

TOPOLOGY OPTIMIZATION OF CABLES, CLOAKS, AND EMBEDDED
LATTICES

A Dissertation
Presented to
The Academic Faculty

By

Emily D. Sanders

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Civil and Environmental Engineering

Georgia Institute of Technology

August 2021

© Emily D. Sanders 2021

TOPOLOGY OPTIMIZATION OF CABLES, CLOAKS, AND EMBEDDED LATTICES

Thesis committee:

Dr. Glaucio H. Paulino
School of Civil and Environmental Engineering
Georgia Institute of Technology

Dr. Phanish Suryanarayana
School of Civil and Environmental Engineering
Georgia Institute of Technology

Dr. Miguel A. Aguiló
Simulation Modeling Sciences
Sandia National Laboratories

Dr. H. Jerry Qi
George W. Woodruff School of Mechanical Engineering
Georgia Institute of Technology

Dr. Anderson Pereira
Department of Mechanical Engineering
Pontifical Catholic University of Rio de Janeiro

Dr. Tomás Zegard
Department of Structural and Geotechnical Engineering
Pontifical Catholic University of Chile

Date approved: June 25, 2021

ACKNOWLEDGMENTS

This dissertation was made possible by the guidance, support, and mentorship of many people. Thanks to my advisor, Prof. Glaucio H. Paulino, for seeing my potential early on, introducing me to new ideas and ways of thinking, and supporting intellectual and professional experiences that many students can only dream of. The freedom you provided to pursue research in a totally unstructured way is not something I was prepared for and it certainly did not come naturally for me, but it created an environment that fostered tremendous intellectual growth.

Thanks to the committee members that played an integral role in my doctoral research. I am grateful for the experiences and insights you graciously shared along the way and the friendships we formed. Prof. Phanish Suryanarayana, you provided new perspectives that made me think differently about my work. Prof. H. Jerry Qi, you welcomed collaborative efforts that allowed me to explore new ideas in the materials aspects of additive manufacturing and placed me in professional situations that challenged me beyond traditional graduate school experiences. Dr. Miguel A. Aguiló, you welcomed me to the Sandia environment, treated me as a colleague, and helped me mature as a researcher through patient mentorship and collaboration. Prof. Anderson Pereira, you promoted informal and frequent discussions that contributed tremendously to my knowledge in the field, developed confidence in my own abilities, and stimulated a great friendship. Prof. Tomás Zegard, you never hesitated to share your intellectual property for everyone's benefit and served as direct inspiration for many aspects of my research, including how to communicate it effectively. Thanks also to collaborator, Prof. Adeildo S. Ramos Jr., for sharing your knowledge, inspiring my research, and becoming a great friend.

Thanks to current and former research group members and visitors, Silvia R. M. Almeida, Devyani Choudhary, Heng Chi, Habeun Choi, Junho Chun, Americo Cunha, Yang Jiang, Madelyn Kosednar, Weichen Li, Ke (Chris) Liu, Javier Vila Morán, Cicero Ribeiro, Kyoungsoo Park, Pradeep Pratapa, Yipin Si, Daniel Spring, Alex Tong, Erol Unal, Peng Wei, Yeming (Paula) Xian, Ying Yu, and Xiaojia Shelly Zhang, for your support and friendship through our shared experiences. Special thanks to those that inspired my trajectory early on, Evgueni Filipov, Sofie Leon, and Tomás Zegard, and those that I worked with closely in research and related projects, Emily Alcazar, Oliver Giraldo-Londoño, Larissa S. Novelino,

Fernando V. Senhora, and Tuo Zhao. Thanks to the support staff at Georgia Tech, especially Andy Udell, Victoria Norvell, Jeremy Stephens, Jamia Lockett, Carol Maddox, Melisa Hubbs, Robert Simon, and Tony Gallego. Our work is not possible without you. Thanks also to the sponsors of my research: Sandia National Laboratories, the National Science Foundation (CMMI grants #1559594 and #1663244; SHAP3D I/UCRC grant IIP-1822141), and the Raymond Allen Jones Chair at Georgia Tech.

Thanks to my parents, Heinz and Andrea Daniels, for never questioning and always supporting me. Thanks to my husband, Luke Sanders, for your constant encouragement, flexibility, and patience.

TABLE OF CONTENTS

Acknowledgments	iii
List of Tables	xi
List of Figures	xiii
Acronyms	xxiv
Summary	xxv
Chapter 1: Introduction and Background	1
1.1 Topology optimization	2
1.2 Additive manufacturing as an enabler of topology optimization	3
Chapter 2: Topology optimization of tension-only cable nets under finite deformations	6
2.1 Overview	6
2.2 Related work	8
2.3 Large deformation kinematics	10
2.3.1 Linear strain measure	12
2.3.2 Logarithmic strain measure	12
2.3.3 Green-Lagrange strain measure	13
2.4 Cable constitutive model	13

2.5	Nonlinear equilibrium equations	15
2.5.1	Linearization	17
2.5.2	Tangent stiffness matrix	17
2.6	Optimization formulation	19
2.6.1	Sensitivity analysis	19
2.6.2	Convexity	20
2.6.3	Optimality conditions	21
2.7	Details of the numerical implementation	22
2.7.1	Damped Newton with line search	22
2.7.2	Design variable update	23
2.7.3	Convergence criterion	24
2.7.4	Reduced order model	24
2.7.5	End filter	25
2.8	Cable net design examples	26
2.8.1	Clamped beam with equal and opposite point loads	27
2.8.2	Pin-supported beam with mid-span load	29
2.8.3	Tangentially loaded donut	31
2.8.4	Spider web inspired cable net	32
Chapter 3: Optimized lattice-based metamaterials for elastostatic cloaking		39
3.1	Overview and related work	39
3.2	Problem setting	41
3.2.1	Target problem	42
3.2.2	Defining the design space	43
3.2.3	Defining the boundary conditions for optimization	46

3.3	Optimization formulation	48
3.3.1	Sensitivity analysis	49
3.3.2	Convexity	51
3.4	Details of the numerical implementation	51
3.4.1	Low stiffness elements and self-equilibrated loading	51
3.4.2	Design variable update	52
3.4.3	Discrete filter	52
3.4.4	Convergence criterion	53
3.5	Elastostatic cloak design studies	53
3.5.1	Comparison of single- and multi-load-direction designs	56
3.5.2	Effect of load directions used in design	62
	Chapter 4: Multi-material topology optimization considering porous, anisotropic materials	66
4.1	Overview	66
4.2	Related work	68
4.3	Problem setting and optimization formulation	70
4.3.1	Continuous sizing problem	71
4.3.2	Discrete problem	73
4.3.3	Material interpolation	74
4.3.4	Volume interpolation	76
4.3.5	Sensitivity analysis	76
4.3.6	Convexity	79
4.4	Details of the numerical implementation	80
4.4.1	Low density regions	80

4.4.2	Design variable update	82
4.4.3	Convergence criteria	83
4.4.4	Handling mixing at material interfaces	83
4.5	3D multi-lattice design examples	87
4.5.1	Lattice-based material properties	87
4.5.2	Cantilever beam - effect of varying anisotropic mechanical properties	88
4.5.3	Cantilever beam - effect of initial guess and continuation scheme	91
Chapter 5: An educational implementation of multi-material topology optimization (PolyMat and PolyAnisoMat)		93
5.1	Overview	93
5.2	Implementation details	94
5.2.1	Input data and PolyScript	94
5.2.2	Constraint specification	96
5.2.3	Passive region specification	98
5.2.4	Initial guess	100
5.2.5	Interpolation functions	100
5.2.6	Analysis functions	101
5.2.7	Sensitivity analysis	102
5.2.8	Design variable update	102
5.2.9	Simplifications for the solid, isotropic case in PolyMat	103
5.3	Efficiency of the 2D implementations	104
5.4	2D multi-material design examples	107
5.4.1	Michell cantilever (PolyMat) - Volume constraint specification	107
5.4.2	Curved beam (PolyMat) - Passive regions	109

5.4.3	Flower and serpentine (PolyMat) - Local volume constraints	111
5.4.4	MBB beam (PolyAnisoMat) - Porous, anisotropic material specification	112
Chapter 6: Optimal and continuous multi-lattice embedding		117
6.1	Overview	117
6.2	Related work	118
6.3	Stereolithography 3D printing	120
6.4	Manufacturing multi-lattice topology optimized parts	122
6.4.1	Assumptions and limitations	122
6.4.2	Interpreting the multi-material topology optimization data	125
6.4.3	Microstructural-material transitions	126
6.4.4	Communicating the microstructure-embedded geometry to the 3D printer	127
6.4.5	Manufactured parts	130
6.5	Effect of transition region length scale	132
6.6	Scaling to larger build volumes	134
6.6.1	Canopy	136
6.6.2	Eiffel Tower-inspired structure	140
Chapter 7: Conclusions and future work		145
Appendices		151
Appendix A: Nonlinear equilibrium solution scheme		152
Appendix B: Design variable update schemes		154
Appendix C: Elastostatic cloaking supplementary information		158
Appendix D: PolyAnisoMat source code		165

Appendix E: PolyMat and PolyAnisoMat library of examples	175
Appendix F: Directional tensile and shear moduli	181
Appendix G: Multi-lattice embedding flowchart	185
References	187

LIST OF TABLES

2.1	Definition of three common strain measures, associated strain energy density functions, and their first and second derivatives with respect to stretch	14
2.2	Optimization input parameters used for the clamped beam	27
2.3	Clamped beam maximum strains and member stresses for varying applied load magnitudes	29
2.4	Optimization input parameters used for the pin-supported beam	29
2.5	Optimization input parameters used for the tangentially loaded donut	32
2.6	Optimization input parameters used for spider web inspired cable net	34
3.1	Lattice parameters for the elastostatic cloak design studies	55
3.2	Number of lattice elements in design region of the initial ground structures and final designs for selected single- and multi-load-case designs (Note: the design region contains 792 lattice elements before ground structure generation)	56
4.1	Optimization input parameters used for the short column (brackets indicate continuation and $R = -1$ indicates no filter).	84
4.2	Optimization input parameters used for the 3D cantilever beam design examples (brackets indicate continuation).	88
5.1	MATLAB variables and their dimensions for the key quantities and derivatives used in <code>PolyAnisoMat</code>	95
5.2	List of fields in the <code>PolyAnisoMat</code> input <code>structs</code> . The fields marked with the superscript \dagger , if empty, are populated inside <code>PolyAnisoMat</code> . The fields marked with the superscript $\dagger\dagger$, are new or modified relative to the single-material implementation in <code>PolyTop</code>	97

5.3	Code runtime breakdown comparison of <code>PolyTop</code> (1 material, 1 constraint), <code>PolyMat</code> (1 material, 1 constraint), <code>PolyMat</code> (5 materials, 2 constraints), <code>PolyAnisoMat</code> (1 material, 1 constraint), and <code>PolyAnisoMat</code> (5 materials, 2 constraints) for the MBB beam design run for 200 iterations (times are in seconds with percentage of total runtime of <code>PolyScript</code> in parentheses) . . .	106
5.4	Optimization input parameters used for 2D multi-material design examples (brackets indicate continuation and $R = -1$ indicates no filter).	107
5.5	Optimization input parameters used for the serpentine problem (brackets indicate continuation and $R = -1$ indicates no filter).	112
6.1	Bounding box dimensions of computer and physical models for each cantilever design (cm)	132
6.2	Optimization input parameters used for the hyperbolic paraboloid canopy and Eiffel tower-inspired structure design examples (brackets indicate continuation).136	
C.1	Lattice parameters used for direct lattice transformation benchmark	159
E.1	Examples provided with <code>PolyMat</code>	176

LIST OF FIGURES

1.1	Topology optimization via (a) the ground structure method for design of structures and components composed of discrete elements and (b) the density-based method for design of continuum structures and components. The top, middle, and bottom rows show the design domain, discretized design spaces, and optimized designs, respectively. The discrete result is obtained using the authors own implementation, but the plotting is done using routines from the educational MATLAB code, GRAND [15]. The continuum result is obtained using the educational MATLAB codes, PolyMesher [16] and PolyTop [17].	4
2.1	Examples of structures with tension-only members: (a) The Millau Viaduct is a cable-stayed bridge in France [image by AAltruibe from Pixabay]; (b) bicycles use lightweight, tension-only elements as pretensioned spokes [image from bikedirect.com]; (c) spider webs have high strength-to-weight ratio [image by Albrecht Fietz from Pixabay].	7
2.2	Hyperelastic constitutive model for cable members: (a) Strain energy density function and (b) corresponding axial stress-strain curve for a linear strain measure.	8
2.3	Illustration of the need for large displacement kinematics with use of the cable material model: (a) Domain, ground structure, and boundary conditions; (b) unbounded solution for the cable material model with small displacement kinematics; (c) undeformed and (d) deformed solution for the cable material model with large displacement kinematics in which equilibrium is achieved in the deformed configuration.	9
2.4	Kinematics for 3D cable member i	11
2.5	Axial Cauchy stress, σ_i , as a function of axial stretch, λ_i , for three different strain measures.	15
2.6	Strain energy density function versus (a) axial strain, ε_i and (b) axial stretch, λ_i , for three common strain measures.	22

2.7	Clamped beam with equal and opposite compressive point loads: (a) Domain, ground structure, and boundary conditions; (b) optimal design considering small deformation kinematics and linear material model (aligned nodes are removed in post-processing); undeformed (left) and deformed (right) topologies considering large deformation kinematics and the cable material model, with (c) $P = 10$ kN, (d) $P = 1000$ kN, and (e) $P = 2000$ kN.	28
2.8	Pin-supported beam with mid-span point load: (a) Domain, ground structure, and boundary conditions; (b) optimal design considering small deformation kinematics and linear material model (aligned nodes are removed in post-processing); undeformed (top) and deformed (bottom) topologies considering large deformation kinematics and the cable material model, with (c) $A_i^{\max} = 1.444 \times 10^{-3}$ m ² for all i and (d) $A_i^{\max} = 1.155 \times 10^{-4}$ m ² for all i	30
2.9	Convergence of the objective function for the pin-supported beam considering large deformation kinematics and the cable material model for the two different design variable upper bounds considered.	31
2.10	Centrally supported, tangentially loaded donut: (a) Domain, boundary conditions, and mesh used to generate the level four ground structure; (b) optimal design considering small deformation kinematics and linear material model (nodes omitted for clarity); and (c) undeformed (left) and deformed (right) topologies considering large deformation kinematics and the cable material model.	33
2.11	Convergence of the objective function for the tangentially loaded donut considering large deformation kinematics and the cable material model.	34
2.12	Domain, boundary conditions, base mesh, and initial ground structure used for the spider web inspired cable net.	35
2.13	Spider web inspired cable net: optimal design considering large deformation kinematics, the cable material model, and the unperturbed nodal mesh with (a) $A_i^{\max} = 0.005$ for all i and (b) $A_i^{\max} = 0.0012$ for all i	37
2.14	Comparison of (a) an orb spider web found in nature [image by Miriam Müller from Pixabay] and the spider web inspired cable nets designed using topology optimization on a (b) unperturbed nodal mesh and (c) perturbed nodal mesh.	38
3.1	Elastostatic cloaking in 2D lattices: (a) Reference lattice; (b) lattice with a prescribed circular hole surrounded by a region in which a cloak should be designed; (c) lattice in which the cloak geometry is defined by a coordinate transformation of the reference lattice nodes.	40

3.2	Lattice model: The lattice elements are modeled as Euler-Bernoulli beam elements that also take axial force. The length, height, and thickness of element ℓ are L_ℓ , h , and t , respectively. In the reference lattice, all elements have the same length, but the length of the lattice elements varies in the design region of the design lattice.	42
3.3	Target problem definition: (a) Reference lattice; (b) design lattice.	42
3.4	Defining the design space for optimization-based elastostatic cloak design: The reference lattice's nodal positions in (a) are modified in (b) to avoid a hole using a coordinate transformation that preserves the reference lattice outside of the design region. Subsequently, in (c), the design space is enriched by defining a highly redundant ground structure within the design region. . .	44
3.5	Illustration of the coordinate transformation used to map the reference lattice's nodal positions into the design region in order to avoid the elliptical hole.	45
3.6	Two ground structures considered in optimization-based elastostatic cloak design: (a) GS1 is constructed to achieve the highest connectivity possible, while avoiding crossing members and preserving lattice symmetry with respect to the reference lattice nodal positions; (b) GS2 contains level 1 connectivity in which an element is generated between all nodes of each hexagon of the lattice, but not from one hexagon to the next. Connectivity of each ground structure with respect to the reference lattice nodal positions is shown in the second row. Hexagons in the insets are colored to help visualize the additional connectivity added with ground structure generation.	46
3.7	Design lattice embedded in extended lattice for (a) $\theta = 10^\circ$ load angle; (b) $\theta = 60^\circ$ load angle. The subset of lattice elements in the surrounding region, \mathcal{S} , depends on the load angle, but those in the design region, \mathcal{C} , are invariant with respect to the load angle.	47
3.8	Design lattices considering the elliptical hole ($k_x = 1.25$) for all load directions considered: In all cases, the solid green lines indicate loaded edges and the green dashed line indicates the direction of uniaxial tension used in design. Shaded cases correspond to loading aligned along a reference lattice line of symmetry. Un-shaded cases correspond to loading aligned away from a reference lattice line of symmetry and in these cases, two pinned supports are included.	54
3.9	Optimization-based cloak designs based on ground structure GS1 for tensile loads oriented at (a) $\theta = 0^\circ$ and (b) $\theta = 30^\circ$, as indicated with green dashed lines. The top row is the design for a circular hole ($k_x = 1$) and the bottom row is the design for an elliptical hole ($k_x = 1.25$). Hexagons in the insets are colored to help visualize the additional connectivity added with ground structure generation.	57

3.10	Optimization-based cloak designs based on ground structures (a) GS1 and (b) GS2 for tensile loads oriented at $\theta_{\text{sym}} = [0^\circ, 30^\circ, 60^\circ, 90^\circ, 120^\circ, 150^\circ]$, as indicated with green dashed lines. The top row is the design for a circular hole ($k_x = 1$) and the bottom row is the design for an elliptical hole ($k_x = 1.25$). Hexagons in the insets are colored to help visualize the additional connectivity added with ground structure generation.	58
3.11	Cloaking metric, Δ , versus load angle, θ , for the lattice with no cloak, DLT cloak, and optimization-based cloaks: In (a) and (b), the cloaks are evaluated under tensile loading for a circular ($k_x = 1$) and an elliptical ($k_x = 1.25$) hole, respectively; in (c) and (d), the cloaks are evaluated under shear loading for a circular ($k_x = 1$) and an elliptical ($k_x = 1.25$) hole, respectively. The optimized cloaks were designed considering tensile loads oriented at $\theta_{\text{sym}} = [0^\circ, 30^\circ, 60^\circ, 90^\circ, 120^\circ, 150^\circ]$	59
3.12	Normalized displacement fields in the x' direction for tensile loading: The left, middle, and right columns show, respectively, the displacement fields of the reference lattice, the lattice with no cloak, and the lattice with GS1 optimized cloak designed for an elliptical hole ($k_x = 1.25$) and tensile loads oriented at $\theta_{\text{sym}} = [0^\circ, 30^\circ, 60^\circ, 90^\circ, 120^\circ, 150^\circ]$	60
3.13	Normalized displacement fields in the y' direction for shear loading: The left, middle, and right columns show, respectively, the displacement fields of the reference lattice, the lattice with no cloak, and the lattice with GS1 optimized cloak designed for an elliptical hole ($k_x = 1.25$) and tensile loads oriented at $\theta_{\text{sym}} = [0^\circ, 30^\circ, 60^\circ, 90^\circ, 120^\circ, 150^\circ]$	61
3.14	Cloaking metric, Δ , versus load angle, θ , for the DLT cloak and optimization-based cloaks (GS1 and GS2) designed for a circular hole ($k_x = 1$): In (a-c), the cloaks are evaluated in tension; in (d-e), the cloaks are evaluated in shear. In the left, middle, and right columns, Δ is reported for the cloaks designed considering the six tensile load cases aligned along the reference lattice's lines of symmetry (θ_{sym}), the six tensile load cases aligned away from the reference lattice's lines of symmetry (θ_{nosym}), and all twelve tensile load cases (θ_{all}), respectively. Solid and dashed lines indicate Δ evaluated for load directions aligned, respectively, along (i.e., $\theta_{\text{sym}} = [0^\circ, 30^\circ, 60^\circ, 90^\circ, 120^\circ, 150^\circ]$) and away from (i.e., $\theta_{\text{nosym}} = [10^\circ, 45^\circ, 80^\circ, 100^\circ, 135^\circ, 170^\circ]$), the reference lattice lines of symmetry.	64

3.15	Cloaking metric, Δ , versus load angle, θ , for the DLT cloak and optimization-based cloaks (GS1 and GS2) designed for an elliptical hole ($k_x = 1.25$): In (a-c), the cloaks are evaluated in tension; in (d-e), the cloaks are evaluated in shear. In the left, middle, and right columns, Δ is reported for the cloaks designed considering the six tensile load cases aligned along the reference lattice's lines of symmetry (θ_{sym}), the six tensile load cases aligned away from the reference lattice's lines of symmetry (θ_{nosym}), and all twelve tensile load cases (θ_{all}), respectively. Solid and dashed lines indicate Δ evaluated for load directions aligned, respectively, along (i.e., $\theta_{\text{sym}} = [0^\circ, 30^\circ, 60^\circ, 90^\circ, 120^\circ, 150^\circ]$) and away from (i.e., $\theta_{\text{nosym}} = [10^\circ, 45^\circ, 80^\circ, 100^\circ, 135^\circ, 170^\circ]$), the reference lattice lines of symmetry.	65
4.1	Density-based, multi-material topology optimization problem setup: (a) Domain and boundary conditions; (b) design variable discretization considering m candidate materials.	67
4.2	Multi-material problem setup: Extended design domain, boundary conditions, and illustration of material partitions making up the optimal shape. . .	71
4.3	Material interpolation functions for two candidate materials ($E_1^0 = 0.5$, $E_2^0 = 1$) in element ℓ with varying choice of penalty parameters, p and γ : (a) For $p = 1$ and $\gamma = 0$, neither intermediate densities nor material mixing are penalized; (b) for $p = 4$ and $\gamma = 0$, the curvature penalizes intermediate densities, but mixing at full densities is likely promoted; (c) for $p = 1$ and $\gamma = 1$, material mixing is penalized at full densities, but not necessarily at intermediate densities; (d) for $p = 4$ and $\gamma = 1$, both intermediate densities and material mixing are penalized.	77
4.4	Material interpolation functions for two candidate materials ($E_1^0 = 0.5$, $E_2^0 = 1$) in element ℓ with continuation (from left to right) on penalty parameters, p and γ : The optimization problem can be initiated as the convex problem at the left, where both intermediate densities and material mixing are allowed, and gradually transformed into the non-convex problem at the right where both intermediate densities and material mixing are penalized.	80
4.5	Short column: (a) Domain, boundary conditions, and constraint specification, where the blue and orange materials have $E_1^0 = 1$ and $E_2^0 = 0.15$, respectively and are controlled by g_1 ($\bar{v}_1 = 0.4$) and g_2 ($\bar{v}_2 = 0.4$), respectively; (b) finite element mesh.	84
4.6	Demonstration of two ways to handle mixing at material interfaces using the short column example: The first row shows the converged solutions after the first continuation step for $R = 0.1$, $R = 0.03$, and $R = -1$ (no filter); the second row shows the converged solutions after the second continuation step in which the filter is turned off; the third and fourth rows show the result of applying post-processing step 1 and post-processing steps 1 and 2, respectively, on the converged solution from the first continuation step.	85

4.7	Effect of porous, anisotropic microstructural-materials: (a) Design considering a single, solid, isotropic material with objective function value, f_0 ; (b-g) designs considering different subsets, indicated to their right, of seven porous, anisotropic microstructural-materials. A single volume constraint is specified in each case such that the total volume of bulk material occupies, at most, a domain volume fraction of $\bar{v} = 0.022$ (the volume constraint is active in all cases and all designs in (a-g) have the same volume of bulk material). Normalized directional tensile and shear moduli (based on homogenized properties) are provided in (h) for the solid, isotropic material and in (i) for the seven porous, anisotropic microstructural-materials. Variable \hat{d}_i^{bar} is bar diameter and f/f_0 is the objective function normalized to that of the structure in (a).	90
4.8	Results considering different initial guesses for the example in Figure 4.7f, without and with continuation on the material interpolation parameters, p and γ : In the top row (a-e), the material interpolation parameters are constant ($p = 3, \gamma = 1$) and the maximum number of iterations is set to 600. In the second row (f - j), the continuation scheme defined in Table 4.2 is used. In (a) and (f), a uniform initial guess is used. For each case in (b - e) and (g - j), one microstructural-material dominates the initial guess with its densities equal to $0.85\bar{v}$ and all other microstructural-material densities equal to $0.05\bar{v}$. The schematic at the bottom indicates the value of the design variables associated with each candidate material at the initial guess. Variable f_0 refers to the objective function value of the solid, isotropic structure in Figure 4.7a.	92
5.1	Constraint and passive region specification: (a) Curved beam domain and code needed to generate the distance values required to define the constraints and/or passive regions; (b) constraint specification and code needed to assemble <code>ElemInd</code> ; (c) constraint and passive region specification and code needed to assemble <code>ElemInd</code> and <code>SElemInd</code> .	99
5.2	CPU time (seconds) for the MBB beam problem with a 300×100 orthogonal mesh run for 200 iterations: (a) A single global volume constraint controlling a varying number of candidate materials; (b) a single candidate material controlled by a varying number of local volume constraints on sub-regions of the domain.	104
5.3	Michell cantilever with circular support to demonstrate various ways the volume constraints can be specified: (a) In case 1, a single global constraint ($\bar{v} = 0.45$) controls all fifteen candidate materials; (b) in case 2, fifteen global constraints ($\bar{v}_j = 0.45/15, j = 1, \dots, 15$) each control a single candidate material; (c) in case 3, thirty local constraints ($\bar{v}_j = 0.45, j = 1, \dots, 30$) that are symmetric about the horizontal center line each control a single candidate material.	110
5.4	Curved beam problem: (a) Domain, boundary conditions, constraint specification, and passive region specification; (b) Young's modulus of the three candidate materials available; (c) optimized design.	111

5.5	Serpentine problem: (a) Domain and boundary conditions; (b) the single region used to define a single global volume constraint with volume fraction, $\bar{v} = 0.5$, (top) and resulting optimized design (bottom); (c) the 84 sub-regions used to define 84 local volume constraints each with volume fraction, $\bar{v}_j = 0.5, j = 1, \dots, 84$, (top) and resulting optimized design (bottom).	113
5.6	Flower problem: (a) Domain, boundary conditions, and constraint specification for the multi-material design; (b) Young's modulus of the five candidate materials available in the multi-material design; (c) multi-material design; (d) corresponding single-material design.	114
5.7	MBB problem with porous, anisotropic candidate microstructural-materials: (a) Domain, boundary conditions, and material/constraint specification; (b) directional tensile (top) and shear (bottom) modulus plots of the candidate microstructural-materials; (c) design.	116
6.1	Stereolithography 3D printing process: (a) Bottom-up approach with point-wise UV laser radiation; (b) top-down approach with pointwise UV laser radiation; (c) top-down approach with layerwise UV radiation masked by LCD display. Images are adapted from [224].	121
6.2	Two-microstructural-material topology optimized cantilever beams considering three different combinations of two microstructural-materials: In all cases, the bulk materials are limited to a domain volume fraction of $\bar{v}_1 = 0.07\hat{v}_1$ and $\bar{v}_2 = 0.03\hat{v}_2$ for microstructural-materials 1 and 2, respectively. Variable \hat{d}_i^{bar} is the unit cell bar diameter.	123
6.3	Various ways to represent multi-material topology optimization data: Each top image shows a portion of the 3D part and each bottom image shows a portion of one slice for the density data represented (a) directly on the underlying FE (hex) mesh; (b) with separate isosurfaces for each material; on a volume (tet) mesh generated inside an isosurface of the composite density data with (c) abrupt material interfaces and (d) functionally graded material interfaces.	126
6.4	Smooth and continuous microstructure connectivity: (a) Two octahedron unit cells with different bar diameter for which connectivity is always guaranteed and a smooth transition is achieved by interpolating the bar diameter; (b) a simple cubic and a truncated octahedron unit cell for which connectivity is, in general, not guaranteed and a smooth and continuous transition is achieved by interpolating the unit cell geometry; (c) a face-x and center-x unit cell for which connectivity is only guaranteed at the unit cell boundaries and a smooth and continuous transition is achieved by using hybrid unit cells composed of the two basic unit cells. Variable \hat{d}_i^{bar} is the unit cell bar diameter.	128

