c = 1 \times 10^6$ . The parameter of interest is the angle of attack,  $\alpha$ , which varies from  $0^\circ$  to  $6^\circ$ . The far field boundary is located  $200c$  away.

#### Behavior of the Space-Parameter Galerkin Formulation

Let us first analyze the behavior of the space-parameter Galerkin formulation of the RANS-SA equations. For this study, the output of interest is set to the mean drag. Figure 8-1 shows the fields of parameter expansion coefficients for select components. The mode 0 fields, which correspond to the mean of the solution fields over the parameter space, is similar to, but different from, the solution field for the  $\alpha = 3^\circ$  case. The differences are due to the nonlinear dependence of the flow field on the angle of attack. The mode 1 fields encode the linear variation in the field quantities with the parameter. The  $x$ -momentum on the upper surface of the airfoil increases with the angle of attack as the flow experiences larger acceleration. The wake region of both the  $x$ -momentum and the SA working variable has a large linear coefficient as the angle of the wake shifts upward with the angle of attack.

Let us now make a quantitative assessment of the convergence property of the spectral parameter expansion for this RANS-SA flow. A fixed  $p = 2$ ,  $\text{dof} = 10,000$  spatial mesh, generated through output-based adaptation, is used for this purpose. Point-wise simulations at several angles of attack show that the mesh commits the  $c_d$  error of less than  $1 \times 10^{-4}$  over the range of angles of attack considered. (This result is presented in the following



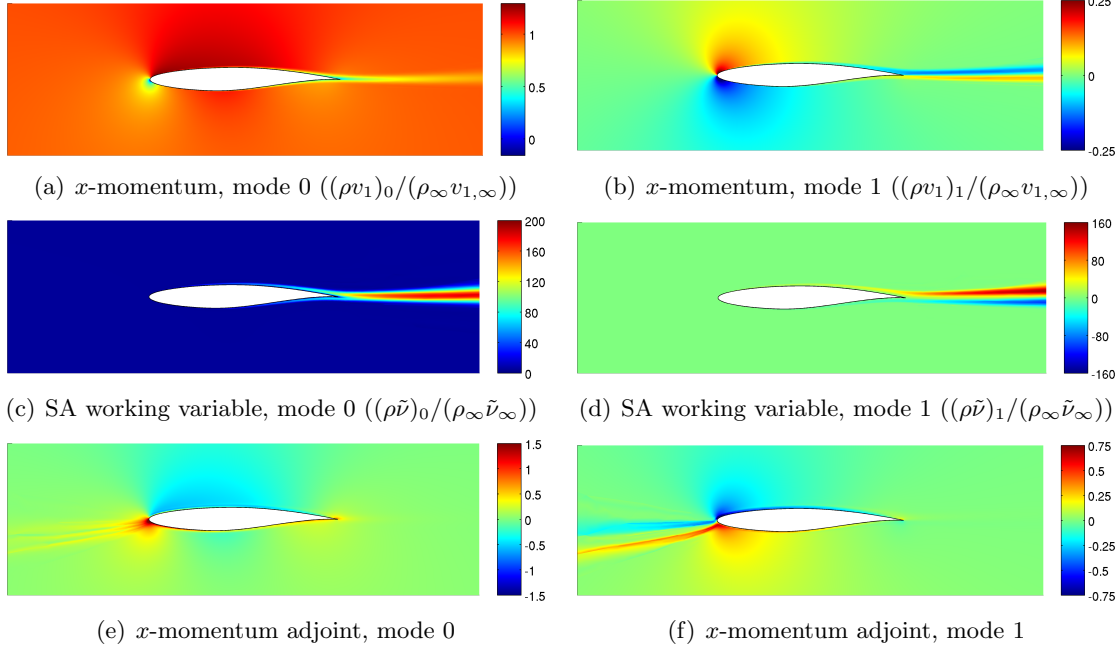


Figure 8-1: Parameter expansion mode strengths of the first two modes for select solution fields of the RAE 2822 case. The output is the mean drag.

section on spatial adaptation.)

The lift curve and the drag polar for the parameter degrees of  $s = 1, \dots, 4$  are shown in Figure 8-2. The results of the point-wise simulation at  $\alpha = 0^\circ, 2^\circ, 4^\circ$ , and  $6^\circ$  on the same spatial mesh are also included for the verification purpose. The figure shows that the lift curve quickly converges even for a small parameter expansion degree,  $s$ . On the other hand, the computation of the drag requires a higher-degree polynomial expansion in the parameter space. Note that the lift and drag computed using a  $s$ -degree polynomial parameter expansion is in general not a  $s$ -degree polynomial due to the nonlinear dependence of the outputs on the solution fields.

Figure 8-3(a) shows the variation in the  $c_d$  error due to insufficient parameter space resolution for the  $s = 1, \dots, 4$  expansions. The reference solution is computed on the same spatial mesh using the  $s = 8$  expansion. The  $c_d$  error in the parameter space is equally distributed; this is not too surprising as the Galerkin projection is employed in the parameter space. Figure 8-3(b) shows the variation in the maximum  $c_d$  error over the parameter space as a function of the parameter polynomial degree. Figure shows that the expansion initially converges rapidly to the true solution, but the convergence stalls for  $s > 4$ . The lack of spectral convergence suggests that the solution over the parameter space



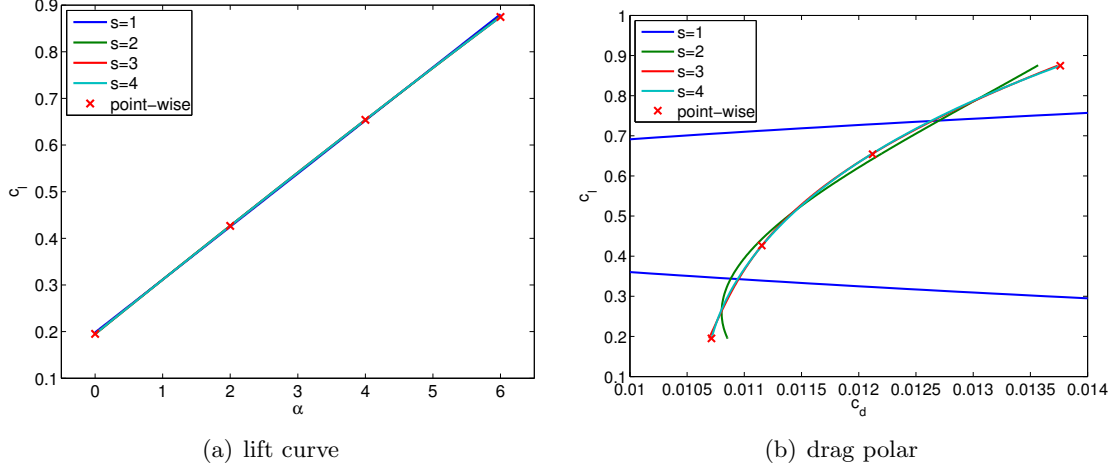


Figure 8-2: The lift curve and drag polar for the RAE 2822 case on a fixed  $p = 2$ ,  $\text{dof} = 10,000$  mesh.

is not smooth. Any singular spatial feature whose location is dependent on the parameter results in singularity in the parameter space; for the RANS-SA flow, the singularity along the outer edge of the turbulent boundary layer and the stagnation streamline in the adjoint solution are potential candidates limiting the regularity in the parameter space. Note that a spatial singularity whose location is independent of the parameter, e.g. the trailing edge singularity, does not influence the regularity of the solution in the parameter space.

Fortunately, Figure 8-3(b) also shows that the limited regularity in the parameter space does not impact the convergence for the maximum  $c_d$  error of greater than 0.1 counts. The result suggests that, for the purpose of drag prediction, resolving some of the low regularity features is not crucial at the accuracy required for a typical engineering simulation of RANS flows. In particular, the  $s = 4$  parameter expansion is sufficient to achieve less than 0.1 counts of drag error with respect to a reference solution computed on the same spatial mesh; thus, the  $s = 4$  parameter expansion is used to assess the quality of the spatial adaptation for the space-parameter Galerkin formulation.

## Spatial Adaptation

We assess the quality of the spatial meshes generated by the space-parameter Galerkin formulation and the space-Galerkin parameter-collocation formulation. For simplicity, we refer to the two formulations by the type of discretization employed in the parameter space, i.e. Galerkin and collocation. Specifically, the parameter space is discretized by



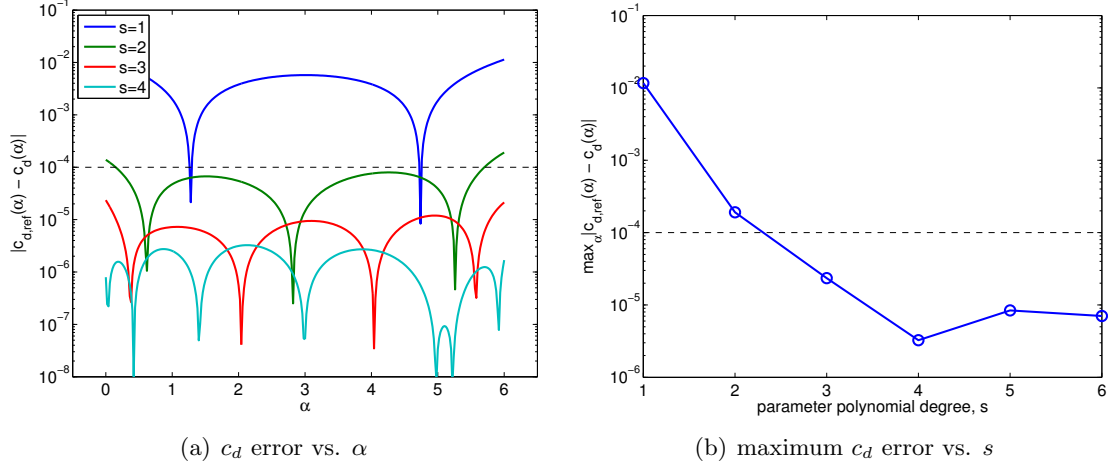


Figure 8-3: Variation in the  $c_d$  error with the parameter expansion degree,  $s$ , for the RAE 2822 case. The reference solution is computed using the  $s = 8$  expansion. Solutions computed on a fixed  $p = 2$ ,  $dof = 10,000$  mesh.

- **Galerkin:**  $s = 4$  polynomial expansion
- **Collocation:** quadrature points of the 3-point Gaussian quadrature rule

Figures 8-4(c) and 8-4(d) show the  $p = 2$ ,  $dof = 10,000$  meshes optimized over  $\alpha \in [0^\circ, 6^\circ]$  using the Galerkin and collocation formulation, respectively. As a comparison, the meshes optimized for  $\alpha = 0^\circ$  and  $\alpha = 4^\circ$  are shown in Figures 8-4(a) and 8-4(b), respectively. For this subsonic configuration and at this error range, neither the wake nor the stagnation streamline is strongly targeted, and all four meshes focus on resolving the boundary layer. One notable difference among the meshes is the element packing at the leading edge. As shown in the zoomed figures, the  $\alpha$ -specific adapted meshes use relatively large elements in the vicinity of the stagnation points, where the boundary layer is absent at the specified angle of attack. On the other hand, the  $\alpha \in [0^\circ, 6^\circ]$ -optimized meshes produce boundary layer packing over the entire leading edge region because, at any given location, the boundary layer is present for some angle of attack. Adapting to the spatial error of the Galerkin and collocation formulations result in similar spatial meshes.

Figure 8-5(a) shows the variation in the error over the range of parameters considered. The reference solution is computed on  $p = 3$ ,  $dof = 80,000$  adapted meshes, each optimized for the specific angle of attack. While the objective is to minimize the true error, the adaptation algorithm works on minimizing the error indicator; for this reason, the variation in the error indicator is also shown in Figure 8-5(b). As expected, the  $\alpha = 0^\circ$ -adapted mesh



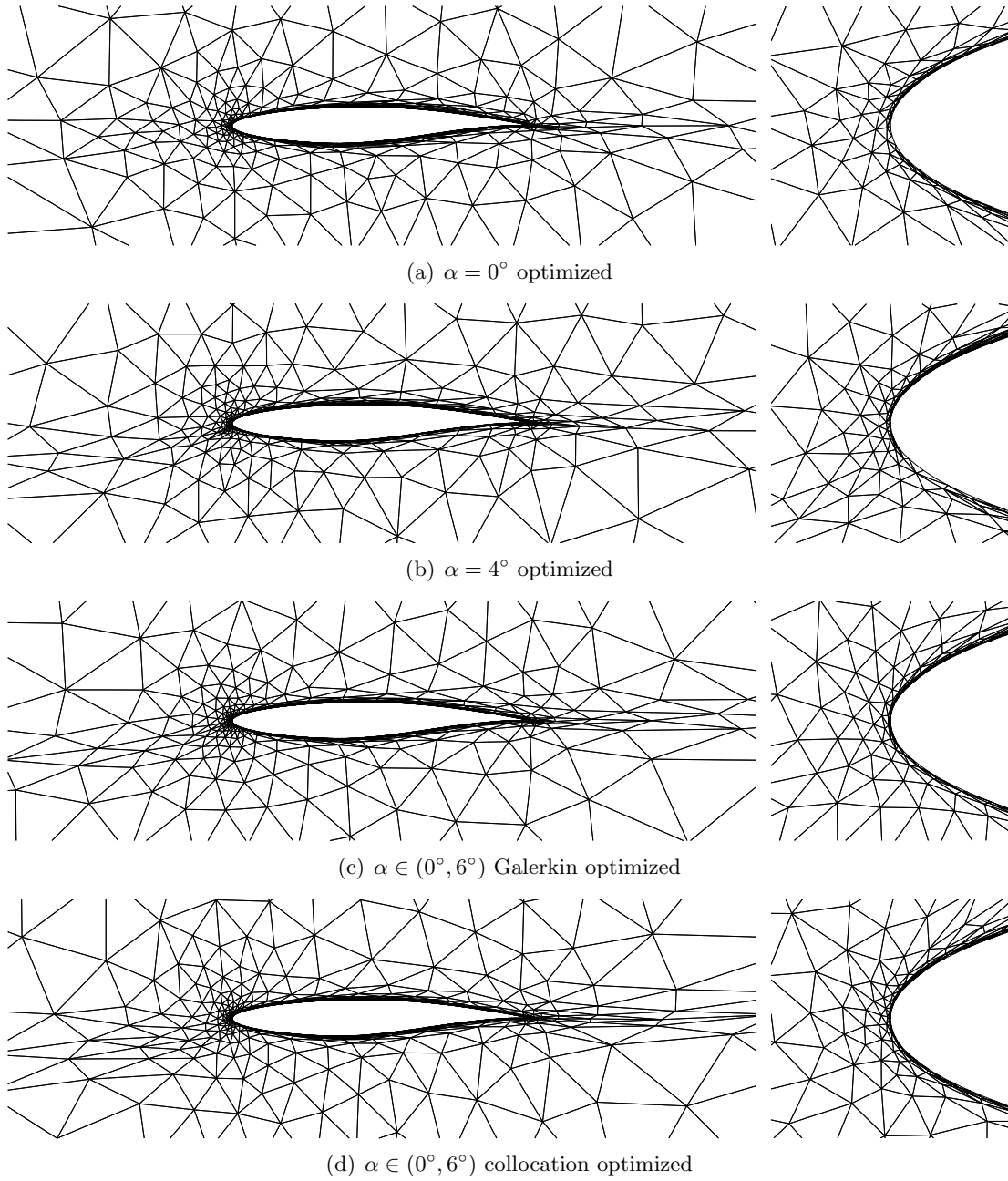


Figure 8-4: Optimized meshes for the RAE 2822 subsonic RANS case. Overview (left) and zoom of the leading edge region  $[-0.03c, 0.03c]^2$  (right).



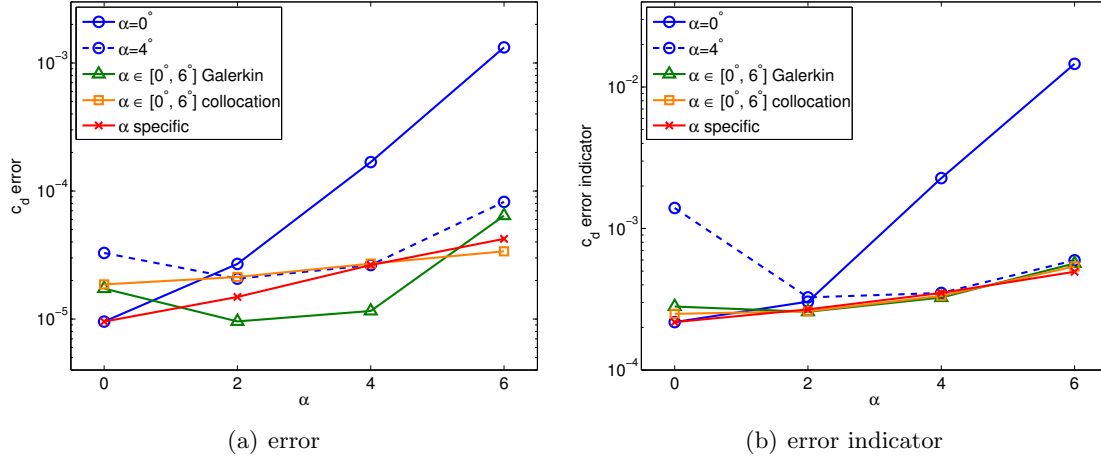


Figure 8-5: Variation in the  $c_d$  error for the RAE 2822 case over  $\alpha \in [0^\circ, 6^\circ]$  using  $\alpha = 0^\circ$ ,  $\alpha = 4^\circ$ , and  $\alpha \in [0^\circ, 6^\circ]$  optimized meshes.

achieves low error and error estimate for  $\alpha = 0^\circ$ ; however, the error grows exponentially with the increase in the angle of attack. The  $\alpha = 4^\circ$ -adapted mesh performs well in practice; however, the increase in the error estimate for the  $\alpha = 0^\circ$  configuration suggests that there may be relevant features in the flow that are not accurately resolved; the accurate output prediction may be due to cancellation of errors from several underresolved features.

A key feature that limits the performance of the  $\alpha$ -specific meshes in off-design conditions is the aforementioned lack of the boundary layer resolution in the leading edge region. In order to accurately compute  $c_d$  at a high angle of attack, the acceleration over the leading edge must be captured accurately. The zoomed in view of Figure 8-4(a) shows that the  $\alpha = 0^\circ$  adapted mesh in particular lacks this leading edge resolution, resulting in an inaccurate drag calculation at a high angle of attack. Thus, even for this simple isolated airfoil case, a subtle difference in the mesh can make a large difference in the quality of the output prediction.

Both Galerkin- and collocation-based parameter-range adapted meshes perform well over the entire range of parameter considered. In fact, for this simple isolated airfoil case, the quality of the output prediction is just as good as those obtained on  $\alpha$ -specific optimized meshes. Note that, for the collocation formulation, the four points used for error assessment are different from the three collocation points used for adaptation. The low errors obtained at the assessment points suggest that the three collocation points sufficiently characterizes the flow behavior over the entire parameter range for the purpose of constructing an efficient



universal mesh.

### 8.4.2 Three-Element MDA High-Lift Airfoil RANS-SA

We consider turbulent flow over a three-element McDonnell Douglas Aerospace (MDA) high-lift airfoil (30P-30N) [89]. The freestream Mach number is  $M_\infty = 0.2$ , the Reynolds number based on the retracted chord is  $Re_c = 9 \times 10^6$ , and the angle of attack varies from  $0^\circ$  to  $24^\circ$ . In order to minimize the finite boundary effect on the force coefficients for this high-lift configuration [8], the farfield boundary is placed  $30000c$  away from the airfoil. Select flow fields at  $\alpha = 8^\circ$  and  $\alpha = 24^\circ$  are shown in Figure 8-6, which depict considerable change in the flow field with the angle of attack. At  $\alpha = 8^\circ$ , the flow is subsonic and there is a region of large separation behind the slat. At  $\alpha = 24^\circ$ , the flow becomes transonic, forming a shock on the suction side of the slat; there is also a region of large separation in the wake.

We again consider the space-parameter Galerkin formulation and the space-Galerkin parameter-collocation formulation. Specifically, the parameter space is discretized by

- **Galerkin:**  $s = 3$  polynomial expansion with 7-point Gaussian quadrature
- **Collocation:** quadrature points of the 7-point Gaussian quadrature rule

The Galerkin formulation uses a relatively low degree polynomial expansion in the parameter space due to computational resource available. Figure 8-7 shows the fields of parameter expansion coefficients of select few components for the Galerkin formulation.

As a comparison, seven  $\alpha$ -specific optimized meshes are generated, where the  $\alpha$  ranges from  $0^\circ$  to  $24^\circ$  in  $4^\circ$  increments. To generate the  $\alpha$ -specific adaptive meshes, MOESS algorithm is first applied to the  $\alpha = 8^\circ$  case, transitioning from the initial mesh consisting of 2343 elements shown in Figure 8-8(a) to the  $8^\circ$ -optimized mesh. Then, to generate a mesh optimized for different angles of attack, adaptation is performed at each angle of attack using the mesh optimized for the previous angle of attack as the starting mesh. For instance, to generate a  $12^\circ$ -optimized mesh, the adaptation starting from the  $8^\circ$ -optimized mesh. As the flow features do not change significantly from one angle of attack to the next, only a few adaptation iterations are necessary to generate the optimal mesh at the new angle of attack. Repeating the process for all angles of attack results in an efficient generation



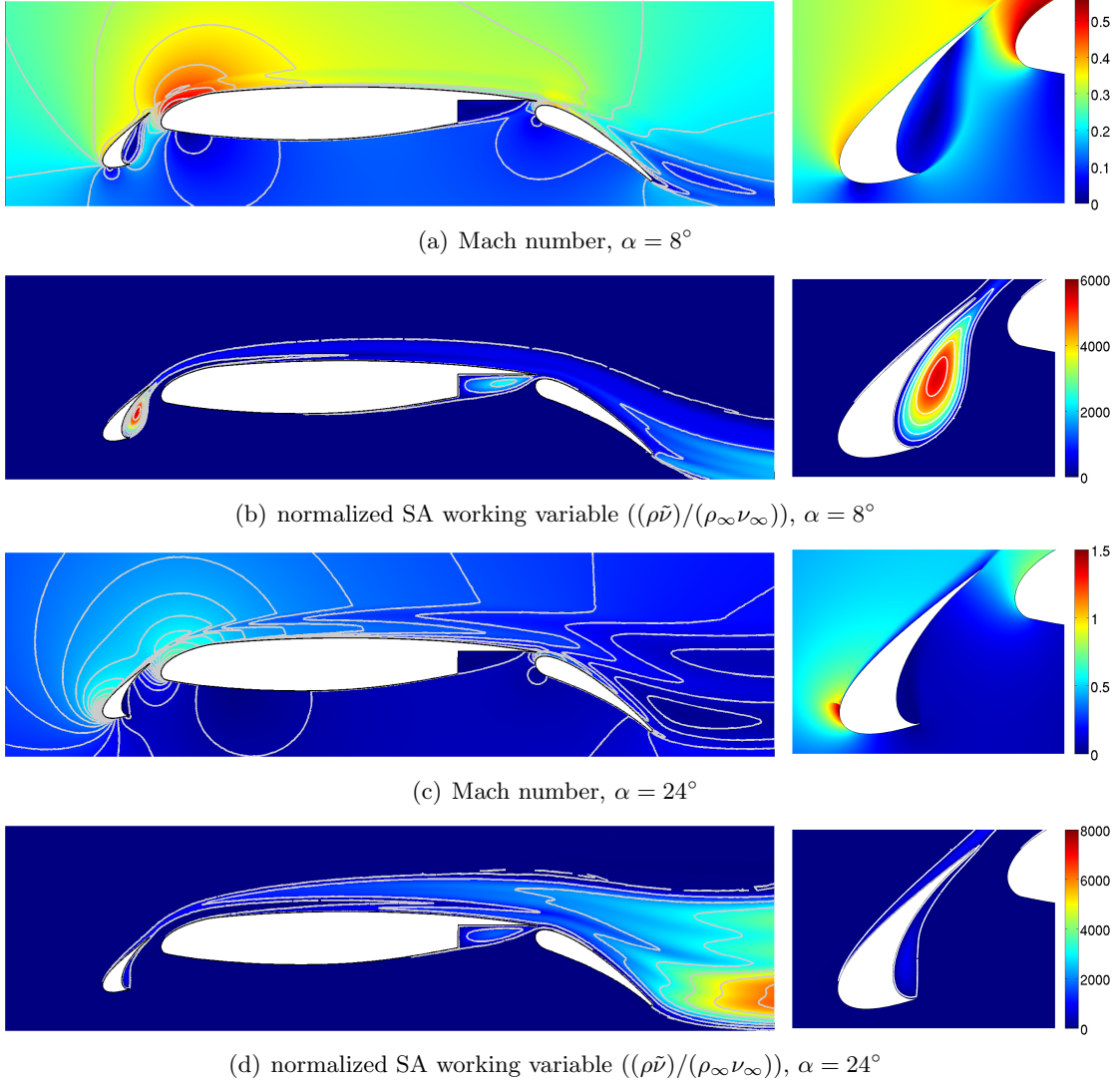


Figure 8-6: The Mach number and normalized SA working variable for the three-element MDA airfoil case at  $\alpha = 8^\circ$  and  $\alpha = 24^\circ$ .



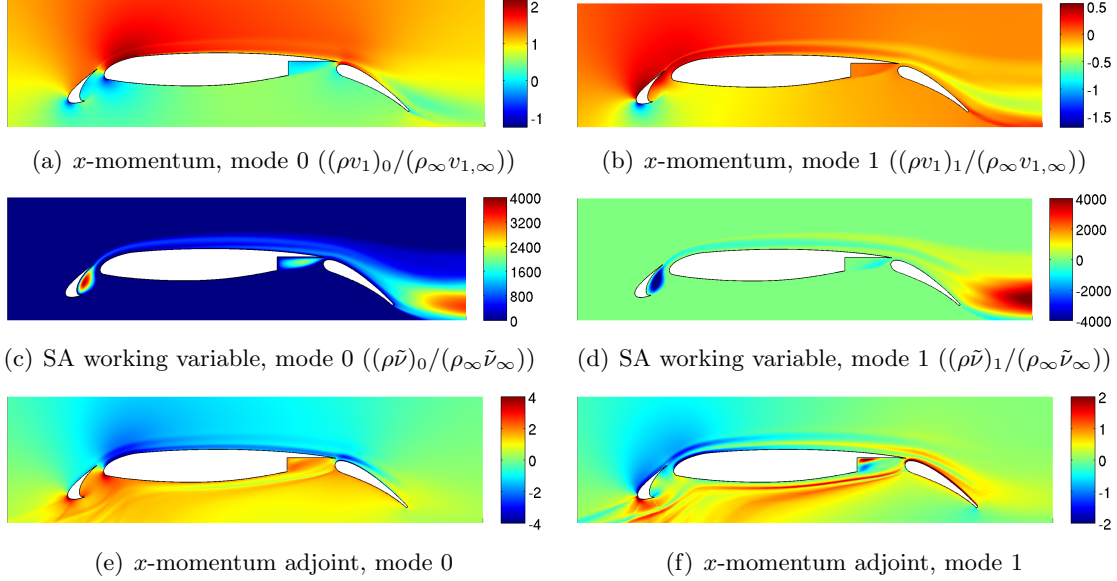


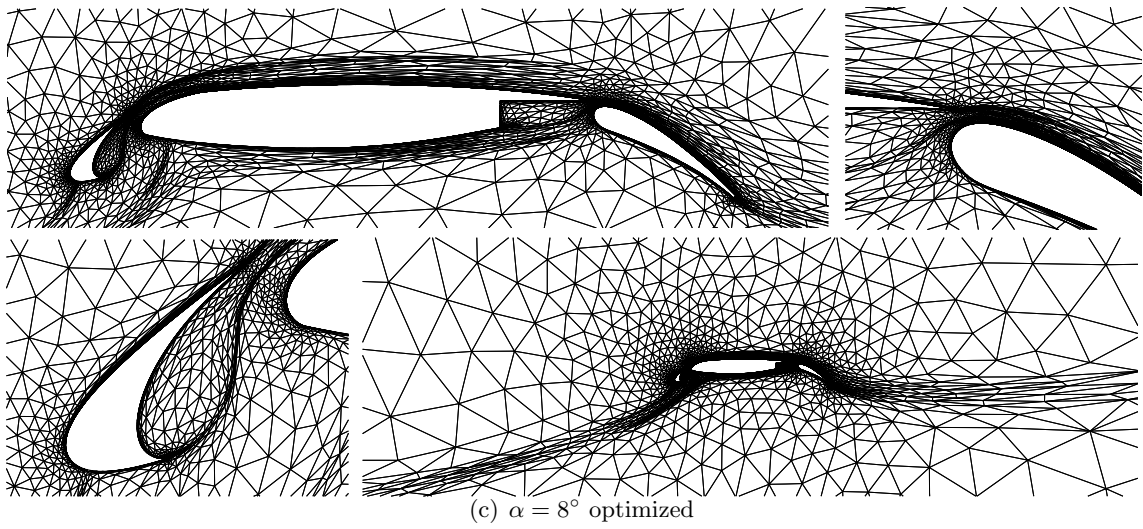
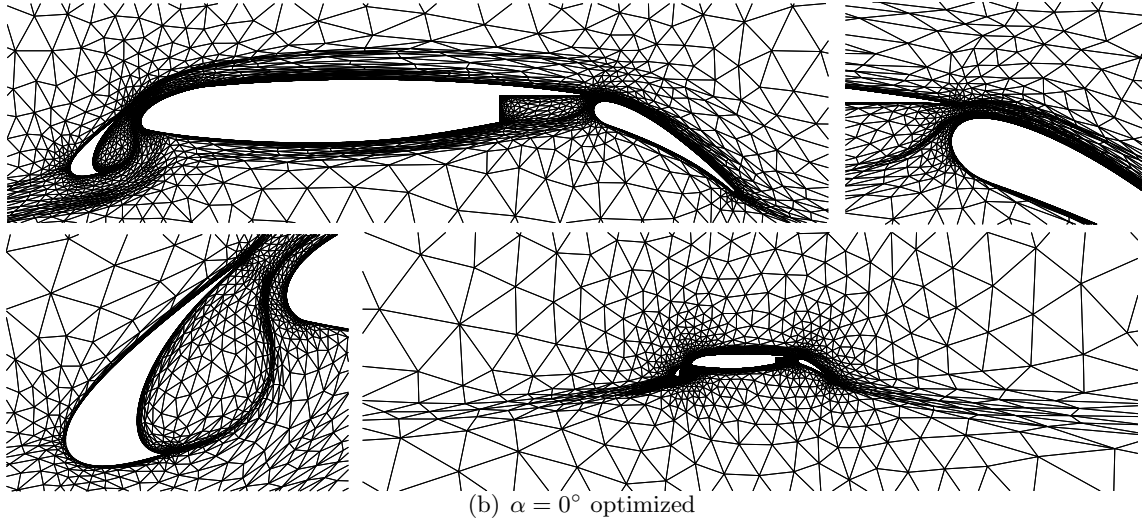
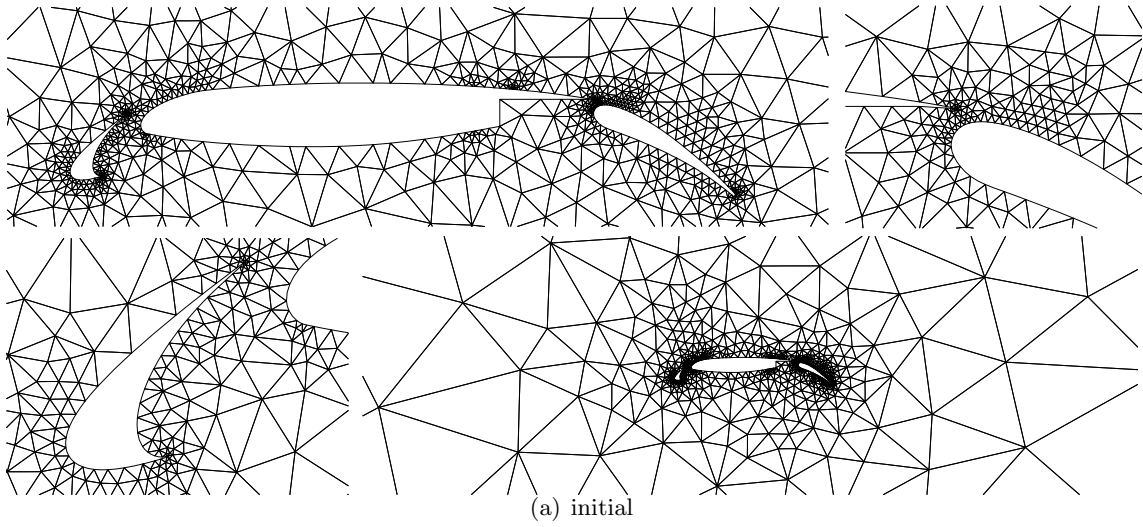
Figure 8-7: Parameter expansion mode strengths of the first two modes for select solution fields of the three-element MDA airfoil case.

of all angle-specific meshes. All results are obtained using the  $p = 2$  DG discretization at approximately 90,000 degrees of freedom.

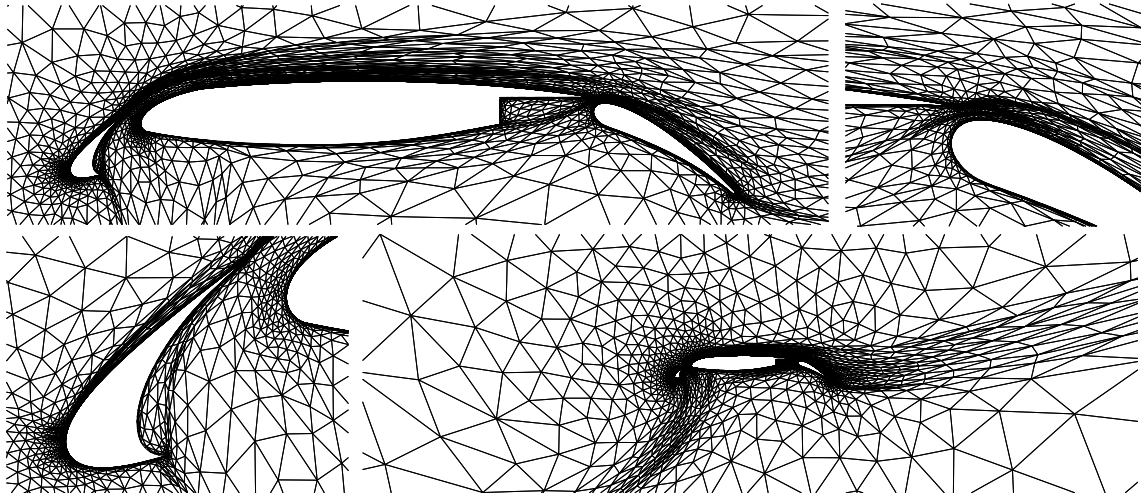
Figures 8-8(b)-8-8(d) show the adapted meshes for select angles of attack. Near the body, all optimized meshes employ highly anisotropic elements to resolve the high Reynolds number boundary layers. On the other hand, the adaptation algorithm targets different off-body features depending on the angle of attack to capture the interaction among the three airfoil elements. At lower angles of attack, the flow separates from the back side of the slat, and the wake must be captured to account for its influence on the main element. At  $\alpha = 24^\circ$ , capturing the acceleration over the front side of the slat, the resulting shock, and the flow separation behind the shock becomes important to accurately calculate the drag. In particular, the flow becomes transonic in the front side of the slat for  $\alpha \geq 20$ ; Figure 8-8(d) shows that the  $\alpha = 24^\circ$ -optimized mesh is refined for this shock. The importance of resolving this shock for accurate calculation of drag will be demonstrated shortly.

Figure 8-8(e) shows the mesh optimized for the mean drag over  $\alpha \in [0^\circ, 24^\circ]$  using the  $s = 3$  Galerkin formulation. Adaptation targets all features important in this parameter range. For instance, the farfield view shows that a sweep of stagnation streamlines and the wakes are resolved. The entire region behind the slat is also refined to track the wake that moves with the angle of attack. Note that this mesh is generated as a consequence of

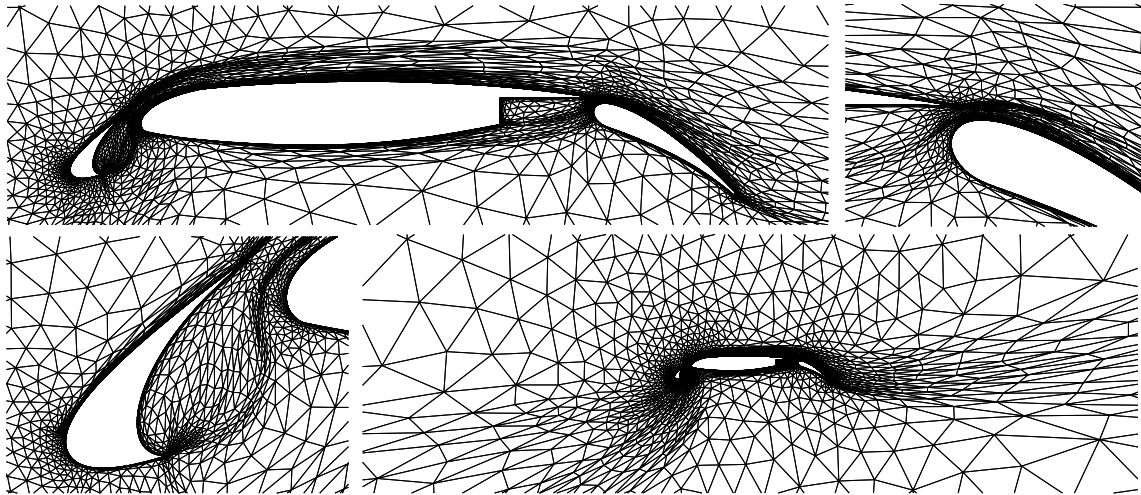




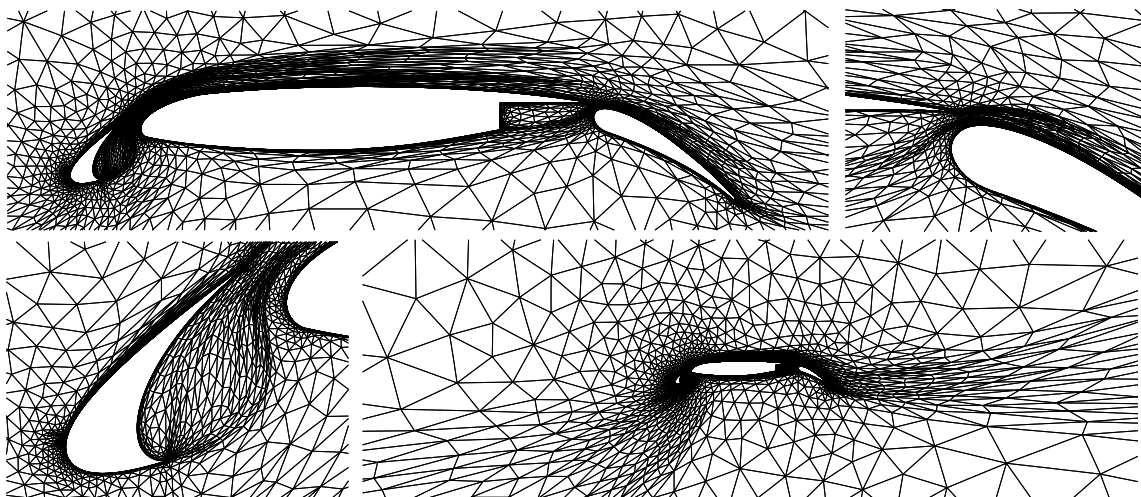




(d)  $\alpha = 24^\circ$  optimized



(e)  $\alpha \in (0^\circ, 24^\circ)$  Galerkin optimized



(f)  $\alpha \in (0^\circ, 24^\circ)$  collocation optimized

Figure 8-8: Select optimized meshes for the three-element MDA airfoil case. ( $p = 2$ ,  $\text{dof} = 90,000$ )



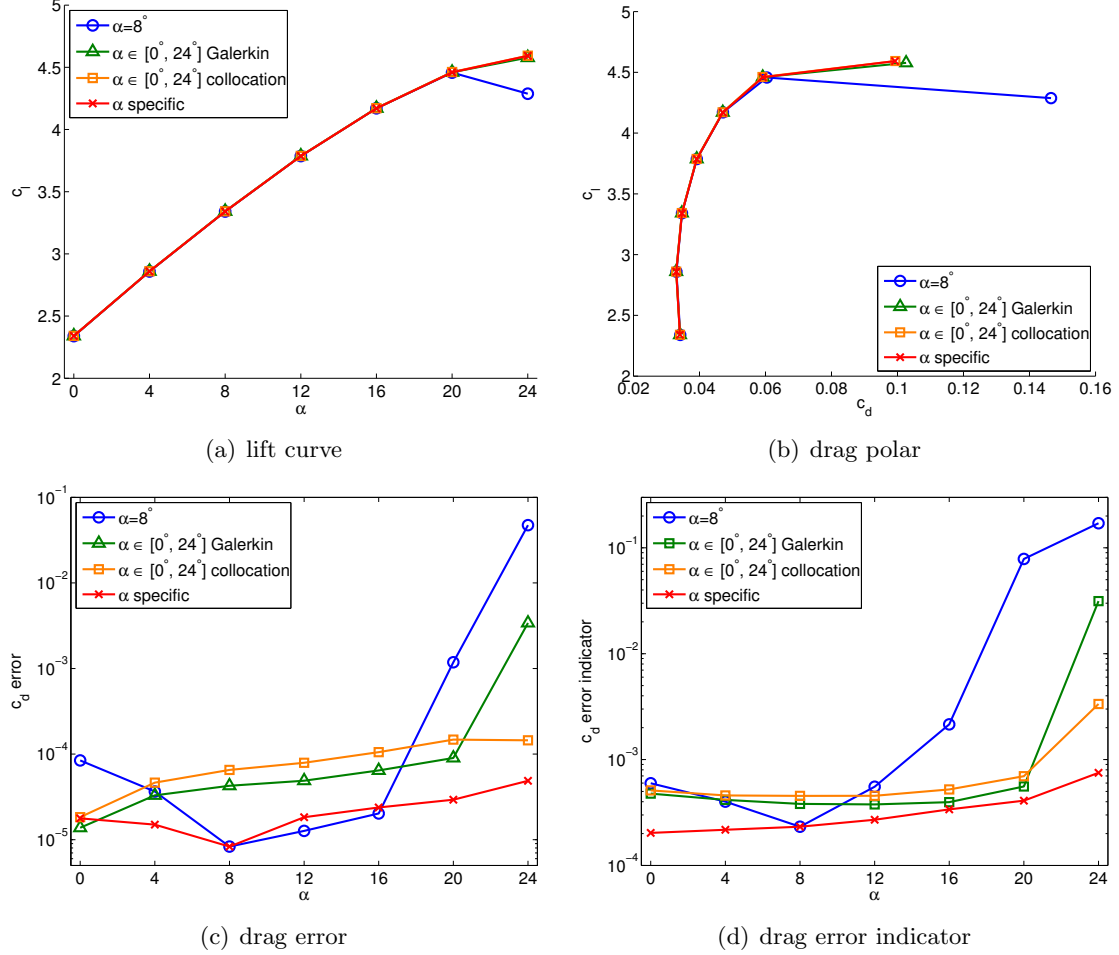


Figure 8-9: The lift curve, drag polar, drag error, and drag error indicator for the three-element MDA case.

trying to control the mean drag error. The error is governed by parameter expansion mode strengths of the primal and dual solutions, some of which are shown in Figure 8-7.