6.5	Line plots showing how the normalized tensile and shear moduli, E_{11}/E^0 and G_{12}/G^0 (in the reference frame and based on homogenized properties), vary over the transitional unit cells between (a) two octahedron unit cells with different bar diameter; (b) a simple cubic and a truncated octahedron unit cell; and (c) a face-x and center-x unit cell.	129
6.6	Multi-microstructure-embedding: In (a) and (b), a portion of one macro-slice and one embedded slice is shown for the abrupt-transition and functionally graded tet meshes, respectively; in (c), the simple cubic, truncated octahedron, and 6 of the 23 transitional unit cells are shown with a portion of the bottom slice of their tessellations (micro-slices).	130
6.7	Continuous multi-microstructure-embedding for two-microstructural-material cantilevers: The cantilevers are composed of (a) two octahedron unit cells with different bar diameter and (b) a face-x and a center-x unit cell. The transitional unit cells making up the macro-to-micro mapping in (a) are obtained by interpolating the bar diameter. Those in (b) are obtained by composing the two unit cells into a set of hybrid unit cells, where the face-x unit cell gradually disappears from one end and the center-x unit cell gradually disappears from the other (with minimum bar diameter limited to 0.065 of the unit cell edge length for manufacturability). In (a), 6 of the 12 transitional unit cells are shown and in (b), 6 of the 11 transitional unit cells are shown.	131
6.8	Cantilever beam designed and manufactured with two microstructural-materials composed of octahedron unit cells with two different bar diameters: The manufactured beam is 14.5 cm tall, the embedded microstructures have unit cells with edge length scaled to 1.5 mm, and the higher and lower volume unit cells have (expected) bar diameter 360 μm and 195 μm , respectively. Scale bars are 1.5 mm.	132
6.9	Cantilever beam designed and manufactured with two microstructural-materials composed of simple cubic and truncated octahedron unit cells: The manufactured beam is 14.5 cm tall, the embedded microstructures have unit cells with edge length scaled to 1.5 mm, and the bars of both unit cell types have (expected) bar diameter 360 μm . Scale bars are 1.5 mm.	133
6.10	Cantilever beam designed and manufactured with two microstructural-materials composed of face-x and center-x unit cells: The manufactured beam is 14.5 cm tall, the embedded microstructures have unit cells with edge length scaled to 1.5 mm, and the bars of both unit cell types have (expected) diameter 300 μm . Scale bars are 1.5 mm.	133

6.11	Illustration of how the transition region length scale, microstructure connectivity, and objective function value are affected by the filter radius, R , used in functional grading: The figure shows a closeup of a portion of the functionally graded tet mesh and associated macro-slices and embedded-slices for $R = 0$ (abrupt interfaces), $R = 0.010$, $R = 0.025$, $R = 0.050$, $R = 0.100$, and $R = 0.200$, where the magnitude of R is with respect to the domain dimensions used during design (see Figure 6.2). The objective function values, f/f_0 , which are normalized to that of the case with $R = 0$, indicate that the length scale of the transition regions does not significantly affect the global elastic properties for which the structure is designed.	135
6.12	Problem description for hyperbolic paraboloid canopy and Eiffel tower-inspired structures: (a) Candidate microstructural materials and associated normalized, directional tensile and shear moduli plots; (b) canopy domain and boundary conditions; (c) Eiffel tower domain and boundary conditions.	137
6.13	Canopy structure: Topology optimized design (top) and manufactured part (bottom) with a (printed) height of 14.4 cm and an (expected) unit cell edge length of 2 mm.	139
6.14	Canopy structure transition regions: (a) Octet unit cells with 300 μm bar diameter to face-x unit cells with 400 μm bar diameter; (b) truncated octahedron unit cells with 400 μm bar diameter to face-x unit cells with 400 μm bar diameter; (c) truncated octahedron unit cells with 400 μm bar diameter to octet unit cells with 300 μm bar diameter; (d) solid to truncated octahedron unit cells with 400 μm bar diameter. Scale bars are 2 mm.	140
6.15	Manufacturing details for canopy structure: (a) Support structures required for printing the canopy structure; (b) one half of the canopy structure before removing the support structure or removing it from the build plate; (c) canopy structure after removing the support and before gluing the pieces together.	141
6.16	Eiffel Tower-inspired structure: Topology optimized design (left) and manufactured part (right). The assembled (printed) structure is 26 cm tall.	142
6.17	Eiffel Tower-inspired structure before assembly.	143
6.18	Additional details of the Eiffel Tower-inspired structure: (a) Cross-section through floor 2; (b) cross-section through base.	144

C.1	Direct lattice transformation benchmark (no optimization): The left side shows results taken from Bückmann et al. [98] and the right side shows the results obtained using the Euler-Bernoulli beam elements and the simplified DLT axial stiffness transformation used here. Strain fields in the horizontal (x) and vertical (y) directions are provided for the reference lattice, lattice without any cloak, and lattice with cloak designed by DLT. The colorbar on the right side is stretched to match that used by Bückmann et al. [98] on the left side; that is, all values greater than 20% of the maximum (or less than 20% of the minimum) are mapped to the maximum value (or minimum value) and all other values are scaled by a factor of 5.	160
C.2	Cloaking metric, Δ , versus load angle, θ , for the no cloak, DLT cloak, and optimization-based cloaks: In (a) and (b), the cloaks are evaluated under tensile loading for a circular ($k_x = 1$) and an elliptical ($k_x = 1.25$) hole, respectively; In (c) and (d), the cloaks are evaluated under shear loading for a circular ($k_x = 1$) and an elliptical ($k_x = 1.25$) hole, respectively. The single-load-case, optimized cloaks were designed with symmetry enforced on the design variables such that the six rotational and mirror symmetries (with respect to the reference lattice nodal positions) are preserved.	161
C.3	Convergence of the objective function, f , for optimization-based elastostatic cloak designs based on ground structure GS1 and considering single load directions, $\theta = 0^\circ$, $\theta = 30^\circ$, $\theta = 60^\circ$, and $\theta = 90^\circ$; and multiple load directions, $\theta_{\text{sym}} = [0^\circ, 30^\circ, 60^\circ, 90^\circ, 120^\circ, 150^\circ]$, $\theta_{\text{nosym}} = [10^\circ, 45^\circ, 80^\circ, 100^\circ, 135^\circ, 170^\circ]$, $\theta_{\text{all}} = [0^\circ, 10^\circ, 30^\circ, 45^\circ, 60^\circ, 80^\circ, 90^\circ, 100^\circ, 120^\circ, 135^\circ, 150^\circ, 170^\circ]$ for the (a) circular hole ($k_x = 1$) and (b) elliptical hole ($k_x = 1.25$). In each case, the objective function value is normalized to its value at the initial design.	162
C.4	Convergence of the cloaking metric, Δ , for the four different single-load-case, GS1 cloak designs evaluated in directions aligned along the reference lattice's lines of symmetry. The left and right columns show results for the circular ($k_x = 1$) and elliptical ($k_x = 1.25$) holes, respectively.	163
C.5	Convergence of the cloaking metric, Δ , for the three different multi-load-case, GS1 cloak designs evaluated in all twelve load directions. The left and right columns show results for the circular ($k_x = 1$) and elliptical ($k_x = 1.25$) holes, respectively. Convergence for cloaks designed considering $\theta_{\text{sym}} = [0^\circ, 30^\circ, 60^\circ, 90^\circ, 120^\circ, 150^\circ]$, $\theta_{\text{nosym}} = [10^\circ, 45^\circ, 80^\circ, 100^\circ, 135^\circ, 170^\circ]$, and $\theta_{\text{all}} = [0^\circ, 10^\circ, 30^\circ, 45^\circ, 60^\circ, 80^\circ, 90^\circ, 100^\circ, 120^\circ, 135^\circ, 150^\circ, 170^\circ]$ are provided in the top, middle, and bottom rows, respectively.	164
F.1	Frame rotation conventions: (a) Rotated (prime) frame initially aligned with reference (unprime) frame; (b) rotation about the x'_3 axis by θ_1 ; (c) rotation about the x'_2 axis by θ_2 ; (d) rotation about the x'_1 axis by θ_3	181

F.2 Directional tensile and shear moduli plots: (a) Truncated octahedron unit cell geometry and corresponding (b) directional tensile moduli; (c) directional shear moduli for various θ_3 rotations of the rotated frame; (d) enveloped directional shear moduli for $\theta_3 = 0, \pi/16, \dots, \pi$ 184

G.1 Flowchart summarizing the overall process from design to manufacturing: On the left side, multi-material topology optimization is performed, the multi-material density data is processed, and the macrostructure is sliced. On the right side, the microstructures associated with the candidate microstructural materials used in topology optimization are sliced and tiled. Finally, the micro-slices are embedded into the macro-slices and sent to the 3D printer. 186

ACRONYMS

AAP	Alternating Active Phase
DLP	Digital Light Processing
DLT	Direct Lattice Transformation
DMO	Discrete Material Optimization
GL	Green Lagrange strain measure
LIN	Linear strain measure
LOG	Logarithmic strain measure
OC	Optimality Criteria
SIMP	Solid Isotropic Material with Penalization
SLA	Stereolithography (vat photopolymerization)
SLS	Selective Laser Sintering
STL	Stereolithography file
UV	Ultraviolet
ZPR	Zhang-Paulino-Ramos Jr.

SUMMARY

Materials play a critical role in the behavior and functionality of natural and engineered systems. For example, the use of cast-iron and steel led to dramatically increased bridge spans per material volume with the move from compression-dominant arch bridges to tensile-capable truss, suspension, and cable-stayed bridges; materials underlie many of the major technological advancements in the auto and aerospace industries that have made cars and airplanes increasingly light, strong, and damage tolerant; and the great diversity of biological materials and bio-composites enable complex biological and mechanical functions in nature. Topology optimization is a computational design method that simultaneously enhances efficiency and design freedom of engineered parts, but is often limited to a single, solid, isotropic, linear-elastic material. To understand how the material space can be tailored to enhance design freedom and/or promote desired mechanical behavior, several topology optimization problems are explored in this dissertation in which the space of available materials is either *relaxed* or *restricted*. Specifically, in a discrete topology optimization setting defined by 1D (truss) elements, tension-only systems are targeted by *restricting* the material space to that of a tension-only material and tailoring a formulation to handle the associated nonlinear mechanics. The discrete setting is then enhanced to handle 2D (beam) elements in pursuit of cloaking devices that hide the effect of a hole or defect on the elastostatic response of lattice systems. In this case the material space is *relaxed* to allow for a continuous range of stiffness and the objective is formulated as a weighted least squares function in which the physically-motivated weights promote global stiffness matching between the cloaked and undisturbed systems. Continuous 2D and 3D structures are also explored in a density-based topology optimization setting in which the material space is *relaxed* to accommodate an arbitrary number of candidate materials in a general continuum mechanics framework that can handle material anisotropy. The theoretical and physical relevance of such framework is highlighted via a continuous embedding scheme that enables manufacturing in the *relaxed* (or *restricted*) design space of lattice-based microstructural-materials. Implications of varying the material design space on the mechanics, mathematics, and computations needed for topology optimization are discussed in detail.

CHAPTER 1

INTRODUCTION AND BACKGROUND

Materials play a critical role in the behavior and functionality of natural and engineered systems. For example, the use of cast-iron and steel led to dramatically increased bridge spans per material volume with the move from compression-dominant arch bridges to tensile-capable truss, suspension, and cable-stayed bridges [1]; materials underlie many of the major technological advancements in the auto and aerospace industries that have made cars and airplanes increasingly light, strong, and damage tolerant [2, 3]; and the great diversity of biological materials and bio-composites enable complex biological and mechanical functions in nature [4]. Topology optimization is a computational design method that simultaneously enhances efficiency and design freedom of engineered parts [5], but is often limited to a single, solid, isotropic, linear-elastic material. This dissertation explores how the material space in topology optimization can be *relaxed* or *restricted* to enhance design freedom and/or promote desired mechanical behavior through topology optimization of cables, cloaks, and embedded lattices. The implications of varying the material design space on the mechanics, mathematics, and computations are discussed in detail.

Specifically, in chapter 2, to achieve maximally stiff cable nets, the material space in a discrete topology optimization setting defined by 1D (truss) elements is *restricted* to that of a tension-only material. In this setting, nonlinear mechanics that accounts for both the nonlinear material behavior and finite-deformation kinematics are shown to be essential to arriving at equilibrated optimized solutions. Additionally, the choice of tension-only material model is shown to have implications on convexity of the optimization problem [6]. The discrete setting is then enhanced in chapter 3 to handle 2D (beam) elements in pursuit of cloaking devices that hide the effect of a hole on the elastostatic response of lattice systems. In this case, the material space is *relaxed* by allowing the stiffness (Young's modulus) to vary in a continuous range from one lattice element to the next. A convex objective is formulated as a weighted least squares function that promotes global stiffness matching between the cloaked and undisturbed systems [7]. Continuous 2D and 3D structures are explored in chapter 4 in a continuum topology optimization setting in which the material

space is *relaxed* to accommodate an arbitrary number of materials in a general framework that can handle material anisotropy. Here, a material interpolation function is introduced to limit material mixing, define the structure’s spatially-varying material properties, and reduce the influence of the initial guess on the final design [8–10]. Detailed in chapter 5 is an educational implementation of the multi-material topology optimization formulation [9] and an extension that accommodates porous, anisotropic candidate materials. Finally, in chapter 6, the theoretical and physical relevance of the proposed multi-material framework is highlighted via a continuous embedding scheme that enables manufacturing in the *relaxed* design space of lattice-based microstructural-materials. A strategy for transitioning between different microstructural-materials is proposed along with a voxel-based embedding scheme that promotes scalability and efficient communication with a 3D printer [10].

1.1 Topology optimization

In topology optimization [5], the geometry of an object is pursued so that it efficiently meets a given performance objective, $f(\mathbf{z}, \mathbf{u})$, and satisfies relevant constraints, $h_i(\mathbf{z}, \mathbf{u}) = 0, i = 1, \dots, J$ and $g_j(\mathbf{z}, \mathbf{u}) \leq 0, j = 1, \dots, K$, that enforce desired physical conditions on the system. Design variables, $\mathbf{z} \in [\underline{\rho}, \bar{\rho}]^N$, define the design within the region of interest and the state variables, \mathbf{u} , represent the physical response of the system. In this dissertation, \mathbf{u} is the displacement field that satisfies the governing equations of static elasticity. A nested formulation is considered, in which the state variables, $\mathbf{u}(\mathbf{z})$, are implicitly defined through equality constraints, $h_i = \delta \Pi(\mathbf{z}, \mathbf{u}(\mathbf{z})) = 0$, that enforce the principle of minimum potential energy. With these definitions and assumptions, the canonical topology optimization formulation explored in this dissertation can be stated as¹

$$\begin{aligned}
 & \min_{\mathbf{z} \in [\underline{\rho}, \bar{\rho}]^N} f(\mathbf{z}, \mathbf{u}(\mathbf{z})) \\
 & \text{s.t.} \quad g_j(\mathbf{z}) \leq 0, \quad j = 1, \dots, K \\
 & \text{with} \quad \mathbf{u}(\mathbf{z}) = \underset{\mathbf{u}}{\operatorname{argmin}} \Pi(\mathbf{z}, \mathbf{u}).
 \end{aligned} \tag{1.1}$$

¹ With the exception of the cloaking formulation in chapter 3, the objective function, f , in Equation 1.1, represents a measure of compliance and the constraint functions, $g_j, j = 1, \dots, K$, represent limits on material volume. Although the material limit constraints are always active in the context of the design-independent loading considered here, they are posed as inequality constraints rather than equality constraints such that they are also relevant in the more general case of design-dependent loading (e.g., self-weight [11]) for which they may be inactive.

As illustrated in Figure 1.1, the formulation in Equation 1.1 is applicable to design of both structures composed of discrete elements (e.g., trusses, beams, cables) and continuum structures. For design of systems composed of discrete elements, an elastic formulation of the ground structure method is adopted [12], in which the design variables, $\mathbf{z} \in [0, \bar{\rho}]^N$, represent cross-sectional or material properties of a set of discrete elements defined in a highly redundant network over the design domain (i.e., a ground structure). For design of continuum structures and components, density-based methods are adopted, where the design variables, $\mathbf{z} \in [0, 1]^N$, represent a density field with the density at each material point, $i = 1, \dots, N$, representing void ($z_i = 0$) or solid ($z_i = 1$) material. Material interpolation functions are used to penalize intermediate densities (i.e., those between 0 and 1) and translate the design variables into stiffness properties of the continuum system as discussed in chapter 4 [13, 14].

To solve Equation 1.1, gradient-based algorithms are adopted (see for example, [18]), in which derivatives of the objective and constraint functions, with respect to the design variables, indicate directions in the design space that lead to an improved objective function value without violating the constraints. Because the derivatives are local in nature, small steps toward an optimal solution are taken in an iterative procedure. With gradient-based algorithms, global optimal solutions can only be guaranteed when the objective and constraint functions are convex. Thus, for each problem of interest explored in this dissertation, convexity of the objective and constraint functions are explored and, if needed, methods are proposed to mitigate the effect of non-convexity on achieving a “good” local solution.

1.2 Additive manufacturing as an enabler of topology optimization

The modern field of topology optimization was initiated by the work of Bendsoe and Kikuchi in 1988 [19], just a few years before additive manufacturing technologies began advancing rapidly [20]. In roughly the past 30 years, significant advances in each field have complemented advances in the other. On one hand, advances in additive manufacturing technologies have provided a platform to physically realize the organic and intricate geometries obtained from topology optimization. At the same time, topology optimization has provided a systematic approach for generating optimized geometries that take full advantage of the capabilities of additive manufacturing technologies [21–23].

Nevertheless, several challenges remain before the two fields will be fully integrated to achieve arbitrary freedom in design with sufficient confidence in part performance. For ex-

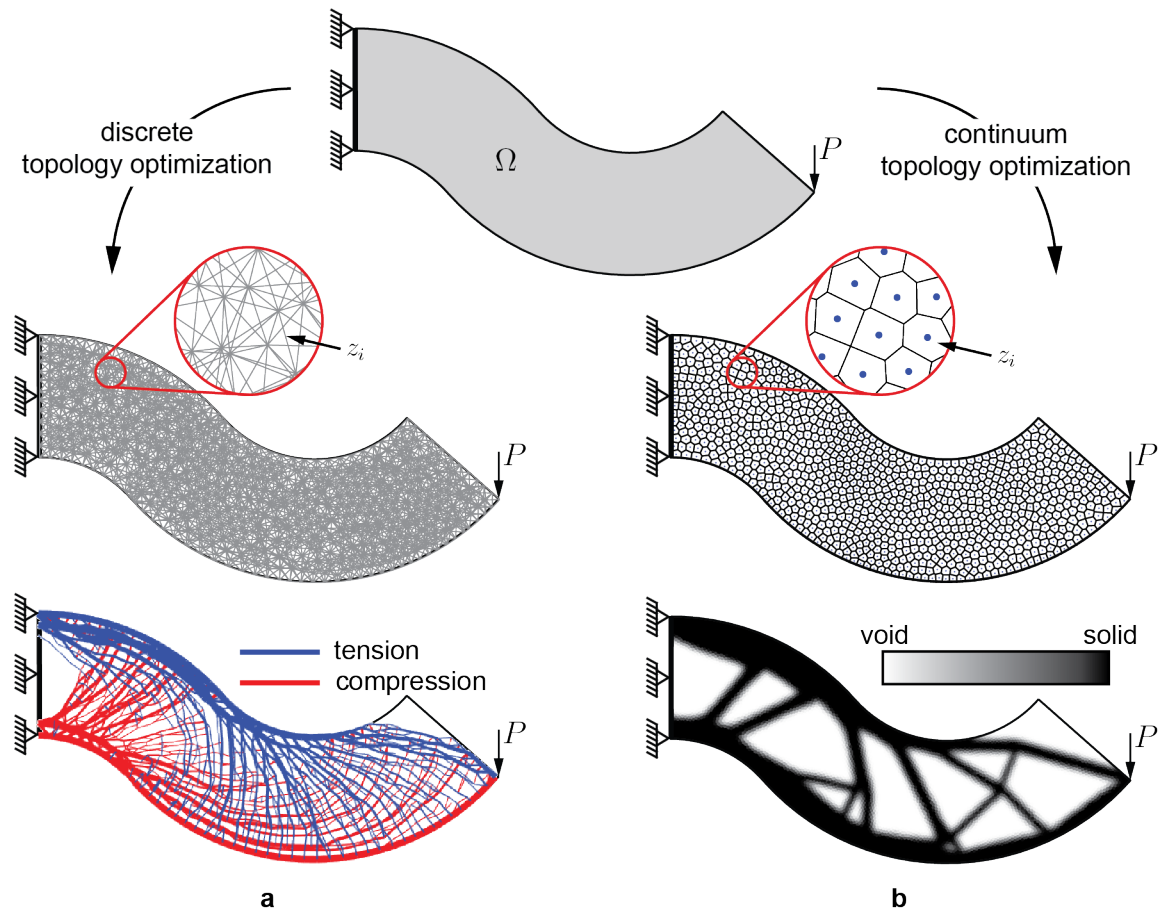


Figure 1.1: Topology optimization via (a) the ground structure method for design of structures and components composed of discrete elements and (b) the density-based method for design of continuum structures and components. The top, middle, and bottom rows show the design domain, discretized design spaces, and optimized designs, respectively. The discrete result is obtained using the authors own implementation, but the plotting is done using routines from the educational MATLAB code, GRAND [15]. The continuum result is obtained using the educational MATLAB codes, PolyMesher [16] and PolyTop [17].

ample, manufacturing constraints (e.g., limited overhang angle or prevention of enclosed voids) often lead to compromised optimality when embedded into the optimization statement [24]. Additionally, a full understanding of how mechanical behavior is influenced by additive manufacturing process parameters, defects, anisotropies, and other uncertainties often inhibits adoption of topology optimized parts for critical industrial applications [22].

Furthermore, advances in one field sometimes outpace complementary advances in the other, as has been the case for multi-material applications. Several multi-material topology optimization formulations have been explored in the literature in recent years [8, 9, 25–30], including the many-material formulation put forth in chapter 4 of this dissertation. Although multi-material additive manufacturing is possible on some systems (e.g., Prusa’s extrusion-based printer with filament switching capabilities, Stratasys’ PolyJet multi-material photopolymer jetting system, Optomec’s Laser Engineered Net Shaping approach to multi-metal parts, multi-vat-photopolymerization techniques [31, 32]), integrity of material interfaces as well as limited and non-continuous ranges of material properties currently impede the usefulness of these multi-material additive manufacturing techniques in practice.

An alternative to multi-material additive manufacturing is to locally control the mechanical properties without changing the base material. For example, in vat-photopolymerization, locally tuned stiffness can be achieved by locally controlling the light intensity during photocure, ultimately leading to spatially-varying cross-link density in the final thermoset polymer [33, 34]. Another approach, explored here, is to blur the lines between material and structure and introduce spatially-varying microarchitectures that can be treated as different materials during design, and when manufactured from a single bulk material, admit spatially-varying properties. The latter approach poses challenges in communicating the highly complex microscale features to the 3D printer in an efficient way and in handling connectivity of the different microarchitectures. Such challenges are considered in chapter 6 of this dissertation.

CHAPTER 2
TOPOLOGY OPTIMIZATION OF TENSION-ONLY CABLE NETS UNDER
FINITE DEFORMATIONS

Frei Otto’s visionary use of tensile components (e.g., cable nets and membranes) pioneered design and construction of lightweight structures and continues to influence minimal design today [35–39]. Modern examples of tensile components in engineered structures include the Millau Viaduct (Figure 2.1a), a cable-stayed bridge in Millau, France that broke span and height records while imposing little interference on the surrounding landscape [40] and bicycle wheels (Figure 2.1b), which use thin, lightweight, pre-tensioned spokes that avoid buckling without the use of bulky and heavy compression struts [41]. Nature also harnesses the structural efficiency of cable networks. For example, spider webs (Figure 2.1c) are tension-only cable nets with an astoundingly high strength-to-weight ratio [42]. In this chapter, optimized cable systems are pursued in a discrete topology optimization setting composed of 1D (truss) elements by *restricting* the material space to that of a tension-only material and formulating a convex topology optimization problem that handles the implications of such restriction on the underlying (nonlinear) mechanics.

2.1 Overview

Inspired by the potential efficiency afforded by tension-only members, maximally stiff cable nets with limited material volume are pursued using topology optimization. To this end, a discrete topology optimization setting is adopted, where Equation 1.1 takes the following specific form

$$\begin{aligned}
 & \min_{\mathbf{A} \in [0, A^{\max}]^N} & f(\mathbf{A}, \mathbf{u}(\mathbf{A})) &= -\Pi_{\min}(\mathbf{A}, \mathbf{u}(\mathbf{A})) \\
 & \text{s.t.} & g(\mathbf{A}) &= \mathbf{L}^T \mathbf{A} - V^{\max} \leq 0 \\
 & \text{with} & \mathbf{u}(\mathbf{A}) &= \underset{\mathbf{u}}{\operatorname{argmin}} \Pi(\mathbf{A}, \mathbf{u}(\mathbf{A})).
 \end{aligned} \tag{2.1}$$

In Equation 2.1, \mathbf{A} is the vector of design variables representing cross-sectional areas of the cable members in the undeformed configuration; \mathbf{u} is the vector of nodal displacements; Π

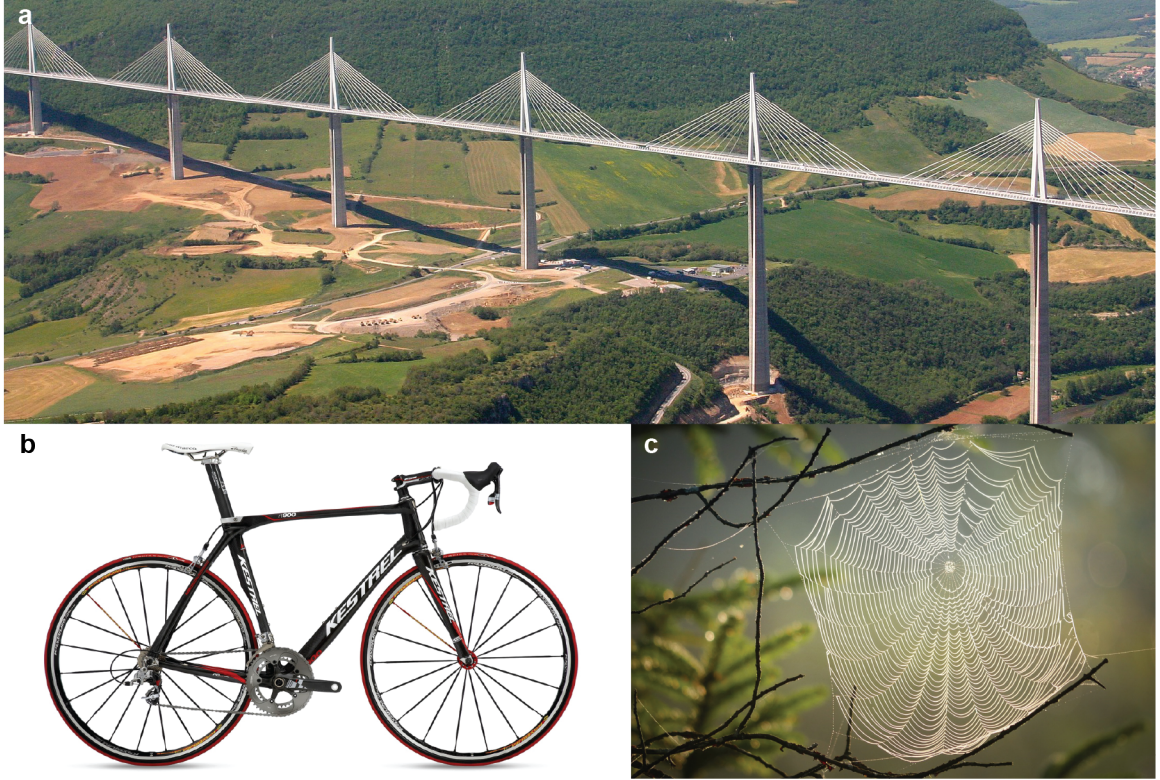


Figure 2.1: Examples of structures with tension-only members: (a) The Millau Viaduct is a cable-stayed bridge in France [image by AAltruibe from Pixabay]; (b) bicycles use lightweight, tension-only elements as pretensioned spokes [image from bikedirect.com]; (c) spider webs have high strength-to-weight ratio [image by Albrecht Fietz from Pixabay].

and Π_{\min} are the total and stationary potential energy of the system, respectively; \mathbf{L} is the vector of cable member lengths in the undeformed configuration; V^{\max} is a limit on the total volume of the cable network in the undeformed configuration; and A^{\max} is the upper bound on each cable member's deformed cross-sectional area. Note that the cable cross-sectional areas are allowed to reduce to zero and the resulting singular system of equilibrium equations is handled using a damped Newton method [43] as described in section 2.7.

The formulation in Equation 2.1 is applicable to structures with both linear and nonlinear elastic material behavior as well as both small and large displacements and deformations. To limit the design to tension-only cable nets, the material space is *restricted* to that with tension-only behavior as illustrated by the constitutive model in Figure 2.2. To motivate the need for finite deformation kinematics in this problem setting, consider design of the simply supported beam in Figure 2.3a, where the ground structure is composed of cable members that can only take tension (e.g., according to the constitutive model in Figure 2.2). Note

that if only small displacements and deformations are considered, no solution can be found (Figure 2.3b) since a tension-only configuration does not exist for the given ground structure and boundary conditions. In contrast, if finite displacements and deformations are included, a tension-only structure that reaches an equilibrium configuration in the deformed shape is found (Figure 2.3c and d). Thus, in this chapter a formulation for topology optimization of tension-only cable nets is put forth, in which the underlying mechanics considers both material and geometric nonlinearities. Implications of the specific choice of strain-energy density function and strain measure on convexity of the optimization problem are discussed in subsection 2.6.2.

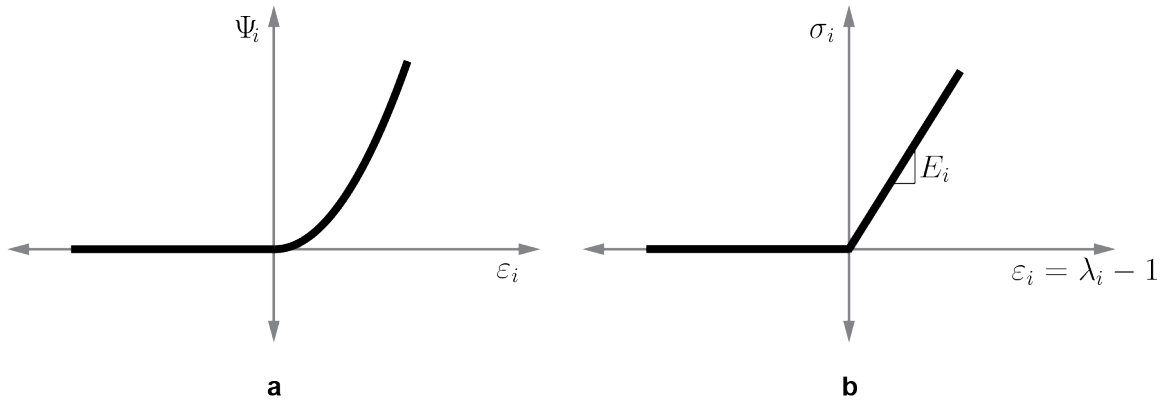


Figure 2.2: Hyperelastic constitutive model for cable members: (a) Strain energy density function and (b) corresponding axial stress-strain curve for a linear strain measure.

2.2 Related work

A number of researchers have explored topology optimization with nonlinear elastic material behavior and finite displacements and/or finite deformations, but most have focused their efforts in the continuum setting [44–60]. The problem of optimizing the layout of discrete cable networks shares some of the same challenges faced in the continuum setting, while others are inherently avoided.

One challenge that arises when nonlinearities are included in the analysis is how to define the objective function to achieve maximum stiffness structures. Minimizing the end-compliance for a fixed load leads to structures that are inefficient for different load magnitudes. In contrast, minimizing the complementary elastic work ensures that the structure can withstand all loads traversed by the load-displacement curve [46]. Other objectives

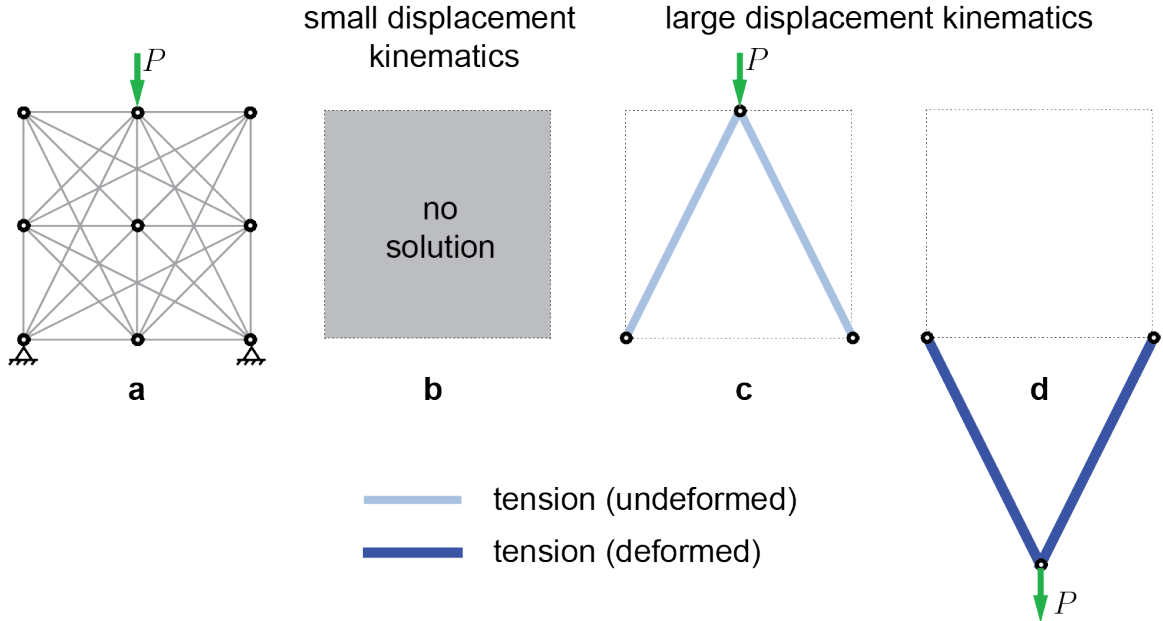


Figure 2.3: Illustration of the need for large displacement kinematics with use of the cable material model: (a) Domain, ground structure, and boundary conditions; (b) unbounded solution for the cable material model with small displacement kinematics; (c) undeformed and (d) deformed solution for the cable material model with large displacement kinematics in which equilibrium is achieved in the deformed configuration.

include minimum strain energy of the final structure and maximum end-stiffness, which corresponds to the tangent of the load-displacement diagram [51]. All of these objective functions require solution of an extra adjoint equation in the sensitivity analysis.

No extra adjoint equation is needed in the case of maximizing the total stationary potential energy, which is equivalent to minimizing compliance for linear problems [53], minimizing complementary energy for material nonlinear problems [61], and minimizing complementary energy for cable networks under finite deformations [62]. For this reason, the total potential energy objective has been adopted for optimization of trusses with nonlinear material behavior [30, 61, 63] and is also adopted in this dissertation for optimal layout design of tension-only (cable) structures under finite displacements and deformations (refer to Equation 2.1).

Another challenge in topology optimization problems that consider large displacements and deformations is the possibility of critical points in the equilibrium path (e.g., buckling, snap-through, and snap-back behavior), which require special techniques (e.g., arc-length methods) to traverse the full nonlinear load-displacement curve [47, 51, 64]. However, tension-only structures designed using the proposed cable topology optimization formulation

inherently avoid these situations since, by construction, the selected strain energy density function, which does not allow compression, prevents limit points in the equilibrium path. Thus, standard Newton-Raphson iterations are sufficient to capture the desired behavior, as discussed in section 2.7.

Another difficulty avoided in the cable problem is handling the numerical difficulties (non-convergence) caused by excessive deformations in low-density regions of the domain, which may lead to an indefinite or negative definite tangent stiffness matrix [46]. Several methods to circumvent this issue have been proposed [44, 50, 54, 56–58]. Perhaps the most widely adopted method is to interpolate between the nonlinear and linear strain-energy density functions so that the low-density elements behave linearly and avoid numerical issues [54, 59]. The proposed tension-only cable formulation naturally avoids this problem since excessive deformations of 1D elements with the selected strain-energy density function does not pose a numerical challenge. In fact, in subsection 2.6.2, the tangent stiffness matrix is shown to be positive-semi definite, which, in addition to preventing numerical difficulties in solution of the nonlinear equilibrium equations, also has implications on the convexity of the optimization problem [61, 63].

2.3 Large deformation kinematics

The kinematics considered here is similar to that of the uniaxial, large displacement, large deformation, pin-jointed truss member provided by Bonet and Wood [65] (see Figure 2.4). In what follows, the terminology “large deformations” is used to encompass both large displacements and finite strains. Upper-case letters are used to describe the undeformed configuration and lower-case letters are used to describe the deformed configuration. As such, the positions of cable member i are described for the undeformed and deformed configurations, respectively, as

$$\mathbf{X}_i = \sum_{j=1}^{n_d} X_i^j \mathbf{E}^j, \text{ and } \mathbf{x}_i = \sum_{j=1}^{n_d} x_i^j \mathbf{e}^j, \quad (2.2)$$

where n_d is the number of spatial dimensions and the coordinate frames for the undeformed (\mathbf{E}^j , $j = 1, \dots, n_d$) and deformed (\mathbf{e}^j , $j = 1, \dots, n_d$) configurations are defined to coincide. Also in Figure 2.4, the displacement, $\mathbf{u}_i = \sum_{j=1}^{n_d} u_i^j \mathbf{E}^j$, of member i is introduced. For later use in the solution scheme, incremental displacements $\Delta \mathbf{u}_i^p$ and $\Delta \mathbf{u}_i^q$ at ends p and q of

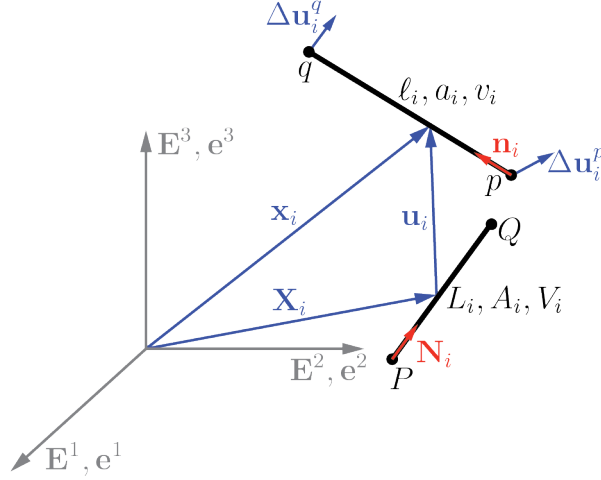


Figure 2.4: Kinematics for 3D cable member i .

deformed member i are also defined.

Based on the coordinates of ends P (p) and Q (q) of the member, the lengths of member i are computed as

$$L_i = \left[(\mathbf{X}_i^Q - \mathbf{X}_i^P) \cdot (\mathbf{X}_i^Q - \mathbf{X}_i^P) \right]^{1/2} \quad \text{and} \quad l_i = [(\mathbf{x}_i^q - \mathbf{x}_i^p) \cdot (\mathbf{x}_i^q - \mathbf{x}_i^p)]^{1/2}, \quad (2.3)$$

where L_i and l_i denote the lengths of member i in the undeformed and deformed configurations, respectively. Similarly, the orientations of member i in the undeformed and deformed configurations, respectively, are defined by unit vectors along their axes,

$$\mathbf{N}_i = \frac{\mathbf{X}_i^Q - \mathbf{X}_i^P}{L_i} \quad \text{and} \quad \mathbf{n}_i = \frac{\mathbf{x}_i^q - \mathbf{x}_i^p}{l_i}. \quad (2.4)$$

Uniaxial strain is assumed, such that the transverse principal stretches, λ_i^2, λ_i^3 , are unity and the fundamental measure of deformation in each cable member is the axial stretch, $\lambda_i^1 = \lambda_i = l_i/L_i$. The Jacobian, $J_i = \lambda_i \lambda_i^2 \lambda_i^3 = \lambda_i$, gives the ratio of the volume in the deformed configuration to the volume in the undeformed configuration,

$$dv_i = J_i dV_i \quad \rightarrow \quad J_i = \frac{v_i}{V_i} = \frac{a_i l_i}{A_i L_i}, \quad (2.5)$$

where the cross-sectional areas of member i are denoted A_i and a_i in the undeformed and deformed configurations, respectively, and the volumes of member i are denoted $V_i = A_i L_i$

and $v_i = a_i \ell_i$ in the undeformed and deformed configurations, respectively. Note that since $J_i = \lambda_i$, the deformed and undeformed cross-sectional areas are equal (i.e., $a_i = A_i$).

In this dissertation, three different commonly-used measures of strain are explored: linear (LIN), logarithmic (LOG), and Green-Lagrange (GL) strain. Each strain measure is derived by integrating the associated measure of instantaneous strain, $d\varepsilon_i^\square$, over the entire deformation from L_i to ℓ_i such that

$$\varepsilon_i^\square = \int_{L_i}^{\ell_i} d\varepsilon_i^\square, \quad (2.6)$$

where the symbol, \square , is a placeholder for strain measures, LIN, LOG, and GL.

2.3.1 Linear strain measure

To derive a linear strain measure, the instantaneous strain in member i , $d\varepsilon_i^{\text{LIN}}$, is defined as the ratio of the instantaneous change in member length to the original member length, i.e.,

$$d\varepsilon_i^{\text{LIN}} = \frac{d\ell_i}{L_i}, \quad (2.7)$$

from which Equation 2.6 is used to derive a linear strain measure,

$$\varepsilon_i^{\text{LIN}} = \int_{L_i}^{\ell_i} \frac{1}{L_i} d\ell_i = \lambda_i - 1. \quad (2.8)$$

2.3.2 Logarithmic strain measure

To derive a logarithmic strain measure, the instantaneous strain in member i , $d\varepsilon_i^{\text{LOG}}$, is defined as the ratio of the instantaneous change in member length to the deformed member length, i.e.,

$$d\varepsilon_i^{\text{LOG}} = \frac{d\ell_i}{\ell_i}, \quad (2.9)$$

from which Equation 2.6 is used to derive a logarithmic strain measure,

$$\varepsilon_i^{\text{LOG}} = \int_{L_i}^{\ell_i} \frac{1}{\ell_i} d\ell_i = \ln \lambda_i. \quad (2.10)$$

2.3.3 Green-Lagrange strain measure

To derive a Green-Lagrange strain measure, the instantaneous strain in member i is defined as

$$d\varepsilon_i^{\text{GL}} = \frac{\ell_i d\ell_i}{L_i^2}, \quad (2.11)$$

from which Equation 2.6 is used to derive a Green-Lagrange strain measure,

$$\varepsilon_i^{\text{GL}} = \int_{L_i}^{\ell_i} \frac{\ell_i}{L_i^2} d\ell_i = \frac{1}{2} (\lambda_i^2 - 1). \quad (2.12)$$

2.4 Cable constitutive model

The hyperelastic strain energy density function (see Figure 2.2a),

$$\Psi_i = \begin{cases} \frac{E_i}{2} (\varepsilon_i^{\square})^2 & \text{if } \lambda_i \geq 1 \\ 0 & \text{otherwise} \end{cases}, \quad (2.13)$$

is chosen to model tension-only behavior, where E_i is a material parameter. By substituting the three strain measures into Equation 2.13, the strain energy density function and its derivatives can be expressed in terms of axial stretch, λ_i . Such quantities are summarized in Table 2.1 for each of the three strain measures considered here. The derivatives will be useful in subsection 2.6.2, when the implications of the choice of strain measure on convexity of the optimization problem are evaluated.

Note that the strain energy density in (tensile) member i can also be computed as

$$\Psi_i = \frac{1}{V_i} \int_{L_i}^{\ell_i} t_i d\ell_i = \frac{1}{V_i} \int_{L_i}^{\ell_i} \sigma_i a_i d\ell_i, \quad (2.14)$$

where the magnitude of internal force in member i is $t_i = \sigma_i a_i$, with σ_i the axial component of Cauchy stress in member i . Then, by substituting each of the strain measures into

Table 2.1: Definition of three common strain measures, associated strain energy density functions, and their first and second derivatives with respect to stretch

strain measure	instantaneous strain, $d\varepsilon_i$	strain, ε_i	strain energy density, Ψ_i	$\frac{\partial \Psi_i}{\partial \lambda_i}$	$\frac{\partial^2 \Psi_i}{\partial \lambda_i^2}$
linear	$\frac{d\ell_i}{L_i}$	$\lambda_i - 1$	$\frac{E_i}{2} (\lambda_i - 1)^2$	$E_i (\lambda_i - 1)$	E_i
logarithmic	$\frac{d\ell_i}{\ell_i}$	$\ln \lambda_i$	$\frac{E_i}{2} (\ln \lambda_i)^2$	$\frac{E_i}{\lambda_i} \ln \lambda_i$	$\frac{E_i}{\lambda_i^2} (1 - \ln \lambda_i)$
Green-Lagrange	$\frac{\ell_i d\ell_i}{L_i^2}$	$\frac{1}{2} (\lambda_i^2 - 1)$	$\frac{E_i}{8} (\lambda_i^2 - 1)^2$	$\frac{E_i \lambda_i}{2} (\lambda_i^2 - 1)$	$\frac{E_i}{2} (3\lambda_i^2 - 1)$

Equation 2.14, σ_i can be computed for each case as follows:

$$\begin{aligned}
 \Psi_i &= \int_0^{\varepsilon_i^{\text{LIN}}} \frac{\sigma_i a_i L_i}{V_i} d\varepsilon_i^{\text{LIN}} \rightarrow \sigma_i = \frac{\partial \Psi_i}{\partial \varepsilon_i^{\text{LIN}}} = \frac{\partial \Psi_i}{\partial \lambda_i} && \text{(linear strain)} \\
 \Psi_i &= \int_0^{\varepsilon_i^{\text{LOG}}} \frac{\sigma_i a_i \ell_i}{V_i} d\varepsilon_i^{\text{LOG}} \rightarrow \sigma_i = \frac{1}{\lambda_i} \frac{\partial \Psi_i}{\partial \varepsilon_i^{\text{LOG}}} = \frac{\partial \Psi_i}{\partial \lambda_i} && \text{(logarithmic strain)} \\
 \Psi_i &= \int_0^{\varepsilon_i^{\text{GL}}} \frac{\sigma_i a_i L_i^2}{V_i \ell_i} d\varepsilon_i^{\text{GL}} \rightarrow \sigma_i = \lambda_i \frac{\partial \Psi_i}{\partial \varepsilon_i^{\text{GL}}} = \frac{\partial \Psi_i}{\partial \lambda_i} && \text{(Green-Lagrange strain).}
 \end{aligned} \tag{2.15}$$

In Equation 2.15, the definition of axial stretch (i.e., $\lambda_i = \ell_i/L_i$), the uniaxial strain assumption (i.e., $a_i = A_i$), and the chain rule (i.e., $\partial \Psi_i / \partial \varepsilon_i = (\partial \Psi_i / \partial \lambda_i) (\partial \lambda_i / \partial \varepsilon_i)$) are used.

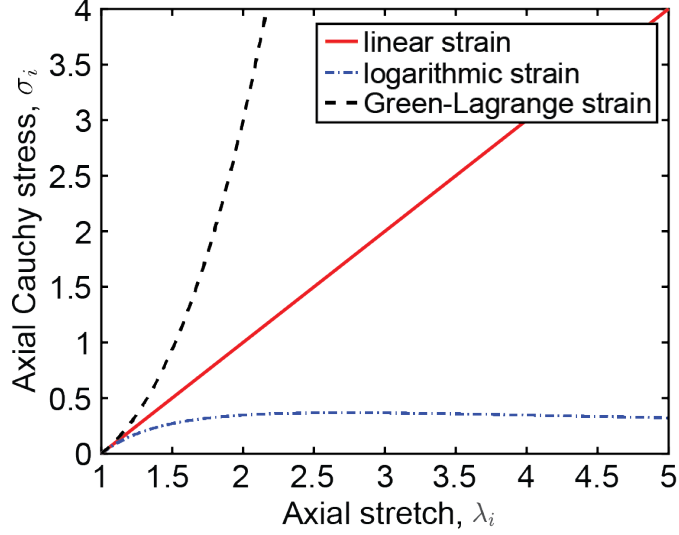


Figure 2.5: Axial Cauchy stress, σ_i , as a function of axial stretch, λ_i , for three different strain measures.

It follows that the constitutive relations, plotted in Figure 2.5, for each strain case are:

$$\begin{aligned}
 \sigma_i &= \begin{cases} E_i \varepsilon_i^{\text{LIN}} & \text{if } \lambda_i \geq 1 \\ 0 & \text{otherwise} \end{cases} \quad (\text{linear strain}) \\
 \sigma_i &= \begin{cases} \frac{E_i}{\lambda_i} \varepsilon_i^{\text{LOG}} & \text{if } \lambda_i \geq 1 \\ 0 & \text{otherwise} \end{cases} \quad (\text{logarithmic strain}) \\
 \sigma_i &= \begin{cases} E_i \lambda_i \varepsilon_i^{\text{GL}} & \text{if } \lambda_i \geq 1 \\ 0 & \text{otherwise} \end{cases} \quad (\text{Green-Lagrange strain}).
 \end{aligned} \tag{2.16}$$

Note that in the case of linear strain, the material parameter, E_i , turns out to be the well-known Young's modulus (see Figure 2.2b). Additionally, the following relation holds, regardless of the selected strain measure:

$$\frac{\partial \sigma_i}{\partial \lambda_i} = \frac{\partial^2 \Psi_i}{\partial \lambda_i^2}. \tag{2.17}$$

2.5 Nonlinear equilibrium equations

Equilibrium of the cable network is enforced by requiring the total potential energy, $\Pi(\mathbf{u})$, to be stationary. The total potential energy, $\Pi(\mathbf{u})$, can be written as the sum of internal

strain energy, $U_I(\mathbf{u})$, and potential of externally applied loads, $U_E(\mathbf{u})$, where

$$U_I(\mathbf{u}) = \sum_{i=1}^N \int_{V_i} \Psi_i(\mathbf{u}) dV = \sum_{i=1}^N V_i \Psi_i(\mathbf{u}) \quad \text{and} \quad U_E(\mathbf{u}) = -\mathbf{F}^T \mathbf{u}. \quad (2.18)$$

In Equation 2.18, N is the number of cable members in the model and \mathbf{F} is the vector of external loads. Then the stationary condition of the potential energy is

$$\begin{aligned} \frac{\partial \Pi(\mathbf{u})}{\partial \mathbf{u}} = 0 &= \frac{\partial U_I(\mathbf{u})}{\partial \mathbf{u}} + \frac{\partial U_E(\mathbf{u})}{\partial \mathbf{u}} \\ &= \sum_{i=1}^N V_i \frac{\partial \Psi_i(\mathbf{u})}{\partial \mathbf{u}} - \frac{\partial(\mathbf{F}^T \mathbf{u})}{\partial \mathbf{u}} \\ &= \sum_{i=1}^N V_i \frac{\partial \Psi_i(\mathbf{u})}{\partial \lambda_i} \frac{\partial \lambda_i}{\partial \mathbf{u}} - \mathbf{F}. \end{aligned} \quad (2.19)$$

Using Equation 2.15, the derivative of the stored strain energy function in Equation 2.19 can be written in terms of the Cauchy stress and only the term,

$$\begin{aligned} \frac{\partial \lambda_i}{\partial \mathbf{u}} &= \frac{1}{L_i} \frac{\partial \ell_i}{\partial \mathbf{u}} \\ &= \frac{1}{L_i} \frac{\partial \ell_i}{\partial \mathbf{x}} \frac{\partial \mathbf{x}}{\partial \mathbf{u}} \\ &= \frac{1}{L_i} \frac{\partial \ell_i}{\partial \mathbf{x}} \frac{\partial(\mathbf{X} + \mathbf{u})}{\partial \mathbf{u}} \\ &= \frac{1}{L_i} \mathbf{b}_i, \end{aligned} \quad (2.20)$$

needs to be computed, where:

$$\mathbf{b}_i = \left\{ \dots \frac{\partial \ell_i}{\partial \mathbf{x}_p} \dots \frac{\partial \ell_i}{\partial \mathbf{x}_q} \dots \right\}^T = \left\{ \dots - \mathbf{n}_i \dots \mathbf{n}_i \dots \right\}^T \quad (2.21)$$

and the dots indicate zeros. Then the stationary condition of the potential energy from Equation 2.19 can be written as

$$\begin{aligned} \frac{\partial \Pi(\mathbf{u})}{\partial \mathbf{u}} = 0 &= \sum_{i=1}^N V_i \sigma_i \frac{1}{L_i} \mathbf{b}_i - \mathbf{F} \\ &= \sum_{i=1}^N a_i \sigma_i \mathbf{b}_i - \mathbf{F} \\ 0 &= \mathbf{T}(\mathbf{u}) - \mathbf{F} = \mathbf{R}(\mathbf{u}), \end{aligned} \quad (2.22)$$

where the uniaxial strain assumption (i.e., $a_i = A_i$) and the definition of internal force in

member i , $t_i = a_i \sigma_i$, are used again. Additionally, the vector of member internal forces is defined as $\mathbf{T}(\mathbf{u})$ and the vector of residual nodal forces is defined as $\mathbf{R}(\mathbf{u})$. Since $\mathbf{T}(\mathbf{u})$ is a function of the deformed configuration, the equilibrium equations in Equation 2.22 are nonlinear. Thus, they need to be linearized and solved iteratively.