Figure 8-8(f) shows the mesh optimized for  $\alpha \in [0^\circ, 24^\circ]$  using the 7-point collocation formulation. The universal mesh is similar to the mesh adapted for the parameter-mean drag using the Galerkin formulation, targeting all features important in the parameter range. All optimized meshes clearly show that MOESS takes full advantage of the arbitrary oriented anisotropy delivered by using simplex elements to resolve off body features.

Figure 8-9 shows the lift curves, drag polars, drag error, and the drag error indicator obtained using a few different approaches. The first approach ( $\alpha = 8^\circ$ ) uses the  $8^\circ$ -optimized mesh for the entire parameter range. The second approach ( $\alpha \in [0^\circ, 24^\circ]$  Galerkin) uses the mean-drag adapted mesh shown in Figure 8-8(e). The third approach ( $\alpha \in [0^\circ, 24^\circ]$



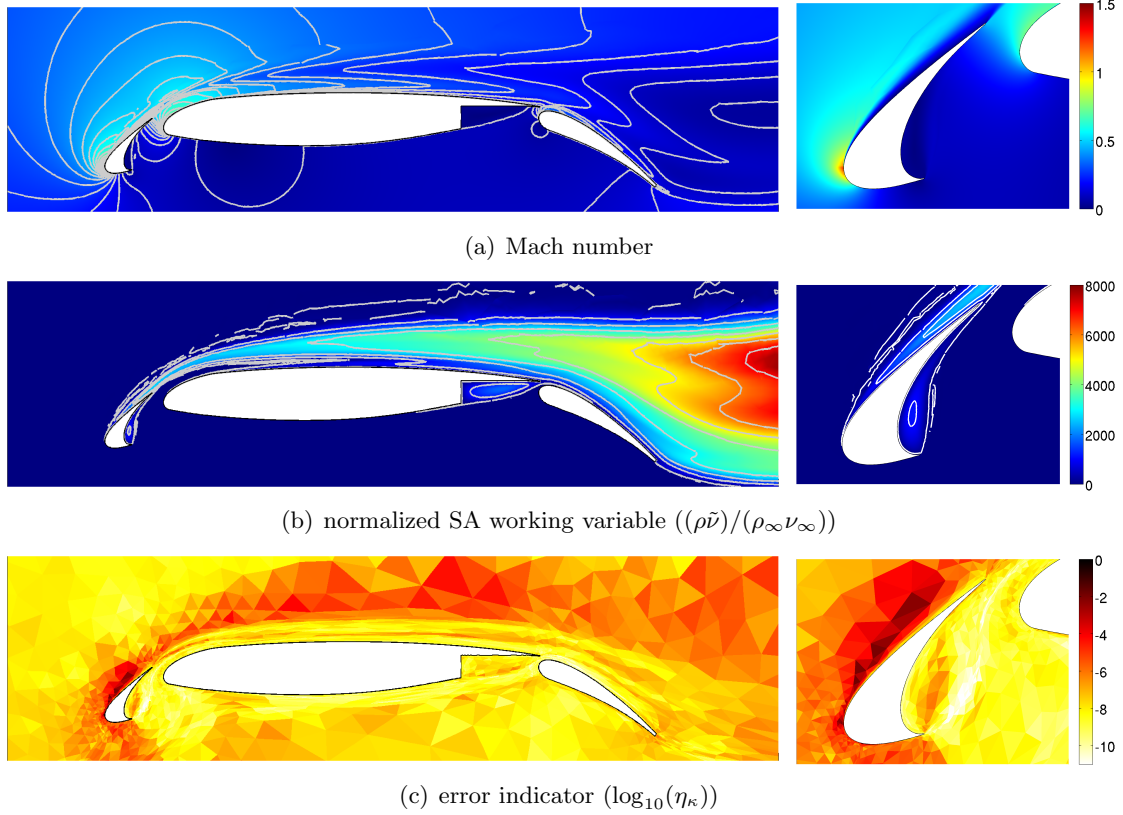


Figure 8-10: The Mach number and normalized SA working variable for the  $\alpha = 24^\circ$  flow computed on the  $\alpha = 8^\circ$  optimized mesh.

collocation) uses the parameter-range adapted mesh shown in Figure 8-8(f). The fourth approach ( $\alpha$  specific) uses a series of adapted meshes, each adapted to the specific angle of attack. Note that a combination of the higher-order discretization ( $p = 2$ ) and  $\alpha$ -specific mesh adaptation maintains the drag error of less than 0.5 counts over the entire range of the angles of attack using only 90,000 degrees of freedom.

As shown in Figure 8-9(c), the  $p = 2$  discretization on the  $8^\circ$  optimized mesh attains less than 1 drag count of error for  $\alpha \in [0^\circ, 16^\circ]$ , but the error grows exponentially with the angle of attack for  $\alpha > 16^\circ$ . In fact, at  $\alpha = 24^\circ$ , the fixed mesh commits a drag error of approximately 500 counts — a relative drag error of approximately 50%. The cause of the error is attributed to a massive artificial separation induced by insufficient mesh resolution on the front side of the slat, as shown in Figure 8-10 (c.f. the  $\alpha = 24^\circ$  reference flow field shown in Figure 8-6). Fortunately, the error indicator correctly identifies not only the lack of confidence in the drag prediction, as shown Figure 8-9(d), but also the regions causing large error, as shown in Figure 8-10(c). The case highlights that a high-order discretization



alone is insufficient to resolve all features present in this complex, multi-element airfoil case.

As shown in Figure 8-9(c), the  $p = 2$  discretization on the parameter-range adapted mesh obtained using the collocation formulation achieves less than 1 drag count of error for all but the  $\alpha = 20^\circ$  case, where the drag error is approximately 1.3 counts. In general, the drag error is 2 to 5 times larger than those obtained on the  $\alpha$ -specific optimized meshes using the same number of degrees of freedom. The complex,  $\alpha$ -dependent interaction of the three airfoil elements makes the construction of a single mesh that works well over the parameter range more challenging than the isolated airfoil case considered in Section 8.4.1. Nevertheless, the drag prediction obtained on the range-optimize mesh is a significant improvement compared to the  $8^\circ$ -optimized mesh.

Figure 8-9(c) shows that the mesh optimized for the mean drag using the Galerkin formulation works well for  $\alpha \in [0^\circ, 20^\circ]$  but is ill-suited for  $\alpha = 24^\circ$ . The large error incurred for the  $\alpha = 24^\circ$  flow is likely due to the low parameter expansion degree of  $s = 3$  employed for the Galerkin projection. Due to the low degree projection, the space-parameter Galerkin system is incapable of capturing the nonlinear parameter dependency, particularly in the front side of the slat. As a result, the mean-drag-adapted mesh for the  $s = 3$  expansion does not provide sufficient resolution in these regions with a strong nonlinear parameter dependence.

It is also worth noting that the  $\alpha$ -specific adaptive  $p = 1$  discretization achieves 3 to 8 drag counts of error — or 15 to 50 times the error obtained using the  $\alpha$ -specific  $p = 2$  discretization with the same number of degrees of freedom. In other words, assuming the asymptotic behavior and the error convergence of  $\mathcal{E} \sim h^{2p} \sim (\text{dof})^{-1}$ , the adaptive  $p = 1$  discretization requires approximately 15 to 50 times more degrees of freedom to achieve the same error level as the adaptive  $p = 2$  discretization. The ability of the high-order discretization to effectively resolve the boundary layer and capture the interaction between the three airfoil elements is a significant advantage over the  $p = 1$  discretization for this high-lift case.

## 8.5 Conclusions

This chapter considered development of an efficient finite-element-based PDE solver that can serve as a backbone of polynomial chaos and reduced order modeling, two technique designed



to accelerate input-output characterization of parametrized PDEs. Using both the space-parameter Galerkin formulation and the space-Galerkin parameter-collocation formulation, MOESS algorithm generated meshes suitable for a wide range of parameters. By casting the adaptation problem as a minimization problem of the error over the space-parameter domain, the versatile adaptive framework enabled straightforward implementation of the spatial error control for parametrized PDEs.

We considered two RANS problems to test the behavior of MOESS applied to the parametrized PDEs and to assess the quality of universal meshes designed for the entire range of parameters. For the isolated airfoil case, both the Galerkin and collocation formulations produced a mesh that works well over the parameter range. In particular, the space-parameter Galerkin formulation generated a  $p = 2$ -spatial  $s = 4$ -parameter mesh that would allow a rapid characterization of  $c_d$  as a function of  $\alpha$ , achieving less than 1 drag count of error for any parameter value. For the multi-element high-lift airfoil case, the collocation formulation generated an efficient finite-element mesh for the angle of attack varying from  $0^\circ$  to  $24^\circ$ . This case also reemphasized the benefit of combining high-order discretizations with mesh adaptivity. Performing parameter-sweep using the  $p = 2$  discretization on a single- $\alpha$ -specific mesh resulted in an inaccurate drag prediction for off-design configurations. Conversely, the adaptive  $p = 1$  discretization was significantly less efficient than the adaptive  $p = 2$  discretization.

While only the spatial adaptivity was considered in this work, a more efficient parametrized PDE solver may be constructed by performing adaptation in the parameter space. In particular, the error control framework of MOESS can be directly applied to the space-parameter Galerkin formulation to enable space-parameter adaptation. One simple extension is to maintain the current tensor product structure of the physical and parameter spaces, but to perform DWR error estimation in both spaces and enrich the space that makes a larger contribution to reducing the error (c.f. [102]). Another approach is to discretize the parameter space using multiple elements (i.e. multi-element polynomial chaos), and use the direct sampling technique to perform anisotropic adaptation in the parameter space. A more sophisticated approach may be to forgo the tensor product structure of the space-parameter space, and use different parameter expansion degree for different spatial elements, i.e. local  $s$ -adaptation in the physical space.







## Chapter 9

# Conclusions

### 9.1 Summary and Conclusions

This thesis presents work toward development of a versatile, adaptive, high-order PDE solver that reliably predicts an engineering output of interest in a fully-automated manner. In particular, we developed an adaptation algorithm, Mesh Optimization via Error Sampling and Synthesis (MOESS). Using the continuous mesh framework, the original intractable optimization problem on the discrete mesh has been relaxed to yield a well-posed optimization problem on a continuous metric field. In the process, we extend the original continuous mesh framework for linear polynomials to arbitrary-degree polynomials. Then, we devised a method for estimating the error functional of the continuous optimization problem. The key strategy used to estimate the error functional is to directly monitor the behavior of the error by solving local problems on anisotropically refined simplices. The strategy eliminates the need to model the true underlying dependencies of the error on the metric configuration, a formidable task that requires estimation of the solution regularity, higher derivatives, and mean-value linearized equation coefficients. An anisotropic error model was developed by incorporating the affine-invariant measure of the metric tensors and by synthesizing the sampled metric-error pairs, yielding a model that is entirely based on the behavior of the *a posteriori* error estimate. Finally, an optimization procedure to solve the surrogate minimization problem based on the proposed error kernel was developed. MOESS offers a number of benefits including: works with any localizable error estimate; handles any discretization order; accounts for both primal and dual solution behaviors; permits arbitrarily oriented anisotropic elements; delivers superior robustness by eliminat-



ing *a priori* error convergence assumptions; and inherits the versatility of the underlying discretization and error estimate.

We demonstrated the versatility and effectiveness of MOESS through various applications. First, the ability of the adaptation framework to produce optimal meshes was verified in the context of  $L^2$  projection error control; to enable the verification, we also derived the optimal anisotropic element size distribution for a few canonical problems using a combination of the continuous relaxation of the anisotropic approximation theory for arbitrary-degree polynomials and calculus of variations. Then, the framework was applied to the advection-diffusion equation and a series of aerodynamic flow problems. The results highlighted the importance of considering both the primal and dual solutions in choosing the elemental anisotropy. In particular, even for problems with an anisotropic primal solution, primal-based anisotropy detection may perform worse than isotropic refinement. Moreover, the appropriate anisotropy is highly  $p$ -dependent. MOESS deduced the appropriate anisotropy for each case, as the error samples implicitly incorporate both the primal and dual solution behaviors. The results also confirmed the ability of the adaptation framework to realize the full-potential of high-order discretizations for practical aerospace problems. In particular, for problems with limited regularity, MOESS offered superior robustness and effectiveness compared to other state-of-the-art adaptive higher-order methods.

Taking advantage of the versatility of the algorithm, we considered adaptation for space-time systems and space-parameter systems. In particular, we realized fully-unstructured space-time adaptivity for linear and nonlinear wave propagation problems. The results demonstrated that the additional computational work required to solve the unified space-time system can be significantly reduced by using space-time anisotropy, which can effectively reduce the dimensionality of the problem. For space-parameter systems, MOESS enabled spatial error control for Galerkin- and collocation-based parameter-space discretizations. The spatial adaptivity facilitates application of polynomial chaos or reduced order modeling to complex, multiscale problems. Combined with MOESS both methods produced universal optimal meshes suitable for a wide range of parameters. The promising results obtained in various examples in this thesis suggest that the adaptation framework is a positive step forward in developing a fully-automated, reliable PDE solver that produces accurate prediction of the performance variables for a wide range of engineering and scientific applications.



## 9.2 Future Work

During the course of this work, we have identified several areas of future research.

- **Extension of the continuous optimization framework to  $hp$ -adaptation**

A natural extension of the current work on anisotropic  $h$ -adaptation is anisotropic  $hp$ -adaptation. The potential of  $hp$ -adaptivity to deliver optimal finite element meshes has been discussed for decades, and its theoretical approximation properties are summarized in, for example, [130], and the references therein. More recently, Georgoulis *et al.* [65] has combined their quadrilateral-based hierarchical subdivision strategy with the regularity estimate developed by Houston and Süli [75] to devise an anisotropic  $hp$ -adaptation strategy for quadrilateral meshes. Leicht and Hartmann have applied a similar strategy to variety of aerodynamic problems in two- and three-dimensions [91]. Their results suggest that the anisotropic  $hp$ -adaptation outperforms their quadrilateral-based anisotropic  $h$ -adaptation, especially for high-fidelity applications.

Several questions must be answered to enable  $hp$ -adaptivity within our simplex-based adaptation framework. First, just as a triangulation has been relaxed to continuous metric field, the concept of element-wise  $p$ -field must be relaxed to yield a continuous  $p$ -field, and rules relating the discrete and continuous fields must be established. Second, the error model must be modified to incorporate the error variation with  $p$ . In particular, sampling in the  $p$ -enriched space is insufficient to fully characterize the behavior with the  $p$ -change; this is due to the fact that a single step of  $p$ -refinement performs better than uniform  $h$ -refinement even for irregular solutions, as proved in [12]. Third, the purpose of  $h$ -refinement in the  $hp$ -context is the containment of singularity effects rather than reducing the error [3], and this interpretation of  $h$ -adaptivity is only implicitly reflected in our current error model.

Most importantly, we must carefully study benefits of  $hp$ -adaptation for fully-unstructured simplex meshes. The efficiency gain, in terms of errors-per-dof, may be smaller than that observed for quadrilateral-based meshes which have fewer spatial adaptation options. A more practical gain may be the improved robustness obtained through using a lower-order discretization in regions with low regularity, especially for nonlinear features such as shocks.



- **Minimizing the degrees of freedom or time of computation for a given error tolerance**

In the current optimization framework, the objective was defined as minimizing the output error for a given number of degrees of freedom. In a practical engineering setting, however, engineers may be more interested in obtaining a solution of a given error tolerance using the least computational effort. One measure of the computational effort is the degrees of freedom. Solving the minimum-dof error-constrained problem is more challenging than the optimization problem considered in this work, as the continuous error model is less accurate than the cost model, making it difficult to impose the constraint. In particular, the error model differs from the cost model in that 1) the error model suffers from inherent noise in the error estimate, and 2) the quality of the error prediction degrades for a large configuration change outside of the sampled configurations. A more robust error estimate may be necessary to successfully solve the minimum-dof error constrained problem. Another measure of computational effort is the computational time. In order to solve the minimum-time, tolerance-constrained problem, a means of estimating the solution time must be developed. In the adaptive context, this may be accomplished using the time history of all previous runs, assuming the solver is sufficiently robust. Then, the current error model may be combined with the time model to minimize the computational time.

- **Improving the robustness of the error estimate**

While the DWR error estimate was sufficiently robust to drive adaptation for a wide range of PDEs considered in this work, the error estimate can significantly underestimate the error on coarse meshes in which solution features are completely under-resolved. While constructing the true bounds — as done for coercive equations in [128, 129] — may not be feasible for general PDEs, improved robustness is highly desired. A recent work on the safeguarded DWR method [108], which augments the standard DWR with a residual-based error estimate, is also only applicable to a limited number of PDEs. One practical approach to improving the robustness of the error estimate within the current sampling-based adaptation framework may be to incorporate the error information gathered in the sampling stage to the error estimate itself. Because the sampling is done on a more refined mesh, this may remedy the underestimation of the error due to the lack of resolution.



- **Higher-order metric field representation and natively curved meshing**

In the current implementation of the continuous optimization framework, the Riemannian metric field was represented as a piecewise linear function on the triangulation. The linear approximation could limit the accuracy of the metric field representation, especially on high aspect-ratio, curved elements. Using a higher-order metric field representation could remedy this problem. However, the current two-step strategy to curved mesh generation — the initial linear mesh generation followed by mesh curving — is insufficient to realize the full benefit of such a higher-order metric representation. A mesh generator capable of generating natively curved elements must be developed.

- **Space-time adaptivity in higher dimensions**

Future work required to make the fully-unstructured space-time adaptivity truly competitive for real world applications has been discussed in Section 7.6. To summarize, first, an efficiency preconditioner that takes advantage of the hyperbolicity of the problem in the temporal direction must be designed. Second, a mesh generator for  $(3 + 1)d$  unstructured space-time meshes must be developed.

- **Space-parameter adaptivity**

Future work to improve the efficiency of the space-parameter discretization has been discussed in Section 8.5. Recommendations include: adaptation of the parameter space by using multiple parameter elements; and unstructured space-parameter adaptation by employing spatially varying  $s$ -adaptivity for the parameter representation.







## Appendix A

# Discontinuous Galerkin Method

This appendix provides details of the discontinuous Galerkin discretization, the solution technique for the resulting discrete nonlinear system, and the output evaluation procedure used in this thesis. Throughout this appendix, we consider a general conservation law of the form

$$\frac{\partial u}{\partial t} + \nabla \cdot \mathcal{F}^{\text{conv}}(u, x, t) - \nabla \cdot \mathcal{F}^{\text{diff}}(u, \nabla u, x, t) = \mathcal{S}(u, \nabla u, x, t) \quad \forall x \in \Omega, t \in I \equiv [t_0, t_f],$$

with the boundary conditions

$$\mathcal{B}(u, \hat{n} \cdot \mathcal{F}^{\text{diff}}(u, \nabla u, x, t), x, t; \text{BC}) = 0, \quad \forall x \in \partial\Omega, t \in I.$$

The diffusive flux is assumed to be a linear function of  $\nabla u(x, t)$ , such that it can be expressed as

$$\mathcal{F}^{\text{diff}}(u, \nabla u, x, t) = \mathcal{K}(u, x, t) \nabla u(x, t),$$

where  $\mathcal{K}$  is the viscosity tensor.



## A.1 Discontinuous Galerkin Discretization

Recall that the weak form associated with the DG approximation of the conservation law is: Find  $u_{h,p} \in V_{h,p}$  such that

$$\mathcal{R}_{h,p}(u_{h,p}, v_{h,p}) = 0, \quad \forall v_{h,p} \in V_{h,p}. \quad (\text{A.1})$$

The semilinear form associated with the spatial residual,  $\mathcal{R}_{h,p} : V_{h,p} \times V_{h,p} \rightarrow \mathbb{R}$ , consists of the convective, diffusive, and source contributions, and may be written as

$$\mathcal{R}_{h,p}(w_{h,p}, v_{h,p}) = \mathcal{R}_{h,p}^{\text{conv}}(w_{h,p}, v_{h,p}) + \mathcal{R}_{h,p}^{\text{diff}}(w_{h,p}, v_{h,p}) + \mathcal{R}_{h,p}^{\text{sour}}(w_{h,p}, v_{h,p}).$$

For notational simplicity, the dependency of the flux and source functions on  $x$  and  $t$  are implied and are not explicitly stated.

The DG discretization of the convective term is given by

$$\begin{aligned} \mathcal{R}_{h,p}^{\text{conv}}(w_{h,p}, v_{h,p}) = & - \sum_{\kappa \in \mathcal{T}_h} \int_{\kappa} \nabla v_{h,p}^T \cdot \mathcal{F}(w_{h,p}) dx + \sum_{f \in \Gamma_i} \int_f (v_{h,p}^+ - v_{h,p}^-)^T \mathcal{H}(w_{h,p}^+, w_{h,p}^-; \hat{n}^+) ds \\ & + \sum_{f \in \Gamma_b} \int_f v_{h,p}^+{}^T \mathcal{H}^b(w_{h,p}^+, u_b(w_{h,p}^+; \text{BC}); \hat{n}^+) ds, \end{aligned}$$

where  $(\cdot)^+$  and  $(\cdot)^-$  denote trace values on the opposite sides of a face  $f$ ,  $\hat{n}^+$  is the normal vector pointing from  $+$  to  $-$ . By convention, the interior side is always the  $+$  side on the boundary faces.  $\mathcal{H}$  and  $\mathcal{H}^b$  are numerical flux functions on interior faces and on boundary, respectively. The boundary state,  $u_b$ , is in general a function of the interior state and the boundary conditions. In this work, the interior face numerical flux function uses the Roe's approximate Riemann solver [126] and takes the form

$$\mathcal{H}(w_{h,p}^+, w_{h,p}^-; \hat{n}^+) = \frac{1}{2} \left( \hat{n}^+ \cdot \mathcal{F}(w_{h,p}^+) + \hat{n}^- \cdot \mathcal{F}(w_{h,p}^-) \right) + \frac{1}{2} |\mathcal{A}^{\text{Roe}}(w_{h,p}^+, w_{h,p}^-; \hat{n}^+)| (w^+ - w^-),$$

where  $\mathcal{A}^{\text{Roe}}$  is the flux Jacobian matrix computed about the Roe's mean state.

The viscous terms are discretized using the second method of Bassi and Rebay (BR2) [25]. For notational convenience, let us define a jump operator,  $[[\cdot]]$ , for a scalar quantity



and a averaging operator,  $\{\cdot\}$ , for a vector quantity, i.e.

$$\llbracket s \rrbracket = s^- \hat{n}^- + s^+ \hat{n}^+ \quad \text{and} \quad \{v\} = \frac{1}{2}(v^+ + v^-)$$

The semilinear form for the diffusive term is given by

$$\begin{aligned} \mathcal{R}_{h,p}^{\text{diff}}(w_{h,p}, v_{h,p}) &= \sum_{\kappa \in \mathcal{T}_h} \int_{\kappa} \nabla v_{h,p}^T \cdot \mathcal{K}(w_{h,p}) \nabla w_{h,p} dx \\ &\quad - \sum_{f \in \Gamma_i} \int_f [\{\mathcal{K}^T(w_{h,p}) \nabla v_{h,p}\}^T \cdot \llbracket w_{h,p} \rrbracket + \llbracket v_{h,p} \rrbracket^T \cdot \{\mathcal{K}(w_{h,p}) (\nabla w_{h,p} + \eta_f r_f(\llbracket w_{h,p} \rrbracket))\}] ds \\ &\quad - \sum_{f \in \Gamma_b} \int_f \left[ (\hat{n}^+ \cdot \mathcal{K}^T(u_b) \nabla v^+)^T (w_{h,p}^+ - u_b) \right. \\ &\quad \left. + v_{h,p}^+{}^T \mathcal{F}^b \left( \hat{n}^+ \cdot (\mathcal{K}(u_b) (\nabla w_{h,p}^+ + \eta_f r_f^b((w_{h,p}^+ - u_b) \hat{n}^+))) \right); \text{BC} \right) \right] ds \end{aligned}$$

where  $\eta_f$  is the BR2 stabilization parameter, and  $r_{h,p}^f$  is the lifting operator associated with the face  $f$ . The lifting operator  $r_{h,p}^f : [V_{h,p}(f)]^d \rightarrow [V_{h,p}]^d$  is defined by

$$\sum_{\kappa \in \kappa_f} \int_{\kappa} \tau_{h,p}^T \cdot r_{h,p}^f(q_{h,p}) dx = - \int_f \{\tau_{h,p}\}^T \cdot q_{h,p} ds, \quad \forall q_{h,p}, \tau_{h,p} \in [V_{h,p}]^d,$$

where  $\kappa_f$  is the set of elements sharing the face  $f$ . The boundary flux function  $\mathcal{F}^b : \mathbb{R}^m \rightarrow \mathbb{R}^m$  takes the diffusive flux based on the interior state as the argument and returns an appropriate diffusive flux by considering the boundary condition.

The source term is discretized using the mixed form presented by Bassi *et al.* [22], which is asymptotically dual-consistent [110]. The semilinear form is given by

$$\mathcal{R}_{h,p}^{\text{sour}}(w_{h,p}, v_{h,p}) = \sum_{\kappa \in \mathcal{T}_h} \int_{\kappa} v_{h,p}^T S(w_{h,p}, \nabla w_{h,p} + r^{\text{glob}}(w_{h,p})) dx,$$

where the global lifting operator is  $r_{h,p}^{\text{glob}} : V_{h,p} \rightarrow [V_{h,p}]^d$  such that

$$\begin{aligned} \sum_{\kappa \in \mathcal{T}_h} \int_{\kappa} \tau_{h,p}^T \cdot r_{h,p}^{\text{glob}}(w_{h,p}) dx &= - \sum_{f \in \Gamma_i} \int_f \{\tau_{h,p}\}^T \cdot \llbracket w_{h,p} \rrbracket ds \\ &\quad - \sum_{f \in \Gamma_b} \int_f \tau_{h,p}^T \cdot \hat{n}(w_{h,p} - u_b) ds, \quad \forall \tau_{h,p} \in [V_{h,p}]^d. \end{aligned}$$



Because the lifting operator is linear on its argument, the global lifting operator is related to the local, face-wise lifting operator by

$$r_{h,p}^{\text{glob}}(w_{h,p}) = \sum_{f \in \Gamma_i} r_{h,p}^f(\llbracket w_{h,p} \rrbracket) + \sum_{f \in \Gamma_b} r_{h,p}^f(\hat{n}(w_{h,p} - u_b)).$$

### A.1.1 Nonlinear Discontinuity Regularization

For problems with strong discontinuities induced by a nonlinear physical mechanism (e.g. shocks in compressible flow), a nonlinear operator that detects and regularizes the discrete solution is added to improve the robustness of the solver. This work uses a jump-based indicator and PDE-based artificial diffusion smoothing introduced by Barter and Darmofal [19] with minor modifications.

The jump-based discontinuity sensor for element  $\kappa$  is given by

$$S_\kappa(w) = \log \left( \frac{1}{|\partial\kappa|} \int_{\partial\kappa} \left| \frac{g(w^+) - g(w^-)}{\frac{1}{2}(g(w^+) + g(w^-))} \right| ds \right),$$

where  $g(w)$  is a scalar quantity suitable for detecting a discontinuity and is dependent on the governing equation of interest. The sensor takes advantage of the relationship between the inter-elemental jump and the strong form of the elemental residual in the DG formulation. To prevent the addition of artificial diffusion in smooth region or addition of excess viscosity, a filter originally developed by Persson and Peraire [120] is applied to yield

$$\bar{S}_\kappa(S_\kappa) = \begin{cases} 0, & S_\kappa \leq S_0(p) - \Delta S \\ \frac{S_{\max}}{2} \left( 1 + \sin \left( \frac{\pi(S_\kappa - S_0)}{2\Delta S} \right) \right), & S_0(p) - \Delta S < S_\kappa \leq S_0 + \Delta S \\ S_{\max}, & S_0(p) + \Delta S < S_\kappa \end{cases},$$

with a polynomial-degree-dependent function  $S_0(p)$  and parameters  $\Delta S = 0.5$  and  $S_{\max} = 1$ .

The element-wise discontinuity sensor is then used as a source term of a diffusive equation which smoothly propagates the effect of discontinuity to generate an artificial diffusivity field,  $\nu_{\text{art}}$ . The artificial-diffusion PDE used in this work is a modified version of the original



equation by Barter and takes the form

$$\begin{aligned} \frac{\partial \nu_{\text{art}}}{\partial t} &= \frac{\partial}{\partial x_i} \left( \frac{\eta_{ij}}{\tau} \frac{\partial \nu_{\text{art}}}{\partial x_j} \right) + \frac{1}{\tau} \left[ \frac{\bar{h}}{p} \lambda_{\max}(u) \bar{S}_{\kappa}(u) - \nu_{\text{art}} \right] \quad \text{in } \Omega \\ \frac{\eta_{ij}}{\tau} \frac{\partial \nu_{\text{art}}}{\partial x_j} n_i &= \sqrt{C_1 C_2 \frac{p \lambda_{\max}}{h_{\min}}} (n_i n_j H_{ij}) (\nu_{\text{art},\infty} - \nu_{\text{art}}) \quad \text{on } \partial\Omega. \end{aligned} \quad (\text{A.2})$$

Here,  $H(x) = \mathcal{M}^{-1/2}(x)$  is the generalized length scale based on the metric-tensor defined in Section 2.3,  $\eta_{ij} = C_2 H_{ik} H_{kj}$  is the diffusion coefficient,  $\tau = h_{\min}/(C_1 p \lambda_{\max}(u))$  is the time scale based on the maximum wave speed,  $\lambda_{\max}(u)$ ,  $h_{\min} = \min_i \lambda_i(H)$  is the minimum (anisotropic) element size, and  $\bar{h} = (\det(H))^{1/d}$  is the volume based element size. The two constants are set to  $C_1 = 3$  and  $C_2 = 5$ . The resulting artificial diffusion field,  $\nu_{\text{art}}$ , is again filtered to completely remove artificial viscosity in the smooth regions and to cap the maximum viscosity. The final filtered artificial viscosity augments the physical viscosity of the governing equation.

Unlike Barter's original equation that used axis aligned bounding boxes to measure the local element sizes, a Riemannian metric field is used in this work to measure the local length scale for the PDE. The new formulation provides consistent propagation of artificial viscosity independent of the coordinate system and enables sharper shock capturing on highly anisotropic elements with arbitrary orientations. Effects of the modification on the solution quality and adaptation efficiency are summarized in Appendix B.

### A.1.2 Solution Method

Upon selecting suitable basis functions for the approximation space,  $V_{h,p}$ , solving Eq. (A.1) becomes a discrete, root-finding problem. The steady-state solution is obtained using a nonlinear solver based on pseudo-time continuation and backward Euler time integration. Given a discrete solution,  $U^n$ , the solution after one time step,  $U^{n+1}$ , is given by solving

$$R_t(U^{n+1}) \equiv M^{\text{tw}}(U^{n+1} - U^n) + R_s(U^{n+1}) = 0, \quad (\text{A.3})$$

where  $R_t(U)$  is the pseudo-unsteady residual,  $M^{\text{tw}}$  is the time-weighted mass matrix, and  $R_s(U)$  is the spatial residual. The  $m$ -th entry of  $R_s(U)$  is the residual evaluated against the  $m$ -th basis function,  $\phi^{(m)}$ , i.e.  $[R_s(U)]_m = \mathcal{R}_{h,p}(U^n) \phi^{(n)}, \phi^{(m)}$ . In the pseudo-time continuation algorithm, the CFL number acts as the global continuation parameter, and



different time step is assigned to each element based on the local characteristic speed and the element size. A single step of Newton's method is used to approximately solve (A.3) at each time step such that

$$U^{n+1} - U^n \approx \Delta U \equiv - \left( M^{\text{tw}} + \frac{\partial R_s}{\partial U} \Big|_{U^n} \right)^{-1} R_s(U^n). \quad (\text{A.4})$$

Computation of the state update,  $\Delta U$ , requires the solution of a large linear system with a block-sparse structure. The linear system in this work is solved with restarted GMRES [127]. In order to improve the convergence of the GMRES algorithm, the linear system is preconditioned with an in-place block-ILU(0) factorization [52] with minimum discarded fill ordering and a coarse  $p = 0$  multigrid correction [121].

The solution process is advanced in time until the 2-norm of the spatial residual,  $\| R_s(U^n) \|_2$ , is less than a specified tolerance. The robustness of the continuation procedure is further enhanced by incorporating two update limiting strategies: a physicality check and a line search over the unsteady residual,  $R_t(U)$ . The physicality check prevents a large change in select states. The line search aims to prevent the nonlinear solver divergence due to the lack of temporal integration accuracy by explicitly controlling the unsteady residual. The details of the limiting strategies are presented in [105, 156].

## A.2 Output Evaluation

This work considers a general output for a conservation law of the form

$$J = \mathcal{J}(u) = \int_{\Omega} j^i(u, \nabla u, x, t) dx + \int_{\partial\Omega} j^b(u, \hat{n} \cdot \mathcal{F}^{\text{diff}}(u, \nabla u, x, t), x, t, ) ds,$$

where  $j^i$  and  $j^b$  are functions specifying the interior and boundary contributions to the output, respectively. In the DG setting, the output is evaluated as

$$J_{h,p} = \mathcal{J}_{h,p}(u_{h,p}),$$



where the output functional,  $J_{h,p} : V_{h,p} \rightarrow \mathbb{R}$ , is given by

$$\begin{aligned} \mathcal{J}_{h,p}(w_{h,p}) &= \int_{\Omega} j^i(w_{h,p}, \nabla w_{h,p} + r^{\text{glob}}(w_{h,p}), x, t) dx \\ &\quad + \int_{\partial\Omega} j^b \left( u_b, \mathcal{F}^b \left( \hat{n}^+ \cdot (\mathcal{K}(u_b)(\nabla w_{h,p}^+ + \eta_f r_f^b((w_{h,p}^+ - u_b)\hat{n}^+))) \right); \text{BC} \right), x, t \right) ds, \end{aligned}$$

where  $r^{\text{glob}}$  is the global lifting operator,  $r_f^b$  is the local lifting operator,  $u_b$  is the boundary state function, and  $\mathcal{F}^b$  is the boundary viscous flux function as specified in Section 2.1.







## Appendix B

# Comparison of Vector- and Tensor-Based Element Sizing for the Shock PDE

This appendix compares the original shock PDE developed by Barter and Darmofal [19], which uses a vector-based element size specification, and the new formulation based on a tensor-based element size specification. The original shock PDE uses a vector-based element size based on the axis-aligned bounding box to specify the element size, i.e. the metric tensor appearing in the shock PDE, Eq. (A.2), is replaced by

$$\mathcal{M} \leftarrow \text{diag}(h_1^{-2}, \dots, h_d^{-2}),$$

where  $h_i$  is the length of the axis-aligned bounding box in the  $i$ -th direction. Section B.1 summarizes the impact of the modification on a fixed mesh, and Section B.2 summarizes its influence on adaptation.

### B.1 Comparison on a Fixed Mesh

In this section, we consider the  $p = 1$  DG discretization of  $M_\infty = \sqrt{2}$  inviscid flow over a NACA 0012 airfoil at  $0^\circ$  angle of attack. The Mach number is chosen such that the Mach angle is  $45^\circ$ . The 7136-element fixed mesh used throughout this section is shown in Figure B-1.



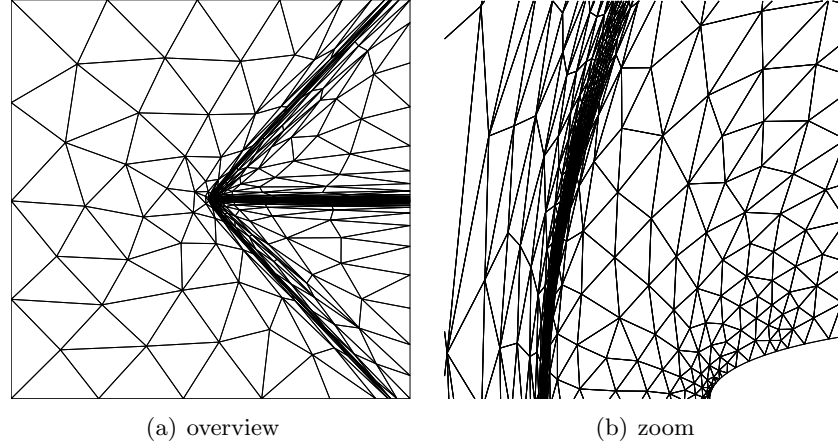


Figure B-1: The 7136-element NACA 0012 mesh used for the fixed mesh tests.

The test consists of solving the supersonic NACA problem on two meshes, the second one tilted by  $45^\circ$  with respect to the first one. The test is designed to check for the dependence of the shock capturing algorithms on the particular coordinate system. Figure B-2 shows the artificial viscosity distribution obtained using the original, vector-based element sizing and the new, tensor-based element sizing. The original shock PDE is coordinate dependent, and the artificial viscosity excessively diffuses when the shock does not align with the coordinate axes (Figure B-2(a)). The modified shock PDE is invariant under coordinate transformation, and the artificial viscosity is tightly confined in the region of the shock.

Figure B-3 compares the effect of sharper, coordinate-independent artificial viscosity distribution on the shock resolution. Figure B-3(a) shows that the original formulation produces a sharp shock along the stagnation streamline since the shock is aligned with one of the coordinate axes (i.e. the  $x_2$ -axis). However, the regularization excessively smears the shock in the curved region, as the shock PDE propagates the artificial viscosity more than necessary when the shock is not axis-aligned. The modified, tensor-based shock PDE produces sharp shock along the entire span of the curved shock. More importantly, it propagates artificial viscosity *consistently* whether the shock is curved. This means that, if the artificial viscosity is tuned for a shock of a particular orientation, it would work for shock of any orientation.