2.5.1 Linearization

Given a solution to Equation 2.22 (i.e., \mathbf{u}_k at iteration k), a new value, $\mathbf{u}_{k+1} = \mathbf{u}_k + \Delta\mathbf{u}_k$, is obtained in terms of an increment, $\Delta\mathbf{u}_k$, by establishing a linear approximation of the residual,

$$\mathbf{R}(\mathbf{u}_{k+1}) = \mathbf{R}(\mathbf{u}_k) + \mathcal{D}\mathbf{R}(\mathbf{u}_k) [\Delta\mathbf{u}_k] = \mathbf{0}, \quad (2.23)$$

where the directional derivative is determined using the chain rule,

$$\begin{aligned} \mathcal{D}\mathbf{R}(\mathbf{u}_k) [\Delta\mathbf{u}_k] &= \left. \frac{d}{d\epsilon} [\mathbf{R}(\mathbf{u}_k + \epsilon\Delta\mathbf{u}_k)] \right|_{\epsilon=0} \\ &= \left. \frac{\partial\mathbf{R}}{\partial\mathbf{u}} \right|_{\mathbf{u}_k} \left. \frac{\partial(\mathbf{u}_k + \epsilon\Delta\mathbf{u}_k)}{\partial\epsilon} \right|_{\epsilon=0} \\ &= \mathbf{K}^t(\mathbf{u}_k) \Delta\mathbf{u}_k. \end{aligned} \quad (2.24)$$

In Equation 2.24 the global tangent stiffness matrix, \mathbf{K}^t , is defined as the derivative of the unbalanced forces with respect to the displacement field. Now, the linear set of equations to solve at each iteration, k , is

$$\mathbf{K}^t(\mathbf{u}_k) \Delta\mathbf{u}_k = -\mathbf{R}(\mathbf{u}_k) = \mathbf{F} - \mathbf{T}(\mathbf{u}_k). \quad (2.25)$$

2.5.2 Tangent stiffness matrix

Since the external force vector is not a function of the deformed configuration, \mathbf{K}^t is derived as the derivative of the internal member forces with respect to the displacement field,

$$\mathbf{K}^t = \left. \frac{\partial\mathbf{T}}{\partial\mathbf{u}} \right|_{\mathbf{u}_k}, \quad (2.26)$$

and can be assembled from the element tangent stiffness matrices,

$$\mathbf{k}_i^t(\mathbf{u}_k) = \begin{bmatrix} \left. (\partial \mathbf{t}_i^p / \partial \mathbf{u}_i^p) \right|_{\mathbf{u}_k} & \left. (\partial \mathbf{t}_i^p / \partial \mathbf{u}_i^q) \right|_{\mathbf{u}_k} \\ \left. (\partial \mathbf{t}_i^q / \partial \mathbf{u}_i^p) \right|_{\mathbf{u}_k} & \left. (\partial \mathbf{t}_i^q / \partial \mathbf{u}_i^q) \right|_{\mathbf{u}_k} \end{bmatrix} = \begin{bmatrix} \mathbf{k}_i^{pp} & \mathbf{k}_i^{pq} \\ \mathbf{k}_i^{qp} & \mathbf{k}_i^{qq} \end{bmatrix}, \quad (2.27)$$

where $\mathbf{k}_i^t(\mathbf{u}_k)$ denotes the element tangent stiffness matrix of element i and the element internal force vector and element displacement vector are, respectively,

$$\mathbf{t}_i(\mathbf{u}_i) = \begin{Bmatrix} \mathbf{t}_i^p \\ \mathbf{t}_i^q \end{Bmatrix} = t_i \begin{Bmatrix} -\mathbf{n}_i \\ \mathbf{n}_i \end{Bmatrix} \quad \text{and} \quad \mathbf{u}_i = \begin{Bmatrix} \mathbf{u}_i^p \\ \mathbf{u}_i^q \end{Bmatrix}, \quad (2.28)$$

for member i with end nodes p and q . In Equation 2.27, it is noted that $\mathbf{k}_i^{pp} = \mathbf{k}_i^{qq} = -\mathbf{k}_i^{pq} = -\mathbf{k}_i^{qp}$, and therefore, $\mathbf{k}_i^t(\mathbf{u}_k)$ can be derived using only one of the four partitions.

For example,

$$\begin{aligned} \mathbf{k}_i^{qq} &= \left. \frac{\partial \mathbf{t}_i^q}{\partial \mathbf{u}_i^q} \right|_{\mathbf{u}_k} = \frac{\partial(\sigma_i a_i \mathbf{n}_i)}{\partial \mathbf{u}_i^q} \\ &= a_i \left. \frac{\partial \sigma_i}{\partial \mathbf{u}_i^q} \right|_{\mathbf{u}_k} \mathbf{n}_i + a_i \sigma_i \left. \frac{\partial \mathbf{n}_i}{\partial \mathbf{u}_i^q} \right|_{\mathbf{u}_k} \\ &= a_i \left. \frac{\partial \sigma_i}{\partial \lambda_i} \frac{\partial \lambda_i}{\partial \ell_i} \frac{\partial \ell_i}{\partial \mathbf{x}_i^q} \frac{\partial \mathbf{x}_i^q}{\partial \mathbf{u}_i^q} \right|_{\mathbf{u}_k} \mathbf{n}_i + a_i \sigma_i \left(\left. \frac{\partial \left(\frac{1}{\ell_i} \right)}{\partial \mathbf{u}_i^q} \right|_{\mathbf{u}_k} (\mathbf{x}_i^q - \mathbf{x}_i^p) + \left. \frac{\partial(\mathbf{x}_i^q - \mathbf{x}_i^p)}{\partial \mathbf{u}_i^q} \right|_{\mathbf{u}_k} \frac{1}{\ell_i} \right) \\ &= \frac{a_i}{L_i} \frac{\partial \sigma_i}{\partial \lambda_i} \mathbf{n}_i \mathbf{n}_i^T + a_i \sigma_i \left(-\frac{(\mathbf{x}_i^q - \mathbf{x}_i^p)}{\ell_i^2} \mathbf{n}_i + \frac{1}{\ell_i} \mathbf{I} \right) \\ &= \frac{a_i}{L_i} \frac{\partial \sigma_i}{\partial \lambda_i} \mathbf{n}_i \mathbf{n}_i^T - \frac{a_i \sigma_i}{\ell_i} \mathbf{n}_i \mathbf{n}_i^T + \frac{a_i \sigma_i}{\ell_i} \mathbf{I} \\ &= \frac{a_i}{L_i} \frac{\partial^2 \Psi_i}{\partial \lambda_i^2} \mathbf{n}_i \mathbf{n}_i^T + \frac{t_i}{\ell_i} (\mathbf{I} - \mathbf{n}_i \mathbf{n}_i^T). \end{aligned} \quad (2.29)$$

In the second line of Equation 2.29, recall that $a_i = A_i$; in the last line of Equation 2.29, recall the definition of internal force, $t_i = a_i \sigma_i$, and the relationship between Ψ_i and σ_i from Equation 2.17. Additionally, in Equation 2.29, the following derivatives are used:

$$\frac{\partial \ell_i}{\partial \mathbf{x}_i^q} = \frac{\mathbf{x}_i^q - \mathbf{x}_i^p}{\ell_i} = \mathbf{n}_i, \quad (2.30)$$

$$\frac{\partial(\mathbf{x}_i^q - \mathbf{x}_i^p)}{\partial \mathbf{u}_i^q} = \frac{\partial \mathbf{x}_i^q}{\partial \mathbf{u}_i^q} = \frac{\partial(\mathbf{X}_i^Q + \mathbf{u}_i^q)}{\partial \mathbf{u}_i^q} = \mathbf{I}, \quad (2.31)$$

and

$$\left. \frac{\partial\left(\frac{1}{\ell_i}\right)}{\partial \mathbf{u}_i^q} \right|_{\mathbf{u}_k} = -\frac{\mathbf{x}_i^q - \mathbf{x}_i^p}{\ell_i^3} = -\frac{1}{\ell_i^2} \mathbf{n}_i, \quad (2.32)$$

where \mathbf{I} is an $n_d \times n_d$ identity matrix.

2.6 Optimization formulation

With the mechanics defined above, the topology optimization formulation in Equation 2.1 becomes specific to maximizing the structural stiffness of cable networks. The stationary total potential energy is chosen to be maximized since it does not require an additional adjoint equation and has been shown equivalent to minimizing end-compliance for linear problems [66], complementary energy for nonlinear problems with small strains [61], and complementary energy for cable networks under large deformations [62]. Next, the sensitivities of this objective function are derived without the need for an adjoint vector, remarks regarding convexity of the formulation are provided, and the optimality conditions are stated.

2.6.1 Sensitivity analysis

The sensitivities of the linear constraint, $g(\mathbf{A})$, in Equation 2.1 with respect to the design variables are

$$\frac{\partial g(\mathbf{A})}{\partial A_i} = L_i \quad i = 1, \dots, N \quad (2.33)$$

and the sensitivities of the objective function, $f(\mathbf{A}, \mathbf{u}(\mathbf{A}))$, in Equation 2.1 with respect to the design variables are

$$\frac{\partial f(\mathbf{A}, \mathbf{u}(\mathbf{A}))}{\partial A_i} = -\frac{\partial \Pi_{\min}(\mathbf{A}, \mathbf{u}(\mathbf{A}))}{\partial A_i} - \frac{\partial \Pi_{\min}(\mathbf{A}, \mathbf{u}(\mathbf{A}))}{\partial \mathbf{u}} \frac{\partial \mathbf{u}(\mathbf{A})}{\partial A_i} \quad i = 1, \dots, N. \quad (2.34)$$

Due to the equilibrium conditions, the second term in Equation 2.34 goes to zero. By expressing Π_{\min} as the sum of the internal strain energy, $U_I(\mathbf{A}, \mathbf{u}(\mathbf{u}))$, and potential of

externally applied loads, $U_E(\mathbf{u}(\mathbf{A}))$, and noting that $U_E(\mathbf{u}(\mathbf{A}))$ is (explicitly) independent of A_i , the sensitivity of the objective function can be written as [66]

$$\frac{\partial f(\mathbf{A}, \mathbf{u}(\mathbf{A}))}{\partial A_i} = -\frac{\partial U_E(\mathbf{A}, \mathbf{u}(\mathbf{A}))}{\partial A_i} = -L_i \Psi_i(\mathbf{u}(\mathbf{A})) \quad i = 1, \dots, N. \quad (2.35)$$

Note that there is no need to calculate an adjoint vector for the selected objective function. Additionally, $L_i \Psi_i(\mathbf{u}(\mathbf{A})) \geq 0$, i.e., the sensitivities of the objective are always non-positive, an observation that demonstrates a clear parallel between the current formulation for maximum stationary potential energy and that of minimum end-compliance. Note also that although the constitutive model is discontinuous at zero strain (see Figure 2.2b), the sensitivity of the objective function is continuous everywhere since it depends only on the continuous strain energy density function (see Figure 2.2a) and not on its derivative.

2.6.2 Convexity

In the case of material nonlinearities with small deformation kinematics, convexity of the objective function in Equation 2.1 was proven for a positive semi-definite tangent stiffness matrix [61, 63], which is shown to hold for (non-strictly) convex hyperelastic material models. Noting that the global tangent stiffness matrix is guaranteed to be positive semi-definite if the element tangent stiffness matrices (from which it is assembled) are positive semi-definite, the proof focuses on the element tangent stiffness matrix. Since the tangent stiffness matrix in Equation 2.27 and Equation 2.29 contains a geometric stiffness term that did not exist in the previous small deformation case, the criteria needed for a positive semi-definite tangent stiffness matrix in the case of finite deformation kinematics needs to be investigated.

Again, only the element tangent stiffness matrix is studied. It is a partitioned matrix (see Equation 2.27) that satisfies conditions such that only the lower left partition, \mathbf{k}_i^{qq} , needs to be checked for positive semi-definiteness [67]. In Equation 2.29, \mathbf{k}_i^{qq} is a sum of two terms. The first term is analogous to the small deformation case and is positive semi-definite given that strain energy density function has non-negative curvature, i.e., $\partial^2 \Psi / \partial \lambda_i^2 \geq 0$. To show positive semi-definiteness of the second term, the principal minors test [68] can be used. The unit vector oriented along member i 's axis can be defined as $\mathbf{n}_i = [\cos \theta_1, \cos \theta_2, \cos \theta_3]^T$ for some angles, $0 \leq \theta_1, \theta_2, \theta_3 \leq 2\pi$, defining the orientation of member i in Cartesian space.

Then,

$$(\mathbf{I} - \mathbf{n}_i \mathbf{n}_i^T) = \begin{bmatrix} 1 - \cos^2 \theta_1 & \cos \theta_2 \cos \theta_1 & \cos \theta_3 \cos \theta_1 \\ & 1 - \cos^2 \theta_2 & \cos \theta_3 \cos \theta_2 \\ \text{symm.} & & 1 - \cos^2 \theta_3 \end{bmatrix}, \quad (2.36)$$

and it is easy to confirm that all principal minors of $(\mathbf{I} - \mathbf{n}_i \mathbf{n}_i^T)$ are non-negative [68]. Thus, for tension-only structures in which $t_i/\ell_i \geq 0$, the stiffness matrix is guaranteed to be positive semi-definite and the same criterion for convexity as in the small deformation case is found, i.e., the optimization problem in Equation 2.1 is convex for (non-strictly) convex hyperelastic material models.

The three strain measures discussed in section 2.3 and section 2.4 are revisited to evaluate whether they lead to a convex optimization problem. From the last column of Table 2.1, it is clear that the convexity requirement, $\partial^2 \Psi_i / \partial \lambda_i^2 \geq 0$, is always satisfied for the cases of linear and Green-Lagrange strain, and is only satisfied for logarithmic strain when $\lambda_i \leq e$. To further illustrate the differences between the three strain measures, the strain energy density function is plotted against axial strain and axial stretch in Figure 2.6a and b, respectively. For the case of logarithmic strain, the strain energy density function is not a convex function of axial stretch, and thus does not meet the curvature requirement needed for a positive semi-definite element tangent stiffness matrix. Although they are different, the general trend of the linear and Green-Lagrange strain measures are similar, and for the conceptual designs pursued here, these two strain measures can be expected to lead to similar designs. For simplicity, the linear strain measure is considered in the remainder of this chapter.

2.6.3 Optimality conditions

The KKT optimality conditions for the convex optimization problem in Equation 2.1 have been derived previously and, for completeness, are stated here as [61, 63]

$$\begin{aligned} \Psi_i(\mathbf{u}(\mathbf{A}^*)) &\geq \Lambda^* && \text{if } A_i^* = A_i^{max}, \\ \Psi_i(\mathbf{u}(\mathbf{A}^*)) &= \Lambda^* && \text{if } 0 < A_i^* < A_i^{max}, \text{ and} \\ \Psi_i(\mathbf{u}(\mathbf{A}^*)) &\leq \Lambda^* && \text{if } A_i^* = 0, \end{aligned} \quad (2.37)$$

where \mathbf{A}^* and Λ^* are the solution and Lagrange multiplier, respectively, at the optimum. Note that when the box constraints are inactive at the optimal point, all cable members have

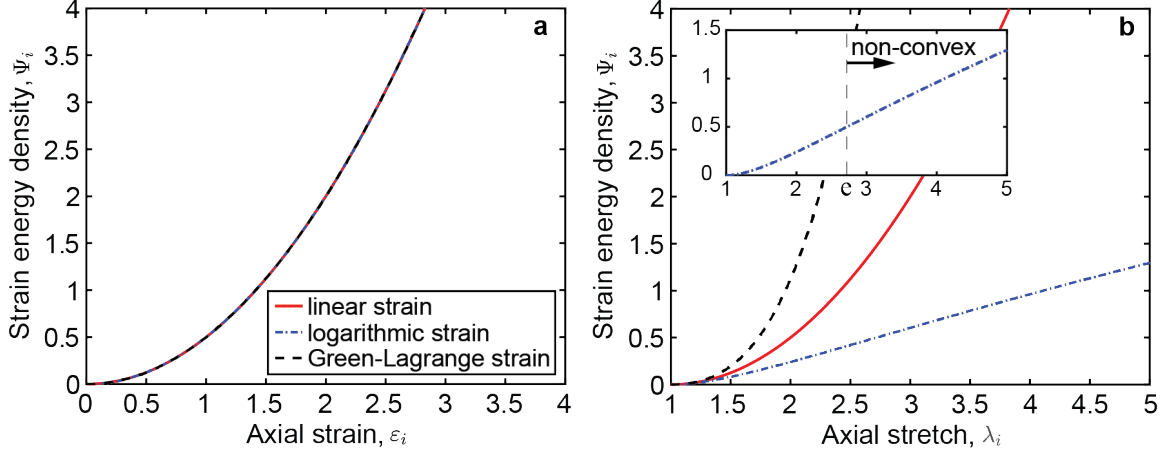


Figure 2.6: Strain energy density function versus (a) axial strain, ε_i and (b) axial stretch, λ_i , for three common strain measures.

equal strain energy, analogously to full stressed design in the linear case with end-compliance objective function.

2.7 Details of the numerical implementation

In this section, details related to implementation of the topology optimization formulation for structures composed of tension-only members are provided. Specifically addressed are the damped Newton algorithm with line search used for efficient solution of the possibly singular nonlinear equilibrium equations, the design variable update scheme, reduction of the problem size during the optimization iterations, and a maximum end filter used to extract clean solutions (after convergence) that satisfy global equilibrium.

2.7.1 Damped Newton with line search

Typically, Newton-Raphson iterations are used to solve the nonlinear equilibrium equations according to the linearization scheme in subsection 2.5.1; however, due to the zero-lower bound on the design variables in Equation 2.1, the tangent stiffness matrix may become singular, preventing solution to Equation 2.25. Thus, a damped Newton method [43] is adopted so that the linearized equilibrium equation of the nonlinear system becomes

$$\mathbf{K}^{t,\eta} \Delta \mathbf{u}_k = -\mathbf{R}(\mathbf{u}_k) = \mathbf{F} - \mathbf{T}(\mathbf{u}_k), \quad (2.38)$$

where $\mathbf{K}^{t,\eta} = \mathbf{K}^t + \eta \mathbf{I}$ and η is the damped Newton parameter defined as $\eta_0 \approx 10^{-12}$ to 10^{-8}

multiplied by the mean of the diagonal of \mathbf{K}^t (a similar regularization scheme was used for small deformation problems [63, 69, 70]). The damped Newton scheme using Equation 2.38 and considering a load control approach is provided in Algorithm 1 of Appendix A. It is noted that Newton methods only converge locally and depending on the constitutive model, the algorithm may diverge [43]. Although the constitutive model used here will not lead to divergence, convergence of the first several iterations of the damped Newton algorithm can be improved by taking strategically sized steps. In this case, the update becomes (see, for example, [71–73])

$$\mathbf{u}_{k+1} = \mathbf{u}_k + \tau_k \Delta \mathbf{u}_k. \quad (2.39)$$

Backtracking line search with an Armijo condition (inexact line search) is used to find the line search parameter, τ_k . The specific line search algorithm used in the implementation [71] is provided in Algorithm 2 of Appendix A.

2.7.2 Design variable update

The design variables are updated in each optimization iteration using the Optimality Criteria (OC) method. The OC method is characterized by a recursive update derived using Lagrangian duality of truncated Taylor approximated subproblems with intervening variables, $\xi_i = A_i^{\alpha_i}$, $\alpha_i < 0$, in the approximation of the objective function [5, 19, 74, 75]. The derivation, provided in Appendix B, leads to the update for iteration $k + 1$,

$$A_i^{k+1} = \begin{cases} A_i^+, & A_i^* \geq A_i^+ \\ A_i^-, & A_i^* \leq A_i^- \\ A_i^*, & \text{otherwise,} \end{cases} \quad (2.40)$$

where A_i^{k+1} is the design at the next iteration and A_i^* is the candidate design for the next iteration that is accepted if it is within bounds,

$$A_i^- = \max(0, A_i^k - M) \quad \text{and} \quad A_i^+ = \min(A_i^{\max}, A_i^k + M) \quad (2.41)$$

defined by box constraints, $\bar{\rho} = 0$ and $\underline{\rho} = A_i^{\max}$; move limit, $M = \beta A_0$ [69]; and initial cross-sectional area, A_0 . The candidate design, A_i^* , is obtained from a fixed-point iteration

of the form

$$A_i^* = A_i^k (B_i)^{\frac{1}{1 - \alpha_i}}, \quad B_i = -\frac{\frac{\partial f}{\partial A_i} \Big|_{\mathbf{A}=\mathbf{A}_k}}{\Lambda \frac{\partial g}{\partial A_i} \Big|_{\mathbf{A}=\mathbf{A}_k}}. \quad (2.42)$$

Note that the recursive nature of the OC method does not allow reappearance of zero-area members.

In Equation 2.42, Λ is the Lagrange multiplier and the quantity $1/(1 - \alpha_i)$ is a damping factor corresponding to a reciprocal approximation when $\alpha_i = -1$. Here α_i is determined using a two-point approximation such that the derivatives of the reciprocal approximation at iteration k match the derivatives of the objective function at iteration $k - 1$. That is,

$$\alpha_i = 1 + \frac{\ln \left(\frac{\frac{\partial f}{\partial A_i} \Big|_{\mathbf{A}=\mathbf{A}^k} / \frac{\partial f}{\partial A_i} \Big|_{\mathbf{A}=\mathbf{A}^{k-1}}}{\ln \left(A_i^k / A_i^{k-1} \right)} \right)}{\ln \left(A_i^k / A_i^{k-1} \right)}. \quad (2.43)$$

In the first iteration, $\alpha_i = -1$ and in subsequent iterations, α_i is computed based on Equation 2.43 with bounds, $-15 \leq \alpha_i \leq -0.1$, to ensure that the primal subproblem remains convex [75, 76].

2.7.3 Convergence criterion

Convergence of the optimization problem is determined based on the change in the design variables. Specifically, the optimization algorithm is aborted when

$$\max_i \left(\frac{|A_i^{k+1} - A_i^k|}{1 + A_i^k} \right) \leq \text{tol}, \quad (2.44)$$

where tol is a prescribed convergence tolerance.

2.7.4 Reduced order model

Since the OC update scheme does not allow zero-area members to reappear, they are removed in each iteration to continually reduce computational cost as convergence toward an optimal solution proceeds [63, 69, 70]. A mapping matrix, \mathbf{Q} , is constructed such that

$$\mathbf{u} = \mathbf{Q}\mathbf{u}^{\text{Top}}, \quad (2.45)$$

where \mathbf{u}^{Top} is the vector of nodal displacements considering only the degrees of freedom associated with the topology, i.e., the set of members in the initial ground structure that have non-null member area. Based on this mapping, the global tangent stiffness matrix, external load vector, and internal load vector associated with the topology, respectively, can be defined as

$$\mathbf{K}^{t,\text{Top}} = \mathbf{Q}^T \mathbf{K}^t \mathbf{Q}, \quad \mathbf{F}^{\text{Top}} = \mathbf{Q}^T \mathbf{F}, \quad \mathbf{T}^{\text{Top}}(\mathbf{u}_k^{\text{Top}}) = \mathbf{Q}^T \mathbf{T}(\mathbf{u}_k). \quad (2.46)$$

Then the associated linearized equilibrium equation of the nonlinear system becomes

$$(\mathbf{K}^{t,\text{Top}} + \eta \mathbf{I}) \Delta \mathbf{u}_k^{\text{Top}} = \mathbf{R}^{\text{Top}}(\mathbf{u}_k) = \mathbf{F}^{\text{Top}} - \mathbf{T}^{\text{Top}}(\mathbf{u}_k^{\text{Top}}), \quad (2.47)$$

with update,

$$\mathbf{u}_{k+1}^{\text{Top}} = \mathbf{u}_k^{\text{Top}} + \xi_k \Delta \mathbf{u}_k^{\text{Top}}. \quad (2.48)$$

2.7.5 End filter

Although a zero lower bound is enforced, the solutions often contain thin members that can be removed with negligible effect on the structural behavior. To clean up the final design, the maximum end filter assigns

$$A_i = \begin{cases} 0 & \text{if } \frac{A_i}{\max(\mathbf{A})} < \alpha_f \\ A_i & \text{otherwise,} \end{cases} \quad (2.49)$$

where α_f is the filter value selected using a bisection algorithm to ensure that, after filtering, the final design satisfies global equilibrium and the increase in the objective is controlled [70].

To determine whether to accept the filtered structure for a given α_f , the global equilibrium error of the filtered structure is first checked to be within a tolerance,

$$\frac{\|\mathbf{R}^{\text{Top}}(\mathbf{u}_k)\|}{\|\mathbf{F}^{\text{Top}}\|} \leq \text{err}^{\text{tol}}, \quad (2.50)$$

where err^{tol} is a tolerance typically chosen to be 10^{-4} . Additionally, the objective function value obtained after filtering, f_{filtered} , is checked to be within a margin of that obtained at

convergence, $f_{\text{converged}}$, i.e.,

$$\Delta f = \frac{(f_{\text{filtered}} - f_{\text{converged}})}{f_{\text{converged}}} \leq f^{\text{tol}}, \quad (2.51)$$

where f^{tol} is a user-prescribed tolerance. If either of Equation 2.50 or Equation 2.51 is not satisfied by a filter value, α_f , greater than $A_i / \max(\mathbf{A}) \forall i$, then no end filter is applied.

2.8 Cable net design examples

In this section four numerical examples are presented to illustrate the capabilities of the proposed formulation for obtaining maximum stiffness cable networks under possibly finite displacements and deformations. The first example highlights the effect of load magnitude on the final design when considering large deformation kinematics. The second example shows that when the design variable upper bound, A_i^{max} , is active, the final topology does not have constant stress or strain energy, as predicted by the KKT optimality conditions. The third example demonstrates that the cable topology optimization formulation may lead to topologies that would be meaningless in the case of small displacement kinematics, but that are well defined in the context of the current nonlinear model. Finally, the formulation is used to design spider web inspired cable nets that have similarities to an orb-web [77]. In all cases, the linear strain measure is considered. Since the problems provided are relatively small scale, symmetry reduction is not considered, but convexity of the formulation implies that symmetry reduction can be used for problems with symmetric domain and boundary conditions [78, 79].

The ground structure “level” reported for each example is based on the definition proposed by Zegard and Paulino [15]. As such, the ground structure is generated on a base mesh in which neighboring nodes are defined as nodes that belong to the same element in the base mesh. Then a level 1 ground structure contains connectivity between all neighboring nodes, level 2 contains connectivity up to the neighbors of the neighbors, level 3 contains connectivity up to the neighbors of the neighbors of the neighbors, and so on. A full-level ground structure contains connectivity between all nodes in the base mesh. In all examples, the longer of two overlapping members in the initial ground structure is not considered. Line thicknesses in the topology plots indicate the diameter of the member normalized to the maximum member diameter, assuming a circular cross-section. Note that, in general,

Table 2.2: Optimization input parameters used for the clamped beam

volume limit, V^{\max}	$2.000 \times 10^{-2} \text{ m}^3$
initial area, A_0	$3.198 \times 10^{-6} \text{ m}^2$
maximum area, A_i^{\max}	$1.600 \times 10^{-3} \text{ m}^2$
move parameter, β	1.000×10^3
convergence tolerance, tol	1.000×10^{-9}
objective tolerance, f^{tol}	1.000×10^{-2}
damped Newton parameter, η	1.000×10^{-8}

line thicknesses cannot be compared between structures, unless indicated otherwise. In all presented results, blue and red indicate members in tension and compression, respectively.

2.8.1 Clamped beam with equal and opposite point loads

Here, a clamped beam subjected to equal and opposite compressive point loads on the top and bottom faces at mid-span of the beam is considered. A full-level ground structure consisting of 251 members with Young’s modulus, $E_i = 7.5 \text{ GPa}$ for all i , is generated based on a 6×3 orthogonal base mesh. The ground structure and boundary conditions are provided in Figure 2.7a. In this example, an end filter with $f^{\text{tol}} = 0.01$ is used to remove thin members remaining at convergence. All other optimization parameters used for this problem are provided in Table 2.2.

Considering small deformation kinematics and a linear material model, the expected solution, shown in Figure 2.7b, is a single vertical member in compression, self-equilibrated by the two equal and opposite loads, where members connected by aligned nodes have been replaced by a single long member (see [70]). In this case, the magnitude of the load P is irrelevant to obtaining the optimal design in Figure 2.7b.

Three different optimal designs obtained considering large deformation kinematics with the cable material model are provided in Figure 2.7c, d, and e. Specific sizing and deformed shapes for $P = 10 \text{ kN}$, $P = 1000 \text{ kN}$, and $P = 2000 \text{ kN}$ are shown, but the critical load that causes a transition between the topologies in Figure 2.7c and d, and Figure 2.7d and e is about 400 kN and 1655 kN, respectively. In this case, the topology and sizing of the stiffest structure is dependent on the magnitude of the load. Note also that all of the members in these designs are in tension, as clearly indicated in the plots of the deformed shapes (right side of Figure 2.7c, d, and e). The maximum strains and stresses for the three (fully-stressed) designs are provided in Table 2.3.

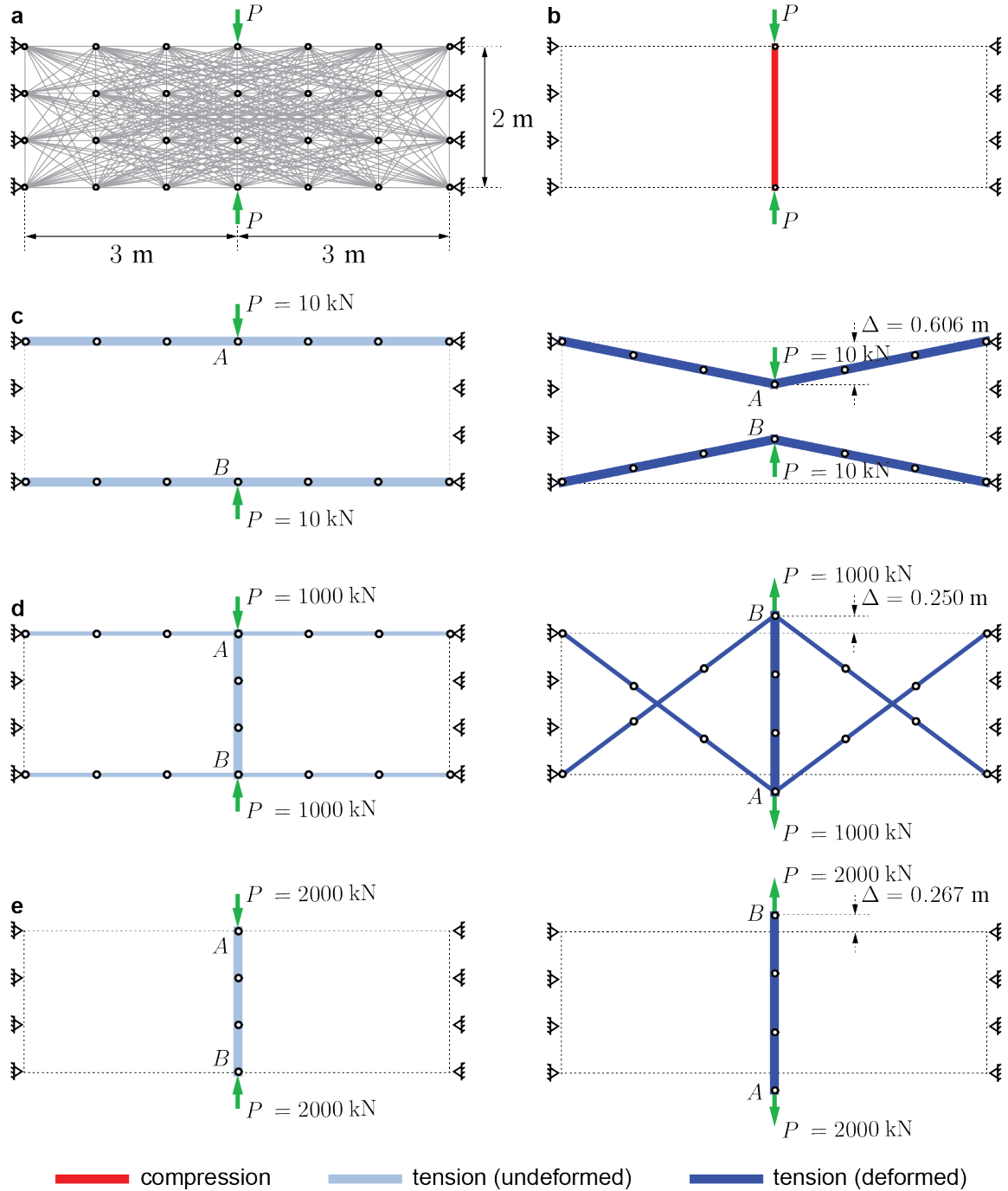


Figure 2.7: Clamped beam with equal and opposite compressive point loads: (a) Domain, ground structure, and boundary conditions; (b) optimal design considering small deformation kinematics and linear material model (aligned nodes are removed in post-processing); undeformed (left) and deformed (right) topologies considering large deformation kinematics and the cable material model, with (c) $P = 10$ kN, (d) $P = 1000$ kN, and (e) $P = 2000$ kN.

Table 2.3: Clamped beam maximum strains and member stresses for varying applied load magnitudes

load (kN)	maximum strain (%)	member stress (MPa) ¹
10	2.02	152
1000	25	1875
2000	26.6	2000

¹ All three structures are fully stressed.

Table 2.4: Optimization input parameters used for the pin-supported beam

volume limit, V^{\max}	$2.000 \times 10^{-3} \text{ m}^3$
initial area, A_0	$2.888 \times 10^{-6} \text{ m}^2$
move parameter, β	1.000×10^3
convergence tolerance, tol	1.000×10^{-9}
objective tolerance, f^{tol}	1.000×10^{-2}
damped Newton parameter, η	1.000×10^{-8}

Note that depending on the load magnitude, a topology identical to that obtained using small deformation kinematics and a linear material model can be obtained (compare the topologies in Figure 2.7b and e). *The difference is that in the case of finite deformations with the cable material model, a large configuration change is needed to obtain an equilibrium position. Notice that in the deformed shape, the location of loaded nodes A and B flip so that tension is induced in the structure.*

2.8.2 Pin-supported beam with mid-span load

In this example, a pin-supported beam with mid-span point load (100 kN) at the top surface is studied. The domain, boundary conditions, and full-level ground structure consisting of 251 members with Young’s modulus, $E_i = 170 \text{ GPa}$ for all i , are provided in Figure 2.8a. Here, solutions based on two different design variable upper bounds, $A_i^{\max} = 1.444 \times 10^{-3} \text{ m}^2$ and $A_i^{\max} = 1.155 \times 10^{-4} \text{ m}^2$ for all i , are investigated. An end filter with $f^{\text{tol}} = 0.01$ is used to remove thin members remaining at convergence. All other optimization parameters used for this problem are provided in Table 2.4.

The solution considering small deformation kinematics and a linear material model is provided in Figure 2.8b, where members connected by aligned nodes have been replaced by a single long member (see [70]). The results considering large deformation kinematics and the cable material model with load $P = 100 \text{ kN}$ are provided in Figure 2.8c and d

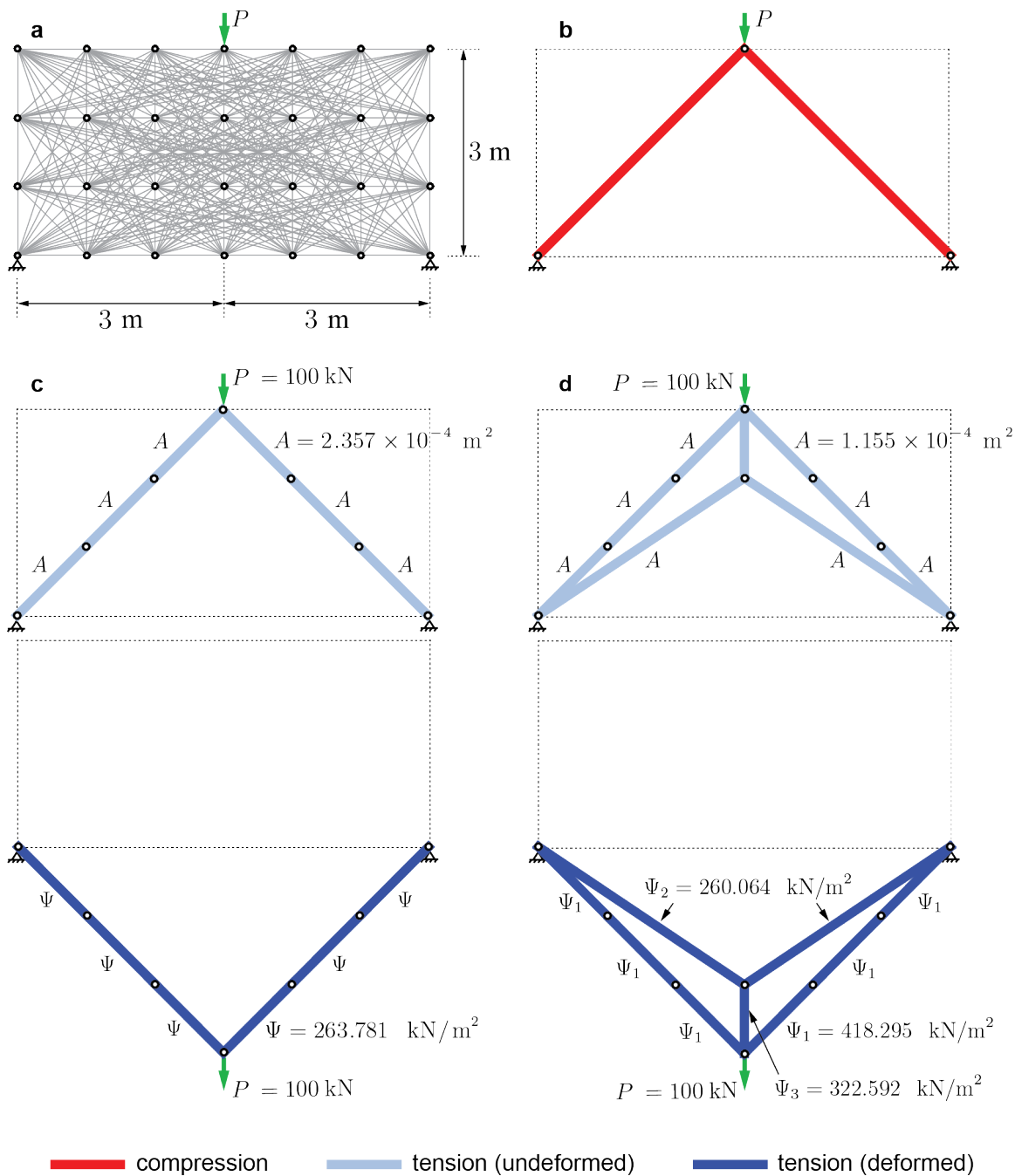


Figure 2.8: Pin-supported beam with mid-span point load: (a) Domain, ground structure, and boundary conditions; (b) optimal design considering small deformation kinematics and linear material model (aligned nodes are removed in post-processing); undeformed (top) and deformed (bottom) topologies considering large deformation kinematics and the cable material model, with (c) $A_i^{\max} = 1.444 \times 10^{-3}\text{ m}^2$ for all i and (d) $A_i^{\max} = 1.155 \times 10^{-4}\text{ m}^2$ for all i .

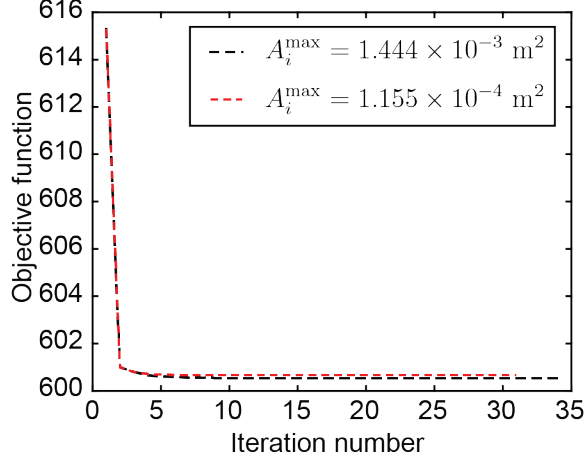


Figure 2.9: Convergence of the objective function for the pin-supported beam considering large deformation kinematics and the cable material model for the two different design variable upper bounds considered.

for $A_i^{\max} = 1.444 \times 10^{-3} \text{ m}^2$ and $A_i^{\max} = 1.155 \times 10^{-4} \text{ m}^2$, respectively, for all i . Notice that the result in Figure 2.8c has the same topology as that obtained from a linear model; however, the internal forces are tension forces rather than compression forces. Also note that the cross-sectional area of each member in Figure 2.8c is below the upper bound and, in agreement with the optimality conditions in Equation 2.37, all members have the same strain energy (fully stressed). In contrast, when the upper bound is reduced to $A_i^{\max} = 1.155 \times 10^{-4} \text{ m}^2$, a different topology, shown in Figure 2.8d, is obtained and the area of each member coincides with the upper bound, leading to a design without constant strain energy (non-fully stressed). Convergence plots for both cases are provided in Figure 2.9.

2.8.3 Tangentially loaded donut

Here, the centrally supported, tangentially loaded, donut shaped domain available with download of GRAND and shown in Figure 2.10a is investigated. Also shown in Figure 2.10a is the base mesh used by GRAND to generate a level 4 ground structure composed of 69,400 members [15]. All members are assigned a Young's modulus of $E_i = 170 \text{ GPa}$ for all i . As in the previous examples, an end filter with $f^{\text{tol}} = 0.01$ is used to remove thin members remaining at convergence. All other optimization parameters are provided in Table 2.5.

Considering small deformation kinematics and the linear material model, an assembly of five structures resembling Michell's solution for a cantilever with circular support are expected [15]. This solution is repeated here in Figure 2.10b with the tension and compression

Table 2.5: Optimization input parameters used for the tangentially loaded donut

volume limit, V^{\max}	$2.000 \times 10^{-3} \text{ m}^3$
initial area, A_0	$2.157 \times 10^{-7} \text{ m}^2$
maximum area, A_i^{\max}	$2.157 \times 10^{-3} \text{ m}^2$
move parameter, β	1.000×10^4
convergence tolerance, tol	1.000×10^{-9}
objective tolerance, f^{tol}	1.000×10^{-2}
damped Newton parameter, η	1.000×10^{-8}

members indicated in blue and red, respectively.

Considering large deformation kinematics and the cable material model with load $P = 100 \text{ kN}$, the structure in Figure 2.10c is obtained, in which all members are in tension. Note that in the deformed configuration, the members making up each tension strand become collinear with the direction of the load, putting the structure in equilibrium. In this case, the configuration found considering finite deformations with the cable material model is different from and totally meaningless for the case of small deformation kinematics. The objective function plotted in Figure 2.11 shows smooth convergence.

2.8.4 Spider web inspired cable net

Inspired by the elegant and efficient cable systems found in nature, a spider web inspired cable net is designed using the proposed formulation. The stress-strain behavior of radial (dragline) and spiral (viscid) spider silk have been empirically parameterized [80]. Although spider silk mechanical properties are dependent on a variety of factors (e.g., type of spider, type of silk, spinning conditions), the spiral (viscid) silk reported by [80] seems to fit well with the constitutive model proposed here as it reaches strains of around 250% and has a stress-strain response with positive curvature (i.e., it satisfies the convexity requirement discussed in subsection 2.6.2).

The goal of this example is not to exactly match the material properties and boundary conditions of a real spider web, but instead, to use the general characteristics of spider webs as inspiration for conceptual design of a cable net. Noting that the design depends on the ratio of the applied load to the stiffness of the system, dimensionless parameters are considered in the design and the effect of varying the design variable upper-bound, A_i^{\max} , for a given volume limit, V^{\max} , is investigated.

With an orb-web in mind, simplified boundary conditions on a 2D circular domain of

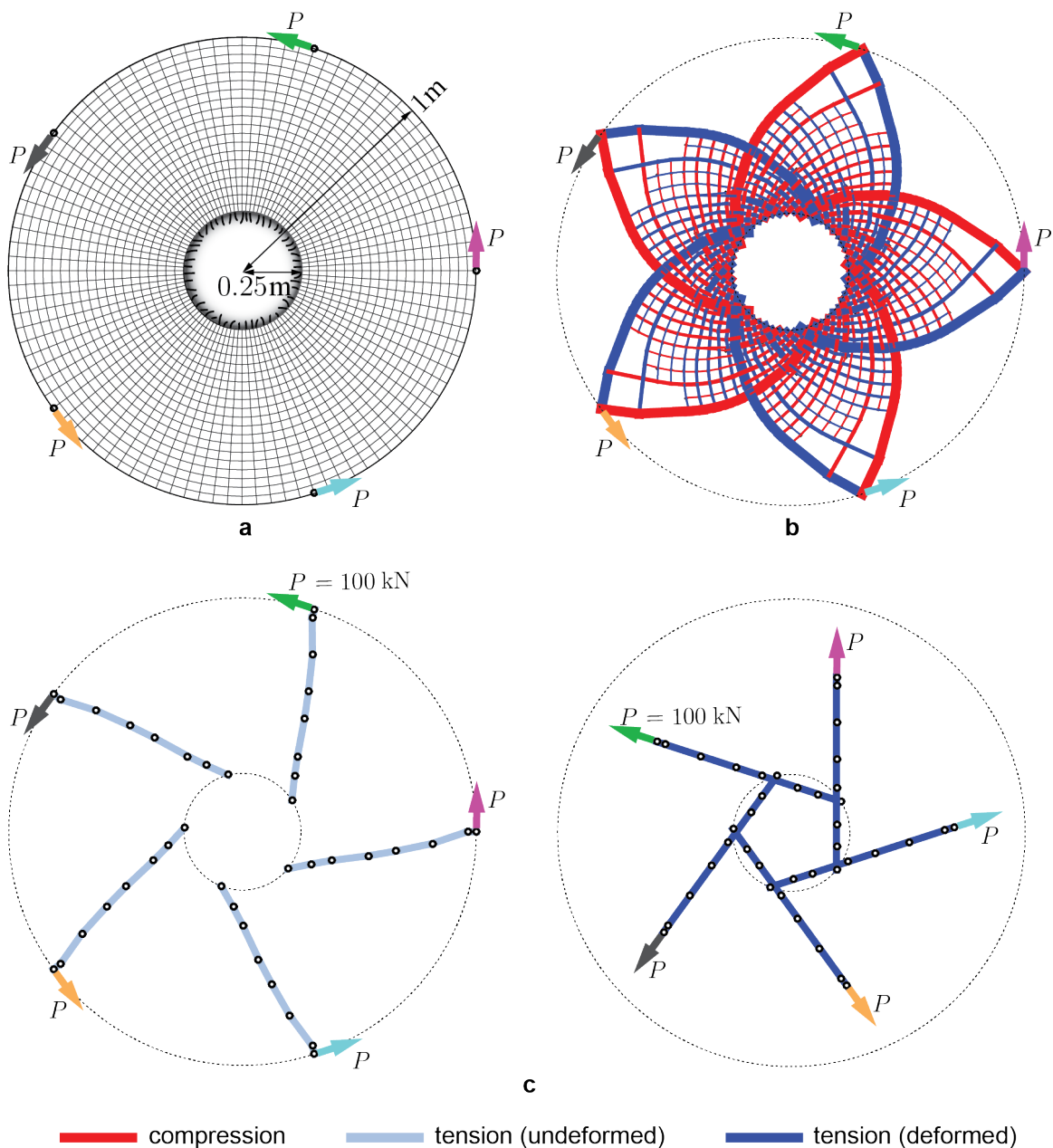


Figure 2.10: Centrally supported, tangentially loaded donut: (a) Domain, boundary conditions, and mesh used to generate the level four ground structure; (b) optimal design considering small deformation kinematics and linear material model (nodes omitted for clarity); and (c) undeformed (left) and deformed (right) topologies considering large deformation kinematics and the cable material model.

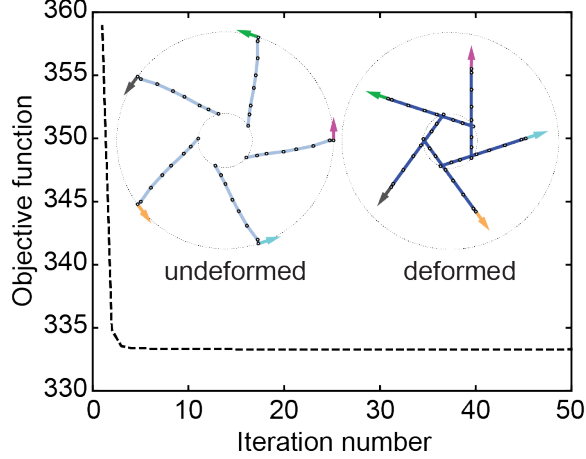


Figure 2.11: Convergence of the objective function for the tangentially loaded donut considering large deformation kinematics and the cable material model.

Table 2.6: Optimization input parameters used for spider web inspired cable net

volume limit, V^{\max}	0.01π
initial area, A_0	1.223×10^{-5}
move parameter, β	1.000×10^2
convergence tolerance, tol	1.000×10^{-9}
objective tolerance, f^{tol}	1.000×10^{-3}
damped Newton parameter, η	1.000×10^{-8}

radius, $r_o = 1$, are considered. Fully fixed supports are placed at 8 equally spaced locations around the circumference and a single out-of-plane point load of magnitude 1 is applied at the center of the domain. GRAND [15] is used to generate a level 3 ground structure on a polar grid with 16 circumferential and 16 radial divisions. Inside a central hole of radius, $r_i = 0.2$, only radial members are allowed, i.e., a restriction zone is defined in GRAND such that additional members are not generated in that region. The initial ground structure contains 5056 members. A summary of the domain, boundary conditions, base mesh, and initial ground structure is provided in Figure 2.12. The volume limit is defined to be 1% of the total in-plane area of the design domain (i.e., $V^{\max} = 0.01\pi r^2$). Additionally, all members are assigned a Young's modulus of $E_i = 1000$ for all i . An end filter with $f^{\text{tol}} = 0.001$ is used to remove thin members remaining at convergence. All other optimization parameters are provided in Table 2.6.

Results considering large deformation kinematics and the cable material model are provided in Figure 2.13a and b considering design variable upper bounds, $A_i^{\max} = 0.005$ and

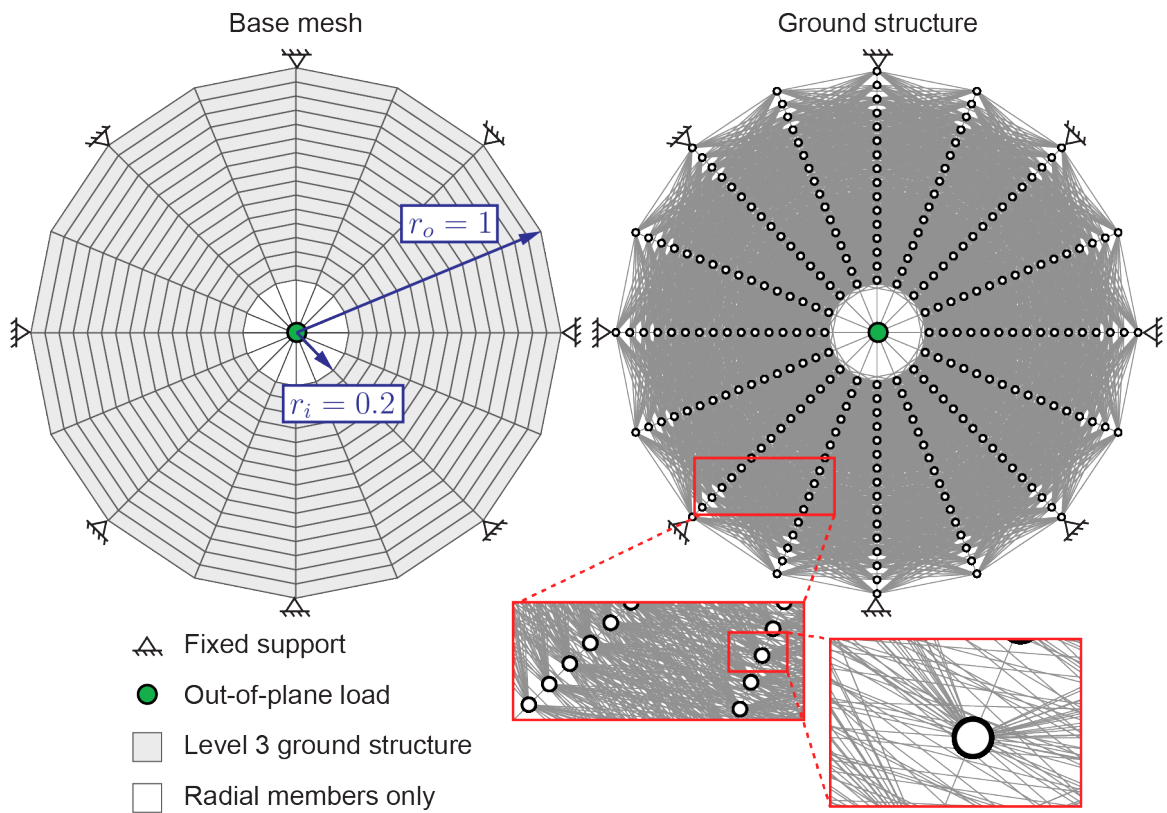


Figure 2.12: Domain, boundary conditions, base mesh, and initial ground structure used for the spider web inspired cable net.

0.0012, respectively, for all i . In Figure 2.13a, the design variable upper bound is not active and in the very simple final design containing 8 radial members with cross-sectional area, $A_i = 0.0039$ for all i , all elements have the same strain energy (fully-stressed). In the other case, the design variable upper bound is active such that some members have cross-sectional area, $A_i = A_i^{\max} = 0.0012$, and additional radial and “circumferential” members are included in the final design to add additional stiffness. A side view showing the deflection of each design confirms that the case with inactive upper bound is stiffer. It is also interesting to note that although the initial ground structure contains many crossing members, the optimal solution prefers radial and nearly circumferential members. In fact, the design in Figure 2.13b is reminiscent of an orb-web with only radial and spiral members (Figure 2.14a). Further, the design clearly prefers higher stiffness for the radial members, which mimics the distinction in real spider webs between the radial (dragline) silk, which is a few orders of magnitude stiffer than the spiral (viscid) silk [80].

Unlike the designs in Figure 2.13, spider webs in nature are imperfect. Notice in Figure 2.14a that neither the radial nor circumferential strands are equally spaced and some of the circumferential members even intersect each other. In an effort to achieve a spider web inspired cable net that is imperfect, like those found in nature, the spider web is redesigned using an initial ground structure with perturbed nodal positions. Starting with the same nodal mesh used to define the initial ground structure of the previous spider web designs (Figure 2.12), the spacing of the radial members is modified by randomly perturbing each of the 16 sets of radial nodes (each radial set of nodes is defined by the same polar angle) by a uniformly distributed random perturbation in the range $[-7.2^\circ, 7.2^\circ]$. Additionally, 1% of the nodes (randomly selected) are perturbed in the radial direction by a uniformly distributed random perturbation in the range $[-0.005 \text{ units}, 0.005 \text{ units}]$. The level 3 ground structure containing only radial members within a radius of $r_i = 0.2$ is regenerated on this perturbed nodal mesh. This time, due to an increased number of overlapping members, the initial ground structure contains a total of 5045 members. The imperfect spider web, designed considering all the same input parameters as the previous spider web designs (Table 2.6) and $A_i^{\max} = 0.0012$, is provided in Figure 2.14c.