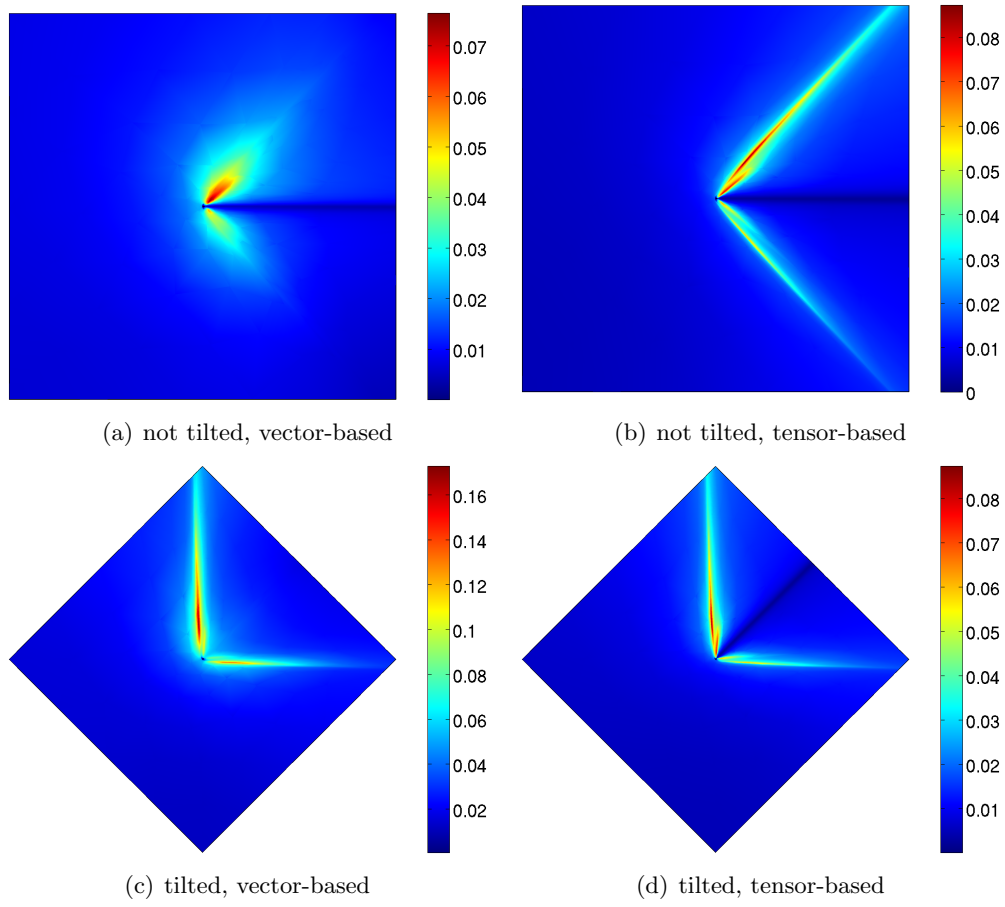


Figure B-2: The artificial viscosity,  $\epsilon$ , for the Euler problem solved at  $M_\infty = \sqrt{2}$ . The two meshes are identical, except that one of them is tilted by  $45^\circ$ .

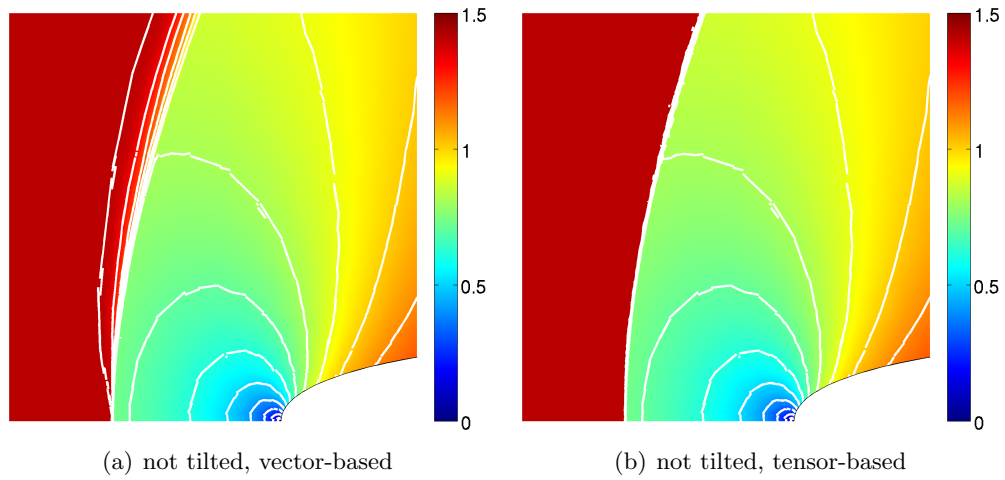


Figure B-3: The Mach number distribution on the same non-tilted meshes for the  $M_\infty = \sqrt{2}$  flow. The contour lines are in 0.1 increments.



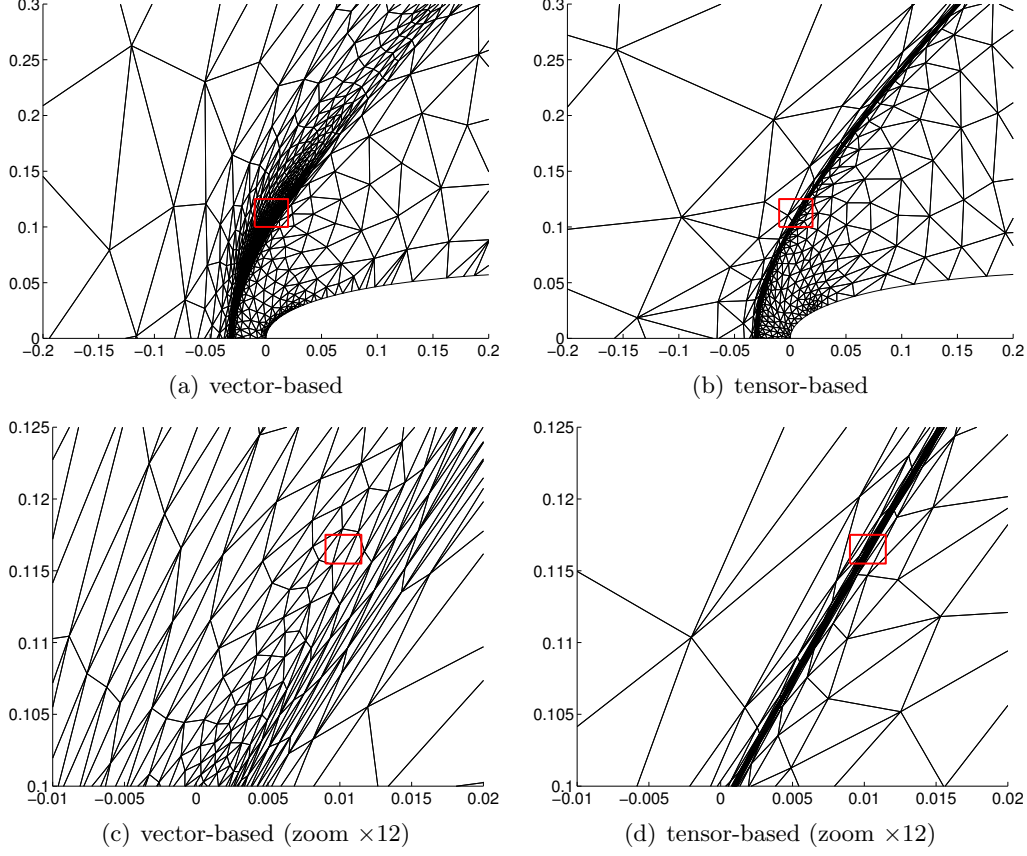


Figure B-4: The adapted meshes obtained using the vector- and tensor-based element size specifications. Each mesh contains approximately 7000 elements.

## B.2 Effects on Adaptation

In this section, we consider the influence of the shock PDEs on the adaptation process. To this end, we compare the adapted meshes obtained for  $M_\infty = 2.0$  flow over a NACA 0012 airfoil using the original, vector-based shock PDE and the modified, tensor-based shock PDE. The  $p = 1$  DG discretization is used for both cases.

Figure B-4 shows adapted meshes. The original shock PDE excessively smears the tilted shock, resulting in the adaptation unable to sharply target the shock. The level of anisotropy is also limited, with less than 30% of the elements having anisotropy above 10. The modified PDE provides consistent level of artificial viscosity to the tilted shock, resulting in the adaptation correctly targeting the shock. A much higher level of anisotropy is attained with 60% of the elements having the anisotropy of between 30 and 300. Thus, the modified, tensor-based formulation of the shock PDE not only improves the shock resolution on a fixed mesh, but also improves the adaptation effectiveness.



## Appendix C

# Regularization of Surface Quantity Distributions

This appendix describes the procedure used to regularize the surface quantity distributions and provides justification for employing such a procedure.

### C.1 Formulation

Suppose we are interested in quantifying the distribution of a surface quantity  $g$ , which in general is a function of the local state and derivative, i.e.  $g = g(u(x), \nabla u(x))$ . Plotting the surface quantity distribution corresponds to evaluating  $g$  at all surface points. This point-wise surface value evaluation problem may be viewed as a “functional” evaluation problem of the form

$$g(x) = \ell_x(u) = \int_{\partial\Omega} \delta(x' - x) g(u(x'), \nabla u(x')) dx',$$

where  $\delta$  is the Dirac delta function. However, it is well-known that this functional  $\ell_x$  arising from the evaluation of a point-wise quantity results in an ill-posed adjoint problem (see, e.g. Giles and Süli [66]). The ill-posed adjoint problem implies that the point-wise quantity does not superconverge, as the adjoint error in the output error representation formula is  $\mathcal{O}(1)$ . In other words, the characterization of the point-wise surface quantity (e.g. skin friction ( $c_f$ )) may be poor even if the prediction of the integral quantity with a well-posed dual problem (e.g. drag) is accurate.



Following the approach pursued in [66], let us define a regularized functional as

$$\ell_{x,h}^{\text{reg}}(u) = \int_{\partial\Omega} \phi_h(x' - x) g(u(x'), \nabla u(x')) dx',$$

where  $\phi_h$  is a regularizer whose support varies with the local element size,  $h$ . In particular, we ensure that  $\phi_h \rightarrow \delta$  as  $h \rightarrow 0$ . This in turn ensures that  $|\ell_x(v) - \ell_{x,h}^{\text{reg}}(v)| \rightarrow 0$  as  $h \rightarrow 0$  for any  $v \in V$ . Error incurred by the regularization procedure may be expressed as

$$|\ell_x(v) - \ell_{x,h}^{\text{reg}}(v_{h,p})| \leq \underbrace{|\ell_x(v) - \ell_{x,h}^{\text{reg}}(v)|}_{\text{regularization error}} + \underbrace{|\ell_{x,h}^{\text{reg}}(v) - \ell_{x,h}^{\text{reg}}(v_{h,p})|}_{\text{approximation error}}.$$

A stronger regularization decreases the approximation error because the regularized functional induces a smoother dual problem, which is easier to approximate; however, a strong regularization increases the regularization error. Thus, the regularizer must be chosen to balance the regularization error and the approximation error. In practice, we have found a Gaussian function with a standard deviation of  $\sigma(x) = h(x)/p$  works well as a regularizer. Here,  $h$  is a smoothly varying function characterizing the local surface element length in two dimensions, and  $p$  is the polynomial order. The choice of the  $h/p$ -scaling regularization scale is motivated by the fact that the resolution of the DG discretization scales as  $h/p$  in pre-asymptotic range, and that we want to produce a regularized dual problem that can be well-approximated by the discretization. The procedure may be generalized to three dimensions by using the metric field projected to the surface as the characteristic surface element size.

## C.2 Results

Let us assess the impact of the regularization procedure using transonic RANS-SA flow over an RAE 2822 airfoil, the case considered in Section 6.3.3. The raw and regularized pressure coefficient ( $c_p$ ) and skin friction coefficient ( $c_f$ ) distributions computed using the  $p = 1$  and  $p = 2$  discretizations with  $\text{dof} = 60,000$  are shown in Figure C-1. The reference distribution, computed on a  $p = 3$ ,  $\text{dof} = 120,000$  mesh, is also shown. As the raw  $c_p$  distributions for both the  $p = 1$  and  $p = 2$  discretizations are already smooth, the regularization procedure has no apparent impact on the  $c_p$  distributions. On the other hand, the raw  $c_f$  distributions



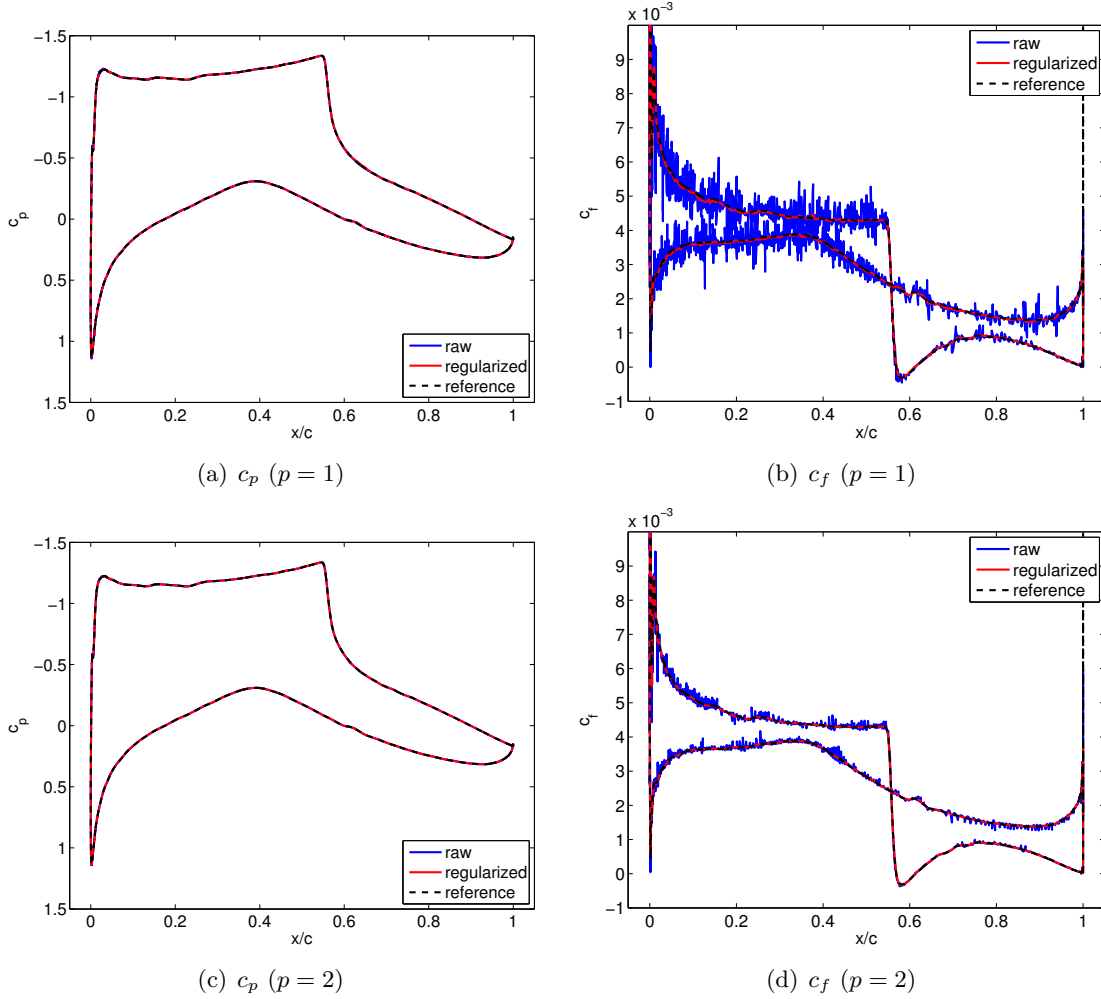


Figure C-1: Comparison of raw and regularized  $c_p$  and  $c_f$  distributions for transonic RANS-SA flow over an RAE 2822 airfoil. The  $p = 1$  and  $p = 2$  discretizations achieve the drag error of  $|c_d - c_d^{\text{ref}}| \approx 7 \times 10^{-6}$  and  $\approx 4 \times 10^{-7}$ , respectively.

are quite noisy, despite the fact that the  $c_d$  errors are  $7 \times 10^{-6}$  for  $p = 1$  and  $4 \times 10^{-7}$  for  $p = 2$ . The poor convergence of  $c_f$  distributions computed on unstructured anisotropic simplex meshes is also reported by Modisette [105]. The regularized  $c_f$  distributions are noticeably smoother than the raw distributions. In fact, the  $p = 2$   $c_f$  plot is essentially indistinguishable from the reference plot for practical engineering purposes. Thus, the regularization procedure improves the prediction of the surface quantity distribution obtained from highly anisotropic simplex-based meshes.







## Appendix D

# Metric-based *A Priori* Error Bounds

This appendix presents details of the metric-based *a priori* error analysis for  $L^2$  projection error and output error provided in Section 2.3.3 and 2.3.4.

### D.1 Anisotropic Polynomial Interpolation Theory

The approximation error analysis presented in this section can be thought of as an extension of the analysis for linear polynomials by Loseille and Alauzet [97, 98] to an arbitrary-degree polynomial space. To extend the analysis, we closely follow the formulation provided by Houston *et al.* [74]. (Related analysis on the higher-degree polynomial approximation error is also provided by Pagnutti and Ollivier-Gooch [112] and Cao [36–38].)

We are concerned with the error that results from approximating a given function with a degree- $p$  polynomial over a region (or an element)  $\kappa$ , i.e. the  $\mathcal{P}^p$  approximation error. Here,  $\kappa$  results from an affine transformation of the reference element,  $\hat{\kappa}$ , with unit length edges, i.e.

$$\kappa = \{x \in \mathbb{R}^d : x = J\hat{x} + x_0, \hat{x} \in \hat{\kappa}\},$$

where  $J \in \mathbb{R}^{d \times d}$  is the Jacobian of the transformation. Specifically, the  $\mathcal{P}^p$  approximation error is defined here as the error incurred by projecting a function onto the polynomial space in the  $L^2$  sense. The  $L^2$  projection of a function  $v \in L^2(\kappa)$  onto  $\mathcal{P}^p(\kappa)$  is denoted by



$\Pi_p v$  and satisfies

$$\Pi_p v = \arg \inf_{v_{h,p} \in \mathcal{P}^p(\kappa)} \|v - v_{h,p}\|_{L^2(\kappa)}.$$

Our goal is to quantify the  $\mathcal{P}^p$  approximation error in terms of metric tensors.

### D.1.1 Notation

Before presenting key results, let us make a few remarks on the notation used in this section. In general, summation on repeated indices is implied except under two circumstances. First, no sum on repeated indices is implied if the indices also appear on the left hand side of the equation. Second, if an index explicitly appears as the index of the summation operator  $\Sigma$ , then the summation on the repeated indices in the argument of the operator is not implied.

### D.1.2 Volume Inequalities

We first develop approximation error bounds over the volume of an anisotropic element starting from the well-known Bramble-Hilbert lemma.

**Lemma D.1** (Approximation on a unit diameter region). *Let  $\hat{\kappa}$  be the reference element of unit diameter. For  $\hat{v} \in H^{k_v}(\hat{\kappa})$ , the  $\mathcal{P}^p$  approximation error is bounded by*

$$|\hat{v} - \Pi_p \hat{v}|_{H^n(\hat{\kappa})} \leq C_{p,d} |\hat{v}|_{H^s(\hat{\kappa})}, \quad n = 0, 1,$$

where  $s \leq \min(p + 1, k_v)$ , and  $C_{p,d}$  is dependent only on the polynomial order  $p$  and the spatial dimension  $d$ .

*Proof.* The proof is provided in, for example, [32]. □

A key approximation result is given in the following theorem, presented by Houston *et al.* [74]:

**Theorem D.2.** *The  $\mathcal{P}^p$  approximation error of a function  $v \in H^{k_v}(\kappa)$  is bounded by*

$$|v - \Pi_p v|_{H^n(\kappa)} \leq C_{p,d} (h_{\min})^{-n} \left( \int_{\kappa} E_{\Sigma}^s(U, \sigma; v) dx \right)^{1/2}, \quad n = 0, 1.$$



Here,  $C_{p,d}$  is the constant of Lemma D.1 that is only dependent on the polynomial degree  $p$  and the dimension  $d$ , and the error kernel  $E_\Sigma^s$  is defined as

$$E_\Sigma^s(U, \sigma; v) \equiv \sum_{i_1=1}^d \cdots \sum_{i_s=1}^d \left( \frac{\partial^s v}{\partial x_{j_1} \cdots \partial x_{j_s}} U_{j_1 i_1} \cdots U_{j_s i_s} \sigma_{i_1} \cdots \sigma_{i_s} \right)^2,$$

where  $s = \min(p+1, k_v)$ ,  $U \in \mathbb{R}^{d \times d}$  is a matrix consisting of left singular vectors of the transformation Jacobian  $J$ ,  $\sigma \in \mathbb{R}^d$  is a set of singular values of  $J$ , and  $h_{\min}$  is the minimum singular value of  $J$ .

*Proof.* The proof follows from anisotropic scaling of the  $H^s$ -norm appearing in the right hand side of Lemma D.1. A proof is provided in [74], but it is repeated here for completeness. For  $n = 0$ , i.e. the  $L^2$  error, the projection error is bounded by

$$\begin{aligned} \|v - \Pi_p v\|_{L^2(\kappa)}^2 &= \int_{\kappa} (v - \Pi_p v)^2(x) dx = \int_{\hat{\kappa}} (\hat{v} - \Pi_p \hat{v})^2(\hat{x}) \det(J) d\hat{x} \\ &\leq C_{p,d}^2 \int_{\hat{\kappa}} \sum_{i_1=1}^d \cdots \sum_{i_s=1}^d \left( \frac{\partial^s \hat{v}}{\partial \hat{x}_{i_1} \cdots \partial \hat{x}_{i_s}} \right)^2 \det(J) d\hat{x} \\ &\leq C_{p,d}^2 \int_{\kappa} \sum_{i_1=1}^d \cdots \sum_{i_s=1}^d \left( \frac{\partial^s v}{\partial x_{j_1} \cdots \partial x_{j_s}} J_{j_1 i_1} \cdots J_{j_s i_s} \right)^2 dx, \end{aligned} \quad (\text{D.1})$$

where  $C_{p,d}$  is the constant of Lemma D.1. For  $n = 1$ , i.e. the  $H^1$  error, the projection error scales as

$$\begin{aligned} |v - \Pi_p v|_{H^1(\kappa)}^2 &= \int_{\kappa} \sum_{l=1}^d \left( \frac{\partial}{\partial x_l} (v - \Pi_p v) \right)^2(x) dx = \int_{\hat{\kappa}} \sum_{l=1}^d \left( \frac{\partial \hat{x}_n}{\partial x_l} \frac{\partial}{\partial \hat{x}_n} (\hat{v} - \Pi_p \hat{v}) \right)^2(\hat{x}) \det(J) d\hat{x} \\ &\leq (h_{\min})^{-2} \int_{\hat{\kappa}} \sum_{l=1}^d \left( \frac{\partial}{\partial \hat{x}_l} (\hat{v} - \Pi_p \hat{v}) \right)^2(\hat{x}) \det(J) d\hat{x} \\ &\leq C_{p,d}^2 (h_{\min})^{-2} \int_{\hat{\kappa}} \sum_{i_1=1}^d \cdots \sum_{i_s=1}^d \left( \frac{\partial^s \hat{v}}{\partial \hat{x}_{i_1} \cdots \partial \hat{x}_{i_s}} \right)^2 \det(J) d\hat{x} \\ &\leq C_{p,d}^2 (h_{\min})^{-2} \int_{\kappa} \sum_{i_1=1}^d \cdots \sum_{i_s=1}^d \left( \frac{\partial^s v}{\partial x_{j_1} \cdots \partial x_{j_s}} J_{j_1 i_1} \cdots J_{j_s i_s} \right)^2 dx \end{aligned}$$

Substitution of the singular decomposition,  $J = U \Sigma V^T$  where  $\Sigma = \text{diag}(\sigma_1, \dots, \sigma_d)$ , into



Eq. (D.1) yields,

$$\begin{aligned}
& |v - \Pi_p v|_{H^n(\kappa)}^2 \\
& \leq C_{p,d}^2 d(h_{\min})^{-2n} \int_{\kappa} \sum_{i_1=1}^d \cdots \sum_{i_s=1}^d \left( \frac{\partial^s v}{\partial x_{j_1} \cdots \partial x_{j_s}} U_{j_1 k_1} \Sigma_{k_1 l_1} V_{i_1 l_1} \cdots U_{j_s k_s} \Sigma_{k_s l_s} V_{i_s l_s} \right)^2 dx \\
& = C_{p,d}^2 d(h_{\min})^{-2n} \int_{\kappa} \sum_{i_1=1}^d \cdots \sum_{i_s=1}^d \left( \frac{\partial^s v}{\partial x_{j_1} \cdots \partial x_{j_s}} U_{j_1 k_1} \Sigma_{k_1 i_1} \cdots U_{j_s k_s} \Sigma_{k_s i_s} \right)^2 dx \\
& = C_{p,d}^2 d(h_{\min})^{-2n} \int_{\kappa} \sum_{i_1=1}^d \cdots \sum_{i_s=1}^d \left( \frac{\partial^s v}{\partial x_{j_1} \cdots \partial x_{j_s}} U_{j_1 i_1} \sigma_{i_1} \cdots U_{j_s i_s} \sigma_{i_s} \right)^2 dx,
\end{aligned}$$

where, following the aforementioned notational convention, the summation of the repeated  $i_1, \dots, i_s$  in the last expression inside the parentheses is not implied. In other words, the integrand is the sum of squared entries of the rank- $s$  tensor indexed by  $i_1, \dots, i_s$ . The first equality follows from the fact that the sum of the squared entries (i.e. the Frobenius norm of the tensor) is invariant under the unitary transformations induced by multiple applications of  $V$ . Recognizing the integrand as the error kernel concludes the proof.  $\square$

Note that the anisotropic  $\mathcal{P}^p$  approximation error bound is a function of not the full  $d \times d$ -dimensional Jacobian matrix; it is a function of  $d$  singular values and a  $d \times d$  unitary matrix, which is  $d(d-1)/2$ -dimensional. In other words, the approximation error is governed by  $d(d-1)/2 + d = d(d+1)/2$  parameters of the affine transformation, rather than the  $d^2$ -dimensional Jacobian. This allows us to encode the anisotropic approximation information into an element-implied metric tensor,  $\mathcal{M}$ . We note that the element-implied metric is closely related to the transformation Jacobian from the reference element  $\hat{\kappa}$  (a unit simplex) to the element  $\kappa$ , i.e.

$$\mathcal{M} = J^{-T} J^{-1}.$$

The proof follows from the fact that the edges of  $\kappa$  are related to that of the unit simplex by  $e_i = J \hat{e}_i$ ,  $i = 1, \dots, d(d+1)/2$ , and the uniqueness of the element-implied metric. The following theorem, expresses the elemental  $\mathcal{P}^p$  approximation error in terms of the elemental metric. (A similar error expression is also derived by Cao in [37].)

**Theorem D.3.** *The  $\mathcal{P}^p$  approximation error of a function  $v \in H^{k_v}(\kappa)$  can be expressed in*



terms of the element-implied metric  $\mathcal{M}_\kappa$  as

$$|v - \Pi_p v|_{H^n(\kappa)} \leq C_{p,d} (h_{\min})^{-n} \left( \int_\kappa E_{\mathcal{M}}^s(\mathcal{M}_\kappa; v) dx \right)^{1/2}, \quad n = 0, 1,$$

where  $s = \min(p+1, k_v)$ ,  $C_{p,d}$  is a constant dependent only on the polynomial degree  $p$  and the dimension  $d$ ,  $h_{\min} = \sigma_{\max}(\mathcal{M}_\kappa)^{-1/2}$  is the minimum element length, the metric-based error kernel  $E_{\mathcal{M}}^s$  is given by

$$E_{\mathcal{M}}^s(\mathcal{M}; v) = \sum_{i_1=1}^d \cdots \sum_{i_s=1}^d \left( \frac{\partial^s v}{\partial x_{j_1} \cdots \partial x_{j_s}} \mathcal{M}_{j_1 i_1}^{-1/2} \cdots \mathcal{M}_{j_s i_s}^{-1/2} \right)^2,$$

and  $\mathcal{M}^{-1/2}$  is the metric square root of  $\mathcal{M}$ .

*Proof.* The metric tensor is related to the singular-value decomposition of the Jacobian by

$$\mathcal{M} = J^{-T} J^{-1} = U \Sigma^{-1} V^T V \Sigma^{-1} U^T = U \Sigma^{-2} U^T.$$

Thus, the  $\mathcal{P}^p$  approximation error bound in Theorem D.2 can be expressed in terms of the metric tensor as

$$\begin{aligned} & |v - \Pi_p v|_{H^n(\kappa)}^2 \\ & \leq C_{p,d}^2 (h_{\min})^{-2n} \int_\kappa \sum_{i_1=1}^d \cdots \sum_{i_s=1}^d \left( \frac{\partial^s v}{\partial x_{j_1} \cdots \partial x_{j_s}} U_{j_1 k_1} \Sigma_{k_1 i_1} \cdots U_{j_s k_s} \Sigma_{k_s i_s} \right)^2 dx \\ & = C_{p,d}^2 (h_{\min})^{-2n} \int_\kappa \sum_{i_1=1}^d \cdots \sum_{i_s=1}^d \left( \frac{\partial^s v}{\partial x_{j_1} \cdots \partial x_{j_s}} U_{j_1 k_1} \Sigma_{k_1 l_1} U_{i_1 l_1} \cdots U_{j_s k_s} \Sigma_{k_s l_s} U_{i_s l_s} \right)^2 dx \\ & = C_{p,d}^2 (h_{\min})^{-2n} \int_\kappa \sum_{i_1=1}^d \cdots \sum_{i_s=1}^d \left( \frac{\partial^s v}{\partial x_{j_1} \cdots \partial x_{j_s}} \mathcal{M}_{j_1 i_1}^{-1/2} \cdots \mathcal{M}_{j_s i_s}^{-1/2} \right)^2 dx, \end{aligned}$$

where the first equality follows from the invariance of the Frobenius norm under successive orthogonal transformations by  $U$ .  $\square$

A straightforward extension of the elemental error result to a triangulation yields the following global approximation error bound.

**Theorem D.4.** *The  $\mathcal{P}^p$  approximation error of a function  $H^{k_v}(\Omega)$  on the triangulation  $\mathcal{T}_h$*



of  $\Omega$  is bounded by

$$|v - \Pi_{h,p} v|_{H^n(\Omega)} \leq C_{p,d} \left[ \sum_{\kappa \in \mathcal{T}_h} \left( (h_{\min})^{-2n} \int_{\kappa} E_{\mathcal{M}}^s(\mathcal{M}_{\kappa}; v) dx \right) \right]^{1/2}, \quad n = 0, 1,$$

where  $s = \min(p+1, k_v)$ ,  $C_{p,d}$  is a constant dependent only on the polynomial degree  $p$  and the dimension  $d$ ,  $h_{\min} = \sigma_{\max}(\mathcal{M}_{\kappa})^{-1/2}$  is the minimum element length,  $E_{\mathcal{M}}^s$  is the error kernels defined in Theorem D.3, and  $\mathcal{M}_{\kappa}$  is the elemental metric tensor associated with  $\kappa$ .

*Proof.* Proof follows from a direct summation of the elemental error in Theorem D.3.  $\square$

### Face Inequalities

Let us now develop a few  $\mathcal{P}^p$  anisotropic error bounds on faces of an element  $\kappa$ . These bounds are used for the *a priori* analysis of the output error in the following section.

**Lemma D.5** (Approximation on a face of a unit diameter element). *Let  $\hat{\kappa}$  be the reference element of unit diameter and  $\hat{f}$  be one of its faces. For  $\hat{v} \in H^{k_v}(\hat{\kappa})$ , the  $\mathcal{P}^p$  approximation error on  $\hat{f}$  is bounded by*

$$|\hat{v} - \Pi_p \hat{v}|_{H^n(\hat{f})} \leq C_{p,d} |\hat{v}|_{H^s(\hat{\kappa})}, \quad n = 0, 1,$$

where  $s \leq \min(p+1, k_v)$ , and  $C_{p,d}$  is dependent only on the polynomial order  $p$  and the spatial dimension  $d$ .

*Proof.* The proof is provided in, for example, [74].  $\square$

The following theorem is a variant of the face inequality shown in [74] expressed in terms of the element-implied metric tensor.

**Theorem D.6.** *The  $\mathcal{P}^p$  approximation error of a function  $v \in H^{k_v}(\kappa)$  on the face  $f$  of the element  $\kappa$  is bounded by*

$$|v - \Pi_p v|_{H^n(f)} \leq C_{p,d} (h_{\min})^{-n} \left( \frac{|f|}{|\kappa|} \right)^{1/2} \left( \int_{\kappa} E_{\mathcal{M}}^s(\mathcal{M}_{\kappa}; v) dx \right)^{1/2}, \quad n = 0, 1,$$

where  $s = \min(p+1, k_v)$ ,  $C_{p,d}$  is a constant only dependent on the polynomial degree  $p$  and the dimension  $d$ ,  $|f|$  is the measure of the face,  $|\kappa|$  is the measure of the element, and  $E_{\mathcal{M}}^s(\mathcal{M}; v)$  is the error kernel defined in Theorem D.3.



*Proof.* The proof follows from anisotropic scaling of the  $H^s$ -norm in the right hand side of Theorem D.6. For  $n = 0$ , i.e. the  $L^2$  error, we have

$$\begin{aligned} \|v - \Pi_p v\|_{L^2(f)}^2 &= \int_f (v - \Pi_p v)^2 ds = \int_{\hat{f}} (\hat{v} - \Pi_p \hat{v})^2 |f| d\hat{s} \\ &\leq C_{p,d} |f| \int_{\hat{\kappa}} \sum_{i_1=1}^d \cdots \sum_{i_s=1}^d \left( \frac{\partial^s \hat{v}}{\partial \hat{x}_{i_1} \cdots \partial \hat{x}_{i_s}} \right)^2 d\hat{x} \\ &\leq C_{p,d} \frac{|f|}{|\kappa|} \int_{\kappa} \sum_{i_1=1}^d \cdots \sum_{i_s=1}^d \left( \frac{\partial^s v}{\partial x_{j_1} \cdots \partial x_{j_s}} J_{j_1 i_1} \cdots J_{j_s i_s} \right)^2 dx, \end{aligned}$$

and replacing the Jacobian  $J$  in the integrand with the metric tensor  $\mathcal{M}$  as in the proof of Theorem D.3 concludes the proof. For  $n = 1$ , we have

$$\begin{aligned} |v - \Pi_p v|_{H^1(f)}^2 &= \int_f \sum_{l=1}^d \left( \frac{\partial}{\partial x_l} (v - \Pi_p v) \right)^2(x) ds \\ &\leq (h_{\min})^{-2} |f| \int_{\hat{f}} \sum_{l=1}^d \left( \frac{\partial}{\partial \hat{x}_l} (\hat{v} - \Pi_p \hat{v}) \right)^2(\hat{x}) d\hat{s} \\ &\leq C_{p,d} (h_{\min})^{-2} |f| \int_{\hat{\kappa}} \sum_{i_1=1}^d \cdots \sum_{i_s=1}^d \left( \frac{\partial^s \hat{v}}{\partial \hat{x}_{i_1} \cdots \partial \hat{x}_{i_s}} \right)^2 d\hat{x} \\ &\leq C_{p,d} (h_{\min})^{-2} \frac{|f|}{|\kappa|} \int_{\kappa} \sum_{i_1=1}^d \cdots \sum_{i_s=1}^d \left( \frac{\partial^s v}{\partial x_{j_1} \cdots \partial x_{j_s}} J_{j_1 i_1} \cdots J_{j_s i_s} \right)^2 dx, \end{aligned}$$

and again replacing the Jacobian  $J$  in the integrand with the metric tensor  $\mathcal{M}$  as in the proof of Theorem D.3 concludes the proof.  $\square$

## D.2 Output Error Bounds

This section provides auxiliary results used to prove the output error bound of the system of linear PDEs, Eq. (2.7), considered in Section 2.3.4. For convenience, the equation is reproduced here:

$$\begin{aligned} \nabla \cdot (\mathcal{A}u) - \nabla \cdot (\mathcal{K} \nabla u) + \mathcal{C}u &= 0, \quad \text{in } \Omega \\ u &= g, \quad \text{on } \partial\Omega, \end{aligned} \tag{D.2}$$



where  $\mathcal{A}_i \in \mathbb{R}^{m \times m}$ ,  $i = 1, \dots, d$ , is the flux Jacobian,  $\mathcal{K}_{ij} \in \mathbb{R}^{m \times m}$ ,  $i, j = 1, \dots, d$  constitute the viscosity tensor, and  $\mathcal{C} \in \mathbb{R}^m$  is the reaction matrix. To proceed with the error analysis, we split the elemental error contribution into convection, diffusion, and source terms, and analyze the terms individually, i.e.

$$\begin{aligned} \eta_\kappa &= |\mathcal{R}'_{h,p}(u - u_{h,p}, (\psi - v_{h,p})|_\kappa)| \\ &\leq |(\mathcal{R}^{\text{conv}}_{h,p})'(u - u_{h,p}, (\psi - v_{h,p})|_\kappa)| + |(\mathcal{R}^{\text{diff}}_{h,p})'(u - u_{h,p}, (\psi - v_{h,p})|_\kappa)| \\ &\quad + |(\mathcal{R}^{\text{sour}}_{h,p})'(u - u_{h,p}, (\psi - v_{h,p})|_\kappa)|, \quad \forall v_{h,p} \in V_{h,p}. \end{aligned}$$

In order to enable the analysis, we make the following key assumption regarding the quality of the DG solution.

**Assumption D.7** (Optimality of the DG solution). *Suppose  $u \in H^{k_u}(\Omega)$  is the solution to the linear advection-diffusion-reaction system, Eq. (D.2). The DG solution to the problem is assumed to be optimal in the sense that the volume and face errors of the solution are comparable to those that result from the  $L^2$  projection of the true solution. Specifically, the degree- $p$  DG solution,  $u_{h,p}$ , is assumed to satisfy,*

$$|u - u_{h,p}|_{H^n(\kappa)} \leq C_{p,d}(h_{\min})^{-n} \int_\kappa E_{\mathcal{M}}^{s_u}(\mathcal{M}_\kappa; u) dx, \quad n = 0, 1,$$

where  $s_u = \min(p + 1, k_u)$ , and  $E_{\mathcal{M}}^s$  is the error kernel defined in Theorem D.3. Similarly, the face errors are assumed to be bounded by

$$|u - u_{h,p}|_{H^n(f)} \leq C_{p,d}(h_{\min})^{-n} \left( \frac{|f|}{|\kappa|} \right)^{1/2} \left( \int_\kappa E_{\mathcal{M}}^{s_u}(\mathcal{M}_\kappa; u) dx \right)^{1/2}, \quad n = 0, 1,$$

where  $s_u = \min(p + 1, k_u)$ .

This is a rather strong assumption regarding the quality of the DG solution. The assumption implies that the DG discretization is stable and is optimal with respect to the volume and face norms. However, the optimality assumption simplifies the output error estimation to that of analyzing the continuity of the bilinear form. (The proof of optimality for a coercive scalar linear equation is provided by Houston *et al.* [74].)

Let us start with the analysis of the convection term.



**Lemma D.8** (Continuity of the convection term of a linear system). *The elemental restriction of the linear convection operator is bounded by*

$$\begin{aligned} |(\mathcal{R}_{h,p}^{\text{conv}})'(w, v|_{\kappa})| &\leq \sum_{k=1}^m \sum_{i=1}^d |\lambda_k^{\mathcal{A}_i}| \| (r_k^{\mathcal{A}_i})^T v \|_{L^2(\kappa)} \| (l_k^{\mathcal{A}_i})^T \frac{\partial w}{\partial x_i} \|_{L^2(\kappa)} \\ &\quad + \sum_{f \in F(\kappa)} \sum_{k=1}^m |\lambda_k^{\mathcal{A}_{\hat{n}}^-}| \| (r_k^{\mathcal{A}_{\hat{n}}^-})^T v \|_{L^2(f)} \| (l_k^{\mathcal{A}_{\hat{n}}^-})^T [w]_{-}^{\pm} \|_{L^2(f)}, \end{aligned}$$

where  $F(\kappa)$  is the set of faces of  $\kappa$ ,  $[w]_{-}^{\pm} = w^{+} - w^{-}$  for interior faces,  $[w]_{-}^{\pm} = w^{+}$  for the boundary faces, and, for an arbitrary matrix  $B$ ,  $\lambda_k^B$ ,  $r_k^B$ , and  $l_k^B$  denote the  $k$ -th eigenvalue, right eigenvector, and left eigenvector, respectively, i.e.  $B = \sum_{k=1}^m \lambda_k^B r_k^B (l_k^B)^T$ .