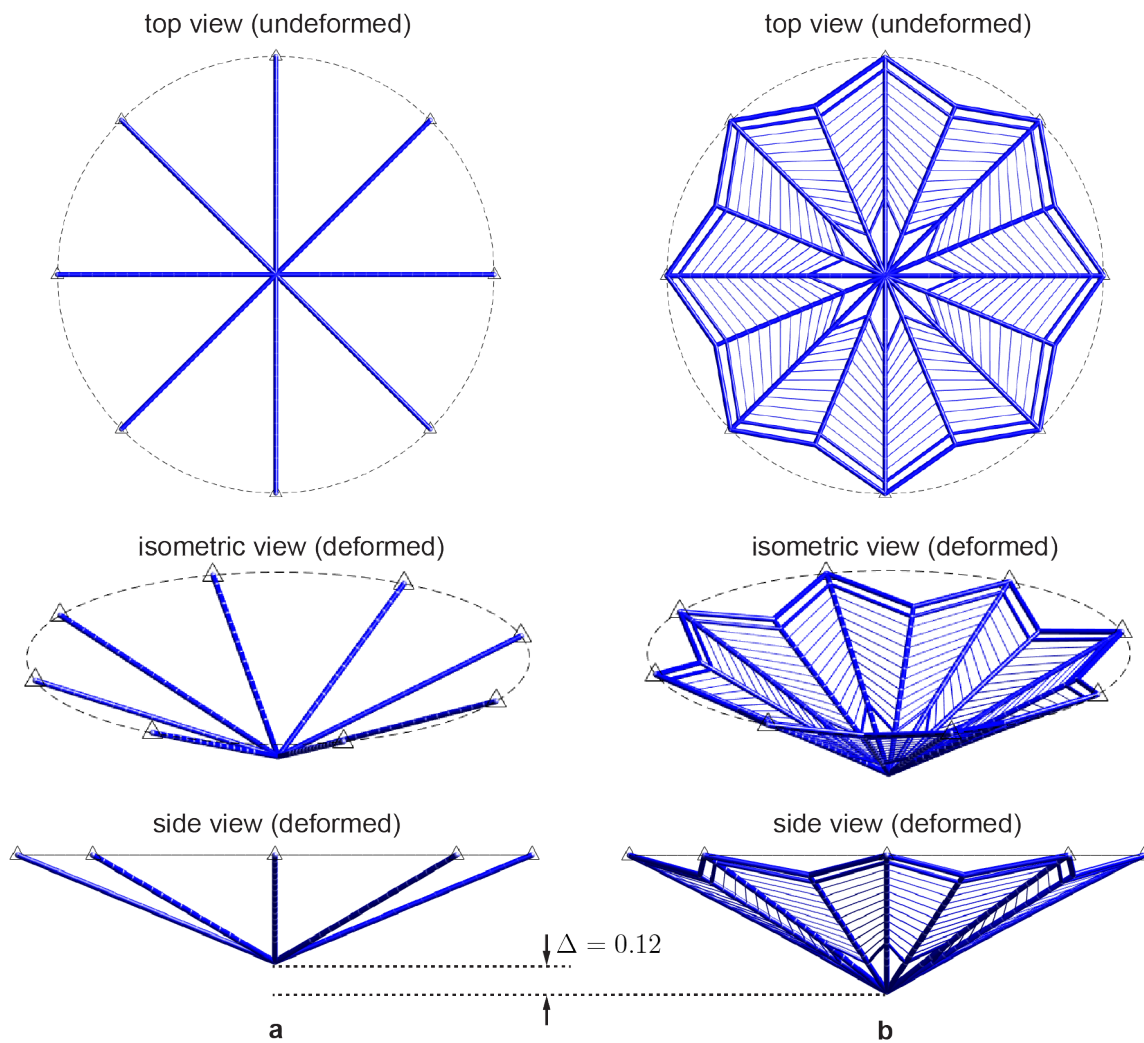


Figure 2.13: Spider web inspired cable net: optimal design considering large deformation kinematics, the cable material model, and the unperturbed nodal mesh with (a) $A_i^{\max} = 0.005$ for all i and (b) $A_i^{\max} = 0.0012$ for all i .

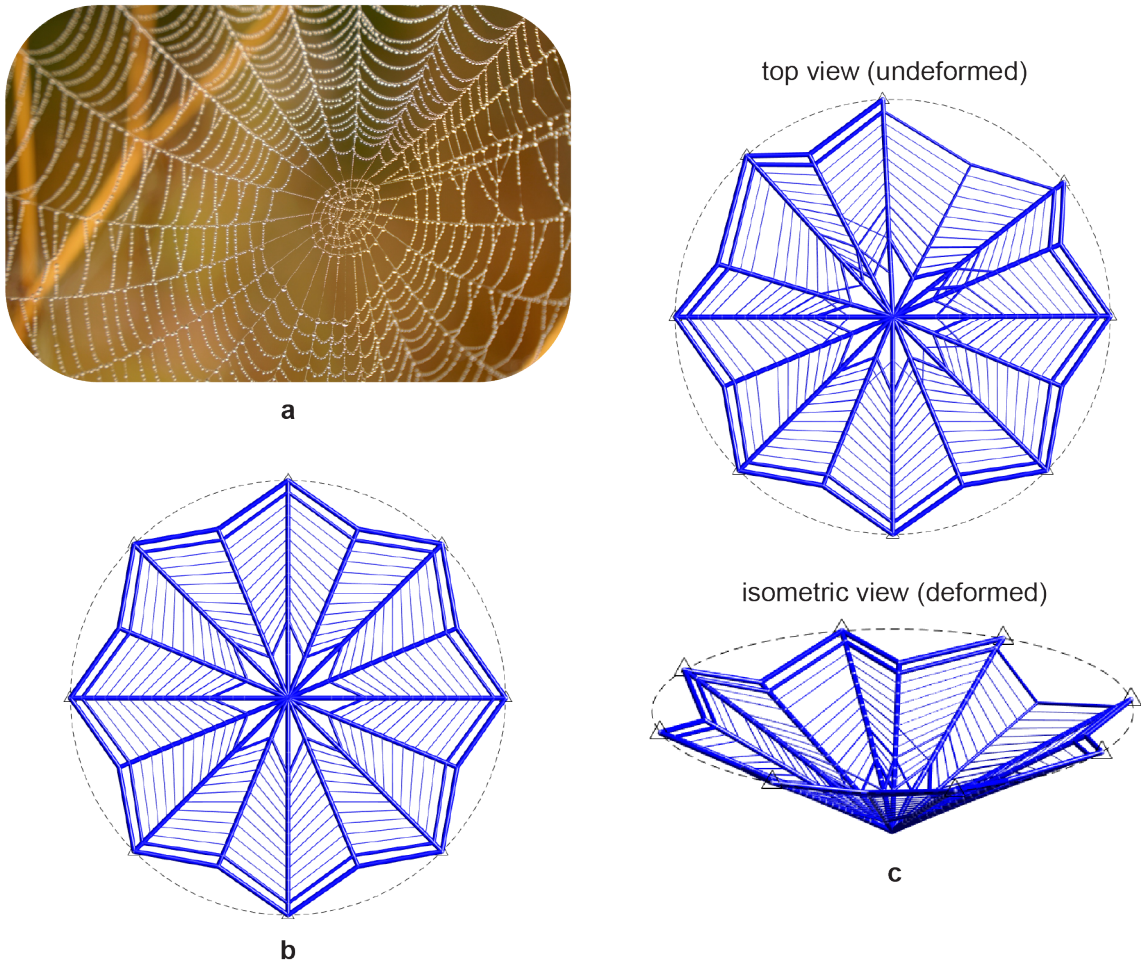


Figure 2.14: Comparison of (a) an orb spider web found in nature [image by Miriam Müller from Pixabay] and the spider web inspired cable nets designed using topology optimization on a (b) unperturbed nodal mesh and (c) perturbed nodal mesh.

CHAPTER 3

OPTIMIZED LATTICE-BASED METAMATERIALS FOR ELASTOSTATIC CLOAKING

Holes adversely affect the stress distribution within a structure, often requiring reinforcement to avoid failure [81]; however, holes typically serve an essential function. For example, windows provide lighting, cavities provide storage, and tunnels enable travel and transport. By eliminating the effect of holes on the elastostatic response of a system, such functionalities can be maintained without compromising structural integrity. Elastostatic cloaking is used to manipulate the displacement, strain, and stress fields induced from static boundary conditions, such that a hole, defect, or object becomes undetectable within an elastic medium. In this chapter, optimized cloaking devices that hide the effect of a hole on the elastostatic response of lattice systems are pursued in a discrete topology optimization setting composed of 2D (beam) elements by *relaxing* the material space to accommodate a continuous range of stiffness and formulating a convex topology optimization problem that promotes global stiffness matching between the cloaked and undisturbed systems.

3.1 Overview and related work

To cloak an object is to make it invisible with respect to some physically observable field. For instance, classical cloaking problems aim to re-direct electromagnetic waves around an object such that the electromagnetic field is undisturbed and the object is effectively made invisible [82, 83]. Relying on form-invariance of the underlying (hyperbolic) equations, cloaking with respect to physically observable fields such as electromagnetic waves [82–84], acoustic-induced fluid waves [85, 86], and quantum mechanical matter waves [87, 88] have traditionally been pursued by applying a coordinate transformation to a given material-parameter distribution and then solving an inverse problem to find a microstructure exhibiting such transformed distribution. Form-invariance has also been exploited to design cloaks in fields governed by time-dependent parabolic equations (e.g., thermodynamics, electrical conduction, and particle diffusion) [89–91], and under special conditions, for fields governed by the elastodynamic wave equation [92–96].

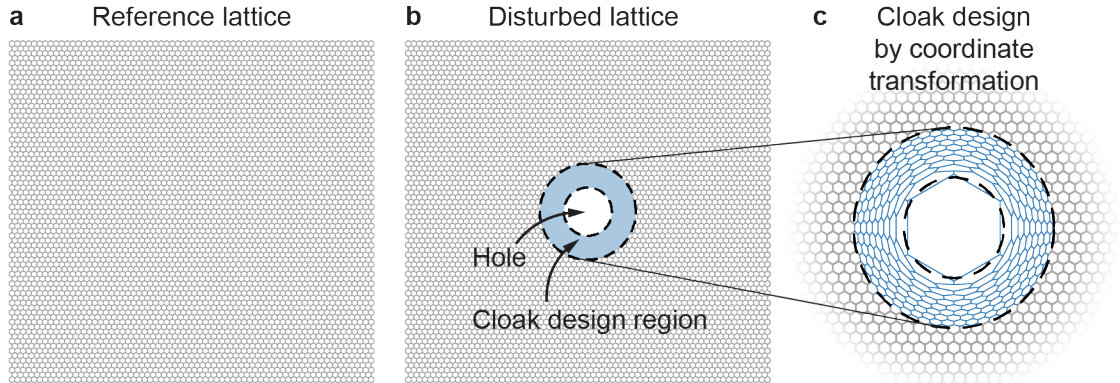


Figure 3.1: Elastostatic cloaking in 2D lattices: (a) Reference lattice; (b) lattice with a prescribed circular hole surrounded by a region in which a cloak should be designed; (c) lattice in which the cloak geometry is defined by a coordinate transformation of the reference lattice nodes.

In contrast to the problems listed above, in which the cloak is used to manipulate wave propagation, elastostatic cloaking is used to manipulate the displacement, strain, and stress fields induced from static boundary conditions, such that a hole, defect, or object becomes undetectable within an elastic medium (see Figure 3.1). The elliptic equations governing elastostatics are not form-invariant under coordinate transformations [92]; and thus, far fewer attempts have been made to achieve elastostatic cloaking [97, 98]. Topology optimization has been pursued for elastostatic cloak design [99, 100], but a formulation in the discrete setting that is capable of achieving multi-directional elastostatic cloaking devices that mitigate the influence of circular or elliptical holes has not been fully explored.

Given a 2D reference lattice (Figure 3.1a), the goal of elastostatic cloaking is to design a cloak (blue design region in Figure 3.1b) around a prescribed defect (hole) so that, under static boundary conditions, the displacement field outside of the cloak matches that of the reference lattice. To accomplish this goal, Bückmann et al. [98] proposed a direct lattice transformation (DLT) approach in which the geometry of the reference lattice nodes are modified to avoid the hole via a coordinate transformation (Figure 3.1c) and the local stiffness of the connecting elements are held constant by choosing the cross-sectional properties appropriately. For problems like electrical or thermal conductivity, in which each lattice element can be described by a single, scalar parameter (i.e., electrical or thermal resistance), a cloak designed via DLT will exhibit perfect cloaking (in a vacuum) as long as the cross-sectional properties needed to achieve the required resistance of the transformed elements are physically viable [90]. While they acknowledge that the DLT idea may not be directly

transferable to cloaking in elastic lattices in which multiple, independent scalars are needed to describe the mechanical behavior of each lattice element, Bückmann et al. [98] design the cross-sectional properties of the transformed lattice elements to preserve their axial stiffness. They demonstrate that the approach leads to approximate elastostatic cloaks that perform well for circular holes of varying size, subjected to shear loading and uniaxial compression along two of the lattice’s lines of symmetry, with and without lateral support.

It is hypothesized that elastostatic cloaking in 2D lattices can be improved beyond that achieved via the DLT approach. Consider approximating the lattice’s mechanical behavior with a network of Euler-Bernoulli beam elements that also resist axial force. For simplicity, assume that the local stiffness of each lattice element can be mapped exactly from the reference lattice to the transformed lattice (i.e., cross-sectional properties can be found that keep the local stiffness of each lattice element constant in the presence of a change in length). Note that the contribution of each lattice element to the global behavior of the lattice is dependent not only on its local stiffness, but also on its orientation in the network [101]. Thus, even if it were possible to perfectly match the element local stiffnesses to the reference, the global stiffness properties of the transformed lattice would still deviate from those of the reference. Thus, rather than preserving local properties, *global system properties are targeted by designing the nodal connectivity and associated stiffness characteristics in the design region using topology optimization.*

3.2 Problem setting

Elastostatic cloaking in 2D lattices is pursued in a discrete topology optimization setting where the lattice elements are modeled as Euler-Bernoulli beam elements that can also take axial force (see Figure 3.2). The ideas are demonstrated using a hexagonal lattice consisting of elements with rectangular cross-section, but the formulation is not specific to any particular lattice geometry or cross-sectional shape. The height, h , thickness, t , and Young’s modulus, E , is the same for all elements in the design lattice; L_ℓ is the length of element ℓ , which may vary in the design region.

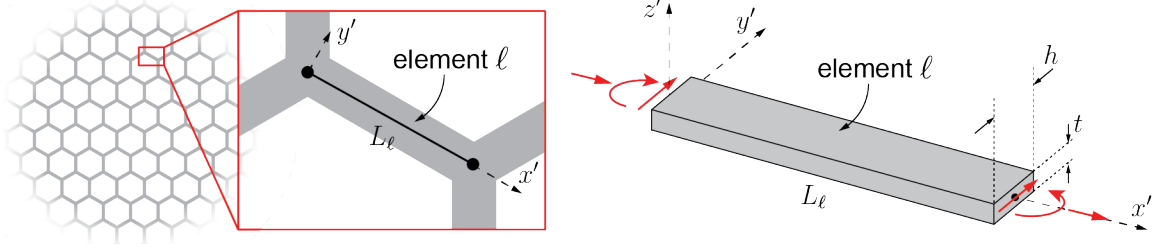


Figure 3.2: Lattice model: The lattice elements are modeled as Euler-Bernoulli beam elements that also take axial force. The length, height, and thickness of element ℓ are L_ℓ , h , and t , respectively. In the reference lattice, all elements have the same length, but the length of the lattice elements varies in the design region of the design lattice.

3.2.1 Target problem

The specific problem pursued in this chapter is outlined in Figure 3.3. The reference lattice contains $N_x \times N_y$ regular hexagons with edge length, L , such that the lattice has width, L_x , and height, L_y . The design lattice is identical to the reference lattice except that an elliptical hole of radius, $R_1(\theta)$, is introduced and enclosed by a circular design region of radius, R_2 .

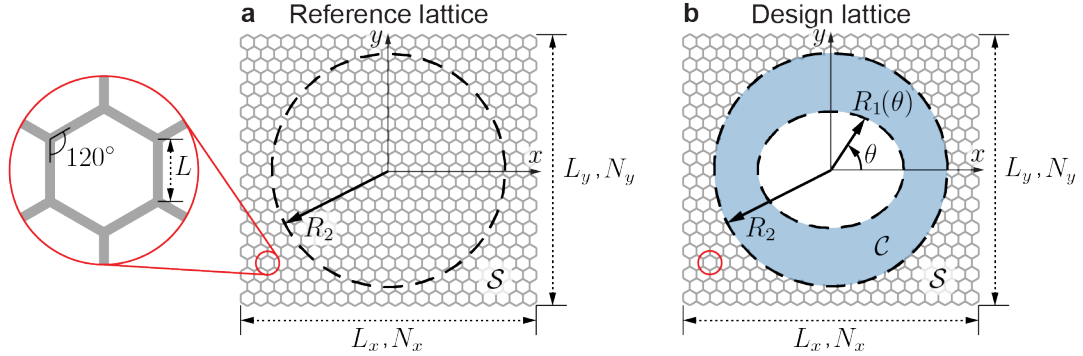


Figure 3.3: Target problem definition: (a) Reference lattice; (b) design lattice.

Stiffness of the lattice elements are scaled by stiffness parameters, $\mathbf{z} = \{z_\ell\}_{\ell=1}^{N_e}$ and $\widehat{\mathbf{z}} = \{\widehat{z}_\ell\}_{\ell=1}^{\widehat{N}_e}$, for the design and reference lattices, respectively, where N_e is the number of elements in the design lattice and \widehat{N}_e is the number of elements in the reference lattice. The element stiffness matrix of element ℓ is $\mathbf{k}_\ell = z_\ell \mathbf{k}_\ell^0(E, t, h, L_\ell)$ and $\widehat{\mathbf{k}}_\ell = \widehat{z}_\ell \widehat{\mathbf{k}}_\ell^0(E, t, h, L)$, for the design and reference lattices, respectively, where \mathbf{k}_ℓ^0 and $\widehat{\mathbf{k}}_\ell^0$, are the unscaled, constant element stiffness matrices (in global coordinates) of element ℓ (refer to McGuire et al. [101]). The set of stiffness parameters for elements with radial coordinates of at least one end node larger than R_2 are members of the surrounding region, \mathcal{S} , and are identical in the

two lattices, i.e., $z_\ell = \widehat{z}_\ell \forall \ell \in \mathcal{S}$. The set of stiffness parameters for elements with radial coordinates of both end nodes between $R_1(\theta)$ and R_2 are members of the design region, \mathcal{C} , and will be used as design variables in optimization.¹

The lattice displacement fields are defined by nodal displacements, $\mathbf{u} = \{u_\ell\}_{\ell=1}^{N_d}$ and $\widehat{\mathbf{u}} = \{\widehat{u}_\ell\}_{\ell=1}^{N_d}$, for the design and reference lattices, respectively, where N_d is the number of degrees of freedom in both the design and reference lattices. The nodal displacements are determined by solving the linear systems, $\mathbf{K}\mathbf{u} = \mathbf{F}$ and $\widehat{\mathbf{K}}\widehat{\mathbf{u}} = \mathbf{F}$, which approximate the elastic response of the design and reference lattices, respectively. The global stiffness matrices, \mathbf{K} and $\widehat{\mathbf{K}}$, of the design and reference lattices are assembled from the element stiffness matrices of each lattice and \mathbf{F} is the vector of applied nodal loads that is the same for both lattices. The set of nodal displacements for nodes with radial coordinates, $r > R_2$, are members of the space of nodal displacements in the surrounding region, \mathcal{S}^u and $\widehat{\mathcal{S}}^u$, for the design and reference lattices, respectively.

With the goal of matching the elastostatic response of the design and reference lattices for nodal coordinates with radius, $r > R_2$, a cloaking metric,

$$\Delta = \frac{\|\mathbf{u}^s - \widehat{\mathbf{u}}^s\|_2}{\|\widehat{\mathbf{u}}^s\|_2}, \quad (3.1)$$

is defined to quantify the cloak's effectiveness [98], where $\mathbf{u}^s = \{u_\ell\}_{\ell \in \mathcal{S}^u}$ and $\widehat{\mathbf{u}}^s = \{\widehat{u}_\ell\}_{\ell \in \widehat{\mathcal{S}}^u}$.

3.2.2 Defining the design space

In a similar spirit to the DLT approach, the design space is defined by first applying a coordinate transformation to the reference lattice's nodal positions such that they avoid the hole, but remain unchanged outside of the design region (see Figure 3.4a and b). The particular coordinate transformation considered here is illustrated in Figure 3.4b. It moves points with radial coordinate, $\widehat{r} < R_1(\theta)$, in the reference lattice, to $R_1(\theta) < r < R_2$ in the design lattice, where $R_1(\theta)$ and R_2 define the inner and outer radial coordinates of the elliptical hole and circular cloak along a ray from the origin and passing through \widehat{r} (see

¹ The choice of a linear element stiffness scale factors for design variables can be interpreted as a material parameter that scales the Young's modulus to achieve a multi-material design, or as a sizing parameter that scales the element thickness to achieve a variable thickness design. The stiffness matrix is a linear function of both the Young's modulus and element thickness (for the case of a rectangular cross-section); other nonlinear design parameters could be targeted, but such choices will have implications on the convexity of the optimization formulation as discussed in subsection 3.3.2.

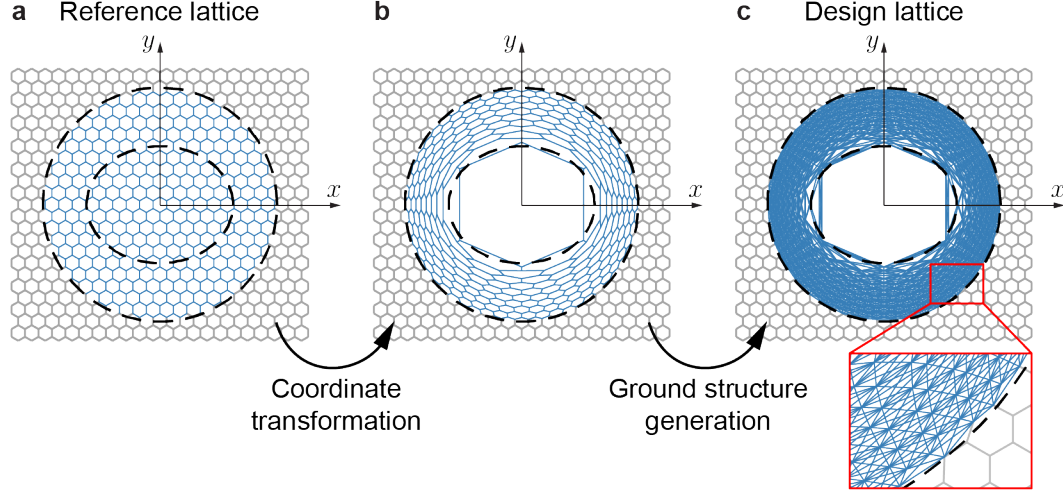


Figure 3.4: Defining the design space for optimization-based elastostatic cloak design: The reference lattice’s nodal positions in (a) are modified in (b) to avoid a hole using a coordinate transformation that preserves the reference lattice outside of the design region. Subsequently, in (c), the design space is enriched by defining a highly redundant ground structure within the design region.

Figure 3.5). Such a coordinate transformation is described by

$$r = R_1(\theta) + \frac{R_2 - R_1(\theta)}{R_2} \hat{r}, \text{ where } R_1(\theta) = \frac{k_x k_y r_1}{\sqrt{k_y^2 \cos^2(\theta) + k_x^2 \sin^2(\theta)}}, \quad (3.2)$$

and k_x , k_y , and r_1 are used to define the semi-major and semi-minor axes of the elliptical hole [102]. Each node of the reference lattice with radial coordinate, $r < R_2$, is transformed according to Equation 3.2 to define the nodal positions of the design lattice.²

To enhance the ability to mimic the global stiffness properties of the reference lattice, the design space is further enriched by defining a ground structure that increases the connectivity in the design region (see Figure 3.4b and c). The ground structures considered here retain all connectivity of the reference lattice and are generated using the ground structure generation tools from the educational code, GRAND [15]. Note that the coordinate transformation stretches the central hexagon radially, and although it ensures that the lattice nodes of this central hexagon avoid the hole, the lattice elements associated with adjacent hexagons may overlap slightly with the hole. These elements are maintained in the design region, but no additional elements are generated within the central hexagon during ground structure generation. Additionally, all ground structures are defined to preserve the symmetries of the

² Other coordinate transformations, for example, a cubic coordinate transformation [103], can also be considered.

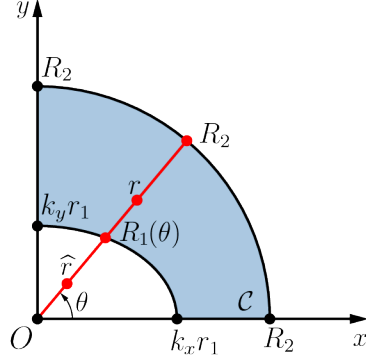


Figure 3.5: Illustration of the coordinate transformation used to map the reference lattice’s nodal positions into the design region in order to avoid the elliptical hole.

reference lattice, although the symmetries do not hold when the coordinate transformation is performed to avoid an elliptical hole (i.e., when $k_x \neq k_y$).

A naive way to select the ground structure is to add a connection between every node in the design region, excluding those that cross the central hexagon. Such a full-level ground structure, as defined by Zegard and Paulino [15], provides the most design freedom; however, a full-level ground structure is extremely dense and will likely lead to designs that are impractical for manufacturing. Instead, two ground structures that are less dense are considered:

1. GS1 is a ground structure constructed to be as dense as possible, without allowing any crossing members and with hexagonal symmetry preserved (with respect to the nodal coordinates of the reference lattice). Ground structure GS1 is shown in Figure 3.6a.
2. GS2 is a level 1 ground structure [15] defined to include connectivity between the nodes of each hexagon, but not from one hexagon to the next. This ground structure may still lead to designs that are difficult to manufacture because it is still relatively dense and contains crossing members. Nevertheless, it is considered in an effort to understand the role of increased design freedom. Ground structure GS2 is shown in Figure 3.6b.

A number of different ground structures can be defined to achieve the goals of GS1, but numerical experiment indicates that the one selected here is representative of this class of ground structures.