*Proof.* The interior and boundary upwinding numerical flux for the linear system is given by

$$\begin{aligned} \mathcal{H}(w^{+}, w^{-}; \hat{n}) &= \mathcal{A}_{\hat{n}}^{+} w^{+} + \mathcal{A}_{\hat{n}}^{-} w^{-} \\ \mathcal{H}^b(w^{+}, g; \hat{n}) &= \mathcal{A}_{\hat{n}}^{+} w^{+} + \mathcal{A}_{\hat{n}}^{-} g, \end{aligned}$$

where  $\mathcal{A}_{\hat{n}} = \mathcal{A}_i \hat{n}_i$  the flux Jacobian in the direction of  $\hat{n}$ ,  $\mathcal{A}_{\hat{n}}^{+}$  and  $\mathcal{A}_{\hat{n}}^{-}$  denote the matrix that results from collecting the modes with positive and negative eigenvalues, respectively. Substitution of the numerical flux functions, restriction to the element  $\kappa$ , and integration by parts yield the local semilinear form

$$\mathcal{R}_{h,p}^{\text{conv}}(w, v|_{\kappa}) = \int_{\kappa} v^T \mathcal{A} \nabla w dx - \int_{\partial\kappa \setminus \partial\Omega} v^T \mathcal{A}_{\hat{n}}^{-} [w]_{-}^{\pm} ds - \int_{\partial\kappa \cap \partial\Omega} v^T \mathcal{A}_{\hat{n}}^{-} (w - g) ds,$$

where  $[w]_{-}^{\pm} = w^{+} - w^{-}$ . Linearization of the semilinear form yields a bilinear form

$$(\mathcal{R}_{h,p}^{\text{conv}})'(w, v|_{\kappa}) = \int_{\kappa} v^T \mathcal{A} \nabla w dx - \int_{\partial\kappa \setminus \partial\Omega} v^T \mathcal{A}_{\hat{n}}^{-} [w]_{-}^{\pm} ds - \int_{\partial\kappa \cap \partial\Omega} v^T \mathcal{A}_{\hat{n}}^{-} w ds.$$

Using the definition of  $[w]_{-}^{\pm}$  and the eigenvalue decompositions, the expression becomes

$$\begin{aligned} (\mathcal{R}_{h,p}^{\text{conv}})'(w, v|_{\kappa}) &= \sum_{k=1}^m \sum_{i=1}^d \int_{\kappa} \lambda_k^{\mathcal{A}_i} ((r_k^{\mathcal{A}_i})^T v) ((l_k^{\mathcal{A}_i})^T \frac{\partial w}{\partial x_i}) dx \\ &\quad - \sum_{f \in F(\kappa)} \sum_{k=1}^m \int_f \lambda_k^{\mathcal{A}_{\hat{n}}^-} ((r_k^{\mathcal{A}_{\hat{n}}^-})^T v) ((l_k^{\mathcal{A}_{\hat{n}}^-})^T [w]_{-}^{\pm}) ds \end{aligned}$$



Invoking the Schwarz inequality on each integral proves the desired result.  $\square$

A combination of the output error representation formula and the continuity of the bilinear form yields the following output error bound for the convection term.

**Theorem D.9** (Output error bound of the linear convection operator). *Let the local restriction of the primal and adjoint solutions to the advection-diffusion-reaction system Eq. (2.7) be  $u \in H^{k_u}(\tilde{\kappa})$  and  $\psi \in H^{k_\psi}(\tilde{\kappa})$ , respectively, where  $\tilde{\kappa}$  consists of element  $\kappa$  and its face-sharing neighbors. Assuming the DG approximation  $u_{h,p} \in V_{h,p}$  satisfies the optimality condition, Assumption D.7, the elemental output error contribution from the convection term is bounded by*

$$\begin{aligned} & |(\mathcal{R}_{h,p}^{\text{conv}})'(u - u_{h,p}, (\psi - v_{h,p})|_\kappa)| \\ & \leq h_{\min}^{-1} \sum_{k=1}^m \sum_{i=1}^d |\lambda_k^{\mathcal{A}_i}| \left( \int_\kappa E_{\mathcal{M}}^{s_u}(\mathcal{M}_\kappa; (r_k^{\mathcal{A}_i})^T u) dx \right)^{1/2} \left( \int_\kappa E_{\mathcal{M}}^{s_\psi}(\mathcal{M}_\kappa; (l_k^{\mathcal{A}_i})^T \psi) dx \right)^{1/2} \\ & + h_{\min}^{-1} \sum_{f \in F(\kappa)} \sum_{k=1}^m |\lambda_k^{\mathcal{A}_{\hat{n}}^-}| \left( \int_\kappa E_{\mathcal{M}}^{s_u}(\mathcal{M}_\kappa; (r_k^{\mathcal{A}_{\hat{n}}})^T u) dx \right)^{1/2} \left( \int_\kappa E_{\mathcal{M}}^{s_\psi}(\mathcal{M}_\kappa; (l_k^{\mathcal{A}_{\hat{n}}})^T \psi) dx \right)^{1/2}, \end{aligned}$$

where  $s_u = \min(p+1, k_u)$ ,  $s_\psi = \min(p+1, k_\psi)$ , and  $h_{\min}^{-1}$  is the minimum singular value of the transformation Jacobian of the element  $\kappa$ .

*Proof.* Substituting  $w = u - u_{h,p}$  and  $v = \psi - \Pi_{h,p}\psi$  into Lemma D.8 and invoking Theorems D.3 and D.6 on the integrals over  $\kappa$  and  $\partial\kappa$ , respectively, results in

$$\begin{aligned} & |(\mathcal{R}_{h,p}^{\text{conv}})'(u - u_{h,p}, (\psi - \Pi_{h,p}\psi)|_\kappa)| \\ & \leq h_{\min}^{-1} \sum_{k=1}^m \sum_{i=1}^d |\lambda_k^{\mathcal{A}_i}| \left( \int_\kappa E_{\mathcal{M}}^{s_u}(\mathcal{M}_\kappa; (r_k^{\mathcal{A}_i})^T u) dx \right)^{1/2} \left( \int_\kappa E_{\mathcal{M}}^{s_\psi}(\mathcal{M}_\kappa; (l_k^{\mathcal{A}_i})^T \psi) dx \right)^{1/2} \\ & + \frac{|f|}{|\kappa|} \sum_{f \in F(\kappa)} \sum_{k=1}^m |\lambda_k^{\mathcal{A}_{\hat{n}}^-}| \left( \int_\kappa E_{\mathcal{M}}^{s_u}(\mathcal{M}_\kappa; (r_k^{\mathcal{A}_{\hat{n}}})^T u) dx \right)^{1/2} \left( \int_\kappa E_{\mathcal{M}}^{s_\psi}(\mathcal{M}_\kappa; (l_k^{\mathcal{A}_{\hat{n}}})^T \psi) dx \right)^{1/2}. \end{aligned}$$

Recognizing  $|f|/|\kappa| \leq h_{\min}^{-1}$  concludes the proof.  $\square$

Before proceeding to bound the diffusive term, let us introduce two useful relationships relating the element and face norms. The first one is an anisotropic extension of the well-known trace scaling result [72], and the second one relates the lifting operator and the face jump.



**Lemma D.10** (Anisotropic trace scaling). *For any  $v \in \mathcal{P}^p(\kappa)$ , the following inverse estimate holds*

$$\|v\|_{L^2(f)} \leq C_{p,d}^{\text{inv}} \left( \frac{|f|}{|\kappa|} \right)^{1/2} \|v\|_{L^2(\kappa)},$$

where  $f$  is one of the faces of  $\kappa$ , and  $C_{p,d}^{\text{inv}}$  is a constant only dependent on the polynomial degree  $p$  and the dimension  $d$ .

*Proof.* The proof is provided in [72], but is repeated here for completeness. On a unit diameter element  $\hat{\kappa}$  with a unit face  $\hat{f}$ ,

$$\|\hat{v}\|_{L^2(\hat{f})} \leq C_{p,d}^{\text{inv}} \|\hat{v}\|_{L^2(\hat{\kappa})}, \quad \forall \hat{v} \in \mathcal{P}^p(\hat{\kappa}),$$

by inverse estimate. Straightforward scaling yields

$$\|v\|_{L^2(f)}^2 = \int_{\hat{f}} \hat{v}^2 |f| d\hat{s} \leq (C_{p,d}^{\text{inv}})^2 |f| \int_{\hat{\kappa}} \hat{v}^2 d\hat{x} = (C_{p,d}^{\text{inv}})^2 \frac{|f|}{|\kappa|} \|v\|_{L^2(\kappa)}^2,$$

which proves the desired result.  $\square$

**Lemma D.11** (Anisotropic  $h$ -scaling of the Lifting Operator). *The BR2 lifting operator is bounded by the face jump according to*

$$\|r_f(\llbracket v \rrbracket)\|_{L^2(f)} \leq C_{p,d}^{\text{inv}} \left( \frac{|f|}{|\kappa|} \right)^{1/2} \|r_f(\llbracket v \rrbracket)\|_{L^2(\kappa)} \leq (C_{p,d}^{\text{inv}})^2 \frac{|f|}{|\kappa|} \|\llbracket v \rrbracket\|_{L^2(f)},$$

where  $C_{p,d}$  is a constant dependent on only polynomial order  $p$  and dimension  $d$ .

*Proof.* The first inequality follows from the trace scaling, Lemma D.10. The second inequality follows from the definition of the lifting operator and the trace scaling, i.e.

$$\begin{aligned} \|r_f(\llbracket v \rrbracket)\|_{L^2(\kappa)}^2 &= \int_{\kappa} r_f(\llbracket v \rrbracket)^2 dx = \int_f r_f(\llbracket v \rrbracket) \llbracket v \rrbracket ds \leq \|r_f(\llbracket v \rrbracket)\|_{L^2(f)} \|\llbracket v \rrbracket\|_{L^2(f)} \\ &\leq C_{p,d}^{\text{inv}} \left( \frac{|f|}{|\kappa|} \right)^{1/2} \|r_f(\llbracket v \rrbracket)\|_{L^2(\kappa)} \|\llbracket v \rrbracket\|_{L^2(f)}, \end{aligned}$$

which proves the desired result.  $\square$

Having proved two auxiliary results concerning the element and face norms, let us now analyze the continuity of the diffusive term.



**Lemma D.12** (Continuity of the diffusive term of a linear system). *The elemental restriction of the linear diffusive operator is bounded by*

$$\begin{aligned}
|(\mathcal{R}_{h,p}^{\text{diff}})'(w, v|_{\kappa})| &\leq \sum_{k=1}^m \sum_{i,j=1}^d |\lambda_k^{\mathcal{K}_{ij}}| \| (r_k^{\mathcal{K}_{ij}})^T \frac{\partial v}{\partial x_i} \|_{L^2(\kappa)} \| (l_k^{\mathcal{K}_{ij}})^T \frac{\partial w}{\partial x_j} \|_{L^2(\kappa)} \\
&\quad + \sum_{k=1}^m \sum_{i,j=1}^d |\lambda_k^{\mathcal{K}_{ij}}| \| (r_k^{\mathcal{K}_{ij}})^T \frac{\partial v}{\partial x_i} \|_{L^2(\partial\kappa)} \| (l_k^{\mathcal{K}_{ij}})^T \llbracket w \rrbracket_j \|_{L^2(\partial\kappa)} \\
&\quad + \sum_{k=1}^m \sum_{i,j=1}^d |\lambda_k^{\mathcal{K}_{ij}}| \| (r_k^{\mathcal{K}_{ij}})^T v \hat{n}_i \|_{L^2(\partial\kappa)} \\
&\quad \left( \| (l_k^{\mathcal{K}_{ij}})^T \left\{ \frac{\partial w}{\partial x_j} \right\} \|_{L^2(\partial\kappa)} + C_{p,d}^{\text{inv}} \eta_f \frac{|f|}{|\kappa|_{\min}} \| (l_k^{\mathcal{K}_{ij}})^T \llbracket w \rrbracket_j \|_{L^2(\partial\kappa)} \right),
\end{aligned}$$

where  $\lambda_k^{\mathcal{K}_{ij}}$ ,  $r_k^{\mathcal{K}_{ij}}$ , and  $l_k^{\mathcal{K}_{ij}}$  are the  $k$ -th eigenvalue, right eigenvector, and left eigenvector of the viscosity matrix  $\mathcal{K}_{ij}$ ,  $|\kappa|_{\min} = \min(|\kappa^+|, |\kappa^-|)$  on  $\partial\kappa \setminus \partial\Omega$ ,  $|\kappa|_{\min} = |\kappa|$  on  $\partial\kappa \cap \partial\Omega$ ,  $\llbracket w \rrbracket_j$  is the  $j$ -th coordinate component of the jump operator on  $\partial\kappa \setminus \partial\Omega$ , and  $\llbracket w \rrbracket_j \equiv w \hat{n}_j$  on  $\partial\kappa \cap \partial\Omega$ .

*Proof.* The local semilinear form for a linear diffusive operator with the Dirichlet boundary condition is

$$\begin{aligned}
\mathcal{R}_{h,p}^{\text{diff}}(w, v|_{\kappa}) &= \int_{\kappa} \nabla v^T \cdot \mathcal{K} \nabla w dx - \int_{\partial\kappa \setminus \partial\Omega} \frac{1}{2} \nabla v^T \cdot \mathcal{K} \llbracket w \rrbracket + v^T \hat{n} \cdot \{ \mathcal{K}(\nabla w + \eta_f r_f(\llbracket w \rrbracket)) \} ds \\
&\quad - \int_{\partial\kappa \cap \partial\Omega} \nabla v^T \cdot \mathcal{K}(w - g) \hat{n} + v^T \hat{n} \cdot \mathcal{K}(\nabla w + \eta_f r_f^b((w - g) \hat{n})) ds.
\end{aligned}$$

Linearization of the semilinear form yields a bilinear form

$$\begin{aligned}
(\mathcal{R}_{h,p}^{\text{diff}})'(w, v|_{\kappa}) &= \int_{\kappa} \nabla v^T \cdot \mathcal{K} \nabla w dx - \int_{\partial\kappa \setminus \partial\Omega} \frac{1}{2} \nabla v^T \cdot \mathcal{K} \llbracket w \rrbracket + v^T \hat{n} \cdot \{ \mathcal{K}(\nabla w + \eta_f r_f(\llbracket w \rrbracket)) \} ds \\
&\quad - \int_{\partial\kappa \cap \partial\Omega} \nabla v^T \cdot \mathcal{K} w \hat{n} + v^T \hat{n} \cdot \mathcal{K}(\nabla w + \eta_f r_f^b(w \hat{n})) ds.
\end{aligned}$$

By defining  $\llbracket w \rrbracket = w \hat{n}$  on  $\partial\kappa \cap \partial\Omega$  and combining the interior and boundary integrals, the expression simplifies to

$$\left| (\mathcal{R}_{h,p}^{\text{diff}})'(w, v|_{\kappa}) \right| \leq \left| \int_{\kappa} \nabla v^T \cdot \mathcal{K} \nabla w dx \right| + \left| \int_{\partial\kappa} \nabla v^T \cdot \mathcal{K} \llbracket w \rrbracket + v^T \hat{n} \cdot \{ \mathcal{K}(\nabla w + \eta_f r_f(\llbracket w \rrbracket)) \} ds \right|.$$



Using the eigenvalue decompositions of  $\mathcal{K}$ , the expression becomes

$$\begin{aligned} \left| (\mathcal{R}_{h,p}^{\text{diff}})'(w, v|_{\kappa}) \right| &\leq \sum_{k=1}^m \sum_{i,j=1}^d \left| \int_{\kappa} \lambda_k^{\mathcal{K}_{ij}} ((r_k^{\mathcal{K}_{ij}})^T \frac{\partial v}{\partial x_i}) ((l_k^{\mathcal{K}_{ij}})^T \frac{\partial w}{\partial x_j}) dx \right| \\ &\quad + \sum_{k=1}^m \sum_{i,j=1}^d \left| \int_{\partial\kappa} \lambda_k^{\mathcal{K}_{ij}} ((r_k^{\mathcal{K}_{ij}})^T \frac{\partial v}{\partial x_i}) ((l_k^{\mathcal{K}_{ij}})^T \llbracket w \rrbracket_j) ds \right| \\ &\quad + \sum_{k=1}^m \sum_{i,j=1}^d \left| \int_{\partial\kappa} \lambda_k^{\mathcal{K}_{ij}} ((r_k^{\mathcal{K}_{ij}})^T v \hat{n}_i) ((l_k^{\mathcal{K}_{ij}})^T \{ \frac{\partial w}{\partial x_j} + \eta_f r_f(\llbracket w \rrbracket_j) \}) ds \right|, \end{aligned}$$

Invoking the Schwarz inequality yields the desired results for the first and second terms. The third term is bounded by Schwarz inequality followed by the lifting operator scaling, Lemma D.11, i.e. after invoking Schwarz inequality, the term involving the lifting operator is bounded by

$$\begin{aligned} &\| (l_k^{\mathcal{K}_{ij}})^T \{ \frac{\partial w}{\partial x_j} + \eta_f r_f(\llbracket w \rrbracket_j) \} \|_{L^2(\partial\kappa)} = \| \{ (l_k^{\mathcal{K}_{ij}})^T \frac{\partial w}{\partial x_j} + \eta_f r_f((l_k^{\mathcal{K}_{ij}})^T \llbracket w \rrbracket_j) \} \|_{L^2(\partial\kappa)} \\ &\leq \| \{ (l_k^{\mathcal{K}_{ij}})^T \frac{\partial w}{\partial x_j} \} \|_{L^2(\partial\kappa)} + \eta_f \| \{ r_f((l_k^{\mathcal{K}_{ij}})^T \llbracket w \rrbracket_j) \} \|_{L^2(\partial\kappa)} \\ &\leq \| \{ (l_k^{\mathcal{K}_{ij}})^T \frac{\partial w}{\partial x_j} \} \|_{L^2(\partial\kappa)} + C_{p,d}^{\text{inv}} \eta_f \frac{|f|}{|\kappa|_{\min}} \| \{ (l_k^{\mathcal{K}_{ij}})^T \llbracket w \rrbracket_j \} \|_{L^2(\partial\kappa)}, \end{aligned}$$

which proves the desired result.  $\square$

**Theorem D.13** (Output error bound of the linear diffusion operator). *Let the local restriction of the primal and adjoint solutions to the advection-diffusion-reaction system Eq. (2.7) be  $u \in H^{k_u}(\tilde{\kappa})$  and  $\psi \in H^{k_\psi}(\tilde{\kappa})$ , respectively, where  $\tilde{\kappa}$  consists of element  $\kappa$  and its face-sharing neighbors. Assuming the DG approximation  $u_{h,p} \in V_{h,p}$  satisfies the optimality condition, Assumption D.7, the elemental output error contribution from the diffusion term is bounded by*

$$\begin{aligned} &\left| (\mathcal{R}_{h,p}^{\text{diff}})'(u - u_{h,p}, (\psi - v_{h,p})|_{\kappa}) \right| \\ &\leq C \sum_{k=1}^m \sum_{i,j=1}^d \frac{|\lambda_k^{\mathcal{K}_{ij}}|}{h_{\min}^2} \left( \int_{\kappa} E_{\mathcal{M}}^{s_u}(\mathcal{M}_{\kappa}; (r_k^{\mathcal{K}_{ij}})^T u) dx \right)^{1/2} \left( \int_{\kappa} E_{\mathcal{M}}^{s_\psi}(\mathcal{M}_{\kappa}; (l_k^{\mathcal{K}_{ij}})^T \psi) dx \right)^{1/2}, \end{aligned}$$

where  $s_u = \min(p+1, k_u)$ ,  $s_\psi = \min(p+1, k_\psi)$ , and  $h_{\min}$  is the minimum singular value of the transformation Jacobians of the elements in  $\tilde{\kappa}$ .



*Proof.* Substituting  $w = u - u_{h,p}$  and  $v = \psi - \Pi_{h,p}\psi$  into Lemma D.12 and invoking Theorems D.3 and D.6 on the integrals over  $\kappa$  and  $\partial\kappa$ , respectively, results in

$$\begin{aligned}
& |(\mathcal{R}_{h,p}^{\text{diff}})'(u - u_{h,p}, (\psi - \Pi_{h,p}\psi)|_{\kappa})| \\
& \leq C_{p,d} \sum_{k=1}^m \sum_{i,j=1}^d |\lambda_k^{\mathcal{K}_{ij}}| \left[ h_{\min}^{-2} \left( \int_{\kappa} E_{\mathcal{M}}^{s_u}(\mathcal{M}_{\kappa}; (r_k^{\mathcal{K}_{ij}})^T u) dx \right)^{1/2} \left( \int_{\kappa} E_{\mathcal{M}}^{s_{\psi}}(\mathcal{M}_{\kappa}; (l_k^{\mathcal{K}_{ij}})^T \psi) dx \right)^{1/2} \right. \\
& \quad + h_{\min}^{-1} d^2 \frac{|f|}{|\kappa|} \left( \int_{\kappa} E_{\mathcal{M}}^{s_u}(\mathcal{M}_{\kappa}; (r_k^{\mathcal{K}_{ij}})^T u) dx \right)^{1/2} \left( \int_{\kappa} E_{\mathcal{M}}^{s_{\psi}}(\mathcal{M}_{\kappa}; (l_k^{\mathcal{K}_{ij}})^T \psi) dx \right)^{1/2} \\
& \quad \left. + d^2 \frac{|f|}{|\kappa|} \left( h_{\min}^{-1} + \frac{C_{p,d}^{\text{inv}} \eta_f |f|}{|\kappa|_{\min}} \right) \left( \int_{\kappa} E_{\mathcal{M}}^{s_u}(\mathcal{M}_{\kappa}; (r_k^{\mathcal{K}_{ij}})^T u) dx \right)^{1/2} \left( \int_{\kappa} E_{\mathcal{M}}^{s_{\psi}}(\mathcal{M}_{\kappa}; (l_k^{\mathcal{K}_{ij}})^T \psi) dx \right)^{1/2} \right].
\end{aligned}$$

Recognizing that  $|f|/|\kappa| \leq h_{\min}^{-1}$  and  $|f|/|\kappa|_{\min} \leq h_{\min}^{-1}$  with the definition of  $h_{\min}$  that includes the face-sharing neighbor elements concludes the proof.  $\square$

Now let us consider the error contribution due to the source term, using the same technique used for the convection and diffusion terms.

**Lemma D.14** (Continuity of the linear source term). *The linear source term is bounded by*

$$|(\mathcal{R}_{h,p}^{\text{sour}})'(w, v|_{\kappa})| \leq \sum_{k=1}^m |\lambda_k^{\mathcal{C}}| \| (r_k^{\mathcal{C}})^T v \|_{L^2(\kappa)} \| (l_k^{\mathcal{C}})^T w \|_{L^2(\kappa)},$$

where  $\lambda_k^{\mathcal{C}}$ ,  $r_k^{\mathcal{C}}$ , and  $l_k^{\mathcal{C}}$  are the  $k$ -th eigenvalue, right eigenvector, and left eigenvector of the reaction matrix  $\mathcal{C}$ .

*Proof.* The elemental restriction of the bilinear form corresponding to the linear source contribution is given by

$$(\mathcal{R}_{h,p}^{\text{sour}})'(w, v|_{\kappa}) = \int_{\kappa} v^T \mathcal{C} w dx.$$

Taking the eigenvalue decomposition of the matrix  $\mathcal{C}$  and invoking the Schwarz inequality yields the desired result.  $\square$

**Theorem D.15** (Output error bound of the linear source term). *Let the elemental restriction of the primal and adjoint solutions to the advection-diffusion-reaction system Eq. (2.7) be  $u \in H^{k_u}(\kappa)$  and  $\psi \in H^{k_{\psi}}(\kappa)$ , respectively. Assuming the DG approximation  $u_{h,p} \in V_{h,p}$*



satisfies the optimality condition, Assumption D.7, the elemental output error contribution from the source term is bounded by

$$\begin{aligned} & |(\mathcal{R}_{h,p}^{\text{sour}})'(u - u_{h,p}, \psi - v_{h,p})| \\ & \leq C \sum_{k=1}^m |\lambda_k^{\mathcal{C}}| \left( \int_{\kappa} E_{\mathcal{M}}^{s_u}(\mathcal{M}_{\kappa}; (r_k^{\mathcal{C}})^T u) dx \right)^{1/2} \left( \int_{\kappa} E_{\mathcal{M}}^{s_{\psi}}(\mathcal{M}_{\kappa}; (l_k^{\mathcal{C}})^T \psi) dx \right)^{1/2}. \end{aligned}$$

where  $s_u = \min(p+1, k_u)$ , and  $s_{\psi} = \min(p+1, k_{\psi})$ .

*Proof.* Substituting  $w = u - u_{h,p}$  and  $v = \psi - \Pi_{h,p}\psi$  into Lemma D.14 and invoking Theorem D.3 yield the desired result.  $\square$

Finally, combining the error bounds for the convection, diffusion, and reaction operators stated in Theorems D.9, D.13, and D.15, respectively, we obtain an elemental *a priori* error bound for the advection-diffusion-reaction system Eq. (2.7).

**Theorem D.16.** *Let the local restriction of the primal and adjoint solutions to the advection-diffusion-reaction system Eq. (2.7) be  $u \in H^{k_u}(\tilde{\kappa})$  and  $\psi \in H^{k_{\psi}}(\tilde{\kappa})$ , respectively, where  $\tilde{\kappa}$  consists of element  $\kappa$  and its face-sharing neighbors. Assuming the DG approximation  $u_{h,p} \in V_{h,p}$  satisfies the optimality condition, Assumption D.7, the elemental output error is bounded by*

$$\begin{aligned} \eta_{\kappa}^{\text{a priori}} & \leq C \left[ \sum_{k=1}^m \sum_{i=1}^d \frac{|\lambda_k^{\mathcal{A}_i}|}{h_{\min}} \left( \int_{\kappa} E_{\mathcal{M}}^{s_u}(\mathcal{M}_{\kappa}; (r_k^{\mathcal{A}_i})^T u) dx \right)^{1/2} \left( \int_{\kappa} E_{\mathcal{M}}^{s_{\psi}}(\mathcal{M}_{\kappa}; (l_k^{\mathcal{A}_i})^T \psi) dx \right)^{1/2} \right. \\ & \quad + \sum_{f \in F(\kappa)} \sum_{k=1}^m \frac{|\lambda_k^{\mathcal{A}_{\hat{n}}}|}{h_{\min}} \left( \int_{\kappa} E_{\mathcal{M}}^{s_u}(\mathcal{M}_{\kappa}; (r_k^{\mathcal{A}_{\hat{n}}})^T u) dx \right)^{1/2} \left( \int_{\kappa} E_{\mathcal{M}}^{s_{\psi}}(\mathcal{M}_{\kappa}; (l_k^{\mathcal{A}_{\hat{n}}})^T \psi) dx \right)^{1/2} \\ & \quad + \sum_{k=1}^m \sum_{i,j=1}^d \frac{|\lambda_k^{\mathcal{K}_{ij}}|}{h_{\min}^2} \left( \int_{\kappa} E_{\mathcal{M}}^{s_u}(\mathcal{M}_{\kappa}; (r_k^{\mathcal{K}_{ij}})^T u) dx \right)^{1/2} \left( \int_{\kappa} E_{\mathcal{M}}^{s_{\psi}}(\mathcal{M}_{\kappa}; (l_k^{\mathcal{K}_{ij}})^T \psi) dx \right)^{1/2} \\ & \quad \left. + \sum_{k=1}^m |\lambda_k^{\mathcal{C}}| \left( \int_{\kappa} E_{\mathcal{M}}^{s_u}(\mathcal{M}_{\kappa}; (r_k^{\mathcal{C}})^T u) dx \right)^{1/2} \left( \int_{\kappa} E_{\mathcal{M}}^{s_{\psi}}(\mathcal{M}_{\kappa}; (l_k^{\mathcal{C}})^T \psi) dx \right)^{1/2} \right] \end{aligned}$$

where  $s_u = \min(p+1, k_u)$ ,  $s_{\psi} = \min(p+1, k_{\psi})$ ,  $h_{\min} = \min_{\kappa \in \tilde{\kappa}} (\sigma_{\max}(\mathcal{M}_{\kappa})^{-1/2})$  is the minimum element length of all the face-sharing neighbors, and  $C$  only depends on the dimension  $d$  and the polynomial degree  $p$ . For an arbitrary matrix  $B$ ,  $\lambda_k^B$ ,  $r_k^B$ , and  $l_k^B$  denote the  $k$ -th eigenvalue, right eigenvector, and left eigenvector, respectively, i.e.  $B = \sum_{k=1}^m \lambda_k^B r_k^B (l_k^B)^T$ .

*Proof.* The proof follows from combining Theorems D.9, D.13, and D.15.  $\square$







## Appendix E

# On DWR Error Estimates for $p$ -Dependent Discretizations

This appendix analyzes the behavior of three variants of the dual-weighted residual (DWR) error estimates applied to the  $p$ -dependent discretization that results from the BR2 discretization of a second-order PDE. Three error estimates are assessed using two metrics: local effectivities and global effectivity. *A priori* error analysis is carried out to study the convergence behavior of the local and global effectivities of the three estimates. Numerical results verify the *a priori* error analysis. This analysis originally appeared in the technical report [154].

### E.1 $p$ -Dependence of DG Discretizations

Let  $u \in V$ , where  $V$  is some appropriate function space, be the weak solution to a general second-order PDE described by the semilinear form  $\mathcal{R}(\cdot, \cdot) : V \times V \rightarrow \mathbb{R}$ . That is,  $u$  satisfies

$$\mathcal{R}(u, v) = 0, \quad \forall v \in V.$$

The space  $V_{h,p}$  is a finite-dimensional space of piecewise polynomial functions of degree at most  $p$  on a triangulation  $\mathcal{T}_h$  of domain  $\Omega \subset \mathbb{R}^n$ , i.e.

$$V_{h,p} \equiv \{v_{h,p} \in L^2(\Omega) \mid v_{h,p}|_{\kappa} \in P^p(\kappa), \forall \kappa \in \mathcal{T}_h\},$$



where  $P^p(\kappa)$  denotes the space of  $p$ -th degree polynomial on element  $\kappa$ . A finite element approximation to the problem,  $u_{h,p} \in V_{h,p}$ , is induced by the semilinear form  $\mathcal{R}_{h,p}(\cdot, \cdot) : V_{h,p} \times V_{h,p} \rightarrow \mathbb{R}$  and satisfies

$$\mathcal{R}_{h,p}(u_{h,p}, v_{h,p}) = 0, \quad \forall v_{h,p} \in V_{h,p}.$$

**Definition E.1** ( $p$ -Dependence). *Let  $q < p$ . A semilinear form  $\mathcal{R}_{h,p}(\cdot, \cdot) : V_{h,p} \times V_{h,p} \rightarrow \mathbb{R}$  is said to be  $p$ -independent if*

$$\mathcal{R}_{h,p}(w_{h,q}, v_{h,q}) = \mathcal{R}_{h,q}(w_{h,q}, v_{h,q}), \quad \forall w_{h,q}, v_{h,q} \in V_{h,q} \subset V_{h,p}.$$

*If a semilinear form is not  $p$ -independent, then it is said to be  $p$ -dependent.*

We now show that the semilinear form arising from the second discretization of Bassi and Rebay (BR2)[25] of a second-order PDE is  $p$ -dependent. For simplicity, let us consider the Poisson equation with homogeneous Dirichlet boundary conditions on domain  $\Omega$ ,

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

The appropriate function space for the problem is  $V = H_0^1(\Omega)$ . The semilinear form is given by

$$\mathcal{R}(w, v) = \ell(w) - a(w, v), \tag{E.1}$$

where the source functional  $\ell \in V'$  and the bilinear form  $a(\cdot, \cdot) : V \times V \rightarrow \mathbb{R}$  are given by

$$\ell(w) = \int_{\Omega} f w dx \quad \text{and} \quad a(w, v) = \int_{\Omega} \nabla v \cdot \nabla w dx.$$

The BR2 discretization of the Poisson equation is given by the semilinear form

$$\mathcal{R}_{h,p}(w_{h,p}, v_{h,p}) = \ell_{h,p}(w_{h,p}) - a_{h,p}(w_{h,p}, v_{h,p}), \tag{E.2}$$



where

$$\begin{aligned}\ell_{h,p}(v_{h,p}) &= \ell(v_{h,p}) \\ a_{h,p}(w_{h,p}, v_{h,p}) &= \sum_{\kappa \in \mathcal{T}_h} \int_{\kappa} \nabla v_{h,p} \cdot \nabla w_{h,p} dx - \sum_{f \in \mathcal{F}_h} \int_f \{ \nabla v_{h,p} \} \cdot \llbracket w_{h,p} \rrbracket + \llbracket v_{h,p} \rrbracket \cdot \{ \nabla w_{h,p} \} ds \\ &\quad + \sum_{f \in \mathcal{F}_h} a_{h,p}^{f, \text{BR2}}(w_{h,p}, v_{h,p})\end{aligned}$$

where  $\mathcal{F}_h$  denotes the set of faces of the triangulation. On the interior faces, the jump operator,  $\llbracket \cdot \rrbracket$ , for a scalar quantity  $x$  is defined by

$$\llbracket x \rrbracket = x^- \hat{n}^- + x^+ \hat{n}^+.$$

and the average operator,  $\{ \cdot \}$ , for a vector quantity  $y$  is defined by

$$\{y\} = \frac{1}{2}(y^- + y^+).$$

Due to the homogeneous Dirichlet boundary condition, the operators on the boundary faces are given by (see e.g. [22] for general case)

$$\llbracket x \rrbracket = x \hat{n} \quad \text{and} \quad \{y\} = y.$$

The BR2 face penalty term for the face  $f \in \mathcal{F}_h$  is given by

$$a_{h,p}^{f, \text{BR2}}(w_{h,p}, v_{h,p}) = - \int_f \beta \llbracket v_{h,p} \rrbracket \cdot \left\{ r_{h,p}^f(\llbracket w_{h,p} \rrbracket) \right\} ds,$$

where the lifting operator,  $r_{h,p}^f(\llbracket w_{h,p} \rrbracket) \in [V_{h,p}^f]^d$ , satisfies

$$\sum_{\kappa \in \kappa_f} \int_{\kappa} g_{h,p} \cdot r_{h,p}^f(\llbracket w_{h,p} \rrbracket) dx = - \int_f \{g_{h,p}\} \cdot \llbracket w_{h,p} \rrbracket ds, \quad \forall g_{h,p} \in [V_{h,p}^f]^d,$$

where  $V_{h,p}^f \equiv \{v_{h,p} \in L^2(\kappa_f) \mid v_{h,p}|_{\kappa} \in P^p(\kappa), \kappa \in \kappa_f\}$  with  $\kappa_f$  denoting the set of elements neighboring face  $\kappa$ . The stability parameter,  $\beta$ , must be set to a number greater than the number of faces for coercivity [10].



**Theorem E.2.** *The BR2 lifting operator,  $r_{h,p}^f(\cdot)$ , is  $p$ -dependent in the sense that*

$$r_{h,q}^f(\llbracket w_{h,q} \rrbracket) \neq r_{h,p}^f(\llbracket w_{h,q} \rrbracket)$$

for some  $w_{h,q} \in V_{h,q}$  with  $q < p$ .

*Proof.* By definition, the lifting operator,  $r_{h,p}^f(\llbracket w_{h,q} \rrbracket)$ , satisfies

$$\sum_{\kappa \in \kappa_f} \int_{\kappa} g_{h,p} \cdot r_{h,p}^f(\llbracket w_{h,q} \rrbracket) dx = - \int_f \{g_{h,p}\} \cdot \llbracket w_{h,q} \rrbracket ds, \quad \forall g_{h,p} \in [V_{h,p}^f]^d.$$

Because  $V_{h,p}^f$  is finite dimensional, there exist basis functions that span  $V_{h,p}^f$ . In particular, let us denote the basis functions that span the restriction of  $V_{h,p}^f$  to  $\kappa$ , one of the elements in  $\kappa_f$ , by  $\{\phi_m\}$ . The dimension of  $V_{h,p}|_{\kappa}$  is  $\mathcal{N}(p)$ , where  $\mathcal{N}(p)$  is the dimension of the  $p$ -th degree polynomial space. For example, for triangular elements,  $\mathcal{N}(p) = (p+1)(p+2)/2$ . We will chose  $\phi_m$  to be a hierarchical orthogonal basis with respect to  $\kappa$ , i.e.,

$$\begin{aligned} \phi_m &\in P^r(\kappa), \quad \forall m \leq \mathcal{N}(r) \\ \int_{\kappa} \phi_n \phi_m dx &= \begin{cases} c_n, & n = m \\ 0, & n \neq m. \end{cases} \end{aligned}$$

The  $i$ -th spatial component of the lifting operator restricted to element  $\kappa$ ,  $r_{h,p}^{f,i}(\llbracket w_{h,q} \rrbracket)|_{\kappa}$ , can be represented as

$$r_{h,p}^{f,i}(\llbracket w_{h,q} \rrbracket)|_{\kappa} = \sum_{n=1}^{\mathcal{N}(p)} B_n^i \phi_n$$

where  $B^i \in \mathbb{R}^{\mathcal{N}(p)}$ . The coefficients,  $B^i$ , of the lifting operator restricted to  $\kappa$  must satisfy the system of algebraic equations

$$\sum_{n=1}^{\mathcal{N}(p)} \left[ \int_{\kappa} \phi_m \phi_n dx \right] B_n^i = -\alpha \int_f \phi_m n_i \cdot \llbracket w_{h,q} \rrbracket ds, \quad \forall m = 1, \dots, \mathcal{N}(p),$$

where  $\alpha = 1/2$  on the interior face and  $\alpha = 1$  on the boundary face. Due to the orthogonality



of the basis functions, we arrive at an explicit expression for the coefficients,

$$B_n^i = -\frac{\alpha}{c_n} \int_f \phi_n \hat{n}_i \cdot \llbracket w_{h,q} \rrbracket ds, \quad n = 1, \dots, \mathcal{N}(p).$$

The face integral term does not vanish in general. In particular,

$$B_n^i = -\frac{\alpha}{c_n} \int_f \phi_n \hat{n}_i \cdot \llbracket w_{h,q} \rrbracket ds \neq 0, \quad n = \mathcal{N}(q) + 1, \dots, \mathcal{N}(p),$$

for some  $\llbracket w_{h,q} \rrbracket \in P^q(f)$ . Having finite coefficients for  $n > \mathcal{N}(q)$ , the lifting operator  $r_{h,p}^{f,i}(\llbracket w_{h,q} \rrbracket)|_\kappa$  is not in the space  $P^q(\kappa)$ . In contrast,  $r_{h,q}^{f,i}(\llbracket w_{h,q} \rrbracket)|_\kappa \in P^q(\kappa)$  by construction. Thus,  $r_{h,q}^{f,i}(\llbracket w_{h,q} \rrbracket) \neq r_{h,p}^{f,i}(\llbracket w_{h,q} \rrbracket)$  and the lifting operator is  $p$ -dependent.  $\square$

As the lifting operator is  $p$ -dependent, the semilinear form arising from the BR2 discretization of a second-order PDE is  $p$ -dependent.

**Remark E.1.** *The interior penalty (IP) DG discretization is also  $p$ -dependent. The bilinear form for the IP method is obtained by replacing the BR2 face penalty term,  $a_{h,p}^{f,BR2}(\cdot, \cdot) : V_{h,p} \times V_{h,p} \rightarrow \mathbb{R}$ , with the IP face penalty term,*

$$a_{h,p}^{f,IP}(w_{h,p}, v_{h,p}) = C^{IP} \int_f \frac{p^2}{h} \llbracket v_{h,p} \rrbracket \cdot \llbracket w_{h,p} \rrbracket ds,$$

*which is  $p$ -dependent due to the explicit presence of the  $p^2$  term.*

## E.2 The Dual-Weighted Residual Error Estimation

In this section, we review the dual-weighted residual (DWR) error estimate of Becker and Rannacher [26, 27] applied to the DG methods.

### E.2.1 Problem Setup

For simplicity, we consider the Poisson equation with homogeneous Dirichlet boundary conditions, as in Section E.1, with a linear output functional of the form

$$\mathcal{J}(w) = \mathcal{J}_{h,p}(w) = -\ell^O(w) = -\int_\Omega g w dx,$$



for some  $g \in L^2(\Omega)$ . Our objective is to quantify

$$\mathcal{E} \equiv \mathcal{J}_{h,p}(u_{h,p}) - \mathcal{J}(u),$$

where  $u \in V$  and  $u_{h,p} \in V_{h,p}$  satisfy the residual expressions Eq. (E.1) and (E.2), respectively. In the DWR framework, the output error is quantified in terms of the adjoint solution,  $\psi$ . For the Poisson problem of interest, the strong form of the dual problem is given by

$$\begin{aligned} -\Delta\psi &= g \quad \text{in } \Omega \\ \psi &= 0 \quad \text{on } \partial\Omega. \end{aligned}$$

Equivalently, the weak form of the dual problem is: Find  $\psi \in V = H_0^1(\Omega)$  such that

$$\mathcal{R}^\psi(v, \psi) = \ell^O(v) - a(v, \psi) = 0, \quad \forall v \in V.$$

Similarly, the finite element approximation to the dual problem is: Find  $\psi_{h,p} \in V_{h,p}$  such that

$$\mathcal{R}_{h,p}^\psi(v_{h,p}, \psi_{h,p}) = \ell^O(v_{h,p}) - a_{h,p}(v_{h,p}, \psi_{h,p}) = 0, \quad \forall v_{h,p} \in V_{h,p}.$$

### E.2.2 Local and Global Consistency Results

Let us develop properties of the discrete primal and dual residual that facilitate the development of error estimates for the DG method.

**Lemma E.3** (Extended Local Consistency). *The semilinear form possesses local consistency in the following sense: Given the true solution,  $u \in V = H^1(\Omega)$ , the residual satisfies*

$$\mathcal{R}_{h,p}(u, v|_\kappa) = 0, \quad \forall v \in H^1(\Omega),$$

where  $v|_\kappa \in L^2(\Omega)$  is understood as the restriction of  $v$  to  $\kappa$  with zero extension in  $\Omega \setminus \kappa$ .

Similarly, given the true adjoint,  $\psi \in V$ , the adjoint residual satisfies

$$R_{h,p}^\psi(v|_\kappa, \psi) = 0, \quad \forall v \in H^1(\Omega).$$



These results are referred to as the extended local primal and dual consistency, respectively, because it encompasses the traditional statement of local consistency for  $v|_\kappa \in V_{h,p}(\kappa) \subset H^1(\kappa)$ .

*Proof.* Since  $u \in H^1(\Omega)$ , all terms related to jumps in  $u$  in the primal residual vanish. The remaining expression is

$$\begin{aligned}\mathcal{R}_{h,p}(u, v|_\kappa) &= \ell(v|_\kappa) - a_{h,p}(u, v|_\kappa) \\ &= \sum_{\kappa' \in \mathcal{T}_h} \int_{\kappa'} f v|_\kappa dx - \sum_{\kappa' \in \mathcal{T}_h} \int_{\kappa'} \nabla v|_\kappa \cdot \nabla u dx + \sum_{f \in \mathcal{F}_h} \int_f \llbracket v|_\kappa \rrbracket \cdot \{\nabla u\} ds \\ &= \int_\kappa f v dx - \int_\kappa \nabla v \cdot \nabla u dx + \int_{\partial\kappa} v \hat{n} \cdot \nabla u ds \\ &= \int_\kappa v(f + \Delta u) dx = 0, \quad \forall v \in V = H^1(\Omega).\end{aligned}$$

Similarly, since  $\psi \in H^1(\Omega)$ , all terms related to jumps in  $\psi$  in the dual residual vanish. The remaining expression is

$$\begin{aligned}\mathcal{R}_{h,p}^\psi(v|_\kappa, \psi) &= \ell^O(v|_\kappa) - a_{h,p}(v|_\kappa, \psi) \\ &= \sum_{\kappa' \in \mathcal{T}_h} \int_{\kappa'} g v|_\kappa dx - \sum_{\kappa' \in \mathcal{T}_h} \int_{\kappa'} \nabla \psi \cdot \nabla v|_\kappa dx + \sum_{f \in \mathcal{F}_h} \int_f \{\nabla \psi\} \llbracket v|_\kappa \rrbracket ds \\ &= \int_\kappa g v dx - \int_\kappa \nabla \psi \cdot \nabla v dx + \int_{\partial\kappa} \nabla \psi \cdot \hat{n} v ds \\ &= \int_\kappa v(g + \Delta \psi) dx = 0, \quad \forall v \in V = H^1(\Omega).\end{aligned}$$

□

**Lemma E.4** (Extended Global Consistency). *Given the true primal solution,  $u \in V = H^1(\Omega)$ , the discrete primal residual is globally consistent in the sense that*

$$\mathcal{R}_{h,p_2}(u, v) = 0, \quad \forall v \in V_{h,p_1} \oplus V, \forall p_1, p_2 \in \mathbb{N}$$

*Similarly, given the true dual solution,  $\psi \in V = H^1(\Omega)$ , the discrete dual residual is globally consistent in the sense that*

$$\mathcal{R}_{h,p_2}^\psi(v, \psi) = 0, \quad \forall v \in V_{h,p_1} \oplus V, \forall p_1, p_2 \in \mathbb{N}$$



*Proof.* First, we note that

$$\begin{aligned} V_{h,p_1} \oplus V &= (\oplus_{\kappa} V_{h,p_1}(\kappa)) \oplus V \subset (\oplus_{\kappa} V_{h,p_1}(\kappa)) \oplus (\oplus_{\kappa} H^1(\kappa)) \\ &= \oplus_{\kappa} (V_{h,p_1}(\kappa) \oplus H^1(\kappa)) = \oplus_{\kappa} H^1(\kappa). \end{aligned}$$

The proof then follows from the extended local consistency. Since  $v = \sum_{\kappa \in \mathcal{T}_h} v|_{\kappa}$ , we have

$$\mathcal{R}_{h,p_2}(u, v) = \sum_{\kappa \in \mathcal{T}_h} \mathcal{R}_{h,p_2}(u, v|_{\kappa}) = 0, \quad \forall v \in \oplus_{\kappa} H^1(\kappa) \supset (V_{h,p_1} \oplus V),$$

where the second equality follows from the extended local consistency, i.e.,  $\mathcal{R}_{h,p_2}(u, v|_{\kappa}) = 0$ ,  $\forall v \in H^1(\kappa)$ . The proof for the global dual consistency is identical.  $\square$

### E.2.3 DWR Error Estimates

**Theorem E.5** (Functional Error Representation Formula). *The error in the finite element approximation of the output,  $\mathcal{J}_{h,p}(u_{h,p})$ , is represented in terms of the adjoint solution,  $\psi \in V$ , by*

$$\mathcal{E} \equiv \mathcal{J}_{h,p}(u_{h,p}) - \mathcal{J}(u) = \mathcal{R}_{h,p}(u_{h,p}, \psi - \psi_{h,p}).$$

*Proof.* Using the definition of the adjoint, we obtain the error representation formula

$$\begin{aligned} \mathcal{E} &\equiv \mathcal{J}_{h,p}(u_{h,p}) - \mathcal{J}(u) = \ell^O(u - u_{h,p}) \\ &= a_{h,p}(u - u_{h,p}, \psi) && \text{(extended global dual consistency)} \\ &= a_{h,p}(u - u_{h,p}, \psi - \psi_{h,p}) && \text{(Galerkin orthogonality)} \\ &= \ell(\psi - \psi_{h,p}) - a_{h,p}(u_{h,p}, \psi - \psi_{h,p}) && \text{(extended global primal consistency)} \\ &= \mathcal{R}_{h,p}(u_{h,p}, \psi - \psi_{h,p}). \end{aligned}$$

Note that  $\psi_{h,p}$  could be replaced by any  $v_{h,p} \in V_{h,p}$  since  $\mathcal{R}_{h,p}(u_{h,p}, v_{h,p}) = 0$ ,  $\forall v_{h,p} \in V_{h,p}$ .  $\square$

**Definition E.6** (Local Functional Error Representation Formula). *The functional output*



error,  $\mathcal{E}$ , is localized to element  $\kappa$  according to

$$\eta_\kappa \equiv \mathcal{R}_{h,p}(u_{h,p}, (\psi - \psi_{h,p})|_\kappa).$$

Let us state a few important properties of the local error  $\eta_\kappa$ . First, the output error is the sum of the local errors, i.e.

$$\mathcal{E} = \sum_{\kappa \in \mathcal{T}_h} \eta_\kappa.$$

Second, the local error representation requires that a local residual, which vanishes with mesh refinement, results from the elemental restriction of test functions. While DG discretizations have this property, continuous Galerkin discretizations do not. For continuous Galerkin discretizations, the global error representation formula must be integrated by parts to yields an expression with the strong form of residual, which vanishes with mesh refinement.

In practice, the true adjoint,  $\psi \in V$ , is not computable. Thus, we replace the adjoint with the surrogate solution obtained on a enriched space, i.e.,  $\psi_{h,\hat{p}} \in V_{h,\hat{p}}$  such that

$$R_{h,\hat{p}}^\psi(v_{h,\hat{p}}, \psi_{h,\hat{p}}) = 0, \quad \forall v_{h,\hat{p}} \in V_{h,\hat{p}},$$

for some  $\hat{p} = p + p_{\text{inc}} > p$ , where  $p_{\text{inc}}$  is the increase in the polynomial degree in the enrichment process.

We now introduce three different forms of the error estimates.

**Definition E.7** (Error Estimate 1). *The error estimate 1 is given by*

$$\begin{aligned} \mathcal{E}^{(1)} &\equiv \mathcal{R}_{h,\hat{p}}(u_{h,p}, \psi_{h,\hat{p}} - \psi_{h,p}) \\ \eta_\kappa^{(1)} &\equiv \mathcal{R}_{h,\hat{p}}(u_{h,p}, (\psi_{h,\hat{p}} - \psi_{h,p})|_\kappa). \end{aligned}$$

The error estimate 1 arises naturally if the discrete formulation of the adjoint is used (see, e.g., [34, 58, 145]).



**Definition E.8** (Error Estimate 2). *The error estimate 2 is given by*

$$\begin{aligned}\mathcal{E}^{(2)} &\equiv \mathcal{R}_{h,\hat{p}}(u_{h,p}, \psi_{h,\hat{p}}) \\ \eta_{\kappa}^{(2)} &\equiv \mathcal{R}_{h,\hat{p}}(u_{h,p}, \psi_{h,\hat{p}}|_{\kappa}).\end{aligned}$$

Error estimate 2 eliminates the need to compute  $\psi_{h,p}$  by appealing to the local Galerkin orthogonality of DG discretizations, and this is one of the error estimates advocated in [55]. However, with the form presented, the local Galerkin orthogonality does not hold due to the  $p$ -dependence of the semilinear form. In particular, while

$$\mathcal{R}_{h,p}(u_{h,p}, v_{h,p}) = 0, \quad \forall v_{h,p} \in V_{h,p},$$

the same does not hold if the  $p$  about which the residual is evaluated is replaced by  $\hat{p} \neq p$ , i.e.,

$$\mathcal{R}_{h,\hat{p}}(u_{h,p}, v_{h,p}) \neq 0, \quad \text{for some } v_{h,p} \in V_{h,p}.$$

This implies that

$$\begin{aligned}\mathcal{E}^{(2)} &\equiv \mathcal{R}_{h,\hat{p}}(u_{h,p}, \psi_{h,\hat{p}}) \neq \mathcal{R}_{h,\hat{p}}(u_{h,p}, \psi_{h,\hat{p}} - \psi_{h,p}) \equiv \mathcal{E}^{(1)} \\ \eta_{\kappa}^{(2)} &\equiv \mathcal{R}_{h,\hat{p}}(u_{h,p}, \psi_{h,\hat{p}}|_{\kappa}) \neq \mathcal{R}_{h,\hat{p}}(u_{h,p}, (\psi_{h,\hat{p}} - \psi_{h,p})|_{\kappa}) \equiv \eta_{\kappa}^{(1)},\end{aligned}$$

and the error estimate 2 is different from error estimate 1.

**Definition E.9** (Error Estimate 3). *The error estimate 3 is given by*

$$\begin{aligned}\mathcal{E}^{(3)} &\equiv \mathcal{R}_{h,p}(u_{h,p}, \psi_{h,\hat{p}}) \\ \eta_{\kappa}^{(3)} &\equiv \mathcal{R}_{h,p}(u_{h,p}, \psi_{h,\hat{p}}|_{\kappa}).\end{aligned}$$

The error estimate 3 is obtained by simply replacing  $\psi$  in the error representation formula by  $\psi_{h,\hat{p}}$ . Note that because the residual is evaluated about  $p$ , the Galerkin orthogonality



holds, and we have

$$\begin{aligned}\mathcal{E}^{(3)} &\equiv \mathcal{R}_{h,p}(u_{h,p}, \psi_{h,\hat{p}}) = \mathcal{R}_{h,p}(u_{h,p}, \psi_{h,\hat{p}} - v_{h,p}) \quad \forall v_{h,p} \in V_{h,p} \\ \eta_\kappa^{(3)} &\equiv \mathcal{R}_{h,p}(u_{h,p}, \psi_{h,\hat{p}}|_\kappa) = \mathcal{R}_{h,p}(u_{h,p}, (\psi_{h,\hat{p}} - v_{h,p})|_\kappa) \quad \forall v_{h,p} \in V_{h,p}.\end{aligned}$$

#### E.2.4 Assessment of the Error Estimates

For each of the error estimates considered, we will develop a bound for the absolute error in the global error estimate

$$|\mathcal{E} - \mathcal{E}^{(i)}|$$

and the absolute error in the local error estimate

$$|\eta_\kappa - \eta_\kappa^{(i)}|.$$

In practice, however, we are more interested in the quality of the error estimates with respect to the true error. In particular, we want to ensure that the error in the error estimate is a small fraction of the true error; otherwise the estimates would be useless. The relative error in the global error estimate  $i$  is given by

$$\theta_{\text{global}}^{(i)} \equiv \frac{|\mathcal{E} - \mathcal{E}^{(i)}|}{|\mathcal{E}|}.$$

The relative error is related to the error effectivity  $I^{\text{eff}}$  defined in, for example, [26, 27] by

$$\frac{|\mathcal{E} - \mathcal{E}^{(i)}|}{|\mathcal{E}|} = \left| \frac{\mathcal{E} - \mathcal{E}^{(i)}}{\mathcal{E}} \right| = \left| 1 - \frac{\mathcal{E}^{(i)}}{\mathcal{E}} \right| = |1 - I^{\text{eff}}|.$$

That is, the relative error measures the deviation of the error effectivity from unity. Ideally, the effectivity of the error estimate should improve with mesh refinement such that  $I^{\text{eff}} \rightarrow 1$  as  $h \rightarrow 0$ . Equivalently, the relative error should ideally vanish as  $h \rightarrow 0$ .

Similarly, the relative error in the local error estimate  $i$  is given by

$$\theta_{\text{local},\kappa}^{(i)} \equiv \frac{|\eta_\kappa - \eta_\kappa^{(i)}|}{|\eta_\kappa|} = \left| 1 - \frac{\eta_\kappa^{(i)}}{\eta_\kappa} \right|.$$



Again, the relative error in the local error estimate measures the deviation of the local error effectivity from unity.

### E.3 *A Priori* Error Analysis

In this section, we perform *a priori* analysis of the three error estimates to establish the bound on the output estimation errors. In particular, we are interested in the convergence of the estimates with grid refinement.

Throughout this section, we will use the notation  $A \lesssim B$  to imply that  $A \leq cB$  for some  $c < \infty$  independent of  $h$ , in order to avoid proliferation of constants. Similarly,  $A \gtrsim B$  implies that  $A \geq cB$  for some  $c > 0$  independent of  $h$ . Moreover,  $A \approx B$  implies that  $A \lesssim B$  and  $B \lesssim A$ .

#### E.3.1 Assumptions

We assume that the DG-FEM approximation to both the primal and the dual problems are optimal in the  $L^2$  sense, i.e.,

$$\begin{aligned} \|u - u_{h,p}\|_{L^2(\kappa)} &\lesssim \|u - \Pi_{h,p}u\|_{L^2(\kappa)}, \quad \forall \kappa \in \mathcal{T}_h \\ \|\psi - \psi_{h,p}\|_{L^2(\kappa)} &\lesssim \|\psi - \Pi_{h,p}\psi\|_{L^2(\kappa)} \quad \forall \kappa \in \mathcal{T}_h, \end{aligned}$$

where  $\Pi_{h,p} : V \rightarrow V_{h,p}$  is the  $L^2$  projection operator such that  $\Pi_{h,p}v \in V_{h,p}$  satisfies

$$\|v - \Pi_{h,p}v\|_{L^2(\Omega)} = \inf_{w_{h,p} \in V_{h,p}} \|v - w_{h,p}\|_{L^2(\Omega)}.$$

Furthermore, we will assume  $u$  and  $\psi$  are analytic for convenience. Under the analyticity assumption, the scaling argument results in the following interpolation results:

$$\begin{aligned} \|v - \Pi_{h,p}v\|_{H^m(\kappa)} &\lesssim h^{p+1-m} \|v\|_{H^{p+1}(\kappa)} \\ \|v - \Pi_{h,p}v\|_{H^m(f)} &\lesssim h^{p+1/2-m} \|v\|_{H^{p+1}(\kappa)}. \end{aligned}$$

#### E.3.2 Useful Relationships

This section introduces lemmas that facilitate the development of the error bounds for the output error estimates.



**Lemma E.10** (Local Residual-Error Mapping). *For all  $p_1, p_2, p_3 \in \mathbb{N}$ , the local dual-weighted residual can be represented as*

$$\mathcal{R}_{h,p_3}(w_{h,p_1}, (\psi - v_{h,p_2})|_\kappa) = a_{h,p_3}(u - w_{h,p_1}, (\psi - v_{h,p_2})|_\kappa), \quad \forall w_{h,p_1} \in V_{h,p_1}, v_{h,p_2} \in V_{h,p_2}.$$

where  $u$  and  $\psi$  are the solutions to the primal and dual problems respectively.

*Proof.* The proof relies on the definition of the primal residual, the extended local consistency (Lemma E.3), and the linearity of the bilinear form, i.e.,

$$\begin{aligned} \mathcal{R}_{h,p_3}(w_{h,p_1}, (\psi - v_{h,p_2})|_\kappa) &\equiv \ell_{h,p_3}((\psi - v_{h,p_2})|_\kappa) - a_{h,p_3}(w_{h,p_1}, (\psi - v_{h,p_2})|_\kappa) \\ &= a_{h,p_3}(u, (\psi - v_{h,p_2})|_\kappa) - a_{h,p_3}(w_{h,p_1}, (\psi - v_{h,p_2})|_\kappa) \\ &= a_{h,p_3}(u - w_{h,p_1}, (\psi - v_{h,p_2})|_\kappa), \quad \forall w_{h,p_1} \in V_{h,p_1}, v_{h,p_2} \in V_{h,p_2}. \end{aligned}$$

□

**Lemma E.11** (Global Residual-Error Mapping). *For all  $p_1, p_2, p_3 \in \mathbb{N}$ , the global dual-weighted residual can be represented as*

$$\begin{aligned} \mathcal{R}_{h,p_3}(w_{h,p_1}, \psi - v_{h,p_2}) &= a_{h,p_3}(u - w_{h,p_1}, \psi - v_{h,p_2}) = R_{h,p_3}^\psi(u - w_{h,p_1}, v_{h,p_2}) \\ &\quad \forall w_{h,p_1} \in V_{h,p_1}, v_{h,p_2} \in V_{h,p_2}. \end{aligned}$$

where  $u$  and  $\psi$  are the solutions to the primal and dual problems respectively.

*Proof.* The first equality follows from the local residual-error mapping, i.e.,

$$\begin{aligned} \mathcal{R}_{h,p_3}(w_{h,p_1}, \psi - v_{h,p_2}) &= \sum_{\kappa \in \mathcal{T}_h} \mathcal{R}_{h,p_3}(w_{h,p_1}, (\psi - v_{h,p_2})|_\kappa) = \sum_{\kappa \in \mathcal{T}_h} a_{h,p_3}(u - w_{h,p_1}, (\psi - v_{h,p_2})|_\kappa) \\ &= a_{h,p_3}(u - w_{h,p_1}, \psi - v_{h,p_2}), \quad \forall w_{h,p_1} \in V_{h,p_1}, v_{h,p_2} \in V_{h,p_2}. \end{aligned}$$

The second equality results from the definition of the adjoint residual and the extended



global consistency, i.e.,

$$\begin{aligned}
R_{h,p_3}^\psi(u - w_{h,p_1}, v_{h,p_2}) &\equiv \ell_{h,p_3}^O(u - w_{h,p_1}) - a_{h,p_3}(u - w_{h,p_1}, v_{h,p_2}) \\
&= a_{h,p_3}(u - w_{h,p_1}, \psi) - a_{h,p_3}(u - w_{h,p_1}, v_{h,p_2}) \\
&= a_{h,p_3}(u - w_{h,p_1}, \psi - v_{h,p_2}), \quad \forall w_{h,p_1} \in V_{h,p_1}, v_{h,p_2} \in V_{h,p_2}.
\end{aligned}$$

□

**Lemma E.12** ( *$h$ -Scaling of the Lifting Operator*). *The BR2 lifting operator is bounded by the face jump according to*

$$\|r_{h,p}^f(\llbracket v \rrbracket)\|_{L^2(\kappa_f)} \lesssim h^{-1/2} \|\llbracket v \rrbracket\|_{L^2(f)}.$$

*Proof.* The lemma is stated in, for example, [33]. Here, we present the proof for completeness. The inequality follows from setting the test function equal to  $r_{h,p}^f(\llbracket v \rrbracket)$  in the definition of the lifting operator, applying the Schwarz inequality, and invoking the trace scaling argument, i.e.,

$$\begin{aligned}
\|r_{h,p}^f(\llbracket v \rrbracket)\|_{L^2(\kappa_f)}^2 &= \int_{\kappa_f} r_{h,p}^f(\llbracket v \rrbracket) \cdot r_{h,p}^f(\llbracket v \rrbracket) dx \\
&= \int_f \{r_{h,p}^f(\llbracket v \rrbracket)\} \cdot \llbracket v \rrbracket ds && \text{(definition of lifting operator)} \\
&\leq \|r_{h,p}^f(\llbracket v \rrbracket)\|_{L^2(f)} \|\llbracket v \rrbracket\|_{L^2(f)} && \text{(Schwarz)} \\
&\lesssim h^{-1/2} \|r_{h,p}^f(\llbracket v \rrbracket)\|_{L^2(\kappa_f)} \|\llbracket v \rrbracket\|_{L^2(f)}. && \text{(trace scaling)}
\end{aligned}$$

Division of the both sides by  $\|r_{h,p}^f(\llbracket v \rrbracket)\|_{L^2(\kappa_f)}$  yields the desired result,

$$\|r_{h,p}^f(\llbracket v \rrbracket)\|_{L^2(\kappa_f)} \lesssim h^{-1/2} \|\llbracket v \rrbracket\|_{L^2(f)}.$$

□

**Remark E.2.** *The face jump is also bounded by the lifting operator as  $h^{-1/2} \|\llbracket v \rrbracket\|_{L^2(f)} \lesssim \|r_{h,p}^f(\llbracket v \rrbracket)\|_{L^2(\kappa_f)}$ . The proof is provided in [33].*

**Lemma E.13** (Local Bilinear Form Error Bound). *Under the optimality assumption, the*



following error bound holds on element  $\kappa$  for all  $p_1, p_2, p_3 \in \mathbb{N}$ :

$$|a_{h,p_3}(u - u_{h,p_1}, (\psi - \psi_{h,p_2})|_{\kappa})| \lesssim h^{p_1+p_2} \|u\|_{H^{p_1+1}(\tilde{\kappa})} \|\psi\|_{H^{p_2+1}(\kappa)},$$

where  $u$  and  $\psi$  are the true solution to the primal and dual problems, respectively, and  $u_{h,p_1}$  and  $\psi_{h,p_2}$  are the DG-FEM approximation to the primal and dual problems, respectively, and  $\tilde{\kappa}$  is the set of elements sharing common face with  $\kappa$ .

*Proof.* Substitution of the expression for the bilinear form yields

$$\begin{aligned} & a_{h,p_3}(u - u_{h,p_1}, (\psi - \psi_{h,p_2})|_{\kappa}) \\ &= \underbrace{\sum_{\kappa'} \int_{\kappa'} \nabla(\psi - \psi_{h,p_2})|_{\kappa} \cdot \nabla(u - u_{h,p_1}) dx}_{(I)} - \underbrace{\sum_f \int_f \{\nabla(\psi - \psi_{h,p_2})|_{\kappa}\} \cdot \llbracket u - u_{h,p_1} \rrbracket ds}_{(II)} \\ & \quad - \underbrace{\sum_f \int_f \llbracket (\psi - \psi_{h,p_2})|_{\kappa} \rrbracket \cdot \{\nabla(u - u_{h,p_1})\} ds}_{(III)} - \underbrace{\sum_f \int_f \beta \llbracket (\psi - \psi_{h,p_2})|_{\kappa} \rrbracket \cdot \left\{ r_{h,p_3}^f(\llbracket u - u_{h,p_1} \rrbracket) \right\} ds}_{(IV)} \end{aligned}$$

Now we bound each one of the braced terms. The interior term becomes

$$\begin{aligned} |(I)| &= \left| \sum_{\kappa'} \int_{\kappa'} \nabla(\psi - \psi_{h,p_2})|_{\kappa} \cdot \nabla(u - u_{h,p_1}) dx \right| = \left| \int_{\kappa} \nabla(\psi - \psi_{h,p_2}) \cdot \nabla(u - u_{h,p_1}) dx \right| \\ &\leq \|\psi - \psi_{h,p_2}\|_{H^1(\kappa)} \|u - u_{h,p_1}\|_{H^1(\kappa)} \lesssim h^{p_1+p_2} \|\psi\|_{H^{p_2+1}(\kappa)} \|u\|_{H^{p_1+1}(\kappa)} \end{aligned}$$

The first face term is bounded by

$$\begin{aligned} |(II)| &= \left| \sum_f \int_f \{\nabla(\psi - \psi_{h,p_2})|_{\kappa}\} \cdot \llbracket u - u_{h,p_1} \rrbracket ds \right| = \left| \int_{\partial\kappa} \alpha \nabla(\psi - \psi_{h,p_2}) \cdot \llbracket u - u_{h,p_1} \rrbracket ds \right| \\ &\leq \|\alpha \nabla(\psi - \psi_{h,p_2})\|_{L^2(\partial\kappa)} \|\llbracket u - u_{h,p_1} \rrbracket\|_{L^2(\partial\kappa)} \\ &\lesssim h^{p_2-1/2} \|\psi\|_{H^{p_2+1}(\kappa)} h^{p_1+1/2} \|u\|_{H^{p_1+1}(\tilde{\kappa})} = h^{p_1+p_2} \|\psi\|_{H^{p_2+1}(\kappa)} \|u\|_{H^{p_1+1}(\tilde{\kappa})}, \end{aligned}$$

where  $\alpha = 1$  if  $f$  is a boundary face, and  $\alpha = 1/2$  if  $f$  is an interior face. The second face



term is bounded in a similar manner as the first term, resulting in

$$\begin{aligned}
|(\text{III})| &= \left| \sum_f \int_f \llbracket (\psi - \psi_{h,p_2})|_\kappa \rrbracket \cdot \{\nabla(u - u_{h,p_1})\} ds \right| = \left| \int_{\partial\kappa} (\psi - \psi_{h,p_2}) \hat{n} \cdot \{\nabla(u - u_{h,p_1})\} ds \right| \\
&\lesssim h^{p_1+p_2} \|\psi\|_{H^{p_2+1}(\kappa)} \|u\|_{H^{p_1+1}(\tilde{\kappa})}
\end{aligned}$$

Finally, we bound the term involving the lifting operator as

$$\begin{aligned}
|(\text{IV})| &= \left| \sum_{f \in \mathcal{F}} \int_f \beta \llbracket (\psi - \psi_{h,p_2})|_\kappa \rrbracket \cdot \left\{ r_{h,p_3}^f(\llbracket u - u_{h,p_1} \rrbracket) \right\} ds \right| \\
&= \left| \sum_{f \in \partial\kappa} \int_f \beta (\psi - \psi_{h,p_2}) \hat{n} \cdot \left\{ r_{h,p_3}^f(\llbracket u - u_{h,p_1} \rrbracket) \right\} \right| && \text{(finite support of } (\psi - \psi_{h,p_2})|_\kappa) \\
&\leq \sum_{f \in \partial\kappa} \beta \|\psi - \psi_{h,p_2}\|_{L^2(f)} \left\| \left\{ r_{h,p_3}^f(\llbracket u - u_{h,p_1} \rrbracket) \right\} \right\|_{L^2(f)} && \text{(Schwarz inequality)} \\
&\lesssim \sum_{f \in \partial\kappa} \|\psi - \psi_{h,p_2}\|_{L^2(f)} h^{-1/2} \left\| \left\{ r_{h,p_3}^f(\llbracket u - u_{h,p_1} \rrbracket) \right\} \right\|_{L^2(\kappa_f)} && \text{(trace scaling)} \\
&\lesssim \sum_{f \in \partial\kappa} \|\psi - \psi_{h,p_2}\|_{L^2(f)} h^{-1} \|\llbracket u - u_{h,p_1} \rrbracket\|_{L^2(f)} && \text{(Lemma E.12)} \\
&\lesssim h^{p_1+p_2} \|\psi\|_{H^{p_2+1}(\kappa)} \|u\|_{H^{p_1+1}(\tilde{\kappa})} && (L^2 \text{ optimality assumption})
\end{aligned}$$

Combining the bounds for (I), (II), (III), and (IV), we obtain the desired result:

$$|\eta_\kappa| \leq |(\text{I})| + |(\text{II})| + |(\text{III})| + |(\text{IV})| \lesssim h^{p_1+p_2} \|\psi\|_{H^{p_2+1}(\kappa)} \|u\|_{H^{p_1+1}(\tilde{\kappa})}.$$

□

**Lemma E.14** (Global Bilinear Form Error Bound). *Under the optimality assumption, the following error bound holds for all  $p_1, p_2, p_3 \in \mathbb{N}$ :*

$$|a_{h,p_3}(u - u_{h,p_1}, \psi - \psi_{h,p_2})| \lesssim h^{p_1+p_2} \|u\|_{H^{p_1+1}(\Omega)} \|\psi\|_{H^{p_2+1}(\Omega)}$$

where  $u_{h,p_1}$  and  $\psi_{h,p_2}$  are the DG-FEM approximation to the primal and dual problems, respectively.



*Proof.* The global error bound is a direct consequence of the local error bound, i.e.,

$$\begin{aligned}
|a_{h,p_3}(u - u_{h,p_1}, \psi - \psi_{h,p_2})| &= \left| \sum_{\kappa \in \mathcal{T}_h} a_{h,p_3}(u - u_{h,p_1}, (\psi - \psi_{h,p_2})|_{\kappa}) \right| \\
&\lesssim \sum_{\kappa \in \mathcal{T}_h} h^{p_1+p_2} \|u\|_{H^{p_1+1}(\tilde{\kappa})} \|\psi\|_{H^{p_2+1}(\kappa)} \\
&\leq h^{p_1+p_2} \left( \sum_{\kappa \in \mathcal{T}_h} \|u\|_{H^{p_1+1}(\tilde{\kappa})} \right) \left( \sum_{\kappa \in \mathcal{T}_h} \|\psi\|_{H^{p_2+1}(\kappa)} \right) \\
&\lesssim h^{p_1+p_2} \|u\|_{H^{p_1+1}(\Omega)} \|\psi\|_{H^{p_2+1}(\Omega)}
\end{aligned}$$

□

### E.3.3 *A Priori* Error Analysis of the True Output Error

In this section, we analyze the convergence behavior of the true output error.

**Theorem E.15** (Convergence of True Error). *Let  $u_{h,p} \in V_{h,p}$  be the DG-FEM solution to the Poisson equation. The local and global error are bounded by*

$$\begin{aligned}
|\eta_{\kappa}| &\lesssim h^{2p} \|u\|_{H^{p+1}(\tilde{\kappa})} \|\psi\|_{H^{p+1}(\kappa)} \\
|\mathcal{E}| &\lesssim h^{2p} \|u\|_{H^{p+1}(\Omega)} \|\psi\|_{H^{p+1}(\Omega)},
\end{aligned}$$

respectively, where  $\tilde{\kappa}$  is the set of elements sharing a common face with  $\kappa$ .

*Proof.* We prove the local convergence bound by invoking the local residual-error mapping, Lemma E.10, for  $w_{h,p_1} = u_{h,p}$  and  $v_{h,p_2} = \psi_{h,p}$  and by applying the local bilinear form error bound, Lemma E.13, for  $p_1 = p_2 = p_3 = p$ , i.e.,

$$|\eta_{\kappa}| \equiv |\mathcal{R}_{h,p}(u_{h,p}, (\psi - \psi_{h,p})|_{\kappa})| = |a_{h,p}(u - u_{h,p}, (\psi - \psi_{h,p})|_{\kappa})| \lesssim h^{2p} \|u\|_{H^{p+1}(\tilde{\kappa})} \|\psi\|_{H^{p+1}(\kappa)}.$$

Similarly, we obtain the global convergence bound by applying the global residual-error mapping, Lemma E.11, for  $w_{h,p_1} = u_{h,p}$  and  $v_{h,p_2} = \psi_{h,p}$  and the global bilinear form error bound, Lemma E.14, for  $p_1 = p_2 = p_3 = p$ , i.e.,

$$|\mathcal{E}| \equiv |\mathcal{R}_{h,p}(u_{h,p}, \psi - \psi_{h,p})| = |a_{h,p}(u - u_{h,p}, \psi - \psi_{h,p})| \lesssim h^{2p} \|u\|_{H^{p+1}(\Omega)} \|\psi\|_{H^{p+1}(\Omega)}.$$



Thus, both the global and local errors superconverge at the rate of  $h^{2p}$ .  $\square$

### E.3.4 *A Priori* Error Analysis of Output Error Estimate 3

In this section, we analyze the convergence behavior of the output error estimate 3.

**Theorem E.16** (Convergence of Local Error Estimate 3). *The error in the local error estimate 3 is bounded by*

$$|\eta_\kappa - \eta_\kappa^{(3)}| \lesssim h^{p+\hat{p}} \|u\|_{H^{p+1}(\tilde{\kappa})} \|\psi\|_{H^{\hat{p}+1}(\kappa)}.$$

*Proof.* By linearity of the semilinear form with respect to the second argument, we have

$$\eta_\kappa - \eta_\kappa^{(3)} = \mathcal{R}_{h,p}(u_{h,p}, \psi|_\kappa) - \mathcal{R}_{h,p}(u_{h,p}, \psi_{h,\hat{p}}|_\kappa) = \mathcal{R}_{h,p}(u_{h,p}, (\psi - \psi_{h,\hat{p}})|_\kappa)$$

From here on, the proof is similar to that of the convergence of the true error. By invoking the local residual-error mapping, Lemma E.10, for  $w_{h,p_1} = u_{h,p}$  and  $v_{h,p_2} = \psi_{h,\hat{p}}$  and by applying the local bilinear form error bound, Lemma E.13, for  $p_1 = p_3 = p$  and  $p_2 = \hat{p}$ , we obtain

$$|\eta_\kappa - \eta_\kappa^{(3)}| = |a_{h,p}(u - u_{h,p}, (\psi - \psi_{h,\hat{p}})|_\kappa)| \lesssim h^{p+\hat{p}} \|u\|_{H^{p+1}(\tilde{\kappa})} \|\psi\|_{H^{\hat{p}+1}(\kappa)}.$$

$\square$

**Corollary E.17.** *Assuming the true local error converges as  $\eta_\kappa \approx h^{2p} \|u\|_{H^{p+1}(\tilde{\kappa})} \|\psi\|_{H^{p+1}(\kappa)}$ , the effectivity of the local error estimate 3 converges to unity as*

$$\theta_{\text{local},\kappa}^{(3)} = \left| 1 - \frac{\eta_\kappa^{(3)}}{\eta_\kappa} \right| = \frac{|\eta_\kappa^{(3)} - \eta_\kappa|}{|\eta_\kappa|} \lesssim \frac{Ch^{p+\hat{p}} \|u\|_{H^{p+1}(\tilde{\kappa})} \|\psi\|_{H^{\hat{p}+1}(\kappa)}}{h^{2p} \|u\|_{H^{p+1}(\tilde{\kappa})} \|\psi\|_{H^{p+1}(\kappa)}} = h^{\hat{p}-p} = h^{p_{\text{inc}}}$$

where  $p_{\text{inc}}$  is the increase in the polynomial degree for the truth surrogate adjoint solve.

**Theorem E.18** (Convergence of Global Error Estimate 3). *The error in the global error estimate 3 is bounded by*

$$|\mathcal{E}^{(3)} - \mathcal{E}| \lesssim h^{p+\hat{p}} \|u\|_{H^{p+1}(\Omega)} \|\psi\|_{H^{\hat{p}+1}(\Omega)}.$$



*Proof.* The convergence of the global error estimate 3 follows from that of the local counterpart, i.e.,

$$\begin{aligned} |\mathcal{E} - \mathcal{E}^{(3)}| &= \left| \sum_{\kappa \in \mathcal{T}_h} (\eta_\kappa - \eta_\kappa^{(3)}) \right| \lesssim \sum_{\kappa \in \mathcal{T}_h} h^{p+\hat{p}} \|u\|_{H^{p+1}(\tilde{\kappa})} \|\psi\|_{H^{\hat{p}+1}(\kappa)} \\ &\lesssim h^{p+\hat{p}} \|u\|_{H^{p+1}(\Omega)} \|\psi\|_{H^{\hat{p}+1}(\Omega)}. \end{aligned}$$

□

**Corollary E.19.** *If  $\mathcal{E} \approx h^{2p} \|u\|_{H^{p+1}(\Omega)} \|\psi\|_{H^{\hat{p}+1}(\Omega)}$ , then the effectivity of the global error estimate 3 converges to unity as*

$$\theta_{\text{global}}^{(3)} = \left| 1 - \frac{\mathcal{E}^{(3)}}{\mathcal{E}} \right| = \frac{|\mathcal{E}^{(3)} - \mathcal{E}|}{|\mathcal{E}|} \lesssim \frac{h^{p+\hat{p}} \|u\|_{H^{p+1}(\Omega)} \|\psi\|_{H^{\hat{p}+1}(\Omega)}}{h^{2p} \|u\|_{H^{p+1}(\Omega)} \|\psi\|_{H^{\hat{p}+1}(\Omega)}} \lesssim h^{\hat{p}-p} = h^{p_{\text{inc}}}.$$

### E.3.5 A Priori Error Analysis of Output Error Estimate 1

In this section, we analyze the convergence behavior of the output error estimate 1.