During ground structure generation, if an element centroid’s radial coordinate, $r > R_2$,

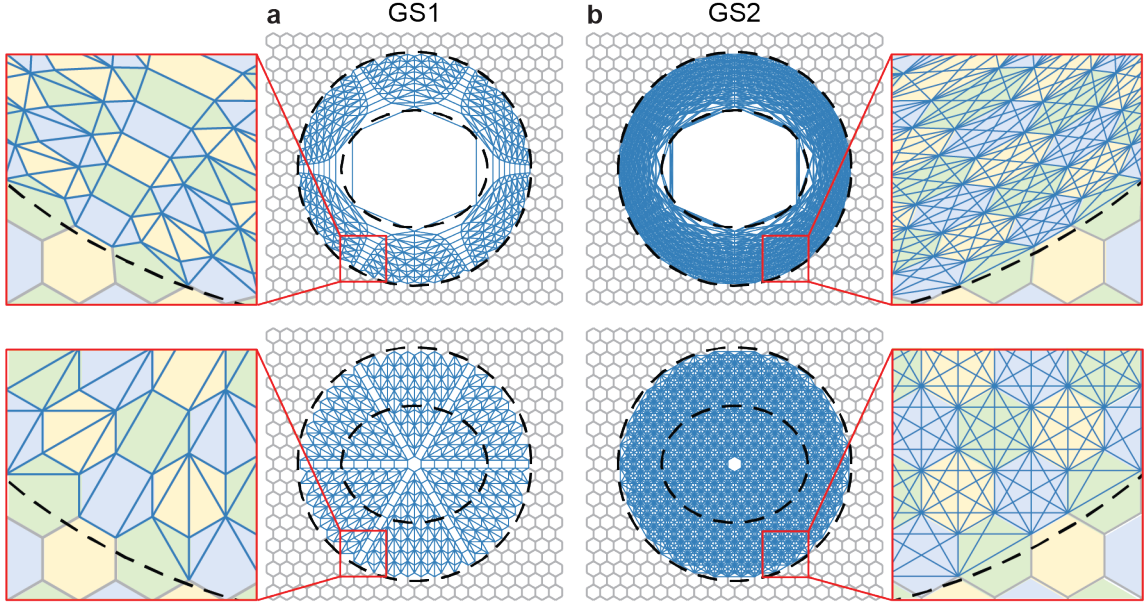


Figure 3.6: Two ground structures considered in optimization-based elastostatic cloak design: (a) GS1 is constructed to achieve the highest connectivity possible, while avoiding crossing members and preserving lattice symmetry with respect to the reference lattice nodal positions; (b) GS2 contains level 1 connectivity in which an element is generated between all nodes of each hexagon of the lattice, but not from one hexagon to the next. Connectivity of each ground structure with respect to the reference lattice nodal positions is shown in the second row. Hexagons in the insets are colored to help visualize the additional connectivity added with ground structure generation.

that element is removed from the ground structure. To more easily understand the connectivity of the two considered ground structures and to see details of the ground structure near the outer radius of the design region, it is illustrative to look at the ground structure connectivity considering the nodal positions of the reference lattice, as illustrated at the bottom of Figure 3.6 for the two considered ground structures. Here, the lattice symmetries also become apparent.

3.2.3 Defining the boundary conditions for optimization

An elastostatic cloak should perform well irrespective of the static boundary conditions imposed on the lattice. For example, a low value of Δ should be achieved whether the lattice is subjected to tension, compression, shear, or any other combination of static loads. By considering multiple load cases, not only will the design lattice be less biased toward a specific set of boundary conditions, but multiple load cases will likely also push the global stiffness characteristics of the design lattice closer to those of the reference lattice. For this

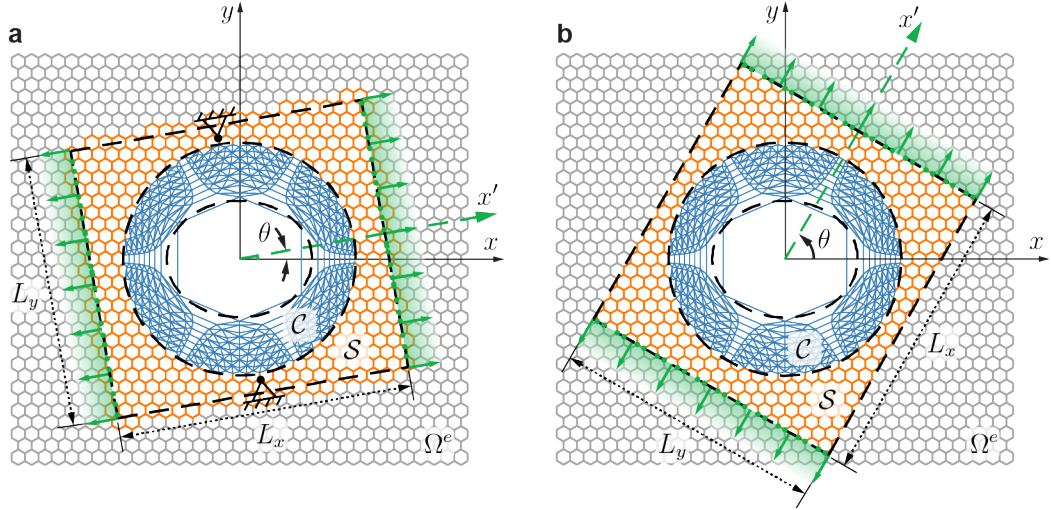


Figure 3.7: Design lattice embedded in extended lattice for (a) $\theta = 10^\circ$ load angle; (b) $\theta = 60^\circ$ load angle. The subset of lattice elements in the surrounding region, \mathcal{S} , depends on the load angle, but those in the design region, \mathcal{C} , are invariant with respect to the load angle.

purpose, various choices and combinations of boundary conditions can be pursued. Perhaps the simplest approach, and the one considered here, is to impose uniaxial tension in several directions to induce different types of loading into each lattice element.

To keep the lattice dimensions, L_x and L_y , constant for each load case, the lattice is embedded in an extended lattice, Ω^e , as shown in Figure 3.7. Depending on the load angle, θ , a different subset of the extended lattice becomes active. Active lattice elements are those shown in orange in the surrounding region, $\mathcal{S} \subseteq \Omega^e$, and those shown in blue in the design region, $\mathcal{C} \subseteq \Omega^e$. The set of lattice elements included in the surrounding region varies depending on the load case, but those in the design region are invariant with respect to the load case.

Lattice elements in the active set are defined as those with centroid falling within the rotated $L_x \times L_y$ bounding box shown with a black dashed line in Figure 3.7. Depending on the load angle, the bounding box may cut through elements of the extended lattice (refer to the case of $\theta = 10^\circ$ in Figure 3.7a). Rather than cutting these lattice elements, a few elements outside of the bounding box are included in the active set to avoid partial hexagons. Additionally, for cases in which the loaded edges of the bounding box do not coincide with existing nodes, additional nodes are inserted and loads are applied to these nodes considering the appropriate tributary width. For these cases, which correspond to tensile loads aligned

away from the reference lattice's lines of symmetry, the uniaxial loads may not be perfectly equal and opposite. To prevent singularities, two pin supports are added at the centerline of the L_x dimension, as indicated in Figure 3.7a. In Figure 3.7b, the load angle, $\theta = 60^\circ$, is aligned with a symmetry axis of the reference lattice; thus, no additional nodes are needed for applying the loads and no support conditions are included.³

3.3 Optimization formulation

Optimal design of elastostatic cloaking devices in 2D lattice systems is pursued using an unconstrained version of Equation 1.1 that takes the specific form,

$$\min_{\{z_\ell\}_{\ell \in \mathcal{C}} \in [0,1]^N} f(\mathbf{z}, \mathbf{u}_1(\mathbf{d}_1), \dots, \mathbf{u}_{N_\theta}(\mathbf{d}_{N_\theta})) = \sum_{i=1}^{N_\theta} (\hat{\mathbf{u}}_i - \mathbf{u}_i(\mathbf{d}_i))^T \mathbf{K}_i(\mathbf{d}_i) (\hat{\mathbf{u}}_i - \mathbf{u}_i(\mathbf{d}_i))$$

with $\mathbf{K}_i(\mathbf{d}_i) \mathbf{u}_i(\mathbf{d}_i) = \mathbf{F}_i(\theta_i)$, $i = 1, \dots, N_\theta$,

(3.3)

where N_θ load cases (directions) are considered in design. The stiffness parameters of the N_e lattice elements in the extended lattice are stored in $\mathbf{z} = \{z_\ell\}_{\ell=1}^{N_e}$. The subset of lattice elements outside of the design region, $\{z_\ell\}_{\ell \in \Omega^e \setminus \mathcal{C}}$, are assigned the same stiffness parameters as the reference lattice elements. The design variables, $\{z_\ell\}_{\ell \in \mathcal{C}} \in [0,1]^N$, are the subset of stiffness parameters corresponding to the N lattice elements in the design region. The subset of active elements for load case i are denoted, $\mathbf{d}_i = \{z_\ell\}_{\ell \in \mathcal{C} \cup \mathcal{S}_i}$, where \mathcal{S}_i is the space of lattice elements in the surrounding region for load case i . The stiffness matrix and displacement vector of the design lattice for load case i are $\mathbf{K}_i(\mathbf{d}_i)$ and $\mathbf{u}_i(\mathbf{d}_i)$, respectively. The displacement field, $\hat{\mathbf{u}}_i$, of the reference lattice is determined by the state equations of the reference lattice, $\hat{\mathbf{K}}_i \hat{\mathbf{u}}_i = \mathbf{F}_i(\theta_i)$, where $\hat{\mathbf{K}}_i$ is the stiffness matrix of the reference lattice for load case i . Applied nodal loads for load case i are stored in the vector, $\mathbf{F}_i(\theta_i)$.

The objective function, f , in Equation 3.3 is a weighted least squares function, where the weights are the design-dependent coefficients of the design lattice's stiffness matrix. Weighted least squares methods are often used when portions of the data should be prioritized more than others [105]. Since cloaking effectiveness is evaluated by only considering the displacements in the surrounding region through the cloaking metric, Δ , one choice of

³ The uniaxial tension loads considered here are self-equilibrated and Tikhonov regularization is used to prevent singularities in the associated system of equations [69, 104] as discussed in subsection 3.4.1.

weights could be one that nullifies terms related to displacements that are not in the surrounding region (i.e., an indicator matrix would replace the stiffness matrix in the objective function in Equation 3.3). Based on insights from cloaking in other fields [82, 83], however, it is clear that the properties of the design region play a critical role in achieving effective cloaking and the response in the design region should also be controlled.

By choosing the stiffness coefficients for weights as in Equation 3.3, the degrees of freedom in the surrounding region will always be prioritized since the stiffness contributions of the lattice elements in the surrounding region are finite (and fixed). The degrees of freedom in the design region will also be prioritized, likely to a greater extent than those in the surrounding region at the beginning of the optimization when connectivity is dense, and will likely lead to changes in the design variables that cause the global stiffness contribution at each degree of freedom to become closer to that contributing to the associated degree of freedom in the reference lattice. Sparsity of the stiffness matrix leads to many null terms in the least squares summation, i.e., terms of the form $(u_i - u_i) K_{ij} (u_j - u_j)$, $i \neq j$ when degrees of freedom i and j are not connected in the lattice system. These terms are likely not as critical in driving the global stiffness of the design lattice toward that of the reference.

Using the stiffness matrix coefficients as weights also endows the objective function with two desirable properties that will be elaborated on in the following two subsections: no solution of an additional adjoint system in computing its sensitivities is needed and the objective function is convex when the stiffness matrix is a linear function of the design variables.

3.3.1 Sensitivity analysis

The adjoint method is used to derive the sensitivities of the objective function, f , in Equation 3.3. For simplicity of notation, the dependence on $\mathbf{d}_i = \{z_\ell\}_{\ell \in \mathcal{C} \cup \mathcal{S}_i}$ is dropped in the following derivations.

The state equations of the design lattice, although implicitly enforced in solution of Equation 3.3, serve as partial differential equation (PDE) constraints. The Lagrangian of

the associated PDE-constrained optimization problem is

$$\mathcal{L}(\mathbf{z}, \mathbf{u}_1, \dots, \mathbf{u}_{N_\theta}, \boldsymbol{\Lambda}_1, \dots, \boldsymbol{\Lambda}_{N_\theta}) = \sum_{i=1}^{N_\theta} \left[(\hat{\mathbf{u}}_i - \mathbf{u}_i)^T \mathbf{K}_i (\hat{\mathbf{u}}_i - \mathbf{u}_i) + \boldsymbol{\Lambda}_i^T (\mathbf{K}_i \mathbf{u}_i - \mathbf{F}_i(\theta_i)) \right], \quad (3.4)$$

where $\boldsymbol{\Lambda}_i$ is the adjoint vector for load case i that can be chosen freely since the second term of the summation in Equation 3.4 vanishes due to equilibrium of the design lattice. Then, the derivative of the objective function with respect to each design variable is

$$\begin{aligned} \frac{\partial f}{\partial z_\ell} = \frac{\partial \mathcal{L}}{\partial z_\ell} = \sum_{i=1}^{N_\theta} \left[(\hat{\mathbf{u}}_i - \mathbf{u}_i)^T \frac{\partial \mathbf{K}_i}{\partial z_\ell} (\hat{\mathbf{u}}_i - \mathbf{u}_i) + \boldsymbol{\Lambda}_i^T \frac{\partial \mathbf{K}_i}{\partial z_\ell} \mathbf{u}_i \right. \\ \left. + (-2\mathbf{K}_i \hat{\mathbf{u}}_i + 2\mathbf{K}_i \mathbf{u}_i + \boldsymbol{\Lambda}_i^T \mathbf{K}_i) \frac{\partial \mathbf{u}_i}{\partial z_\ell} \right], \quad \forall \ell \in \mathcal{C}. \end{aligned} \quad (3.5)$$

To avoid computing $\partial \mathbf{u}_i / \partial z_\ell$ in the third term of the summation, the adjoint vector is chosen to be

$$\boldsymbol{\Lambda}_i = 2(\hat{\mathbf{u}}_i - \mathbf{u}_i), \quad (3.6)$$

such that the adjoint equation pre-multiplying $\partial \mathbf{u}_i / \partial z_\ell$ in Equation 3.5 vanishes. Note that $\boldsymbol{\Lambda}_i$ is a function of known quantities and does not require an additional linear solve.

With the adjoint vectors known, the derivative of the objective function in Equation 3.5 is simplified to be

$$\frac{\partial f}{\partial z_\ell} = \sum_{i=1}^{N_\theta} (\hat{\mathbf{u}}_i - \mathbf{u}_i(\mathbf{d}_i))^T \mathbf{k}_\ell^0 (\hat{\mathbf{u}}_i + \mathbf{u}_i(\mathbf{d}_i)), \quad \forall \ell \in \mathcal{C}. \quad (3.7)$$

In Equation 3.7, it has been noted that the stiffness matrix of the design lattice can be written as a linear function of the design variables,

$$\mathbf{K}_i = \sum_{\ell \in \mathcal{C} \cup \mathcal{S}_i} z_\ell \mathbf{k}_\ell^0(E, t, h, L_\ell), \quad \text{such that} \quad \frac{\partial \mathbf{K}_i}{\partial z_\ell} = \mathbf{k}_\ell^0, \quad \forall \ell \in \mathcal{C} \cup \mathcal{S}_i. \quad (3.8)$$

3.3.2 Convexity

To investigate convexity of the objective function, f , its Hessian,

$$\frac{\partial^2 f}{\partial z_\ell \partial z_k} = \frac{\partial}{\partial z_k} \left[\sum_{i=1}^{N_\theta} (\hat{\mathbf{u}}_i^T \mathbf{k}_\ell^0 \hat{\mathbf{u}}_i + \hat{\mathbf{u}}_i^T \mathbf{k}_\ell^0 \mathbf{u}_i - \mathbf{u}_i^T \mathbf{k}_\ell^0 \hat{\mathbf{u}}_i - \mathbf{u}_i^T \mathbf{k}_\ell^0 \mathbf{u}_i) \right], \quad (3.9)$$

is considered. Note that the term in brackets is an expanded form of Equation 3.7. The second and third terms of the summation in Equation 3.9 cancel out, the first term goes to zero since it is the derivative of a constant, and the last term is the second derivative of the well-known structural compliance that is ubiquitous in topology optimization. Due to the proof of convexity of structural compliance by Svanberg [106], the Hessian of f is positive-definite, and thus, f is convex. Convexity holds because of the assumption that the stiffness matrix of the design lattice can be written as a linear function of the design variables (see Equation 3.8) and there is a one-to-one mapping between nodes in the reference and design lattices.

Although the objective function is convex and, in theory, a global optimum can be found, it is unlikely that perfect cloaking can be achieved with this approach, if at all. Cloaking capabilities in this setting are limited by physical constraints of the design space, e.g., the ground structure connectivity and box constraints on the design variables.

3.4 Details of the numerical implementation

In this section, details related to implementation of the topology optimization formulation for design of elastostatic cloaking devices in 2D lattices are provided. Specifically addressed are Tikhonov regularization to handle low stiffness elements and self-equilibrated loads, the design variable update scheme, the discrete filtering scheme used to remove thin members during the optimization [69], and the convergence criterion used to stop the iterative optimization algorithm.

3.4.1 Low stiffness elements and self-equilibrated loading

When the design variables approach zero, the stiffness matrix, $\mathbf{K}_i(\mathbf{d}_i)$, can become ill-conditioned or singular. To avoid ill-conditioning, the state equations are derived by introducing Tikhonov regularization [104], such that the total potential energy of the system for

load case i is written as

$$\Pi_i(\mathbf{u}_i(\mathbf{d}_i), \mathbf{d}_i) = \frac{1}{2} \mathbf{u}_i(\mathbf{d}_i)^T \mathbf{K}_i(\mathbf{d}_i) \mathbf{u}_i(\mathbf{d}_i) - \mathbf{F}_i^T \mathbf{u}_i(\mathbf{d}_i) + \frac{\eta}{2} \mathbf{u}_i(\mathbf{d}_i)^T \mathbf{u}_i(\mathbf{d}_i). \quad (3.10)$$

The last term in equation Equation 4.33 is the Tikhonov regularization term and η is the Tikhonov parameter defined as $\eta_0 \approx 10^{-12}$ to 10^{-8} multiplied by the mean of the diagonal of $\mathbf{K}_i(\mathbf{d}_i)$. Then according to the principal of minimum potential energy, $\partial \Pi_i / \partial \mathbf{u}_i = 0$ implies that the discretized state equations become $(\mathbf{K}_i(\mathbf{d}_i) + \eta \mathbf{I}) \mathbf{u}_i(\mathbf{d}_i) = \mathbf{F}_i$, where \mathbf{I} is the identity matrix. Tikhonov regularization has also been shown to handle self-equilibrated loads [69], and is used here to regularize the uniaxial tension problems that are specified without support boundary conditions.

3.4.2 Design variable update

The design variables are updated in each optimization iteration using a steepest descent algorithm such that the candidate design variables at iteration $k + 1$ are

$$\tilde{z}_\ell^{(k+1)} = z_\ell^{(k)} - \tau^{(k)} \frac{\partial f}{\partial z_\ell^{(k)}}, \quad \forall \ell \in \mathcal{C}, \quad (3.11)$$

where inexact line search is used to find the step size, $\tau^{(k)}$, at iteration k [71]. In Equation 3.11, $\tilde{z}_\ell^{(k+1)}$ is the value of the design variable associated with lattice element ℓ before the (modified) discrete filter [69] described in the next section is employed.

3.4.3 Discrete filter

A modified discrete filter [69] is employed in each optimization iteration such that members with design variables falling below a given threshold, α_f , are pushed to zero. That is,

$$z_\ell^{(k+1)} = \begin{cases} 0 & \text{if } \frac{\tilde{z}_\ell^{(k+1)}}{\max(\{\tilde{z}_\ell\}_{\ell \in \mathcal{C}}^{(k+1)})} < \alpha_f \\ \tilde{z}_\ell^{(k+1)} & \text{otherwise.} \end{cases}, \quad \forall \ell \in \mathcal{C} \quad (3.12)$$

If the filter operation would cause the lattice to become un-equilibrated in a given iteration, the filter operation is skipped in that step. The lattice is said to satisfy equilibrium if the

error, $\|\mathbf{K}_i(\mathbf{d}_i) \mathbf{u}_i(\mathbf{d}_i) - \mathbf{F}(\theta_i)\|/\|\mathbf{F}(\theta_i)\| \leq \text{err}^{\text{tol}}$ (typically $\text{err}^{\text{tol}} = 10^{-4}$).

3.4.4 Convergence criterion

The optimization algorithm is stopped based on stagnation of the cloaking metric, Δ . For load case, i , define

$$\Delta_i = \frac{\|\mathbf{u}_i^s - \widehat{\mathbf{u}}_i^s\|_2}{\|\widehat{\mathbf{u}}_i^s\|_2}; \quad (3.13)$$

for iteration, k , define

$$\delta\Delta_{\max}^{(k)} = \max\left(\left(\Delta_1^{(k)} - \Delta_1^{(k-1)}\right), \dots, \left(\Delta_{N_\theta}^{(k)} - \Delta_{N_\theta}^{(k-1)}\right)\right), \text{ and} \quad (3.14)$$

$$\delta\Delta_{\min}^{(k)} = \min\left(|\Delta_1^{(k)} - \Delta_1^{(k-1)}|, \dots, |\Delta_{N_\theta}^{(k)} - \Delta_{N_\theta}^{(k-1)}|\right). \quad (3.15)$$

The algorithm is said to have converged when $\delta\Delta_{\max}^{(k)} \geq 10^{-2}$ or when $\delta\Delta_{\min}^{(k)} \leq 2 \times 10^{-4}$. The first criterion is needed because although the objective function, f , is convex, Δ is non-convex and may increase over the optimization iterations.

3.5 Elastostatic cloak design studies

The studies included in this section consider the target problem described in Figure 3.3, with unitless lattice parameters provided in Table 3.1. The stiffness parameters in the surrounding region are $z_\ell = \widehat{z}_\ell = 0.8 \forall \ell \in \Omega^e \setminus \mathcal{C}$ and the initial value of each design variable is $z_\ell = 0.8 \forall \ell \in \mathcal{C}$. The discrete filter threshold used during the optimization iterations is $\alpha_f = 10^{-4}$. All load cases used in design consider a tensile pressure of $p = 0.8925$ in the positive and negative x' direction as shown in Figure 3.7, applied as nodal loads distributed using an appropriate tributary width. Although only designed for tensile loads, the cloaks are also evaluated in shear. Shear loads are obtained by rotating the tensile loads by 90° counterclockwise. The active portion of the extended lattice for each of the considered load directions are provided in Figure 3.8, where the solid green lines indicate loaded edges.

Optimization-based cloaks are designed considering various single- and multi-load-direction cases. The single-load-direction designs are indicated by the angle, θ , for which they were designed. The multi-load-direction designs are indicated according to the following conven-

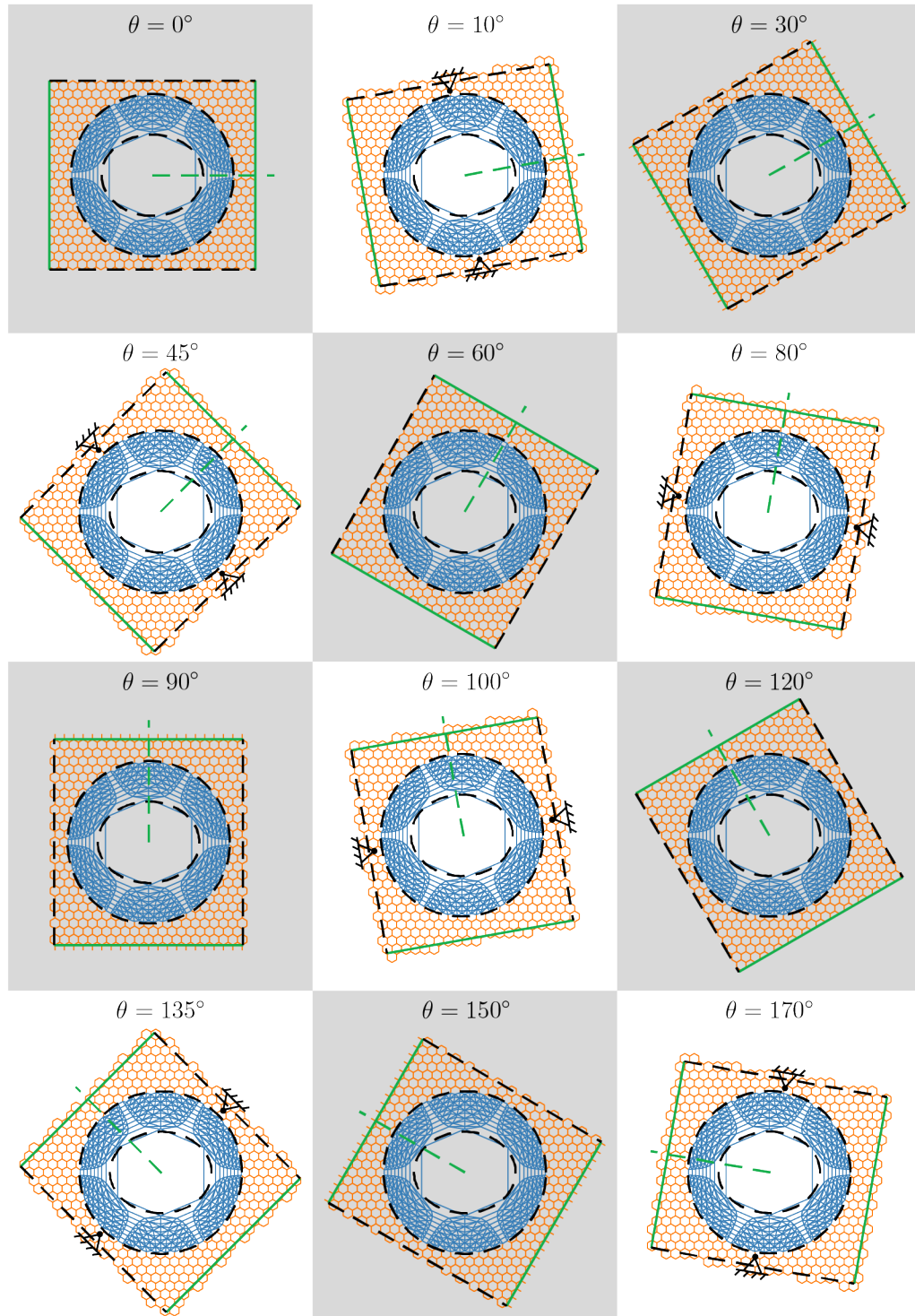


Figure 3.8: Design lattices considering the elliptical hole ($k_x = 1.25$) for all load directions considered: In all cases, the solid green lines indicate loaded edges and the green dashed line indicates the direction of uniaxial tension used in design. Shaded cases correspond to loading aligned along a reference lattice line of symmetry. Un-shaded cases correspond to loading aligned away from a reference lattice line of symmetry and in these cases, two pinned supports are included.

Table 3.1: Lattice parameters for the elastostatic cloak design studies

lattice width, L_x	152.04
lattice height, L_y	139.65
reference hexagon edge length, L	4
number of hexagons along lattice width, N_x	22
number of hexagons along lattice height, N_y	12
hole radius, r_1	30
hole semi-major axis parameter, k_x	1 or 1.25
hole semi-minor axis parameter, k_y	1
design region outer radius, R_2	60
Young’s modulus, E	100
beam thickness, t	0.8
beam height, h	0.8

tion:

1. $\theta_{\text{sym}} = [0^\circ, 30^\circ, 60^\circ, 90^\circ, 120^\circ, 150^\circ]$ includes load directions aligned along the reference lattice’s lines of symmetry (shaded cases in Figure 3.8).
2. $\theta_{\text{nosym}} = [10^\circ, 45^\circ, 80^\circ, 100^\circ, 135^\circ, 170^\circ]$ includes load directions aligned away from the reference lattice’s lines of symmetry (un-shaded cases in Figure 3.8).
3. $\theta_{\text{all}} = [0^\circ, 10^\circ, 30^\circ, 45^\circ, 60^\circ, 80^\circ, 90^\circ, 100^\circ, 120^\circ, 135^\circ, 150^\circ, 170^\circ]$ includes all load directions defined in Figure 3.8.