**Theorem E.20** (Convergence of Local Error Estimate 1). *The error in the local error estimate 1 is bounded by*

$$|\eta_\kappa - \eta_\kappa^{(1)}| \lesssim h^{2p} \|\psi\|_{H^{p+1}(\kappa)} \|u\|_{H^{p+1}(\tilde{\kappa})}$$

*Proof.* Expanding the difference in the local error using the error representation formula,

$$\begin{aligned} \eta_\kappa - \eta_\kappa^{(1)} &= \mathcal{R}_{h,p}(u_{h,p}, (\psi - \psi_{h,p})|_\kappa) - \mathcal{R}_{h,\hat{p}}(u_{h,p}, (\psi_{h,\hat{p}} - \psi_{h,p})|_\kappa) \\ &= \underbrace{\mathcal{R}_{h,p}(u_{h,p}, (\psi - \psi_{h,p})|_\kappa) - \mathcal{R}_{h,\hat{p}}(u_{h,p}, (\psi - \psi_{h,p})|_\kappa)}_{\text{(I)}} + \underbrace{\mathcal{R}_{h,\hat{p}}(u_{h,p}, (\psi - \psi_{h,\hat{p}})|_\kappa)}_{\text{(II)}} \end{aligned}$$

Term (II) can be bounded following a similar argument as that used to bound  $\eta_\kappa - \eta_\kappa^{(3)}$ . By invoking the local residual-error mapping, Lemma E.10, for  $w_{h,p_1} = u_{h,p}$  and  $v_{h,p_2} = \psi_{h,\hat{p}}$  and by applying the local bilinear form error bound, Lemma E.13, for  $p_1 = p$  and



$p_2 = p_3 = \hat{p}$ , we obtain

$$\begin{aligned} |(\text{II})| &= |\mathcal{R}_{h,\hat{p}}(u_{h,p}, (\psi - \psi_{h,\hat{p}})|_\kappa)| = |a_{h,\hat{p}}(u - u_{h,p}, (\psi - \psi_{h,\hat{p}})|_\kappa)| \\ &\lesssim h^{p+\hat{p}} \|u\|_{H^{p+1}(\tilde{\kappa})} \|\psi\|_{H^{\hat{p}+1}(\kappa)}. \end{aligned}$$

The only difference in the terms constituting (I) stems from the difference in the lifting spaces. Thus, term (I) can be expressed as

$$\begin{aligned} |(\text{I})| &= \left| - \sum_{f \in \partial\kappa} \left( \int_f \beta[(\psi - \psi_{h,p})|_\kappa] \cdot \left\{ r_{h,\hat{p}}^f(\llbracket u - u_{h,p} \rrbracket) \right\} ds \right. \right. \\ &\quad \left. \left. - \int_f \beta[(\psi - \psi_{h,p})|_\kappa] \cdot \left\{ r_{h,p}^f(\llbracket u - u_{h,p} \rrbracket) \right\} ds \right) \right| \\ &= \left| - \sum_{f \in \partial\kappa} \int_f \beta[(\psi - \psi_{h,p})|_\kappa] \cdot \left( \left\{ r_{h,\hat{p}}^f(\llbracket u - u_{h,p} \rrbracket) - r_{h,p}^f(\llbracket u - u_{h,p} \rrbracket) \right\} \right) ds \right| \\ &\leq \sum_{f \in \partial\kappa} \beta \|\psi - \psi_{h,p}\|_{L^2(f)} \left\| \left\{ r_{h,\hat{p}}^f(\llbracket u - u_{h,p} \rrbracket) - r_{h,p}^f(\llbracket u - u_{h,p} \rrbracket) \right\} \right\|_{L^2(f)} \\ &\lesssim \sum_{f \in \partial\kappa} \|\psi - \psi_{h,p}\|_{L^2(f)} h^{-1/2} \left\| \left\{ r_{h,\hat{p}}^f(\llbracket u - u_{h,p} \rrbracket) - r_{h,p}^f(\llbracket u - u_{h,p} \rrbracket) \right\} \right\|_{L^2(\kappa_f)} \\ &\lesssim \sum_{f \in \partial\kappa} \|\psi - \psi_{h,p}\|_{L^2(f)} h^{-1} \|\llbracket u - u_{h,p} \rrbracket\|_{L^2(f)} \\ &\lesssim h^{2p} \|\psi\|_{H^{p+1}(\kappa)} \|u\|_{H^{p+1}(\tilde{\kappa})} \end{aligned}$$

Combining the bounds for (I) and (II), we obtain

$$\begin{aligned} |\eta_\kappa^{(1)} - \eta_\kappa| &\leq |(\text{I})| + |(\text{II})| \lesssim h^{2p} \|\psi\|_{H^{p+1}(\kappa)} \|u\|_{H^{p+1}(\tilde{\kappa})} + h^{p+\hat{p}} \|u\|_{H^{p+1}(\tilde{\kappa})} \|\psi\|_{H^{\hat{p}+1}(\kappa)} \\ &\lesssim h^{2p} \|\psi\|_{H^{p+1}(\kappa)} \|u\|_{H^{p+1}(\tilde{\kappa})} \end{aligned}$$

□

**Corollary E.21.** *If  $\eta_\kappa \approx h^{2p} \|u\|_{H^{p+1}(\tilde{\kappa})} \|\psi\|_{H^{p+1}(\kappa)}$ , then the local effectivity does not converge to unity as the mesh is refined, i.e.,*

$$\theta_{\text{local},\kappa}^{(1)} = \left| 1 - \frac{\eta_\kappa^{(1)}}{\eta_\kappa} \right| = \frac{|\eta_\kappa - \eta_\kappa^{(1)}|}{|\eta_\kappa|} \lesssim \frac{h^{2p} \|\psi\|_{H^{p+1}(\kappa)} \|u\|_{H^{p+1}(\tilde{\kappa})}}{h^{2p} \|\psi\|_{H^{p+1}(\kappa)} \|u\|_{H^{p+1}(\tilde{\kappa})}} \lesssim 1.$$

**Theorem E.22** (Convergence of Global Error Estimate 1). *The error in the global error*



estimate 1 is bounded by

$$|\mathcal{E} - \mathcal{E}^{(1)}| \lesssim h^{2p} \|\psi\|_{H^{p+1}(\Omega)} \|u\|_{H^{p+1}(\Omega)}.$$

*Proof.* The convergence of the global error estimate 1 follows from that of the local counterpart, i.e.,

$$\begin{aligned} |\mathcal{E} - \mathcal{E}^{(1)}| &= \left| \sum_{\kappa \in \mathcal{T}_h} (\eta_\kappa - \eta_\kappa^{(1)}) \right| \lesssim \sum_{\kappa \in \mathcal{T}_h} h^{2p} \|u\|_{H^{p+1}(\tilde{\kappa})} \|\psi\|_{H^{p+1}(\kappa)} \\ &\lesssim h^{2p} \|u\|_{H^{p+1}(\Omega)} \|\psi\|_{H^{p+1}(\Omega)}. \end{aligned}$$

□

**Corollary E.23.** *If  $\mathcal{E} \approx h^{2p} \|\psi\|_{H^{p+1}(\Omega)} \|u\|_{H^{p+1}(\Omega)}$ , then the global effectivity does not converge to unity as the mesh is refined, i.e.,*

$$\theta_{\text{global}}^{(1)} = \left| 1 - \frac{\mathcal{E}^{(1)}}{\mathcal{E}} \right| = \frac{|\mathcal{E} - \mathcal{E}^{(1)}|}{|\mathcal{E}|} \lesssim \frac{h^{2p} \|\psi\|_{H^{p+1}(\Omega_h)} \|u\|_{H^{p+1}(\Omega_h)}}{h^{2p} \|\psi\|_{H^{p+1}(\Omega_h)} \|u\|_{H^{p+1}(\Omega_h)}} \lesssim 1.$$

### E.3.6 A Priori Error Analysis of Output Error Estimate 2

In this section, we analyze the convergence behavior of the output error estimate 2.

**Theorem E.24** (Convergence of Local Error Estimate 2). *The error in the local error estimate 2 is bounded by*

$$|\eta_\kappa - \eta_\kappa^{(2)}| \lesssim h^p \|u\|_{H^{p+1}(\tilde{\kappa})} \|\psi\|_{H^1(\kappa)}.$$

*Proof.* We will first bound the local error estimate,  $\eta_\kappa^{(2)}$ . By the definition of the primal residual and the linearity of the bilinear form,

$$\eta_\kappa^{(2)} = \mathcal{R}_{h,\hat{p}}(u_{h,p}, \psi_{h,\hat{p}}|_\kappa) = \ell(\psi_{h,\hat{p}}|_\kappa) - a_{h,\hat{p}}(u_{h,p}, \psi_{h,\hat{p}}|_\kappa) = a_{h,\hat{p}}(u - u_{h,p}, \psi_{h,\hat{p}}|_\kappa).$$

As  $a_{h,\hat{p}}(u - u_{h,p}, v_{h,p}) \neq 0$  in general for  $\hat{p} > p$ , we cannot subtract  $\psi_{h,\hat{p}}|_\kappa$  from the second



argument. The substitution of the BR2 bilinear form to the expression for  $\eta_\kappa^{(2)}$  yields

$$\begin{aligned} \eta_\kappa^{(2)} = & \underbrace{\int_\kappa \nabla(u - u_{h,p}) \cdot \nabla(\psi_{h,\hat{p}}) dx}_{(I)} - \underbrace{\int_{\partial\kappa} \{\nabla\psi_{h,\hat{p}}|_\kappa\} \cdot \llbracket u - u_{h,p} \rrbracket ds}_{(II)} \\ & - \underbrace{\int_{\partial\kappa} \llbracket \psi_{h,\hat{p}}|_\kappa \rrbracket \cdot \{\nabla(u - u_{h,p})\} ds}_{(III)} - \underbrace{\sum_{f \in \partial\kappa} \int_f \beta \llbracket \psi_{h,\hat{p}}|_\kappa \rrbracket \cdot \left\{ r_{h,p}^f(\llbracket u - u_{h,p} \rrbracket) \right\} ds}_{(IV)} \end{aligned}$$

The interior term is bounded by

$$|(I)| \leq \|u - u_{h,p}\|_{H^1(\kappa)} \|\psi_{h,\hat{p}}\|_{H^1(\kappa)} \lesssim h^p \|u\|_{H^{p+1}(\kappa)} \|\psi_{h,\hat{p}}\|_{H^1(\kappa)}.$$

The first face term is bounded by

$$\begin{aligned} |(II)| & \leq \|\{\nabla\psi_{h,\hat{p}}|_\kappa\}\|_{L^2(\partial\kappa)} \|\llbracket u - u_{h,p} \rrbracket\|_{L^2(\partial\kappa)} \lesssim h^{-1/2} \|\nabla\psi_{h,\hat{p}}\|_{L^2(\kappa)} h^{p+1/2} \|u\|_{H^{p+1}(\tilde{\kappa})} \\ & \lesssim h^p \|\psi_{h,\hat{p}}\|_{H^1(\kappa)} \|u\|_{H^{p+1}(\tilde{\kappa})}. \end{aligned}$$

The second face term is bounded by

$$\begin{aligned} |(III)| & \leq \|\llbracket \psi_{h,\hat{p}}|_\kappa \rrbracket\|_{L^2(\partial\kappa)} \|\{\nabla(u - u_{h,p})\}\|_{L^2(\partial\kappa)} \lesssim h^{-1/2} \|\psi_{h,\hat{p}}\|_{L^2(\kappa)} h^{p+1/2} \|u\|_{H^{p+1}(\kappa)} \\ & \lesssim h^p \|\psi_{h,\hat{p}}\|_{L^2(\kappa)} \|u\|_{H^{p+1}(\kappa)}. \end{aligned}$$

The term involving the lifting operator is bounded by

$$\begin{aligned} |(IV)| & = \sum_{f \in \partial\kappa} \int_f \beta \llbracket \psi_{h,\hat{p}}|_\kappa \rrbracket \cdot \left\{ r_{h,p}^f(\llbracket u - u_{h,p} \rrbracket) \right\} ds \\ & \leq \sum_{f \in \partial\kappa} \beta \|\psi_{h,\hat{p}}\|_{L^2(f)} \left\| \left\{ r_{h,p}^f(\llbracket u - u_{h,p} \rrbracket) \right\} \right\|_{L^2(f)} \\ & \lesssim \sum_{f \in \partial\kappa} \|\psi_{h,\hat{p}}\|_{L^2(f)} h^{-1/2} \left\| \left\{ r_{h,p}^f(\llbracket u - u_{h,p} \rrbracket) \right\} \right\|_{L^2(\kappa_f)} \\ & \lesssim \sum_{f \in \partial\kappa} \|\psi_{h,\hat{p}}\|_{L^2(f)} h^{-1} \|\llbracket u - u_{h,p} \rrbracket\|_{L^2(f)} \\ & \lesssim h^p \|\psi_{h,\hat{p}}\|_{L^2(\kappa)} \|u\|_{H^{p+1}(\tilde{\kappa})}. \end{aligned}$$



Combining the bounds for (I), (II), (III), and (IV), we obtain

$$|\eta_\kappa^{(2)}| \leq |(\text{I})| + |(\text{II})| + |(\text{III})| + |(\text{IV})| \lesssim h^p \|u\|_{H^{p+1}(\tilde{\kappa})} \|\psi_{h,\hat{p}}\|_{H^1(\kappa)}.$$

We further note that

$$\begin{aligned} \|\psi_{h,\hat{p}}\|_{H^1(\kappa)} &= \|\psi_{h,\hat{p}} - \psi + \psi\|_{H^1(\kappa)} \leq \|\psi_{h,\hat{p}} - \psi\|_{H^1(\kappa)} + \|\psi\|_{H^1(\kappa)} \\ &\lesssim h^{\hat{p}} \|\psi\|_{H^{\hat{p}+1}(\kappa)} + \|\psi\|_{H^1(\kappa)} \lesssim \|\psi\|_{H^1(\kappa)} \end{aligned}$$

for  $h$  sufficiently small. Thus, we obtain the bound for  $\eta_\kappa^{(2)}$  in terms of  $u$  and  $\psi$ , i.e.,

$$|\eta_\kappa^{(2)}| \lesssim h^p \|u\|_{H^{p+1}(\tilde{\kappa})} \|\psi\|_{H^1(\kappa)}$$

An immediate consequence of this result is that

$$\begin{aligned} |\eta_\kappa^{(2)} - \eta_\kappa| &\lesssim |h^p \|u\|_{H^{p+1}(\tilde{\kappa})} \|\psi\|_{H^1(\kappa)} - h^{2p} \|u\|_{H^{p+1}(\tilde{\kappa})} \|\psi\|_{H^{p+1}(\kappa)}| \\ &\lesssim h^p \|u\|_{H^{p+1}(\tilde{\kappa})} \|\psi\|_{H^1(\kappa)}. \end{aligned}$$

□

**Corollary E.25.** *If  $\eta_\kappa \approx h^{2p} \|u\|_{H^{p+1}(\tilde{\kappa})} \|\psi\|_{H^{p+1}(\kappa)}$ , then the effectivity of the local error estimate 3 diverges in the sense that*

$$\theta_{\text{local},\kappa}^{(2)} = \left| 1 - \frac{\eta_\kappa^{(2)}}{\eta_\kappa} \right| = \frac{|\eta_\kappa - \eta_\kappa^{(2)}|}{|\eta_\kappa|} \lesssim \frac{h^p \|u\|_{H^{p+1}(\tilde{\kappa})} \|\psi\|_{H^1(\kappa)}}{h^{2p} \|u\|_{H^{p+1}(\tilde{\kappa})} \|\psi\|_{H^{p+1}(\kappa)}} \lesssim h^{-p},$$

i.e., the local error estimator degrades (relative to the true local error) as the mesh is refined.

**Theorem E.26** (Convergence of Global Error Estimate 2). *The error in the global error estimate 2 is bounded by*

$$|\mathcal{E} - \mathcal{E}^{(2)}| \lesssim h^{2\hat{p}} \|u\|_{H^{p+1}(\Omega)} \|\psi\|_{H^{\hat{p}+1}(\Omega)}$$

*Proof.* Unlike the analysis for the global error estimate 1 and 3, simply summing the local error estimator bounds results in a loose bound. Thus, we will pursue a different approach



to obtain a tighter bound. We first note that

$$\mathcal{R}_{h,p}(u_{h,p}, v) = \mathcal{R}_{h,\hat{p}}(u_{h,p}, v), \quad \forall v \in H^1(\Omega), \forall p, \hat{p} \in \mathbb{N},$$

as the lifting operator is always multiplied by the jump in the second argument and  $\llbracket v \rrbracket = 0$ ,  $\forall v \in H^1(\Omega)$ . In particular, we can rewrite the true error representation as

$$\mathcal{E} = \mathcal{R}_{h,p}(u_{h,p}, \psi) = \mathcal{R}_{h,\hat{p}}(u_{h,p}, \psi)$$

The error in the global error estimate becomes

$$\begin{aligned} \mathcal{E} - \mathcal{E}^{(2)} &= \mathcal{R}_{h,\hat{p}}(u_{h,p}, \psi) - \mathcal{R}_{h,\hat{p}}(u_{h,p}, \psi_{h,\hat{p}}) \\ &= \mathcal{R}_{h,\hat{p}}(u_{h,p}, \psi - \psi_{h,\hat{p}}) && \text{(linearity)} \\ &= R_{h,\hat{p}}^\psi(u - u_{h,p}, \psi_{h,\hat{p}}) && \text{(Lemma E.11 for } w_{h,p} = u_{h,p}, v_{h,p} = \psi_{h,\hat{p}}) \\ &= \inf_{v_{h,\hat{p}} \in V_{h,\hat{p}}} R_{h,\hat{p}}^\psi(u - u_{h,p} - v_{h,\hat{p}}, \psi_{h,\hat{p}}) && \text{(dual Galerkin orthogonality)} \\ &= \inf_{v_{h,\hat{p}} \in V_{h,\hat{p}}} a_{h,\hat{p}}(u - v_{h,\hat{p}}, \psi - \psi_{h,\hat{p}}) && \text{(Lemma E.11 for } w_{h,p} = v_{h,\hat{p}}, v_{h,p} = \psi_{h,\hat{p}}) \end{aligned}$$

By applying the global bilinear form error bound, Lemma E.14, for  $p_1 = p_2 = p_3 = \hat{p}$ , we obtain

$$|\mathcal{E}^{(2)} - \mathcal{E}| = |a_{h,\hat{p}}(u - v_{h,\hat{p}}, \psi - \psi_{h,\hat{p}})| \lesssim h^{2\hat{p}} \|u\|_{H^{\hat{p}+1}(\Omega_h)} \|\psi\|_{H^{\hat{p}+1}(\Omega_h)}$$

□

**Corollary E.27.** *If  $\mathcal{E} \approx h^{2p} \|\psi\|_{H^{p+1}(\Omega)} \|u\|_{H^{p+1}(\Omega)}$ , then the effectivity of the global error estimate 2 converges to unity as*

$$\theta_{\text{global}}^{(2)} = \left| 1 - \frac{\mathcal{E}^{(2)}}{\mathcal{E}} \right| = \frac{|\mathcal{E} - \mathcal{E}^{(2)}|}{|\mathcal{E}|} \lesssim \frac{h^{2\hat{p}} \|u\|_{H^{p+1}(\Omega_h)} \|\psi\|_{H^{\hat{p}+1}(\Omega_h)}}{h^{2p} \|\psi\|_{H^{\hat{p}+1}(\Omega)} \|u\|_{H^{p+1}(\Omega)}} \lesssim h^{2(\hat{p}-p)} = h^{2p_{\text{inc}}}$$

### E.3.7 Summary of *A Priori* Error Analysis

Table E.1 summarizes the result of the *a priori* error analysis. The table shows that neither the local nor global effectivity of the estimate 1 approaches unity as  $h \rightarrow 0$ . The estimate 2 results in a superconvergent global estimate; however, the local error effectivity



(a) local estimates			
	$\eta_\kappa^{(i)}$	$ \eta_\kappa^{(i)} - \eta_\kappa $	$\theta_{\text{local},\kappa} =  1 - \eta_\kappa^{(i)}/\eta_\kappa $
1	$\mathcal{R}_{h,\hat{p}}(u_{h,p}, (\psi_{h,\hat{p}} - \psi_{h,p}) _\kappa)$	$h^{2p} \ u\ _{H^{p+1}(\tilde{\kappa})} \ \psi\ _{H^{p+1}(\kappa)}$	$h^0$
2	$\mathcal{R}_{h,\hat{p}}(u_{h,p}, \psi_{h,\hat{p}} _\kappa)$	$h^p \ u\ _{H^{p+1}(\tilde{\kappa})} \ \psi\ _{H^1(\kappa)}$	$h^{-p}$
3	$\mathcal{R}_{h,p}(u_{h,p}, \psi_{h,\hat{p}} _\kappa)$	$h^{p+\hat{p}} \ u\ _{H^{p+1}(\tilde{\kappa})} \ \psi\ _{H^{\hat{p}+1}(\kappa)}$	$h^{p_{\text{inc}}}$
true	$\mathcal{R}_{h,p}(u_{h,p}, \psi _\kappa)$	-	-
(b) global estimates			
	$\mathcal{E}^{(i)}$	$ \mathcal{E}^{(i)} - \mathcal{E} $	$\theta_{\text{global}} =  1 - \mathcal{E}^{(i)}/\mathcal{E} $
1	$\mathcal{R}_{h,\hat{p}}(u_{h,p}, \psi_{h,\hat{p}} - \psi_{h,p})$	$h^{2p} \ u\ _{H^{p+1}(\Omega)} \ \psi\ _{H^{p+1}(\Omega)}$	$h^0$
2	$\mathcal{R}_{h,\hat{p}}(u_{h,p}, \psi_{h,\hat{p}})$	$h^{2\hat{p}} \ u\ _{H^{\hat{p}+1}(\Omega)} \ \psi\ _{H^{\hat{p}+1}(\Omega)}$	$h^{2p_{\text{inc}}}$
3	$\mathcal{R}_{h,p}(u_{h,p}, \psi_{h,\hat{p}})$	$h^{p+\hat{p}} \ u\ _{H^{p+1}(\Omega)} \ \psi\ _{H^{\hat{p}+1}(\Omega)}$	$h^{p_{\text{inc}}}$
true	$\mathcal{J}_{h,p}(u_{h,p}) - \mathcal{J}(u) = \mathcal{R}_{h,p}(u_{h,p}, \psi)$	-	-

Table E.1: Summary of the local and global error estimate convergence.

diverges with mesh refinement, and thus the estimator is not suited for driving adaptation. The estimate 3 is the only estimate whose effectivity converges to unity both locally and globally as  $h \rightarrow 0$ .

## E.4 Numerical Results

This section provides numerical verification of the *a priori* error analysis results presented in Section E.3. In particular, we apply the three error estimates to a one dimensional Poisson problem given by

$$\begin{aligned}
-\frac{d^2 u}{dx^2} &= \exp(x)(1+x), \quad \text{on } (0,1), \\
u(0) &= u(1) = 0,
\end{aligned}$$

and the functional output of interest,

$$\mathcal{J}(u) = \int_0^1 \sin(\pi x) u(x) dx.$$

Note that the analytical solution to the primal and dual problems are given by

$$u = (\exp(x) - 1)(1 - x) \quad \text{and} \quad \psi = \sin(\pi x),$$

both of which are in  $C^\infty$  and have finite and non-vanishing measures in  $H^m(\Omega)$ ,  $\forall m \in \mathbb{N}$ .



We will use two different metrics to assess the performance of the error estimates. The first measure is the relative error in the global estimate as defined earlier, i.e.

$$\theta_{\text{global}}^{(i)} \equiv \frac{|\mathcal{E} - \mathcal{E}^{(i)}|}{|\mathcal{E}|} = \left| 1 - \frac{\mathcal{E}^{(i)}}{\mathcal{E}} \right|,$$

where  $\mathcal{E}$  is the true error and  $\mathcal{E}^{(i)}$  is the error estimate provided by the estimator  $i$ . Recall that the relative error is equivalent to the deviation of the error effectivity from unity. The second measure is the agglomerated local effectivity, which is a single measure intended to capture the effectivity of the local, element-wise error estimates. The agglomerated local effectivity is defined by

$$\theta_{\text{local}}^{(i)} \equiv \left| 1 - \frac{\mathcal{E}_{\text{agg}}^{(i)}}{\mathcal{E}_{\text{agg}}} \right| = \frac{|\mathcal{E}_{\text{agg}} - \mathcal{E}_{\text{agg}}^{(i)}|}{|\mathcal{E}_{\text{agg}}|}$$

where

$$\mathcal{E}_{\text{agg}} \equiv \sum_{\kappa \in \mathcal{T}_h} |\eta_{\kappa}| \quad \text{and} \quad \mathcal{E}_{\text{agg}}^{(i)} \equiv \sum_{\kappa \in \mathcal{T}_h} |\eta_{\kappa}^{(i)}|.$$

Note that this is different from the relative local error  $\theta_{\text{local},\kappa}^{(i)}$  associated with each element  $\kappa$ , but it is an agglomerated measure of the quality of the local estimates.

#### E.4.1 True Output Error

We first analyze the behavior of the true error, measured in the standard sense and in the agglomerated local sense. Figure E-1 shows the convergence results for  $p = 1, 2, 3, 4$ . Since both the primal and dual solutions are infinitely smooth, Theorem E.15 predicts the superconvergence of both the local and global errors at the rate of  $h^{2p}$ . The numerical result confirms the analysis. Since the solutions have well-behaved higher order derivatives, the convergence with grid refinement is very smooth. We note that  $p = 4$  solution achieves machine precision accuracy using just 16 elements; while this is an encouraging result, it makes the assessment of the error estimates more difficult, as the results are affected by the finite precision arithmetics. Thus,  $p = 3$  and  $p = 4$  results are sometimes truncated or omitted, if the results have been deemed polluted by rounding errors.



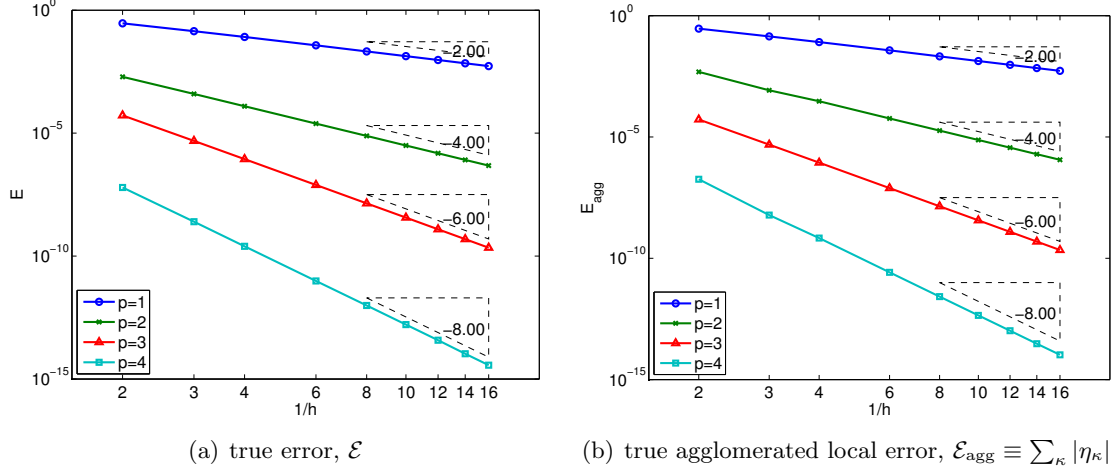


Figure E-1: The convergence of the true output error.

#### E.4.2 Output Error Estimate 1

By the *a priori* error analysis, Theorem E.20 and E.22, we expect

$$\theta_{\text{local}}^{(1)} \equiv \left| 1 - \frac{\mathcal{E}_{\text{agg}}^{(1)}}{\mathcal{E}} \right| \lesssim h^0 \quad \text{and} \quad \theta_{\text{global}}^{(1)} \equiv \left| 1 - \frac{\mathcal{E}^{(1)}}{\mathcal{E}} \right| \lesssim h^0,$$

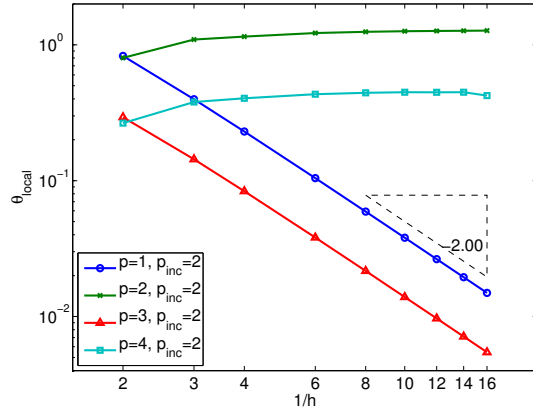
i.e., neither the local nor the global effectivity converge to unity with grid refinement. Figure E-2 shows the convergence of the local and global effectivity of error estimate 1. The result must be interpreted carefully, as the *a priori* error analysis results are upper bound and the cancellation can give a false sense of convergence. For example, Figure E-2(a) and E-2(c) show that the local and global effectivities converge to unity for odd  $p$  but not for even  $p$ . The cause of this odd-even behavior is unclear, but similar results have been observed in [67, 110]. In these cases, we should always compare the worst convergence rate with the *a priori* analysis, i.e. the even results for this case. The numerical experiment confirms that the effectivity of the error estimate 1 does not converge to unity in either the local or the global sense.

#### E.4.3 Output Error Estimate 2

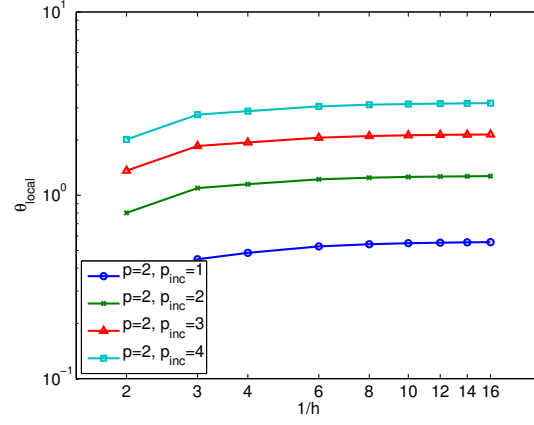
By the *a priori* error analysis, Theorem E.24 and E.26, we expect

$$\theta_{\text{local}}^{(2)} \equiv \left| 1 - \frac{\mathcal{E}_{\text{agg}}^{(2)}}{\mathcal{E}} \right| \lesssim h^{-p} \quad \text{and} \quad \theta_{\text{global}}^{(2)} \equiv \left| 1 - \frac{\mathcal{E}^{(2)}}{\mathcal{E}} \right| \lesssim h^{2p_{\text{inc}}},$$

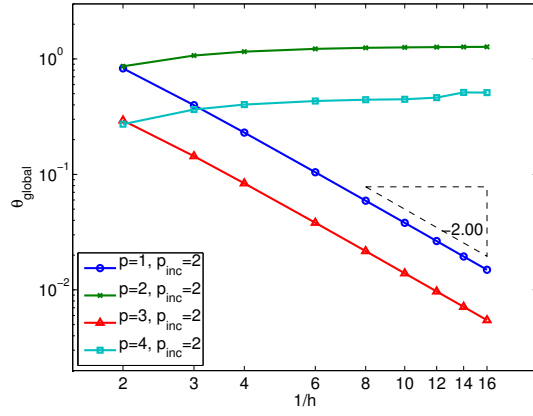




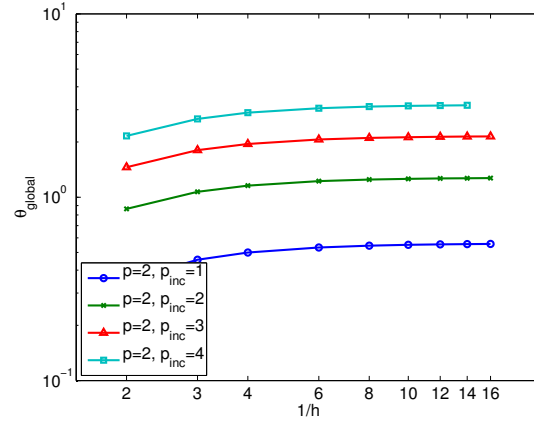
(a) local error effectivity, varying  $p$



(b) local error effectivity, varying  $p_{\text{inc}}$



(c) global error effectivity, varying  $p$



(d) global error effectivity, varying  $p_{\text{inc}}$

Figure E-2: The local and global effectivity of the error estimate 1.



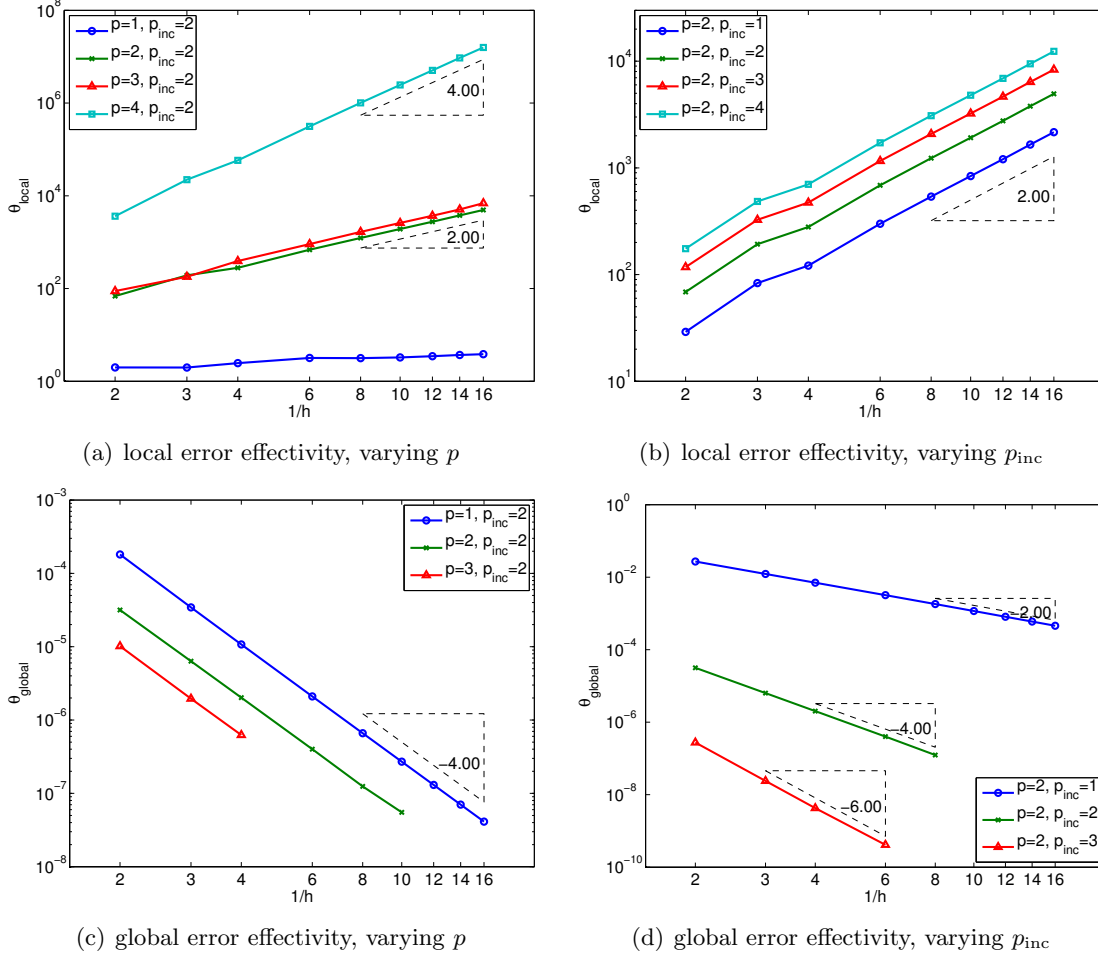


Figure E-3: The local and global effectivity of the error estimate 2.

i.e., the local error effectivity diverges at the rate of  $h^{-p}$ , but the global error effectivity superconverges at the rate of  $h^{2p_{inc}}$ . The divergence of the local effectivity is captured in Figure E-3(a) and E-3(b). In particular, the local effectivity diverges at the rate of  $h^{-2}$  and  $h^{-4}$  for  $p = 2$  and  $4$ , respectively. The local effectivity is not a function of  $p_{inc}$  as  $p_{inc} = 1, 2, 3, 4$  all diverges at the rate of  $h^{-2}$  for  $p = 2$ . On the other hand, Figure E-3(c) and E-3(d) show that the global effectivity exhibit superconvergence. In particular, the global effectivity convergence rate is a function of  $p_{inc}$  showing the convergence rates of  $h^2$ ,  $h^4$ , and  $h^6$  for  $p_{inc} = 1, 2$ , and  $3$ , respectively. The global effectivity convergence rate is not a function of  $p$ , as  $p_{inc} = 2$  results in the convergence rate of  $h^4$  for all  $p = 1, 2, 3$ . These results are consistent with the *a priori* analysis.



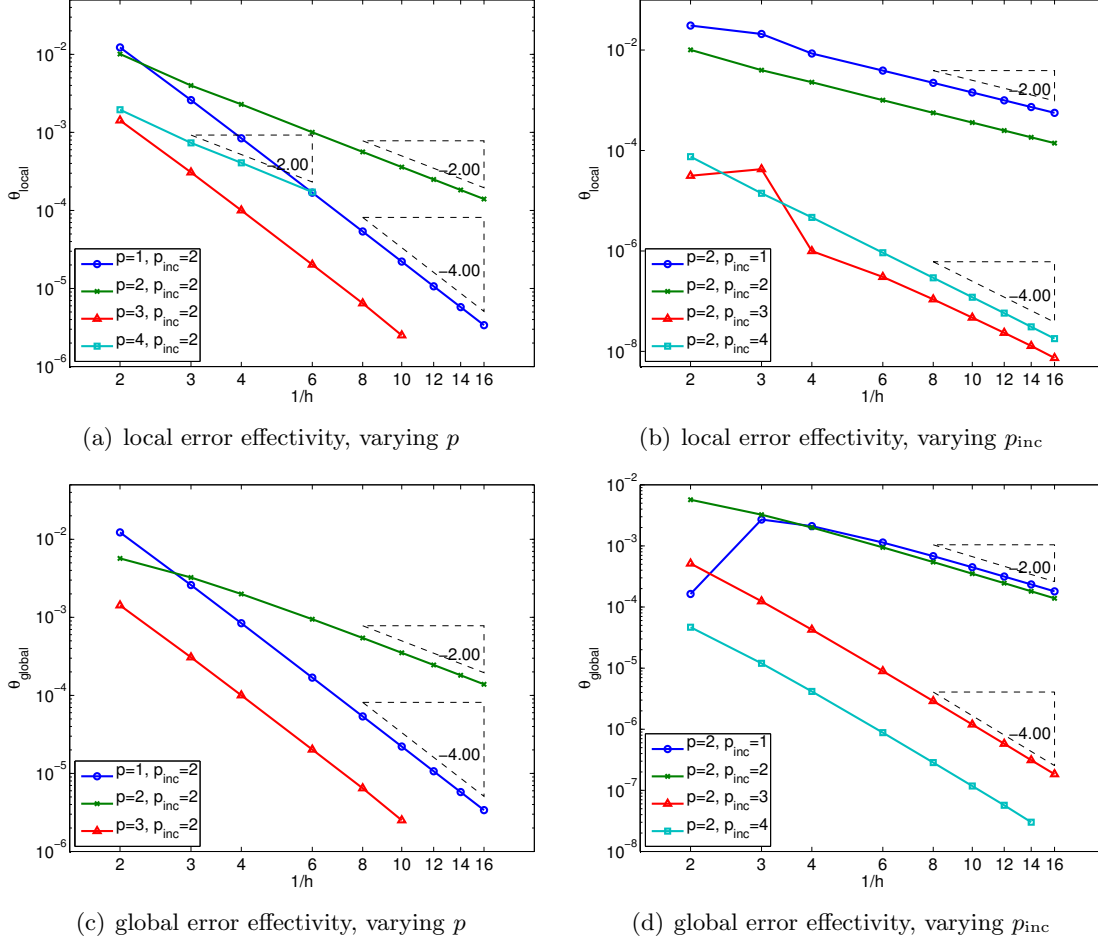


Figure E-4: The local and global effectivity of the error estimate 3.

#### E.4.4 Output Error Estimate 3

By the *a priori* error analysis, Theorem E.16 and E.18, we expect

$$\theta_{\text{local}} \equiv \left| 1 - \frac{\mathcal{E}_{\text{agg}}^{(3)}}{\mathcal{E}_{\text{agg}}} \right| \lesssim h^{p_{\text{inc}}} \quad \text{and} \quad \theta_{\text{global}} \equiv \left| 1 - \frac{\mathcal{E}^{(3)}}{\mathcal{E}} \right| \lesssim h^{p_{\text{inc}}},$$

i.e., both the local and global error effectivities converge at the rate of  $h^{p_{\text{inc}}}$ . Figure E-4(a) shows that  $p_{\text{inc}} = 2$  results in the local effectivity convergence of  $h^2$  for  $p = 2, 4$ . Figure E-4(b) shows that the convergence rate improves to  $h^4$  for  $p_{\text{inc}} = 4$ . The same behavior is shown for the global effectivity in Figure E-4(c) and E-4(d), converging at the rate of at least  $h^{p_{\text{inc}}}$ .



## E.5 Conclusion

This appendix analyzed the behavior of three variants of the DWR error estimates applied to a  $p$ -dependent discretization. We showed that the BR2 discretization of second-order PDEs results in a  $p$ -dependent discretization due to the presence of the  $p$ -dependent lifting operator. Then, we analyzed three commonly used variants of DWR error estimates. The *a priori* error analysis showed that the effectivity of error estimate 1—which naturally results from the discrete interpretation of the adjoint—converges in neither the local nor global sense. Error estimate 2 exhibited superconvergent global effectivity; however, its local effectivity diverges, making it unsuited for grid adaptation. The effectivity of error estimate 3 converges both in the local and global sense, making it an attractive choice for both error estimation and adaptation. A simple one-dimensional Poisson problem numerically verified the *a priori* error analysis.







## Appendix F

# Properties of the Adaptation Algorithm

### F.1 Relationship between Step Matrix and the Change in Approximability

In designing our surrogate error model and optimization algorithm, we advocated the use of the step matrix  $S$  (either elemental or vertex) rather than using the metric tensor  $\mathcal{M}$  directly. This is because the magnitude of the entries of a step matrix  $S$  is closely related to the change in the anisotropic approximability of the space associated with  $\mathcal{M}_0$  and  $\mathcal{M}(S) \equiv \mathcal{M}_0^{1/2} \exp(S) \mathcal{M}_0^{1/2}$ , as stated in Section 3.2.1. Here we prove the relationship Eq. (3.6) between the change in the anisotropic approximability and the entries of the step matrix  $S$ .

The change in the approximability in a given direction, or the ratio of the directional lengths between the configurations induced by  $\mathcal{M}_0$  and  $\mathcal{M}(S)$ , is

$$\frac{h(e; \mathcal{M}(S))}{h(e; \mathcal{M}_0)} = \left( \frac{e^T \mathcal{M}_0 e}{e^T \mathcal{M}_0^{1/2} \exp(S) \mathcal{M}_0^{1/2} e} \right)^{1/2}.$$

The lower bound of the ratio, i.e. the maximum increase in the approximability, is related



to the eigenvalues of  $S$  by

$$\begin{aligned} \min_{e \in \mathbb{R}^d \setminus 0} \frac{h(e; \mathcal{M}(S))}{h(e; \mathcal{M}_0)} &= \min_{e \in \mathbb{R}^d \setminus 0} \left( \frac{e^T \mathcal{M}_0 e}{e^T \mathcal{M}_0^{1/2} \exp(S) \mathcal{M}_0^{1/2} e} \right)^{1/2} = \min_{f \in \mathbb{R}^d \setminus 0} \left( \frac{f^T f}{f^T \exp(S) f} \right)^{1/2} \\ &= (\lambda_{\max}(\exp(S)))^{-1/2} = \exp\left(-\frac{1}{2} \lambda_{\max}(S)\right), \end{aligned}$$

where  $\lambda_{\max}(S)$  denotes the maximum eigenvalue of  $S$ . Similarly, the upper bound of the ratio can be expressed as

$$\max_{e \in \mathbb{R}^d \setminus 0} \frac{h(e; \mathcal{M}(S))}{h(e; \mathcal{M}_0)} = \exp\left(-\frac{1}{2} \lambda_{\min}(S)\right),$$

where  $\lambda_{\min}(S)$  denotes the minimum eigenvalue of  $S$ . Thus, we can control the maximum increase or decrease in the approximability by controlling the maximum and minimum eigenvalue of  $S$ , respectively. In particular, because

$$\lambda_{\min}^2(S) \leq \|S\|_F^2 \quad \text{and} \quad \lambda_{\max}^2(S) \leq \|S\|_F^2,$$

the magnitude of the entries in  $S$  is a good indicator of the maximal change in the approximability in moving from  $\mathcal{M}_0$  to  $\mathcal{M}(S)$ . Thus, expressing the manipulation in terms of the step tensor  $S \in \text{Sym}_d$  and mapping the tensor to  $\mathcal{M}(S) \in \text{Sym}_d^+$  via the exponential map not only eliminates the potential of generating a null-tensor but also provides a convenient means of controlling the change in the anisotropic approximability.

## F.2 Inclusion of the Isotropic Error Model

As mentioned in Section 3.2.3, our anisotropic error model  $\eta_\kappa(S_\kappa) = \eta_{\kappa_0} \exp(\text{tr}(R_\kappa S_\kappa))$  is a generalization of the familiar isotropic error relationship based on the power law,

$$\eta_\kappa^{\text{iso}}(h) = \eta_{\kappa_0} \left( \frac{h}{h_0} \right)^{r_\kappa^{\text{iso}}}, \quad (\text{F.1})$$

where  $r_\kappa^{\text{iso}}$  is the convergence rate. In particular, the behavior of the error model under isotropic scaling is consistent with that of the isotropic error model in the following sense. The isotropic metric  $\mathcal{M}$  for mesh size  $h$  is given by  $\mathcal{M} = h^{-2}I$ . The step tensor required



to change from an isotropic tensor  $\mathcal{M}_0 = h_0^{-2}I$  to  $\mathcal{M} = h^{-2}I$  is

$$S_\kappa = \log \left( \mathcal{M}_0^{-1/2} \mathcal{M} \mathcal{M}_0^{-1/2} \right) = \log (h_0^2 h^{-2} I) = -2 \log \left( \frac{h}{h_0} \right) I$$

We note that the trace-free part  $\tilde{S}_\kappa$  vanishes as expected, and the isotropic part is  $s_\kappa = -2 \log(h/h_0)$ . Substitution of the step tensor into the local error model yields

$$\eta_\kappa(S_\kappa) = \eta_\kappa \left( -2 \log \left( \frac{h}{h_0} \right) I \right) = \eta_{\kappa_0} \exp \left( -2dr_\kappa \log \left( \frac{h}{h_0} \right) \right) = \eta_{\kappa_0} \left( \frac{h}{h_0} \right)^{-2dr_\kappa}.$$

If we define  $r_\kappa^{\text{iso}} = -2dr_\kappa$ , then we recover the isotropic error relationship Eq. (F.1). Thus, our error model can be thought of as an extension of the scalar error model to anisotropic deformations.

### F.3 Invariance of the Sampling Quality

One of the important features of the proposed error model and sampling strategy is that the quality of the error reconstruction does not degrade on highly anisotropic elements. Recalling the error reconstruction operates on the step matrices  $\{S_{\kappa_i}\}_{i=1}^{n_{\text{config}}}$  in the tangent space, the property requires that the set of step matrices does not become degenerate on a highly anisotropic configuration. In fact, we will show that  $\{S_{\kappa_i}\}_{i=1}^{n_{\text{config}}}$  are invariant with respect to the current configuration  $\mathcal{M}_{\kappa_0}$  up to orthogonal transformation, which does not influence the quality of reconstruction. The invariance is a consequence of the local coordinate system induced by the affine-invariant metric. We note that, if the error reconstruction is performed directly using the coefficients of the metric tensor, the error reconstruction would become ill-posed as the step tensors  $\mathcal{M}_{\kappa_0} - \mathcal{M}_{\kappa_i}$  becomes degenerate on highly anisotropic elements.

Let us denote the metric tensor associated with the unit reference element by  $\widehat{\mathcal{M}}_0$ . By definition,  $\widehat{\mathcal{M}}_0 = I$ . Let us denote the mapping of the unit reference element to an element obtained by the  $i$ -th local mesh operation of the reference element by  $\widehat{J}_i$ . The tensor corresponding to the split reference element is

$$\widehat{\mathcal{M}}_i = \widehat{J}_i^{-T} \widehat{\mathcal{M}}_0 \widehat{J}_i^{-1}.$$



The step tensor from  $\widehat{\mathcal{M}}_0$  to  $\widehat{\mathcal{M}}_i$  is

$$\widehat{S}_i = \log(\widehat{\mathcal{M}}_0^{-1/2} \widehat{\mathcal{M}}_i \widehat{\mathcal{M}}_0^{-1/2}) = \log(\widehat{\mathcal{M}}_i)$$

Let us now consider the step tensor from an arbitrary configuration  $\mathcal{M}_{\kappa_0}$  to the configuration obtained by the  $i$ -th local mesh operation,  $\mathcal{M}_{\kappa_i}$ . Let us denote the mapping from the unit reference triangle,  $\widehat{\mathcal{M}}_0$ , to  $\mathcal{M}_{\kappa_0}$  by  $J$  and the singular value decomposition of  $J$  by  $J = U\Sigma V^T$ . Then,  $\mathcal{M}_{\kappa_0}$  can be expressed as

$$\mathcal{M}_{\kappa_0} = J^{-T} \widehat{\mathcal{M}}_0 J^{-1} = (U\Sigma^{-1}V^T)I(V\Sigma^{-1}U^T) = U\Sigma^{-2}U^T.$$

Similarly, using the mapping  $J$ , we can express the configuration obtained by  $i$ -th mesh operation as

$$\mathcal{M}_{\kappa_i} = J^{-T} \widehat{\mathcal{M}}_i J^{-1} = U\Sigma^{-1}V^T \widehat{\mathcal{M}}_i V\Sigma^{-1}U^T.$$

The step matrix from  $\mathcal{M}_{\kappa_0}$  to  $\mathcal{M}_{\kappa_i}$  is

$$\begin{aligned} S_{\kappa_i} &= \log(\mathcal{M}_{\kappa_0}^{-1/2} \mathcal{M}_{\kappa_i} \mathcal{M}_{\kappa_0}^{-1/2}) \\ &= \log((U\Sigma^{-2}U^T)^{-1/2} (U\Sigma^{-1}V^T \widehat{\mathcal{M}}_i V\Sigma^{-1}U^T) (U\Sigma^{-2}U^T)^{-1/2}) \\ &= UV^T \log(\widehat{\mathcal{M}}_i) VU^T = (VU^T)^T \widehat{S}_i (VU^T) \end{aligned}$$

The step matrix from  $\mathcal{M}_{\kappa_0}$  to  $\mathcal{M}_{\kappa_i}$  is related to the step matrix from  $\widehat{\mathcal{M}}_0$  to  $\widehat{\mathcal{M}}_i$  by the orthogonal transformation induced by  $VU^T$ . Thus, as long as the samples  $\{\widehat{\mathcal{M}}_i\}_{i=1}^{n_{\text{config}}}$  are chosen such that the linear error reconstruction problem is well-posed on the reference element, the linear fitting problem on  $\mathcal{M}_{\kappa_0}$  is well-posed. In other words, the quality of the error model reconstruction is preserved even on high aspect ratio elements encountered in anisotropic adaptation.

## F.4 Invariance under Coordinate Transformation

In this section, we show that the tensor field optimization algorithm presented is independent of the particular coordinate representation of the tensors. The property means that



the same physical problem represented in two different coordinate systems would produce the identical sequences of the tensor fields with respect to the physical problem.

Let us consider two coordinate systems,  $x$  and  $\bar{x}$ , that are related by the mapping

$$\bar{x} = g(x) = \alpha U x + \bar{x}_0,$$

where  $U$  is a  $d \times d$  orthogonal matrix,  $\alpha > 0$  is the coordinate scaling factor, and  $\bar{x}_0 \in \mathbb{R}^d$  is the coordinate shift. A mesh defined in terms of  $x$  can be represented in the coordinate system  $\bar{x}$  by mapping each nodal coordinate according to  $\bar{x} = g(x)$ . Then, the tensor field represented in the coordinate system  $\bar{x}$ ,  $\{\bar{\mathcal{M}}\}_{\bar{x} \in \Omega}$ , is related to that of the coordinate system  $x$ ,  $\{\mathcal{M}\}_{x \in \Omega}$ , by

$$\bar{\mathcal{M}}(\bar{x}) = \alpha^{-2} U \mathcal{M}(x) U^T.$$

Now let us work through the adaptation procedure and show that it is invariant under coordinate transformation.

The first step of adaptation is local sampling. The elemental step tensor in  $\bar{x}$ ,  $\bar{S}_{\kappa_i}$ , is related to that in  $x$ ,  $S_{\kappa_i}$ , by

$$\begin{aligned} \bar{S}_{\kappa_i} &= \log \left( \bar{\mathcal{M}}_{\kappa_0}^{-1/2} \bar{\mathcal{M}}_{\kappa_i} \bar{\mathcal{M}}_{\kappa_0}^{-1/2} \right) \\ &= \log \left( (\alpha^{-2} U \mathcal{M}_{\kappa_0} U^T)^{-1/2} (\alpha^{-2} U \mathcal{M}_{\kappa_i} U^T) (\alpha^{-2} U \mathcal{M}_{\kappa_0} U^T)^{-1/2} \right) \\ &= U \log \left( \mathcal{M}_{\kappa_0}^{-1/2} \mathcal{M}_{\kappa_i} \mathcal{M}_{\kappa_0}^{-1/2} \right) U^T = U S_{\kappa_i} U^T, \end{aligned}$$

where we have identified the step matrix in  $x$  coordinate system as  $S_{\kappa_i} = \mathcal{M}_{\kappa_0}^{-1/2} \mathcal{M}_{\kappa_i} \mathcal{M}_{\kappa_0}^{-1/2}$ . We also map the change in the error to the logarithmic space, i.e.  $f_{\kappa_i} = \log(\eta_{\kappa_i}/\eta_{\kappa})$ . Here, because the two coordinate systems represent the same physical system, we assume that  $\eta_{\kappa_i}/\eta_{\kappa_0}$  evaluates to the same value for all  $i = 1, \dots, n_{\text{config}}$  and  $\kappa \in \mathcal{T}_h$ , resulting in the same  $\{f_{\kappa_i}\}_{i=1}^{n_{\text{config}}}$  for both coordinate systems. To identify the rate matrix in the transformed coordinate system,  $\bar{R}_{\kappa}$ , we solve the minimization problem

$$\bar{R}_{\kappa} = \arg \min_{\bar{Q} \in \text{Sym}_d} \sum_{i=1}^{n_{\text{config}}} (f_{\kappa_i} - \text{tr}(\bar{Q} \bar{S}_{\kappa_i})) = \arg \min_{\bar{Q} \in \text{Sym}_d} \sum_{i=1}^{n_{\text{config}}} (f_{\kappa_i} - \text{tr}(\bar{Q} U S_{\kappa_i} U^T)).$$



Recalling  $R_\kappa$  in the original coordinate system is the solution to

$$R_\kappa = \arg \min_{Q \in \text{Sym}_d} \sum_{i=1}^{n_{\text{config}}} (f_{\kappa_i} - \text{tr}(QS_{\kappa_i})).$$

and noting that similarity transforms do not alter the value of trace, we immediately recognize the solution to the minimization problem on the transformed coordinate is related to that of the original coordinate by

$$\bar{R}_\kappa = UR_\kappa U^T.$$

As a result, the two error models are identical in the sense that

$$\bar{\eta}_\kappa(\bar{S}_\kappa) = \eta_{\kappa_0} \exp(\text{tr}(\bar{R}_\kappa \bar{S}_\kappa)) = \eta_{\kappa_0} \exp(\text{tr}(UR_\kappa U^T U S_\kappa U)) = \eta_{\kappa_0} \exp(\text{tr}(R_\kappa S_\kappa)) = \eta_\kappa(S_\kappa).$$

Similarly, the cost model is identical because

$$\bar{\rho}_\kappa(\bar{S}_\kappa) = \rho_{\kappa_0} \exp\left(\frac{1}{2}\text{tr}(\bar{S}_\kappa)\right) = \rho_{\kappa_0} \exp\left(\frac{1}{2}\text{tr}(US_\kappa U^T)\right) = \rho_{\kappa_0} \exp\left(\frac{1}{2}\text{tr}(S_\kappa)\right) = \rho_\kappa(S_\kappa).$$

Finally, to create the new vertex representation of the metric field, we solve the optimization problem on the surrogate model. Recall that the optimization algorithm relies entirely on the gradient of the surrogate error and cost functions. Let us denote the surrogate error and cost functions in the transformed space by  $\bar{\mathcal{E}}(\{\bar{S}_\nu\})$  and  $\bar{\mathcal{C}}(\{\bar{S}_\nu\})$ . Because the error and cost models are invariant under the coordinate transformation, their derivatives are related by simple coordinate transformations,

$$\frac{\partial \bar{\mathcal{E}}}{\partial \bar{S}_\nu} = U \frac{\partial \mathcal{E}}{\partial S_\nu} U^T \quad \text{and} \quad \frac{\partial \bar{\mathcal{C}}}{\partial \bar{S}_\nu} = U \frac{\partial \mathcal{C}}{\partial S_\nu} U^T.$$

Consequently, the proposed gradient descent algorithm produces the vertex step matrices in the transformed coordinate,  $\{\bar{S}_\nu\}$ , which are related to that solved in the original coordinate by

$$\bar{S}_\nu = US_\nu U^T.$$



The exponential map of the step matrices in the transformed coordinate yields

$$\begin{aligned}
\bar{\mathcal{M}}_\nu (\bar{S}_\nu) &= \bar{\mathcal{M}}_{\nu_0}^{1/2} \exp (\bar{S}_\nu) \bar{\mathcal{M}}_{\nu_0}^{1/2} \\
&= (\alpha^{-2} U \mathcal{M}_{\nu_0} U^T)^{1/2} \exp (U S_\nu U^T) (\alpha^{-2} U \mathcal{M}_{\nu_0} U^T)^{1/2} \\
&= \alpha^{-2} U \mathcal{M}_{\nu_0}^{1/2} \exp (S_\nu) \mathcal{M}_{\nu_0}^{1/2} U^T = \alpha^{-2} U \mathcal{M} (S_\nu) U^T.
\end{aligned}$$

Because the relationship between  $\bar{\mathcal{M}}_\nu (\bar{S}_\nu)$  and  $\mathcal{M} (S_\nu)$  is identical to the transformation of the tensor for the two coordinate systems, the two updated tensors  $\{\mathcal{M}_\nu\}_{\nu \in \mathcal{V}}$  and  $\{\bar{\mathcal{M}}_\nu\}_{\nu \in \mathcal{V}}$  represent the same physical tensor fields. Thus, our adaptation algorithm is invariant under coordinate transformation.







## Appendix G

# On Gradient Descent in the Metric Tensor Space

In the mesh optimization algorithm presented in Chapter 3, we assumed that the error is a function of the Riemannian metric field and solved an optimization problem on the metric field. The particular anisotropic error model employed in the optimization process employed an affine-invariant description of metric tensors, i.e. symmetric positive definite matrices. This appendix presents a few error models considered in designing a gradient-based optimization algorithm in the metric tensor space.

### G.1 The Choice of Metric

To perform a gradient-based optimization, we must first endow the metric tensor space with a “metric” (i.e. a sense of distance). To avoid confusions, the metric tensor used for interpolation is referred to as “tensor” whereas the “metric” is used to describe the distance measure with which the tensor space is equipped.

While the choice of metric is arbitrary, we note the following desirable properties:

1. A null element of the tensor space, i.e.  $\det(\mathcal{M}) = 0$ , is infinite distance from any elements. This guarantees that null elements are not reached in the gradient descent algorithm.
2. Metric resulting from edge split operations,  $\{\mathcal{M}_i\}_{i=1}^{m_d}$ , is equidistant from the original metric  $\mathcal{M}_0$ . The property ensures that the quality of the samples are independent



of the current configuration and, in particular, quality does not degrade on highly anisotropic elements.

3. The characterization of the metric does not rely on a particular decomposition of  $\mathcal{M}$ , e.g. the area  $A$ , the aspect ratio  $\mathcal{R}$ , and the orientation  $\theta$  in two dimensions. The decomposition-based characterization renders generalization to higher dimensions non-trivial.

Let us now present three error models that result from choosing different metrics.

### G.1.1 Frobenius Norm

Treating tensors as matrices with no additional properties, we may define the vector pointing from  $\mathcal{M}_0$  to  $\mathcal{M}$  by

$$S^F \equiv \mathcal{M} - \mathcal{M}_0$$

and the distance by taking the Frobenius norm, i.e.

$$d^F(\mathcal{M}, \mathcal{M}_0) = \|\mathcal{M} - \mathcal{M}_0\|_F.$$

However, we immediately conclude that metric  $d_F(\cdot, \cdot)$  has none of the aforementioned desired properties. For instance, the distance to the zero tensor is finite. The metric is unsuitable for our purpose and will not be considered any further.

### G.1.2 Log-Euclidean Framework

The second metric we consider is Log-Euclidean metric, proposed by Arsigny *et al.* [11].

The metric is defined by

$$d^{\text{LE}}(\mathcal{M}, \mathcal{M}_0) = \|\log(\mathcal{M}) - \log(\mathcal{M}_0)\|_F,$$

where  $\log(\cdot)$  is the matrix logarithm. Because the tensors are symmetric positive definite, the matrix logarithm is well defined. Note that this metric places the null tensor infinite distance from any other tensors. Furthermore, the distance is invariant under scalar scaling



and orthogonal transformations, i.e.

$$\begin{aligned} d^{\text{LE}}(\alpha\mathcal{M}, \alpha\mathcal{M}_0) &= d^{\text{LE}}(\mathcal{M}, \mathcal{M}_0), \quad \forall \alpha \in \mathbb{R}^+ \\ d^{\text{LE}}(U\mathcal{M}U^T, U\mathcal{M}_0U^T) &= d^{\text{LE}}(\mathcal{M}, \mathcal{M}_0), \quad \forall U \in \{V \in \mathbb{R}^d : VV^T = I\} \end{aligned}$$

Given the metric, it is natural to define the vector pointing from  $\mathcal{M}_0$  to  $\mathcal{M}$  by

$$S^{\text{LE}} \equiv \log(\mathcal{M}) - \log(\mathcal{M}_0) \in \text{Sym}_d. \quad (\text{G.1})$$

Similarly, we measure the distance between two errors  $\eta_0 \in \mathbb{R}^+$  and  $\eta \in \mathbb{R}^+$  in the logarithmic space, i.e.

$$f^{\text{LE}} \equiv \log(\eta) - \log(\eta_0) \in \mathbb{R}. \quad (\text{G.2})$$

With these choices, the linear error model based on the Log-Euclidean measure is given by

$$f^{\text{LE}} = \text{tr}(R^{\text{LE}} S^{\text{LE}})$$

or, equivalently,

$$\eta(\mathcal{M}) = \eta_0 \exp\left(\text{tr}\left(R^{\text{LE}}(\log(\mathcal{M}) - \log(\mathcal{M}_0))\right)\right),$$

where  $R^{\text{LE}} \in \text{Sym}_d$  is the parameter of the log-Euclidean-based model. Note that the log-Euclidean model is a generalization of the standard isotropic error model. That is, if we choose  $\mathcal{M}_0 = h_0^{-2}I$ ,  $\mathcal{M} = h^{-2}I$ , and  $R^{\text{LE}} = -(r/2d)I$ , then

$$\eta(h) = \eta_0 \left(\frac{h}{h_0}\right)^r.$$

The parameter  $R^{\text{LE}} \in \text{Sym}_d$  is deduced from regression. Given metric-error pairs  $\{\mathcal{M}_i, \eta_i\}_{i=0}^{n_{\text{config}}}$ , we can readily compute the log-Euclidean description of the metric-error pairs,  $\{S_i^{\text{LE}}, f_i^{\text{LE}}\}_{i=1}^{n_{\text{config}}}$ , using Eq. (G.1) and Eq. (G.2), and find

$$R^{\text{LE}} = \arg \min_{Q \in \text{Sym}_d} \sum_{i=1}^{n_{\text{config}}} \left(f_i^{\text{LE}} - \text{tr}(QS_i^{\text{LE}})\right)^2,$$



which entails finding the least-squares solution to a  $n_{\text{config}}\text{-by-}d(d+1)/2$  linear system. If we perform steepest descent by choosing  $S^{\text{LE}} = -\alpha R^{\text{LE}}$  for some  $\alpha > 0$  in Eq. (G.1), the resulting updated tensor is given by

$$\tilde{\mathcal{M}} = \exp(\log(\mathcal{M}_0) - \alpha R^{\text{LE}}). \quad (\text{G.3})$$

### G.1.3 Affine-Invariant Framework

The third metric we consider is affine-invariant metric, introduced by Pennec *et al.* [117] and defined by

$$d^{\text{AI}}(\mathcal{M}, \mathcal{M}_0) = \|\log(\mathcal{M}_0^{-1/2} \mathcal{M} \mathcal{M}_0^{-1/2})\|_F.$$

This metric also places the null element infinite distance from any other elements. Furthermore, the distance is invariant under the action of any linear transformation, i.e.

$$d^{\text{AI}}(A\mathcal{M}A^T, A\mathcal{M}_0A^T) = d^{\text{AI}}(\mathcal{M}, \mathcal{M}_0), \quad \forall A \in \mathbb{R}^{d \times d}$$

Given the metric, it is natural to define the vector from  $\mathcal{M}_0$  to  $\mathcal{M}$  by

$$S^{\text{AI}} = \log(\mathcal{M}_0^{-1/2} \mathcal{M} \mathcal{M}_0^{-1/2}) \in \text{Sym}_d. \quad (\text{G.4})$$

Similarly, we measure the distance between two errors  $\eta_0 \in \mathbb{R}^+$  and  $\eta \in \mathbb{R}^+$  in the logarithmic space as in the log-Euclidean case, i.e.

$$f^{\text{AI}} \equiv \log(\eta) - \log(\eta_0) \in \mathbb{R}. \quad (\text{G.5})$$

With these choices, the linear error model based on the affine-invariant measure is given by

$$f^{\text{AI}} = \text{tr}(R^{\text{AI}} S^{\text{AI}})$$

or, equivalently,

$$\eta(\mathcal{M}) = \eta_0 \exp\left(\text{tr}\left(R^{\text{AI}} \log\left(\mathcal{M}_0^{-1/2} \mathcal{M} \mathcal{M}_0^{-1/2}\right)\right)\right),$$



where  $R^{\text{AI}} \in \text{Sym}_d$  is the parameter of the affine-invariant-based model. As shown in Section F.2, the affine-invariant model is also a generalization the standard isotropic error model.

As in the log-Euclidean model, the parameter  $R^{\text{AI}} \in \text{Sym}_d$  is deduced from regression. Given metric-error pairs  $\{\mathcal{M}_i, \eta_i\}_{i=0}^{n_{\text{config}}}$ , we can readily compute the affine-invariant description of the metric-error pairs,  $\{S_i^{\text{AI}}, f_i^{\text{AI}}\}_{i=1}^{n_{\text{config}}}$ , using Eq. (G.4) and Eq. (G.5), and find

$$R^{\text{AI}} = \arg \min_{Q \in \text{Sym}_d} \sum_{i=1}^{n_{\text{config}}} (f_{\kappa_i}^{\text{AI}} - \text{tr}(QS_{\kappa_i}^{\text{AI}}))^2,$$

which entails finding the least-squares solution to a  $n_{\text{config}}$ -by- $d(d+1)/2$  linear system. If we perform steepest descent by choosing  $S^{\text{AI}} = -\alpha R^{\text{AI}}$  for some  $\alpha > 0$  in Eq. (G.4), the resulting update is

$$\tilde{\mathcal{M}} = \mathcal{M}_0^{1/2} \exp(-\alpha R^{\text{AI}}) \mathcal{M}_0^{1/2}. \quad (\text{G.6})$$

## G.2 Single Step Descent Test

Note that both the Log-Euclidean metric and the affine-invariant metrics are invariant under scalar multiplication, i.e.

$$d(\mathcal{M}, \mathcal{M}_0) = d(\alpha \mathcal{M}, \alpha \mathcal{M}_0), \quad \forall \mathcal{M}_0, \mathcal{M} \in \text{Sym}_d^+, \forall \alpha \in \mathbb{R}^+.$$

Thus, we only need to consider the effect of non-scaling operations. In particular, for  $2 \times 2$  tensors, we need to consider rotation and aspect ratio deformations.

Throughout this section, we visually assess the quality of the descent algorithms. In order to do so, we perform a step of gradient descent using Eq. (G.3) and Eq. (G.6), starting from a given metric  $\mathcal{M}_0$  and a prescribed error vector  $\eta_i/\eta_0$ . Then we visualize resulting updated metric in three different measures. First is the relative change from the original configuration measured in the log-Euclidean sense, i.e.

$$\exp(\log(\mathcal{M}) - \log(\mathcal{M}_0)) \in \text{Sym}_d^+.$$



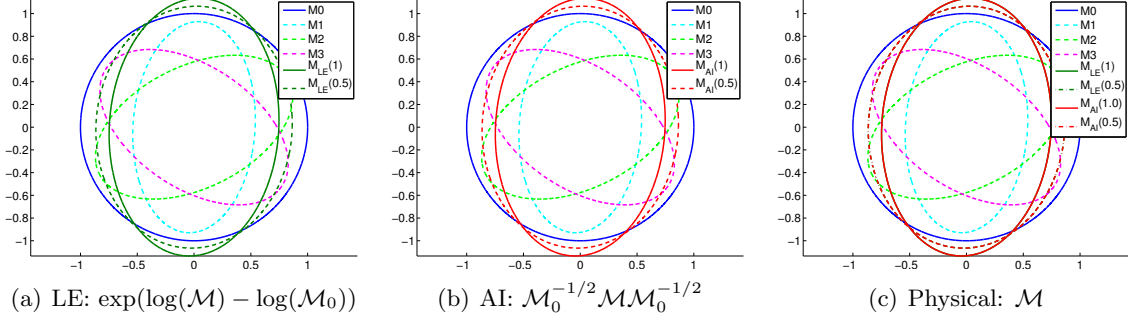


Figure G-1: LE and AI gradient descent from  $\mathcal{M}_0 = I$  for  $\eta/\eta_0 = (0.5, 1.0, 1.0)$ .

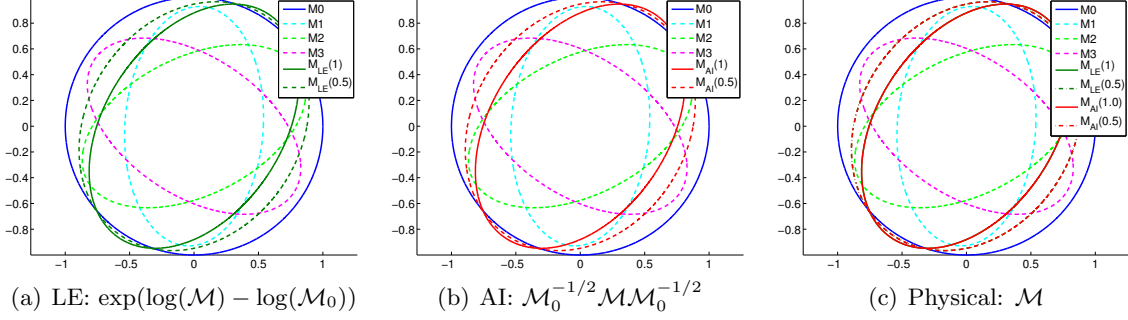


Figure G-2: LE and AI gradient descent from  $\mathcal{M}_0 = I$  for  $\eta/\eta_0 = (0.5, 0.5, 1.0)$ .

Note that the original tensor,  $\mathcal{M}_0$ , maps to the identity, which is visualized as a unit circle. Second measure is the relative change from the original configuration measured in the affine invariant sense, i.e.

$$\mathcal{M}_0^{-1/2} \mathcal{M} \mathcal{M}_0^{-1/2} \in Sym_d^+.$$

The third measure is the metric measured in the physical space, i.e. with respect to the identity tensor.

### G.2.1 Action from the Identity Tensor

We first compare the action of the gradient descent algorithm starting from the identity tensor, i.e.  $\mathcal{M}_0 = I$ . Figure G-1 shows the results of taking a step of gradient descent for an error configuration  $\eta/\eta_0 = (0.5, 1.0, 1.0)$ . Because the error only decreases for the first configuration, we expect the gradient descent to step in the direction of  $\mathcal{M}_1$ . Figure confirms that both log-Euclidean and affine-invariant error models achieve this; in fact, the two models are identical when  $\mathcal{M}_0 = I$  and thus the models produce identical results.



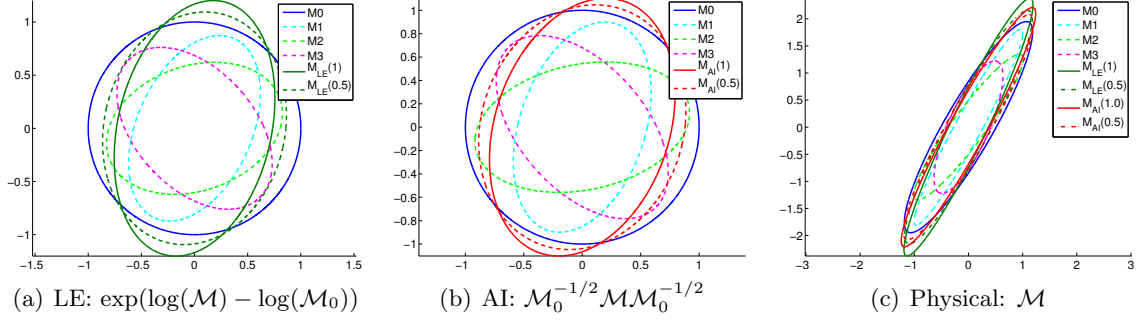


Figure G-3: LE and AI gradient descent from  $\mathcal{R}(\mathcal{M}_0) = 5$  for  $\eta/\eta_0 = (0.5, 1.0, 1.0)$ .

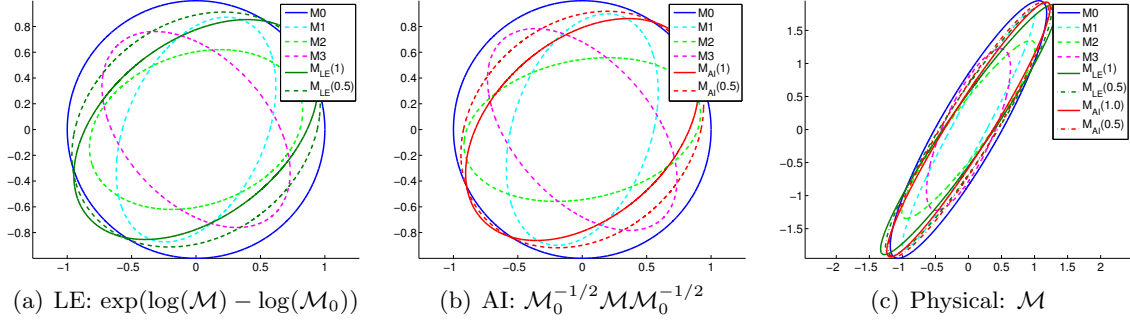


Figure G-4: LE and AI gradient descent from  $\mathcal{R}(\mathcal{M}_0) = 5$  for  $\eta/\eta_0 = (0.5, 0.5, 1.0)$ .

Figure G-2 shows the results of taking a step for an error configuration  $\eta/\eta_0 = (0.5, 0.5, 1.0)$ . For this configuration, we expect gradient descent to produce a tensor that is in some sense an average of  $\mathcal{M}_1$  and  $\mathcal{M}_2$ . The figure confirms that this is the case for both models.

## G.2.2 Action from an $\mathcal{R} = 5$ Tensor

We repeat the same test but starting from the original tensor  $\mathcal{M}_0$  with an aspect ratio of  $\mathcal{R} = 5$ . As before, we first consider an error configuration  $\eta/\eta_0 = (0.5, 1.0, 1.0)$ . The correct behavior is to step in the direction of  $\mathcal{M}_1$ . Figure G-3 shows that the affine-invariant model steps exactly in the direction of  $\mathcal{M}_1$ ; this is not surprising because the action in the step taken by the model is invariant to  $\mathcal{M}_0$ . On the other hand, the log-Euclidean framework takes a step that is a combination of  $\mathcal{M}_1$  and  $\mathcal{M}_2$ . Similarly, Figure G-4 shows the affine-invariant model exhibiting the desired behavior, whereas the log-Euclidean model produces a non-ideal result.



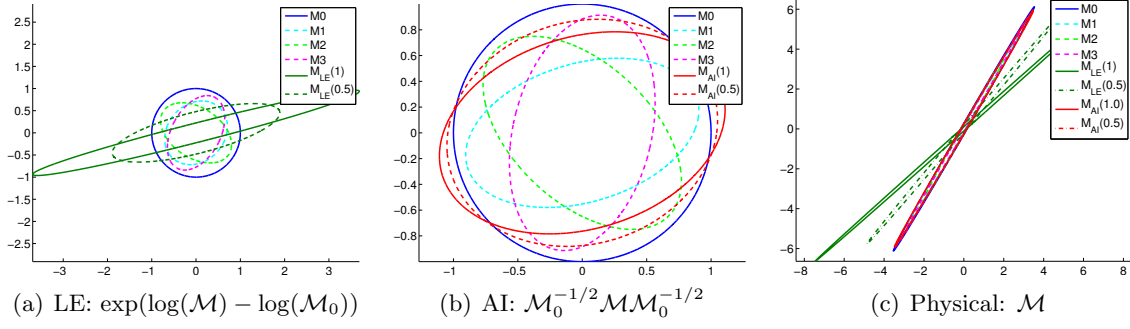


Figure G-5: LE and AI gradient descent from  $\mathcal{R}(\mathcal{M}_0) = 50$  for  $\eta/\eta_0 = (0.5, 1.0, 1.0)$ .

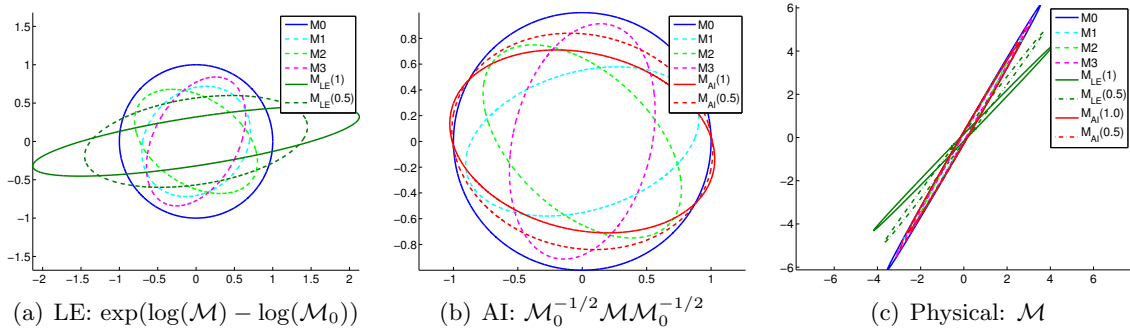


Figure G-6: LE and AI gradient descent from  $\mathcal{R}(\mathcal{M}_0) = 50$  for  $\eta/\eta_0 = (0.5, 0.5, 1.0)$ .

### G.2.3 Action from an $\mathcal{R} = 50$ Tensor

To study the effect of increased aspect ratio, we now consider  $\mathcal{M}_0$  with the aspect ratio of  $\mathcal{R} = 50$ . Figures G-5 and G-6 show that the affine-invariant model produces identical behavior to the two previous cases. On the other hand, the log-Euclidean model produces undesirable result. Figure G-5(a) shows that the two samples from  $\mathcal{M}_2$  and  $\mathcal{M}_3$  become nearly degenerate; as a result, the gradient reconstruction is unstable and gradient descent produces an undesirable metric.

Comparison of the results obtained for  $\mathcal{M}_0$  with  $\mathcal{R} = 1, 5$ , and  $50$  show that the action of the affine-invariant model is invariant to  $\mathcal{M}_0$ , producing the desired results for any aspect ratio. On the other hand, the log-Euclidean model becomes unstable for higher aspect ratios. Thus, the error model based on the log-Euclidean metric is unsuited for a gradient-based mesh optimization.



### G.3 Multi- Step Descent Test

We now compare the performance of the descent algorithms based on the log-Euclidean and affine-invariant error models. In particular, we assume that the error is of the form

$$\eta(\mathcal{H}) = \eta_{\text{opt}}[\sigma_{\max}(\mathcal{H}\mathcal{H}_{\text{opt}}^{-1})]^r,$$

where where  $\mathcal{H} = \mathcal{M}^{-1/2}$  is the generalized element size,  $r$  is the convergence rate,  $\sigma_{\max}(\cdot)$  is the maximum singular-value operator, and  $\mathcal{H}_{\text{opt}}$  is the tensor having  $\det(\mathcal{H}_{\text{opt}}) = 1$  and gives minimum error  $\eta_{\text{opt}}$ . Clearly, the minimum is attained at  $\mathcal{H} = \mathcal{H}_{\text{opt}}$ .

Specifically, the test runs as follows: Given an optimal configuration  $\{\mathcal{M}_{\text{opt}}, \eta_{\text{opt}}\}$  and the initial metric  $\mathcal{M}^{(0)}$ ,

1. Set  $i = 0$
2. Construct a triangle  $\Delta^{(i)}$  which conforms to  $\mathcal{M}^{(i)}$  (not unique).
3. Split the  $j$ -th edge of the triangle  $\Delta^{(i)}$  and compute the metric tensor,  $\mathcal{M}_j^{(i)}$ , and the error,  $\eta_j^{(i)}$ , associated with the split configurations for  $j = 1, \dots, 3$ .
4. Perform a step of gradient descent :  $\tilde{\mathcal{M}}^{(i+1)} = \text{descent}(\{\mathcal{M}_j^{(i)}, \eta_j^{(i)}\}_{j=0}^4)$
5. Isotropically scale  $\tilde{\mathcal{M}}^{(i+1)}$  such that  $\det(\mathcal{M}^{(i+1)}) = 1$ .
6. Set  $i \leftarrow i + 1$ , go to 2.

We consider three options for the descent step, Step 4.

- **Minimum:** take  $\mathcal{M}_i$  that gives the minimum error
- **Log-Euclidean:** take a step using the log-Euclidean model, Eq. (G.3), with  $\alpha = 1$
- **Affine-Invariant:** take a step using the affine-invariant model, Eq. (G.6), with  $\alpha = 1$

#### G.3.1 From the Identity Tensor to an $\mathcal{R} = 2$ Tensor

Figure G-7 shows the result of the multi-step gradient descent test starting from the identity tensor toward an optimal tensor with  $\mathcal{R} = 2$ . Both the sequences of the tensors generated and the error history are shown. The descent algorithm based on choosing the minimum error does not converge to the optimal tensor, as the algorithm always steers toward one of



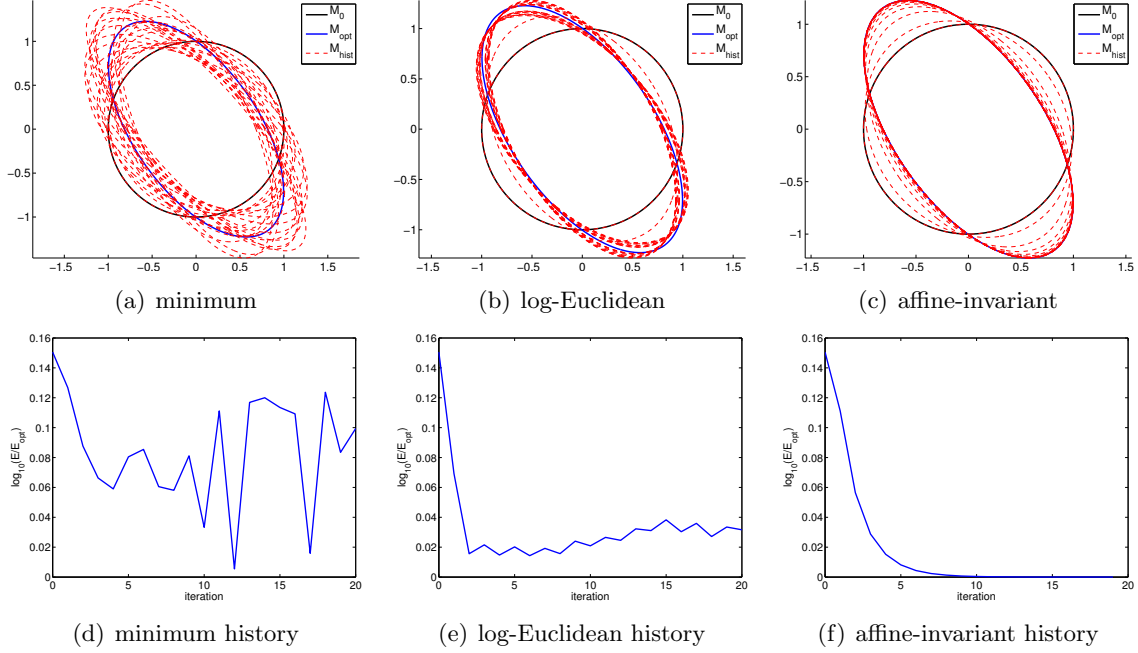


Figure G-7: Multi-step descent test from the identity tensor to an  $\mathcal{R} = 2$  tensor.

the three split configurations. The log-Euclidean solution also fluctuates about the optimal metric, but the fluctuation is smaller than that using the minimum descent. The affine-invariant metric approaches the optimal configuration smoothly, resulting in a monotonic error convergence.

### G.3.2 From the Identity Tensor to an $\mathcal{R} = 20$ Tensor

Figure G-8 shows the result of the multi-step gradient descent test starting from the identity tensor toward an optimal tensor with  $\mathcal{R} = 20$ . Similar to the  $\mathcal{R} = 2$  case, the minimum-metric descent results in fluctuation about the optimal configuration. The log-Euclidean descent does not work, even for this moderate aspect ratio; the result is consistent with the behavior observed in the single-step tests in Section G.2. The affine-invariant descent again converges monotonically to the optimal configuration; the number of steps required is higher than that for the  $\mathcal{R} = 2$  case.

### G.3.3 From an $\mathcal{AR} = 5$ Tensor to an $\mathcal{R} = 20$ Tensor

Figure G-9 shows the result of the multi-step gradient descent test starting from an initial metric with  $\mathcal{R} = 5$  toward an optimal tensor with  $\mathcal{R} = 20$  at a different orientation.



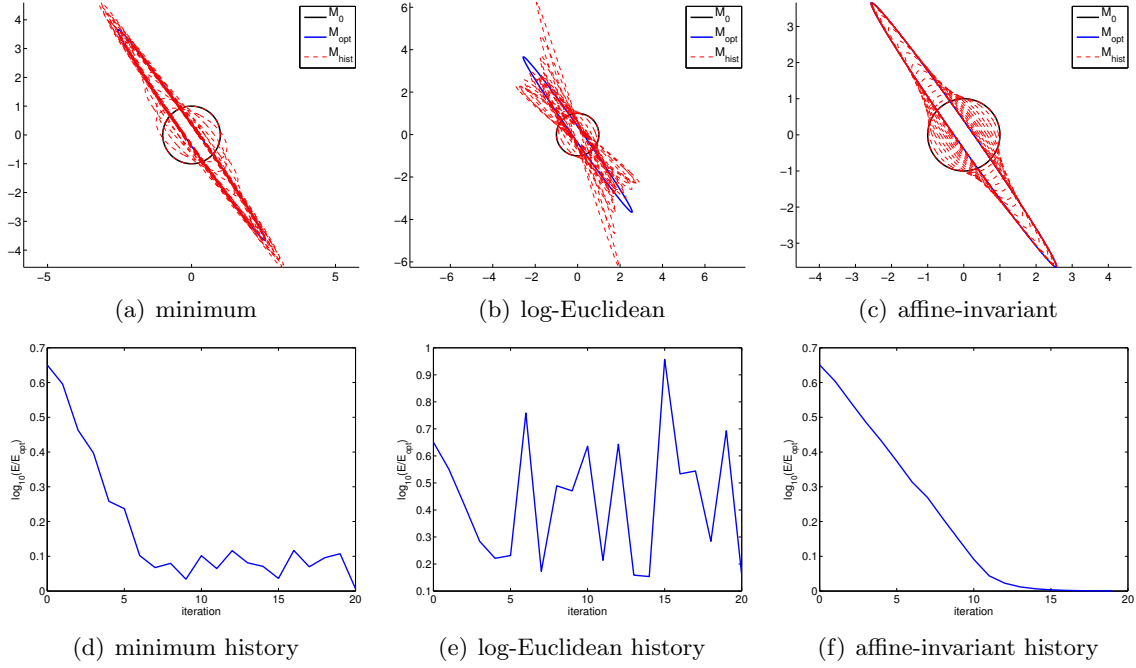


Figure G-8: Multi-step descent test from the identity tensor to an  $\mathcal{R} = 20$  tensor.

Similar to the previous case, the log-Euclidean descent algorithm does not work even for this moderate aspect ratio. The affine invariant descent algorithm morphs the metric smoothly, simultaneously adjusting the aspect ratio and the orientation.



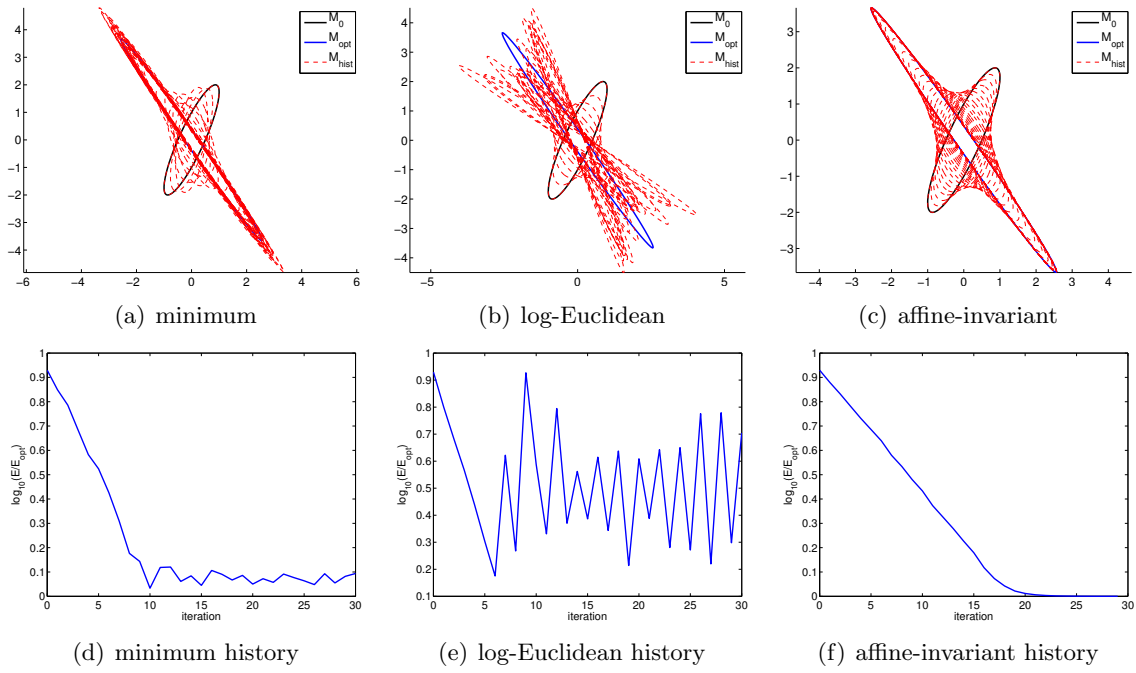


Figure G-9: Multi-step descent test from an  $\mathcal{R} = 5$  tensor to an  $\mathcal{R} = 20$  tensor.



# Bibliography

- [1] Ainsworth, M. and Oden, J. T. *A Posteriori Error Estimation in Finite Element Analysis*. John Wiley and Sons, 2000.
- [2] Ainsworth, M. and Oden, J. T. “A posteriori error estimation in finite element analysis.” *Comput. Methods Appl. Mech. Engrg.*, 142:1–88, 1997.
- [3] Ainsworth, M. and Senior, B. “An adaptive refinement strategy for *hp*-finite element computations.” *Appl. Numer. Math.*, 26:165–178, 1998.
- [4] Alauzet, F., Frey, P. J., George, P. L., and Mohammadi, B. “3D transient fixed point mesh adaptation for time-dependent problems: Application to CFD simulations.” *J. Comput. Phys.*, 222(2):592–623, 2007.
- [5] Alauzet, F., Belme, A., and Dervieux, A. “Anisotropic Goal-Oriented Mesh Adaptation for Time Dependent Problems.” In Quadros, W. R., editor, *Proceedings of the 20th International Meshing Roundtable*, pages 99–121. Springer Berlin Heidelberg, 2012.
- [6] Allmaras, S. R. *A coupled Euler/Navier-Stokes algorithm for 2-D unsteady transonic shock/boundary-layer interaction*. PhD thesis, Massachusetts Institute of Technology, 1989.
- [7] Allmaras, S. R. and Giles, M. B. “A second-order flux split scheme for the unsteady 2-D Euler equations on arbitrary meshes.” AIAA 1987-1119-CP, 1987.
- [8] Allmaras, S. R., Venkatakrishnan, V., and Johnson, F. T. “Farfield Boundary Conditions for 2-D Airfoils.” AIAA 2005-4711, 2005.
- [9] Allmaras, S. R., Bussoletti, J. E., Hilmes, C. L., Johnson, F. T., Melvin, R. G., Tinoco, E. N., Venkatakrishnan, V., Wigton, L. B., and Young, D. P. “Algorithm Issues and Challenges Associated with the Development of Robust CFD Codes.” In Buttazzo, G. and Frediani, A., editors, *Variational Analysis and Aerospace Engineering*. Springer Science+Business Media, 2009.
- [10] Arnold, D. N., Brezzi, F., Cockburn, B., and Marini, L. D. “Unified analysis of discontinuous Galerkin methods for elliptical problems.” *SIAM J. Numer. Anal.*, 39(5):1749–1779, 2002.
- [11] Arsigny, V., Fillard, P., Pennec, X., and Ayache, N. “Log-Euclidean metrics for fast and simple calculus on diffusion tensors.” *Magnetic Resonance in Medicine*, 56: 411–421, 2006.



- [12] Babuska, I., Szabo, B. A., and Katz, I. N. “The  $p$ -version of the finite element method.” *SIAM J. Numer. Anal.*, 18(3):515–545, 1981.
- [13] Babuška, I. and Reinboldt, W. C. “ $A$ -posteriori error estimates for the finite element method.” *Internat. J. Numer. Methods Engrg.*, 12:1597–1615, 1978.
- [14] Bangerth, W., Geiger, M., and Rannacher, R. “Adaptive Galerkin finite element methods for the wave equation.” *Comput. Methods Appl. Math.*, 10(1):3–48, 2010.
- [15] Bangerth, W. and Rannacher, R. “Finite element approximation of the acoustic wave equation: error control and mesh adaptation.” *East-West Journal of Numerical Mathematics*, 7(4):263–282, 1999.
- [16] Bangerth, W. and Rannacher, R. *Adaptive Finite Element Methods for Differential Equations*. Birkhäuser Verlag, Basel, 2003.
- [17] Bar-Yoseph, P. “Space-time discontinuous finite element approximations for multi-dimensional nonlinear hyperbolic systems.” *Computational Mechanics*, 5:145–160, 1989.
- [18] Barter, G. E. *Shock Capturing with PDE-Based Artificial Viscosity for an Adaptive, Higher-Order, Discontinuous Galerkin Finite Element Method*. PhD thesis, Massachusetts Institute of Technology, Department of Aeronautics and Astronautics, June 2008.
- [19] Barter, G. E. and Darmofal, D. L. “Shock capturing with PDE-based artificial viscosity for DGFEM: Part I, Formulation.” *J. Comput. Phys.*, 229(5):1810–1827, 2010.
- [20] Barth, T. J. “Recent developments in high-order  $k$ -exact reconstruction on unstructured meshes.” AIAA 1993-0668, 1993.
- [21] Barth, T. J. “Numerical Methods for Gasdynamic Systems on Unstructured Meshes.” In Kroner, D., Olhberger, M., and Rohde, C., editors, *An Introduction to Recent Developments in Theory and Numerics for Conservation Laws*, pages 195 – 282. Springer-Verlag, 1999.
- [22] Bassi, F., Crivellini, A., Rebay, S., and Savini, M. “Discontinuous Galerkin solution of the Reynolds averaged Navier-Stokes and  $k$ - $\omega$  turbulence model equations.” *Comput. & Fluids*, 34:507–540, May-June 2005.
- [23] Bassi, F. and Rebay, S. “High-order accurate discontinuous finite element solution of the 2D Euler equations.” *J. Comput. Phys.*, 138(2):251–285, 1997.
- [24] Bassi, F. and Rebay, S. “A high-order discontinuous finite element method for the numerical solution of the compressible Navier-Stokes equations.” *J. Comput. Phys.*, 131:267–279, 1997.
- [25] Bassi, F. and Rebay, S. “GMRES discontinuous Galerkin solution of the compressible Navier-Stokes equations.” In Cockburn, K. and Shu, editors, *Discontinuous Galerkin Methods: Theory, Computation and Applications*, pages 197–208. Springer, Berlin, 2000.



- [26] Becker, R. and Rannacher, R. “A feed-back approach to error control in finite element methods: Basic analysis and examples.” *East-West J. Numer. Math.*, 4:237–264, 1996.
- [27] Becker, R. and Rannacher, R. “An optimal control approach to a posteriori error estimation in finite element methods.” In Iserles, A., editor, *Acta Numerica*. Cambridge University Press, 2001.
- [28] Belhamadia, Y., Fortin, A., and E.Chamberland. “Three-dimensional anisotropic mesh adaptation for phase change problems.” *J. Comput. Phys.*, 201:753–770, 2004.
- [29] Berkooz, G., Holmes, P., and Lumley, J. L. “The proper orthogonal decomposition in the analysis of turbulent flows.” *Annual Review of Fluid Mechanics*, 25:539–575, 1993.
- [30] Borouchaki, H., George, P. L., Hecht, F., Laug, P., and Saltel, E. “Delaunay mesh generation governed by metric specifications. Part I algorithms.” *Finite Elem. Anal. Des.*, 25(1-2):61–83, 1997.
- [31] Bottasso, C. L. “Anisotropic mesh adaptation by metric-driven optimization.” *Internat. J. Numer. Methods Engrg.*, 60(3):597–639, 2004.
- [32] Brenner, S. C. and Scott, L. R. *The Mathematical Thoery of Finite Element Methods, Third Edition*. Springer, New York, 2008.
- [33] Brezzi, F., Manzini, M., Marini, D., Pietra, P., and Russo, A. “Discontinuous finite elements for diffusion problems.” In *Francesco Brioschi (1824-1897) convegno di studi matematici, October 22-23, 1997*, Ist. Lomb. Acc. Sc. Lett., Incontro di studio N. 16, pages 197–217, 1999.
- [34] Burgess, N. K. and Mavriplis, D. J. “An  $hp$ -adaptive discontinuous Galerkin solver for aerodynamic flows on mixed-element meshes.” AIAA 2011-490, 2011.
- [35] Cao, W. “On the error of linear interpolation and the orientation, aspect ratio, and internal angles of a triangle.” *SIAM J. Numer. Anal.*, 43(1):19–40, 2005.
- [36] Cao, W. “Anisotropic measures of third order derivatives and the quadratic interpolation error on triangular elements.” *SIAM J. Sci. Comput.*, 29(2):756–781, 2007.
- [37] Cao, W. “An interpolation error estimate on anisotropic meshes in  $R^n$  and optimal metrics for mesh refinement.” *SIAM J. Numer. Anal.*, 45(6):2368–2391, 2007.
- [38] Cao, W. “An interpolation error estimate in  $\mathcal{R}^2$  based on the anisotropic measures of higher order derivatives.” *Math. Comp.*, 77(261):265–286, 2008.
- [39] Castro-Díaz, M. J., Hecht, F., Mohammadi, B., and Pironneau, O. “Anisotropic unstructured mesh adaptation for flow simulations.” *Internat. J. Numer. Methods Fluids*, 25:475–491, 1997.
- [40] Ceze, M. and Fidkowski, K. J. “Output-driven anisotropic mesh adaptation for viscous flows using discrete choice optimization.” AIAA 2010-170, 2010.
- [41] Chavent, G. and Salzano, G. “A finite element method for the 1D water flooding problem with gravity.” *J. Comput. Phys.*, 42:307–344, 1982.



- [42] Cockburn, B., Hou, S., and Shu, C. W. “TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws IV: The multidimensional case.” *Math. Comp.*, 54:545–581, 1990.
- [43] Cockburn, B., Karniadakis, G., and Shu, C. “The development of discontinuous Galerkin methods.” In *Lecture Notes in Computational Science and Engineering*, volume 11. Springer, 2000.
- [44] Cockburn, B., Lin, S. Y., and Shu, C. W. “TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws III: One dimensional systems.” *J. Comput. Phys.*, 84:90–113, 1989.
- [45] Cockburn, B. and Shu, C. W. “TVB Runge-Kutta local projection discontinuous Galerkin finite element method for scalar conservation laws II: General framework.” *Math. Comp.*, 52:411–435, 1989.
- [46] Cockburn, B. and Shu, C. W. “The local discontinuous Galerkin method for time-dependent convection-diffusion systems.” *SIAM J. Numer. Anal.*, 35(6):2440–2463, December 1998.
- [47] Cockburn, B. and Shu, C. W. “The Runge-Kutta discontinuous Galerkin finite element method for conservation laws V: Multidimensional systems.” *J. Comput. Phys.*, 141:199–224, 1998.
- [48] Cockburn, B., Dong, B., and Guzman, J. “A superconvergent LDG-hybridizable Galerkin method for second-order elliptic problems.” *Math. Comp.*, 77(264):1887–1916, 2008.
- [49] Collins, M., Vecchio, F., Selby, R., and Gupta, P. “Failure of an offshore platform.” *Concrete International*, 19(8):28–35, 1997.
- [50] Demkowicz, L., Devloo, P., and Oden, J. T. “On an  $h$ -type mesh-refinement strategy based on minimization of interpolation errors.” *Comput. Methods Appl. Mech. Engrg.*, 53:67–89, 1985.
- [51] Dervieux, A., Leservoisier, D., Paul-Louis, and Coudière, Y. “About theoretical and practical impact of mesh adaptation on approximation of functions and PDE solutions.” *Internat. J. Numer. Methods Fluids*, 43:507–516, 2003.
- [52] Diosady, L. T. and Darmofal, D. L. “Preconditioning methods for discontinuous Galerkin solutions of the Navier-Stokes equations.” *J. Comput. Phys.*, 228:3917–3935, 2009.
- [53] Drela, M. Personal Communication via email, November 2010.
- [54] Eriksson, K. and Johnson, C. “Error estimates and automatic time step control for nonlinear parabolic problems, I.” *SIAM J. Numer. Anal.*, 24(1):12–23, 1987.
- [55] Fidkowski, K. and Darmofal, D. “Review of output-based error estimation and mesh adaptation in computational fluid dynamics.” *AIAA Journal*, 49(4):673–694, 2011.
- [56] Fidkowski, K. J. “A High-Order Discontinuous Galerkin Multigrid Solver for Aerodynamic Applications.” Masters thesis, Massachusetts Institute of Technology, Department of Aeronautics and Astronautics, June 2004.



- [57] Fidkowski, K. J. *A Simplex Cut-Cell Adaptive Method for High-Order Discretizations of the Compressible Navier-Stokes Equations*. PhD thesis, Massachusetts Institute of Technology, Department of Aeronautics and Astronautics, June 2007.
- [58] Fidkowski, K. J. and Darmofal, D. L. “A triangular cut-cell adaptive method for higher-order discretizations of the compressible Navier-Stokes equations.” *J. Comput. Phys.*, 225:1653–1672, 2007.
- [59] Fidkowski, K. J. and Luo, Y. “Output-based spacetime mesh adaptation for the compressible NavierStokes equations.” *J. Comput. Phys.*, 230(14):5753 – 5773, 2011.
- [60] Fidkowski, K. J. and Roe, P. L. “An entropy adjoint approach to mesh refinement.” *SIAM J. Sci. Comput.*, 32(3):1261–1287, 2010.
- [61] Formaggia, L., Micheletti, S., and Perotto, S. “Anisotropic mesh adaptation with applications to CFD problems.” In Mang, H. A., Rammerstorfer, F. G., and Eberhardsteiner, J., editors, *Fifth World Congress on Computational Mechanics*, Vienna, Austria, July 7-12 2002.
- [62] Formaggia, L., Perotto, S., and Zunino, P. “An anisotropic a-posteriori error estimate for a convection-diffusion problem.” *Comput. Visual Sci.*, 4:99–104, 2001.
- [63] George, P. L., Hecht, F., and Vallet, M. G. “Creation of internal points in Voronoi’s type method. Control adaptation.” *Advances in Engineering Software and Workstations*, 13:303–312, 1991.
- [64] Georgoulis, E. H., Hall, E., and Houston, P. “Discontinuous Galerkin methods on *hp*-anisotropic meshes I: A priori error analysis.” *Int. J. of Computing Science and Mathematics*, 1(2–4), 2007.
- [65] Georgoulis, E. H., Hall, E., and Houston, P. “Discontinuous Galerkin methods on *hp*-anisotropic meshes II: A posteriori error analysis and adaptivity.” *Appl. Numer. Math.*, 59:2179–2194, 2009.
- [66] Giles, M. B. and Süli, E. “Adjoint methods for PDEs: a posteriori error analysis and postprocessing by duality.” In *Acta Numerica*, volume 11, pages 145–236, 2002.
- [67] Harriman, K., Houston, P., Senior, B., and Süli, E. “hp-version discontinuous Galerkin methods with interior penalty for partial differential equations with nonnegative characteristic form.” Report NA 02/21, Oxford University Computing Lab Numerical Analysis Group, 2002.
- [68] Hartmann, R. “Adjoint consistency analysis of discontinuous Galerkin discretizations.” *SIAM J. Numer. Anal.*, 45(6):2671–2696, 2007.
- [69] Hartmann, R. “Adaptive FE methods for conservation equations.” In Freistühler, H. and Warnecke, G., editors, *Hyperbolic Problems: Theory, Numerics, Applications: Eighth International Conference in Magdeburg, February, March 2000*, volume 141 of *International series of numerical mathematics*, pages 495–503. Birkhäuser, Basel, 2001.
- [70] Hartmann, R. *Adaptive Finite Element Methods for the Compressible Euler Equations*. PhD thesis, University of Heidelberg, 2002.



- [71] Hartmann, R. and Houston, P. “Adaptive discontinuous Galerkin finite element methods for the compressible Euler equations.” *J. Comput. Phys.*, 183(2):508–532, 2002.
- [72] Hartmann, R. and Houston, P. “Error estimation and adaptive mesh refinement for aerodynamic flows.” In Deconinck, H., editor, *VKI LS 2010-01: 36<sup>th</sup> CFD/ADIGMA course on hp-adaptive and hp-multigrid methods, Oct. 26-30, 2009*. Von Karman Institute for Fluid Dynamics, Rhode Saint Genèse, Belgium, 2009.
- [73] Hecht, F. “BAMG: Bidimensional Anisotropic Mesh Generator.” 1998.  
<http://www-rocq1.inria.fr/gamma/cdrom/www/bamg/eng.htm>.
- [74] Houston, P., Georgoulis, E. H., and Hall, E. “Adaptivity and a posteriori error estimation for DG methods on anisotropic meshes.” *International Conference on Boundary and Interior Layers*, 2006.
- [75] Houston, P. and Süli, E. “A note on the design of *hp*-adaptive finite element methods for elliptic partial differential equations.” *Comput. Methods Appl. Mech. Engrg.*, 194: 229–243, 2005.
- [76] Hughes, T. J. R. “A simple scheme for developing upwind finite elements.” *Internat. J. Numer. Methods Engrg.*, 12:1359–1365, 1978.
- [77] Hughes, T. J. R., Franca, L. P., and Hulbert, G. M. “A new finite element formulation for computational fluid dynamics: VIII. The Galerkin/least-squares method for advective-diffusive equations.” *Comput. Methods Appl. Mech. Engrg.*, 73:173–189, 1989.
- [78] Hughes, T. J. R., Franca, L. P., and Mallet, M. “A new finite element formulation for computational fluid dynamics: VI. Convergence analysis of the generalized SUPG formulation for linear time-dependent multi-dimensional advective-diffusive systems.” *Comput. Methods Appl. Mech. Engrg.*, 63(1):97–112, 1987.
- [79] Hughes, T. J. R. and Hulbert, G. M. “Space-time finite element methods for elastodynamics: Formulations and error estimates.” *Comput. Methods Appl. Mech. Engrg.*, 66:339–363, 1988.
- [80] Hughes, T. J. R. and Mallet, M. “A new finite element formulation for computational fluid dynamics: III The generalized streamline operator for multidimensional advective-diffusive systems.” *Comput. Methods Appl. Mech. Engrg.*, 58:305–328, 1986.
- [81] Hughes, T. J. R. and Mallet, M. “A new finite element formulation for computational fluid dynamics: IV A discontinuity capturing operator for multidimensional advective-diffusive systems.” *Comput. Methods Appl. Mech. Engrg.*, 58:329–336, 1986.
- [82] Hughes, T. J. R., Mallet, M., and Mizukami, A. “A new finite element formulation for computational fluid dynamics: II Beyond SUPG.” *Comput. Methods Appl. Mech. Engrg.*, 54:341–355, 1986.
- [83] Hughes, T. J. R. and Tezduyar, T. E. “Finite element methods for first-order hyperbolic systems with particular emphasis on the compressible Euler equations.” *Comput. Methods Appl. Mech. Engrg.*, 45:217–284, 1984.



- [84] Jakobsen, B. and Rosendahl, F. “The Sleipner platform accident.” *Structural Engineering International*, 4(3):190–193, 1994.
- [85] Johnson, C. and Pitkäranta, J. “An analysis of the discontinuous Galerkin method for a scalar hyperbolic equation.” *Math. Comp.*, 46:1–26, 1986.
- [86] Johnson, C. “Error estimates and adaptive time-step control for a class of one-step methods for stiff ordinary differential equations.” *SIAM J. Numer. Anal.*, 25(4):908–926, 1988.
- [87] Johnson, C. “Discontinuous Galerkin finite element methods for second order hyperbolic problems.” *Comput. Methods Appl. Mech. Engrg.*, 107:117–129, 1993.
- [88] Jones, W. T., Nielsen, E. J., and Park, M. A. “Validation of 3d adjoint based error estimation and mesh adaptation for sonic boom prediction.” AIAA 2006-1150, 2006.
- [89] Klausmeyer, S. M. and Lin, J. C. “Comparative results from a CFD challenge over a 2D three-element high-lift airfoil.” NASA Technical Memorandum 112858, 1997.
- [90] Korczak, K. Z. and Patera, A. T. “An isoparametric spectral element method for solution of the Navier-Stokes equations in complex geometry.” *J. Comput. Phys.*, 62:361–382, 1984.
- [91] Leicht, T. and Hartmann, R. “Error estimation and hp-adaptive mesh refinement for discontinuous Galerkin methods.” In Wang, Z. J., editor, *Adaptive High-Order Methods in Computational Fluid Dynamics*, pages 67–94. World Science Books, 2011.
- [92] Leicht, T. and Hartmann, R. “Anisotropic mesh refinement for discontinuous Galerkin methods in two-dimensional aerodynamic flow simulations.” *Internat. J. Numer. Methods Fluids*, 56:2111–2138, 2008.
- [93] Leicht, T. and Hartmann, R. “Error estimation and anisotropic mesh refinement for 3d laminar aerodynamic flow simulations.” *J. Comput. Phys.*, 229:7344–7360, 2010.
- [94] Li, X., Shephard, M. S., and Beal, M. W. “3D anisotropic mesh adaptation by mesh modification.” *Comput. Methods Appl. Mech. Engrg.*, 194:4915–4950, 2005.
- [95] Lin, G., Su, C.-H., and Karniadakis, G. E. “Predicting shock dynamics in the presence of uncertainties.” *J. Comput. Phys.*, 217:260–276, 2006.
- [96] Loseille, A., Dervieux, A., and Alauzet, F. “Fully anisotropic goal-oriented mesh adaptation for 3d steady Euler equations.” *J. Comput. Phys.*, 229:2866–2897, 2010.
- [97] Loseille, A. and Alauzet, F. “Continuous mesh framework part I: well-posed continuous interpolation error.” *SIAM J. Numer. Anal.*, 49(1):38–60, 2011.
- [98] Loseille, A. and Alauzet, F. “Continuous mesh framework part II: validations and applications.” *SIAM J. Numer. Anal.*, 49(1):61–86, 2011.
- [99] Loseille, A., Dervieux, A., Frey, P., and Alauzet, F. “Achievement of global second order mesh convergence for discontinuous flows with adapted unstructured meshes.” AIAA 2007-4186, 2007.



- [100] Lu, J. *An a Posteriori Error Control Framework for Adaptive Precision Optimization Using Discontinuous Galerkin Finite Element Method*. PhD thesis, Massachusetts Institute of Technology, Cambridge, Massachusetts, 2005.
- [101] Machiels, L., Patera, A. T., Peraire, J., and Maday, Y. “A General Framework for Finite Element A Posteriori Error Control: Application to Linear and Nonlinear Convection-Dominated Problems.” In Baines, M., editor, *Numerical Methods for Fluid Dynamics (ICFD, Oxford, UK)*, 1998.
- [102] Mathelin, L., , and Maître, O. L. “Dual-based *a posteriori* error estimate for stochastic finite element methods.” *Comm. Appl. Math. Comp. Sci.*, 2(1):83–115, 2007.
- [103] Mavriplis, D. J. “Results from the 3rd Drag Prediction Workshop using the NSU3D unstructured mesh solver.” AIAA 2007-256, 2007.
- [104] Michal, T. and Krakos, J. “Anisotropic mesh adaptation through edge primitive operations.” AIAA 2012-159, 2012.
- [105] Modisette, J. M. *An Automated Reliable Method for Two-Dimensional Reynolds-averaged Navier-Stokes Simulations*. PhD thesis, Massachusetts Institute of Technology, Department of Aeronautics and Astronautics, September 2011.
- [106] Najm, H. N. “Uncertainty quantification and polynomial chaos techniques in computational fluid dynamics.” *Annu. Rev. Fluid Mech.*, 41:35–52, 2009.
- [107] Nemec, M. and Aftosmis, M. J. “Adjoint error estimation and adaptive refinement for embedded-boundary Cartesian meshes.” AIAA 2007-4187, 2007.
- [108] Nochetto, R. H., Veeger, A., and Verani, M. “A safeguarded dual weighted residual method.” *IMA J. Numer. Anal.*, 29:126–140, 2009.
- [109] Oliver, T. and Darmofal, D. “Impact of turbulence model irregularity on high-order discretizations.” AIAA 2009-953, 2009.
- [110] Oliver, T. A. *A Higher-Order, Adaptive, Discontinuous Galerkin Finite Element Method for the Reynolds-averaged Navier-Stokes Equations*. PhD thesis, Massachusetts Institute of Technology, Department of Aeronautics and Astronautics, June 2008.
- [111] Ollivier-Gooch, C. and Altena, M. V. “A high-order-accurate unstructured mesh finite-volume scheme for the advection-diffusion equation.” *J. Comput. Phys.*, 181(2): 729–752, 2002.
- [112] Pagnutti, D. and Ollivier-Gooch, C. “A generalized framework for high order anisotropic mesh adaptation.” *Computers and Structures*, 87(11-12):670 – 679, 2009.
- [113] Pain, C. C., Umpleby, A. P., d.Oliveira, C. R. E., and Goddard, A. J. H. “Tetrahedral mesh optimisation and adaptivity for steady-state and transient finite element calculations.” *Comput. Methods Appl. Mech. Engrg.*, 190:3771–3796, 2001.
- [114] Park, M. A. *Anisotropic Output-Based Adaptation with Tetrahedral Cut Cells for Compressible Flows*. PhD thesis, Massachusetts Institute of Technology, Department of Aeronautics and Astronautics, 2008.



- [115] Patera, A. T. “A Spectral Element Method for fluid dynamics: laminar flow in a channel expansion.” *J. Comput. Phys.*, 54:468–488, 1984.
- [116] Patera, A. T. and Peraire, J. “A General Lagrangian Formulation for the Computation of *A-Posteriori* Finite Element Bounds.” In Barth, T. J. and Deconinck, H., editors, *Error Estimation and Adaptive Discretization Methods in Computational Fluid Dynamics*, pages 159–206. Springer-Verlag, 2002.
- [117] Pennec, X., Fillard, P., and Ayache, N. “A Riemannian framework for tensor computing.” *Int. J. Comput. Vision*, 66(1):41–66, 2006.
- [118] Peraire, J., Nguyen, N., and Cockburn, B. “An embedded discontinuous Galerkin method for the compressible Euler and Navier-Stokes equations.” AIAA 2011–3228, 2011.
- [119] Peraire, J., Vahdati, M., Morgan, K., and Zienkiewicz, O. C. “Adaptive remeshing for compressible flow computations.” *J. Comput. Phys.*, 72:449–466, 1987.
- [120] Persson, P.-O. and Peraire, J. “Sub-cell shock capturing for discontinuous Galerkin methods.” AIAA 2006-0112, 2006.
- [121] Persson, P.-O. and Peraire, J. “Newton-GMRES preconditioning for Discontinuous Galerkin discretizations of the Navier-Stokes equations.” *SIAM J. Sci. Comput.*, 30(6):2709–2722, 2008.
- [122] Persson, P.-O. and Peraire, J. “Curved mesh generation and mesh refinement using Lagrangian solid mechanics.” AIAA 2009-0949, 2009.
- [123] Reed, W. H. and Hill, T. R. “Triangular mesh methods for the neutron transport equation.” Report LA-UR-73-479, Los Alamos Scientific Laboratory, 1973.
- [124] Richter, G. R. “An optimal-order error estimate for the discontinuous Galerkin method.” *Math. Comp.*, 50:75–88, 1988.
- [125] Richter, T. “*A posteriori* error estimation and anisotropy detection with the dual-weighted residual method.” *Internat. J. Numer. Methods Fluids*, 62:90–118, 2010.
- [126] Roe, P. L. “Approximate Riemann solvers, parameter vectors, and difference schemes.” *J. Comput. Phys.*, 43(2):357–372, 1981.
- [127] Saad, Y. *Iterative Methods for Sparse Linear Systems*. Society for Industrial and Applied Mathematics, 1996.
- [128] Sauer-Budge, A. M., Bonet, J., Huerta, A., and Peraire, J. “Computing bounds for linear functionals of exact weak solutions to Poisson’s equation.” *SIAM J. Numer. Anal.*, 42(4):1610–1630, 2004.
- [129] Sauer-Budge, A. M. and Peraire, J. “Computing bounds for linear functionals of exact weak solutions to the advection-diffusion-reaction equation.” *SIAM J. Sci. Comput.*, 26(2):636–652, 2004.
- [130] Schwab, C. *p- and hp- Finite Element Methods*. Oxford Science Publications, Great Clarendon Street, Oxford, UK, 1998.



- [131] Shakib, F., Hughes, T. J. R., and Johan, Z. “A new finite element formulation for computational fluid dynamics: X. The compressible Euler and Navier-Stokes equations.” *Comput. Methods Appl. Mesh. Eng.*, 89:141–219, 1991.
- [132] Shewchuk, J. R. “Triangle: Engineering a 2D quality mesh generator and Delaunay triangulator.” In Lin, M. C. and Manocha, D., editors, *Applied Computational Geometry: Towards Geometric Engineering*, pages 203–222. Springer-Verlag, 1996.
- [133] Shewchuk, J. R. “What Is a Good Linear Finite Element? Interpolation, Conditioning, Anisotropy, and Quality Measures.” Report, University of California at Berkeley, 2002.
- [134] Si, H. “TetGen: A Quality Tetrahedral Mesh Generator and Three-Dimensional Delaunay Triangulator.” Weierstrass Institute for Applied Analysis and Stochastics, 2005. <http://tetgen.berlios.de>.
- [135] Sirovich, L. “Turbulence and the dynamics of coherent structures. I - Coherent structures. II - Symmetries and transformations. III - Dynamics and scaling.” *Quarterly of Applied Mathematics*, 45:561–571, October 1987.
- [136] Sod, G. “A survey of several finite difference methods for systems of nonlinear hyperbolic conservation laws.” *J. Comput. Phys.*, 27:1–31, 1978.
- [137] Spalart, P. R. and Allmaras, S. R. “A one-equation turbulence model for aerodynamics flows.” AIAA 1992-0439, January 1992.
- [138] Süli, E. and Houston, P. “Adaptive finite element approximation of hyperbolic problems.” In Barth, T., Griebel, M., Keyes, D. E., Nieminen, R. M., Roose, D., and Schlick, T., editors, *Lecture Notes in Computational Science and Engineering: Error Estimation and Adaptive Discretization Methods in Computational Fluid Dynamics*, volume 25. Springer, Berlin, 2002.
- [139] Sun, H. “Impact of triangle shapes using high-order discretizations and direct mesh adaptation for output error.” Masters thesis, Massachusetts Institute of Technology, Computation for Design and Optimization Program, 2009.
- [140] Tam, A., Ait-Ali-Yahia, D., Robichaud, M. P., Moore, M., Kozel, V., and Habashi, W. G. “Anisotropic mesh adaptation for 3d flows on structured and unstructured grids.” *Comput. Methods Appl. Mech. Engrg.*, 189:1205–1230, 2000.
- [141] Tryoen, J., Maître, O. L., Ndjinga, M., and Ern, A. “Intrusive Galerkin methods with upwinding for uncertain nonlinear hyperbolic systems.” *J. Comput. Phys.*, 229: 6485–6511, 2010.
- [142] van der Vegt, J. J. W. and van der Ven, H. “Space-time discontinuous Galerkin finite element method with dynamic grid motion for inviscid compressible flows.” *J. Comput. Phys.*, 182:546–585, 2002.
- [143] van Leer, B. and Nomura, S. “Discontinuous Galerkin for diffusion.” AIAA 2005-5108, 2005.



- [144] Vassberg, J. C., DeHaan, M. A., and Sclafani, T. J. “Grid generation requirements for accurate drag predictions based on OVERFLOW calculations.” AIAA 2003-4124, 2003.
- [145] Venditti, D. A. *Grid Adaptation for Functional Outputs of Compressible Flow Simulations*. PhD thesis, Massachusetts Institute of Technology, Cambridge, Massachusetts, 2002.
- [146] Venditti, D. A. and Darmofal, D. L. “Adjoint error estimation and grid adaptation for functional outputs: Application to quasi-one-dimensional flow.” *J. Comput. Phys.*, 164(1):204–227, 2000.
- [147] Venditti, D. A. and Darmofal, D. L. “Grid adaptation for functional outputs: application to two-dimensional inviscid flows.” *J. Comput. Phys.*, 176(1):40–69, 2002.
- [148] Venditti, D. A. and Darmofal, D. L. “Anisotropic grid adaptation for functional outputs: Application to two-dimensional viscous flows.” *J. Comput. Phys.*, 187(1): 22–46, 2003.
- [149] Veroy, K., Prud’homme, C., Rovas, D. V., and Patera, A. T. “*A posteriori* error bounds for reduced-basis approximation of parametrized noncoercive and nonlinear elliptic partial differential equations.” AIAA 2003–3847, 2003.
- [150] Wang, L. and Mavriplis, D. J. “Implicit solution of the unsteady Euler equations for high-order accurate discontinuous Galerkin discretizations.” AIAA 2006-0109, 2006.
- [151] Wang, Z. J. “Spectral (finite) volume method for conservation laws on unstructured grids. Basic formulation.” *J. Comput. Phys.*, 178:210–251, 2002.
- [152] Wang, Z. J. “1st International Workshop on High-Order CFD Methods.” <http://zjw.public.iastate.edu/hiocfd.html>, 2012.
- [153] Wintzer, M., Nemec, M., and Aftosmis, M. J. “Adjoint-based adaptive mesh refinement for sonic boom prediction.” AIAA 2008-6593, 2008.
- [154] Yano, M. and Darmofal, D. “On dual-weighted residual error estimates for  $p$ -dependent discretizations.” ACDL Report TR-11-1, Massachusetts Institute of Technology, 2011.
- [155] Yano, M. and Darmofal, D. “An optimization framework for anisotropic simplex mesh adaptation: application to aerodynamic flows.” AIAA 2012–0079, January 2012.
- [156] Yano, M., Modisette, J. M., and Darmofal, D. “The importance of mesh adaptation for higher-order discretizations of aerodynamic flows.” AIAA 2011–3852, June 2011.