All optimization-based cloaks are designed to minimize the objective function, f , but cloak effectiveness is measured in terms of the cloaking metric, Δ .⁴ Using Δ , the optimization-based cloak designs are compared to the “no cloak” case in which all lattice elements with either of its end nodes within a radius, R_1 , are removed and no cloak is designed around the resulting hole. The optimization-based cloak designs are also compared to those designed via a DLT approach similar to the one proposed by Bückmann et al. [98]. Recall that the DLT approach defines the nodal positions in the design region by applying the coordinate transformation in Equation 3.2 to the nodal positions of the reference lattice, without changing the connectivity. After the coordinate transformation, the cross-sectional properties of each lattice element in the design region are designed so that their axial stiffness is unchanged in the presence of the element’s change in length. Bückmann et al. [98]

⁴ Convergence plots for both f and Δ are provided in Appendix D for the GS1 optimization-based cloak designs.

design such axial stiffness by defining a non-uniform cross-section of each lattice element in the design region. Here, a simpler approach is considered in which the stiffness of lattice element, $\ell \in \mathcal{C}$, is scaled by L/L_ℓ , where L is the length of each lattice element in the reference lattice and L_ℓ is the length of lattice element ℓ after the coordinate transformation. It is noted that this scaling approach does not observe the lower and upper bounds on the design variables used in the optimization problem. In this sense, it provides more design freedom in local element stiffness than the optimization-based approach. A benchmark is provided in Appendix C to demonstrate that the Euler-Bernoulli beam model and the axial stiffness transformation considered here provide a DLT cloak with similar performance to that obtained by Bückmann et al. [98].

3.5.1 Comparison of single- and multi-load-direction designs

The topology optimization formulation in Equation 3.3 is used to design a cloak around a circular hole ($k_x = 1$) and an elliptical hole ($k_x = 1.25$). Using ground structure GS1, designs considering a single load direction in the optimization are compared to designs considering multiple load directions in the optimization. Single-load-case designs are obtained for tensile loading oriented at $\theta = 0^\circ$, $\theta = 30^\circ$, $\theta = 60^\circ$, and $\theta = 90^\circ$. Multi-load-case designs are obtained for tensile loading oriented along the reference lattice’s six lines of symmetry, i.e., $\theta_{\text{sym}} = [0^\circ, 30^\circ, 60^\circ, 90^\circ, 120^\circ, 150^\circ]$.

Table 3.2: Number of lattice elements in design region of the initial ground structures and final designs for selected single- and multi-load-case designs (Note: the design region contains 792 lattice elements before ground structure generation)

	initial ground structure	final design ($k_x = 1$)	final design ($k_x = 1.25$)
GS1, $\theta = 0^\circ$	1500	1244	1320
GS1, $\theta = 30^\circ$	1500	1188	1252
GS1, $\theta = 60^\circ$	1500	1244	1248
GS1, $\theta = 90^\circ$	1500	1188	1228
GS1, $\theta_{\text{sym}} = [0^\circ, 30^\circ, 60^\circ, 90^\circ, 120^\circ, 150^\circ]$	1500	1140	1238
GS2, $\theta_{\text{sym}} = [0^\circ, 30^\circ, 60^\circ, 90^\circ, 120^\circ, 150^\circ]$	3078	1752	2024

The elastostatic cloak designs based on ground structure GS1 and considering the single load directions, $\theta = 0^\circ$ and $\theta = 30^\circ$, are asymmetric as shown in Figure 3.9a and b, respectively. The elastostatic cloak design based on ground structure GS1 and considering

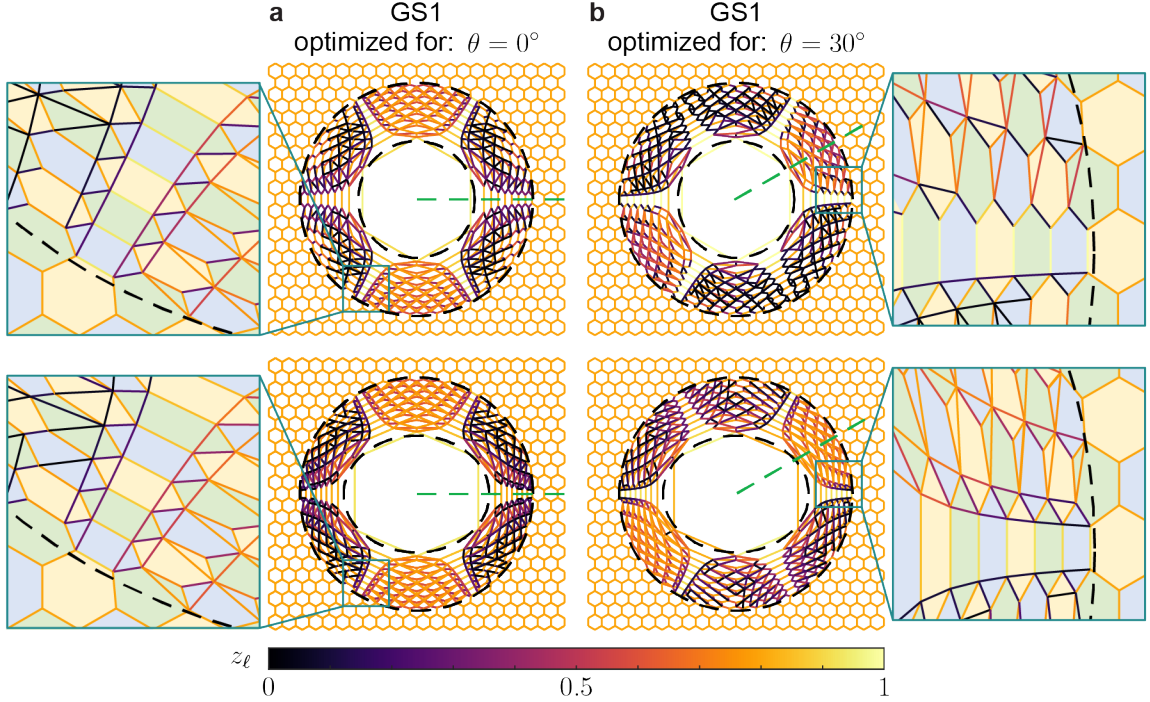


Figure 3.9: Optimization-based cloak designs based on ground structure GS1 for tensile loads oriented at (a) $\theta = 0^\circ$ and (b) $\theta = 30^\circ$, as indicated with green dashed lines. The top row is the design for a circular hole ($k_x = 1$) and the bottom row is the design for an elliptical hole ($k_x = 1.25$). Hexagons in the insets are colored to help visualize the additional connectivity added with ground structure generation.

multiple load cases (θ_{sym}) is shown in Figure 3.10a. The symmetrically-selected, multiple load cases considered here tend to preserve the symmetries of the reference lattice, although the cloak designed for the elliptical hole is clearly not symmetric.⁵ The number of lattice elements in the design region for each of the single-load-case and multi-load-case designs described here are provided in Table 3.2.

The multi-directional performance of each elastostatic cloak design is evaluated by computing the cloaking metric, Δ , for tension and shear loads applied at $\theta = 0^\circ$, $\theta = 30^\circ$, $\theta = 60^\circ$, $\theta = 90^\circ$, $\theta = 120^\circ$, and $\theta = 150^\circ$. In Figure 3.11a and b, the cloaks designed for the circular ($k_x = 1$) and elliptical ($k_x = 1.25$) hole, respectively, are evaluated in tension. In Figure 3.11c and d, the cloaks designed for the circular ($k_x = 1$) and elliptical ($k_x = 1.25$) hole, respectively, are evaluated in shear (i.e., for loading conditions not used in design). The effectiveness of the single- and multi-load-case designs considering ground structure GS1 are also compared to the lattice without any cloak around the hole (no cloak), the

⁵ Appendix C includes a study showing how the single-load-case designs perform when symmetry is enforced on the design variables during optimization.

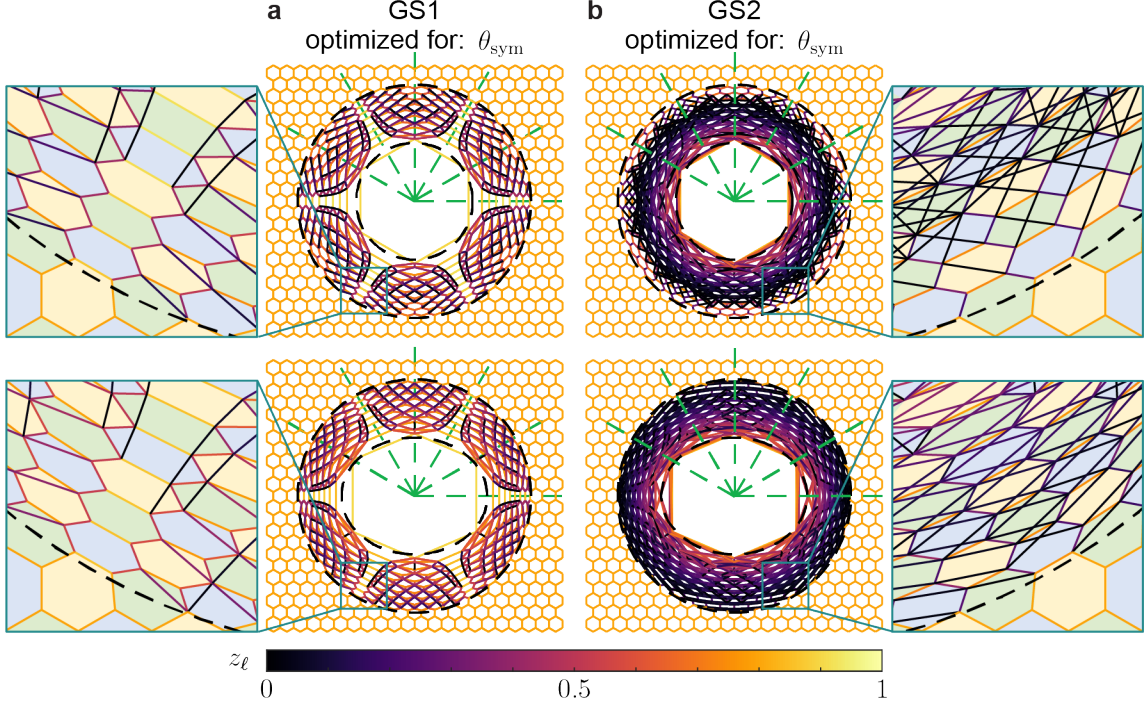


Figure 3.10: Optimization-based cloak designs based on ground structures (a) GS1 and (b) GS2 for tensile loads oriented at $\theta_{\text{sym}} = [0^\circ, 30^\circ, 60^\circ, 90^\circ, 120^\circ, 150^\circ]$, as indicated with green dashed lines. The top row is the design for a circular hole ($k_x = 1$) and the bottom row is the design for an elliptical hole ($k_x = 1.25$). Hexagons in the insets are colored to help visualize the additional connectivity added with ground structure generation.

lattice with cloak designed using the DLT approach, and the lattice with cloak designed using the multi-load-case, optimization-based approach considering ground structure GS2 (designs for the GS2 cloaks are shown in Figure 3.10).

The value of Δ for the single-load-case designs, shown in dashed, colored lines in Figure 3.11, indicate that single-load-case designs may be biased toward the load case for which they were designed. In Figure 3.11a and b, each of the single-load-case designs have the smallest value of Δ for the direction used in design. In some cases, Δ becomes as large as the case with no cloak when loaded in directions not used in design. In other cases (e.g., $\theta = 0^\circ$ and $\theta = 60^\circ$ in the case of a circular hole in Figure 3.11a), the cloaks obtained from a single load case perform well in all directions.

Figure 3.11a and b also show that the multi-load-case designs (θ_{sym}) are effective in all tensile directions used in design, with relatively constant Δ over load angle. They also outperform the DLT cloak in all load directions considered here. Figure 3.11c and d show that the multi-load-case cloaks also obscure the effect of the hole on the displacement field

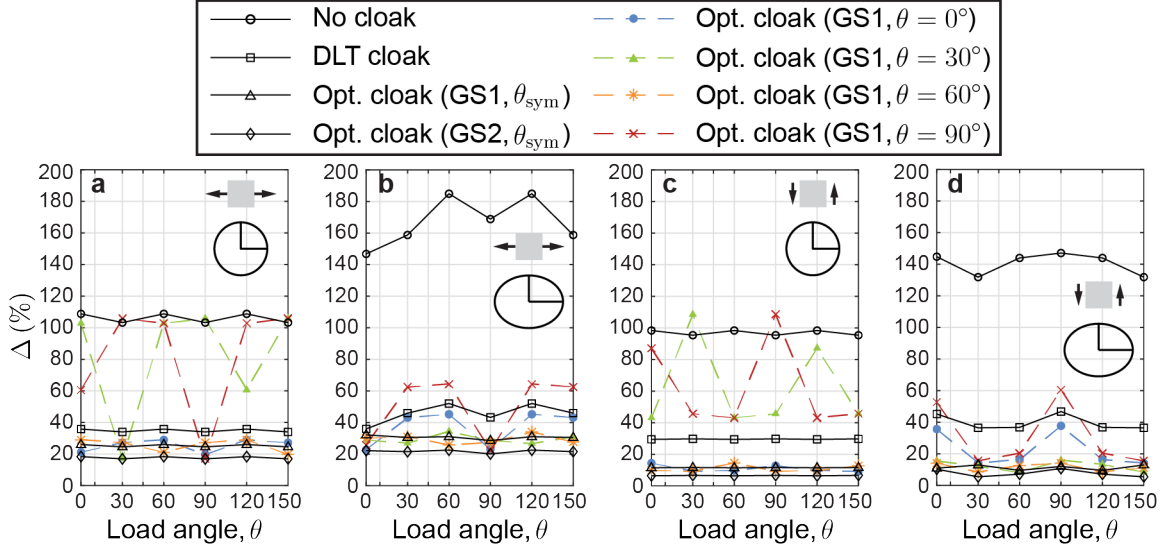


Figure 3.11: Cloaking metric, Δ , versus load angle, θ , for the lattice with no cloak, DLT cloak, and optimization-based cloaks: In (a) and (b), the cloaks are evaluated under tensile loading for a circular ($k_x = 1$) and an elliptical ($k_x = 1.25$) hole, respectively; in (c) and (d), the cloaks are evaluated under shear loading for a circular ($k_x = 1$) and an elliptical ($k_x = 1.25$) hole, respectively. The optimized cloaks were designed considering tensile loads oriented at $\theta_{\text{sym}} = [0^\circ, 30^\circ, 60^\circ, 90^\circ, 120^\circ, 150^\circ]$.

when loaded in shear, for which they were not designed, and although the DLT cloaks are also effective in shear, the optimization-based cloaks outperform the DLT cloaks to an even greater extent in shear than in tension. The superiority of the multi-load-case, optimization-based cloaks over the DLT cloak in shear may be because the DLT approach focuses on local *axial* stiffness of the lattice elements, whereas the optimization-based approach targets global stiffness characteristics of the lattice as a system. Lastly, because it allows for more freedom during design, the GS2 cloak outperforms the GS1 cloak in all cases, as expected.

The metric, Δ , is a global measure of the cloak's performance. To understand the performance locally, the normalized displacement fields in the x' direction under tensile loading are plotted for the reference, the case of no cloak, and the GS1 optimization-based cloak designed for an elliptical hole ($k_x = 1.25$) in Figure 3.12. Similarly, the normalized displacement fields in the y' direction under shear loading are plotted in Figure 3.13. In both cases, it is clear that the cloak obscures the effect of the hole on the displacement fields.

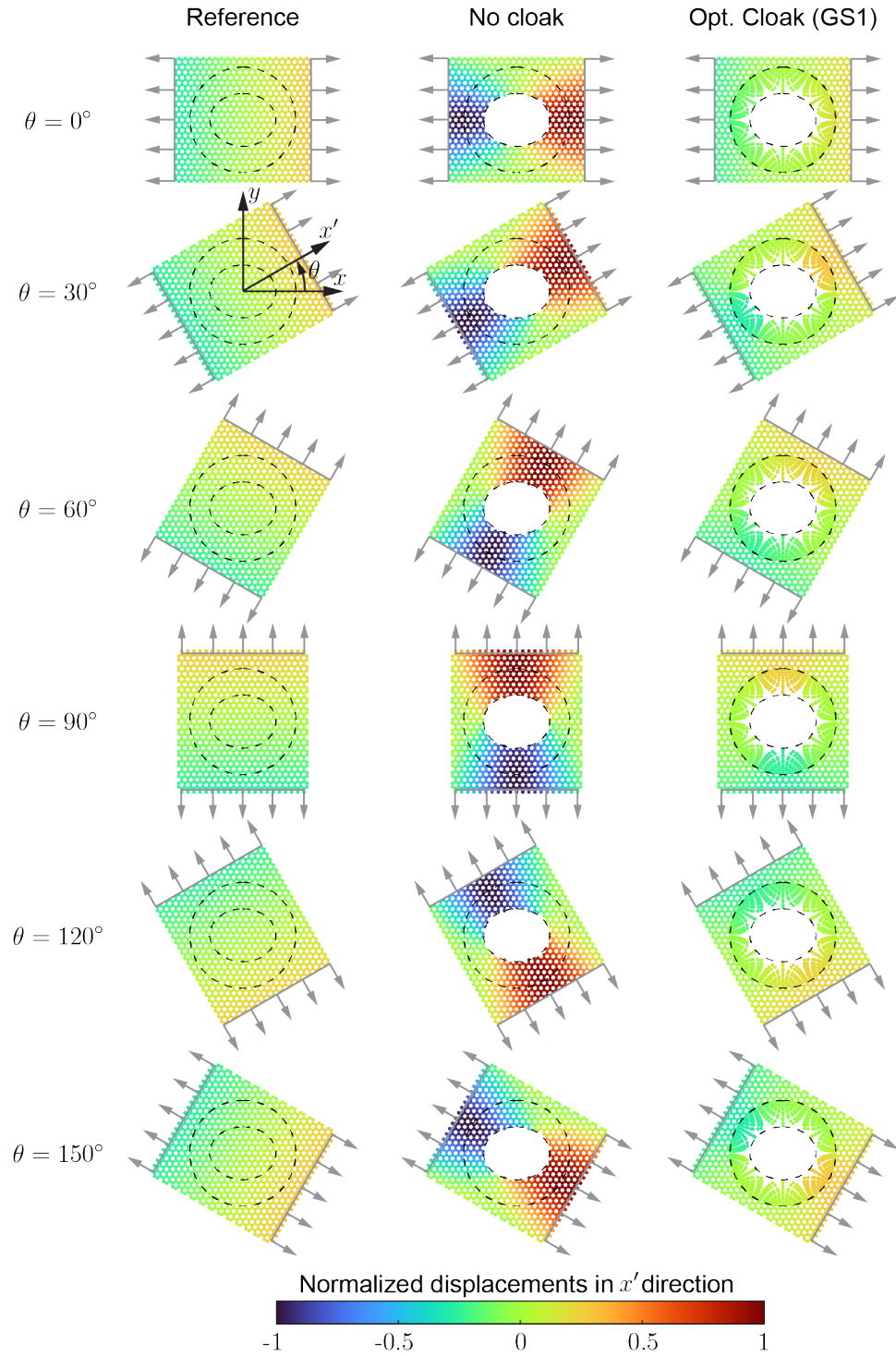


Figure 3.12: Normalized displacement fields in the x' direction for tensile loading: The left, middle, and right columns show, respectively, the displacement fields of the reference lattice, the lattice with no cloak, and the lattice with GS1 optimized cloak designed for an elliptical hole ($k_x = 1.25$) and tensile loads oriented at $\theta_{\text{sym}} = [0^\circ, 30^\circ, 60^\circ, 90^\circ, 120^\circ, 150^\circ]$.

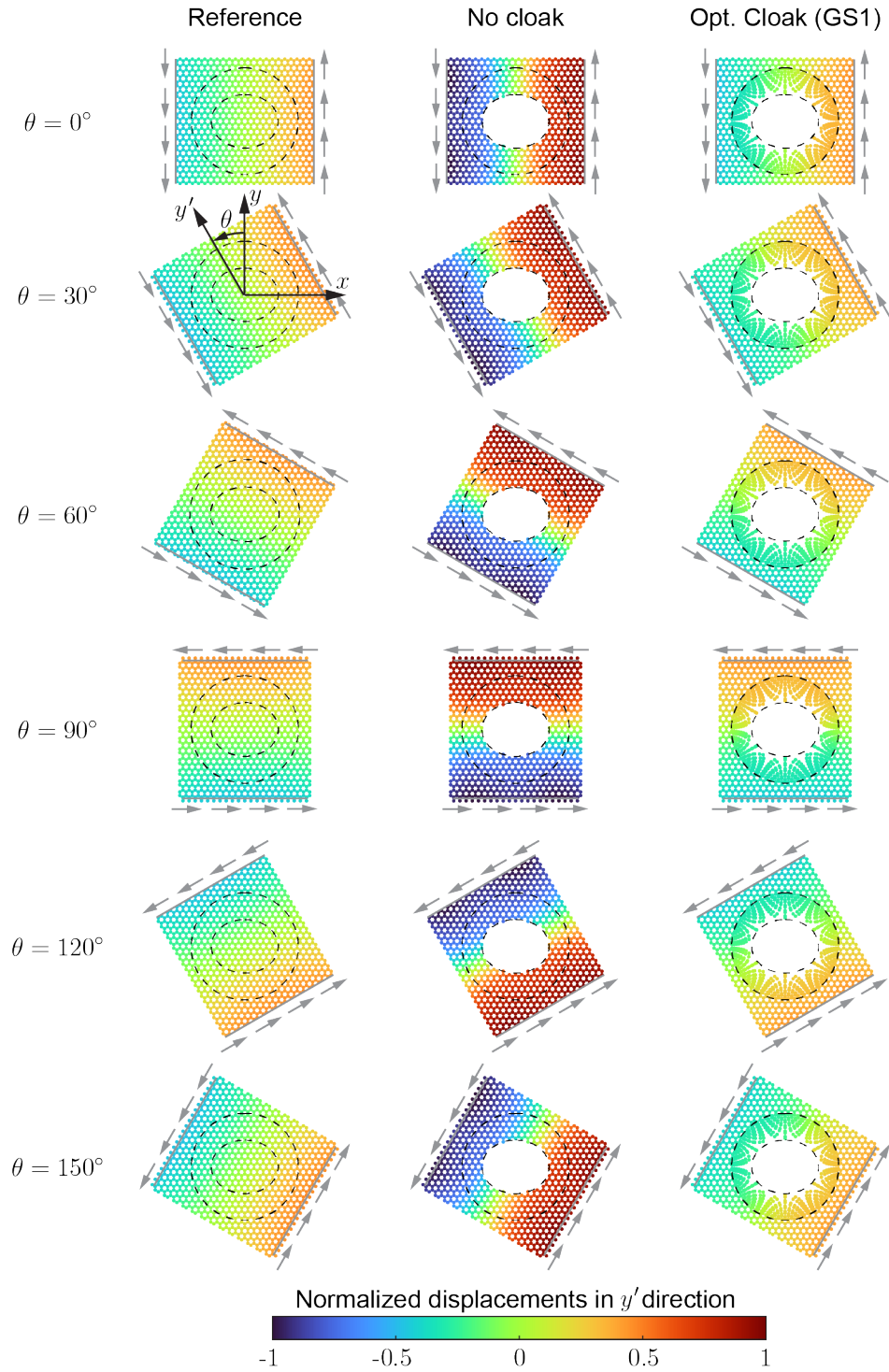


Figure 3.13: Normalized displacement fields in the y' direction for shear loading: The left, middle, and right columns show, respectively, the displacement fields of the reference lattice, the lattice with no cloak, and the lattice with GS1 optimized cloak designed for an elliptical hole ($k_x = 1.25$) and tensile loads oriented at $\theta_{\text{sym}} = [0^\circ, 30^\circ, 60^\circ, 90^\circ, 120^\circ, 150^\circ]$.

3.5.2 Effect of load directions used in design

In the previous sub-section, the load directions considered in multi-load-case design were aligned with the reference lattice's lines of symmetry (i.e., $\theta_{\text{sym}} = [0^\circ, 30^\circ, 60^\circ, 90^\circ, 120^\circ, 150^\circ]$). Here, load directions aligned away from the reference lattice's lines of symmetry are considered in design (i.e., $\theta_{\text{nosym}} = [10^\circ, 45^\circ, 80^\circ, 100^\circ, 135^\circ, 170^\circ]$). Additionally, designs obtained considering θ_{sym} and θ_{nosym} are evaluated for the directions not used in design and compared to designs obtained considering all twelve load directions simultaneously (i.e., $\theta_{\text{all}} = [0^\circ, 10^\circ, 30^\circ, 45^\circ, 60^\circ, 80^\circ, 90^\circ, 100^\circ, 120^\circ, 135^\circ, 150^\circ, 170^\circ]$). As before, all load cases used in design are tensile loads, but the cloaks are also evaluated for effectiveness in shear.

In the left, middle, and right columns of Figure 3.14, the cloaking metric, Δ , is reported for the cloaks designed considering a circular hole and the six tensile load cases aligned along the reference lattice's lines of symmetry (θ_{sym}), the six tensile load cases aligned away from the reference lattice's lines of symmetry (θ_{nosym}), and all twelve tensile load cases (θ_{all}), respectively. In the top row, Δ is reported for the cloaks evaluated in tension and in the bottom row, Δ is reported for the cloaks evaluated in shear. In all plots, Δ is reported for all twelve load directions to understand how these multi-load-case designs behave in directions for which they were not designed. Since the lattice has different stiffness along its lines of symmetry than away from its lines of symmetry, solid lines are used to show the response of each lattice when loaded along its lines of symmetry (θ_{sym}) and dashed lines are used to show the response of each lattice when loaded away from its lines of symmetry (θ_{nosym}).

The top row of Figure 3.14 shows that the GS1 cloak performs better in tension for all load directions when designed for tensile loads aligned away from the lattice's lines of symmetry (θ_{nosym}), whereas the GS2 cloak generally performs better in tension in load directions for which it was designed. The GS2 cloak has the most design freedom, and as expected, always outperforms the GS1 and DLT cloaks. The GS1 cloak always outperforms the DLT cloak in tension for loads aligned along the lattice's lines of symmetry (θ_{sym} , solid lines), but only outperforms the DLT cloak in tension for loads aligned away from the lattice's lines of symmetry (θ_{nosym} , dashed lines) when it is designed for those load cases only (Figure 3.14b). When all twelve load cases are considered (θ_{all} , Figure 3.14c), the performance of the optimized cloaks is typically somewhere between that of either of the six-load-case results. In general, the optimized cloaks are effective in tension in all directions

investigated here. In the case of shear (bottom row of Figure 3.14), the GS1 and GS2 cloaks are always superior to the DLT cloak and the GS2 cloak is always superior to the GS1 cloak. Cloak performance is best in shear when designed for tension in all twelve load directions (θ_{all} , Figure 3.14e).

Figure 3.15 provides a similar study for the case of an elliptical hole and similar trends are observed. The superiority in shear of the GS2 cloak over the GS1 cloak is less clear for the case of an elliptical hole. Nevertheless, all optimized cloaks considered here for an elliptical hole are effective in directions for which they were not designed and are superior to the DLT cloak with the exception of the GS1 cloak when evaluated in tension for directions aligned away from the lattice's lines of symmetry (θ_{nosym} , dashed lines in top row of Figure 3.15a and c).

As demonstrated in the study comparing single-load-case and multi-load-case designs in subsection 3.5.1, it is apparent that the choice of boundary conditions plays a critical role in mimicking the stiffness characteristics of the reference lattice. One load case is clearly not sufficient, but the multi-load-case study here indicates that too many load cases, in some cases, can also degrade cloaking performance. Additional studies are needed to determine how to best choose the set of boundary conditions considered in design (e.g., load directions, number of load cases, type of loading conditions, type of support conditions).

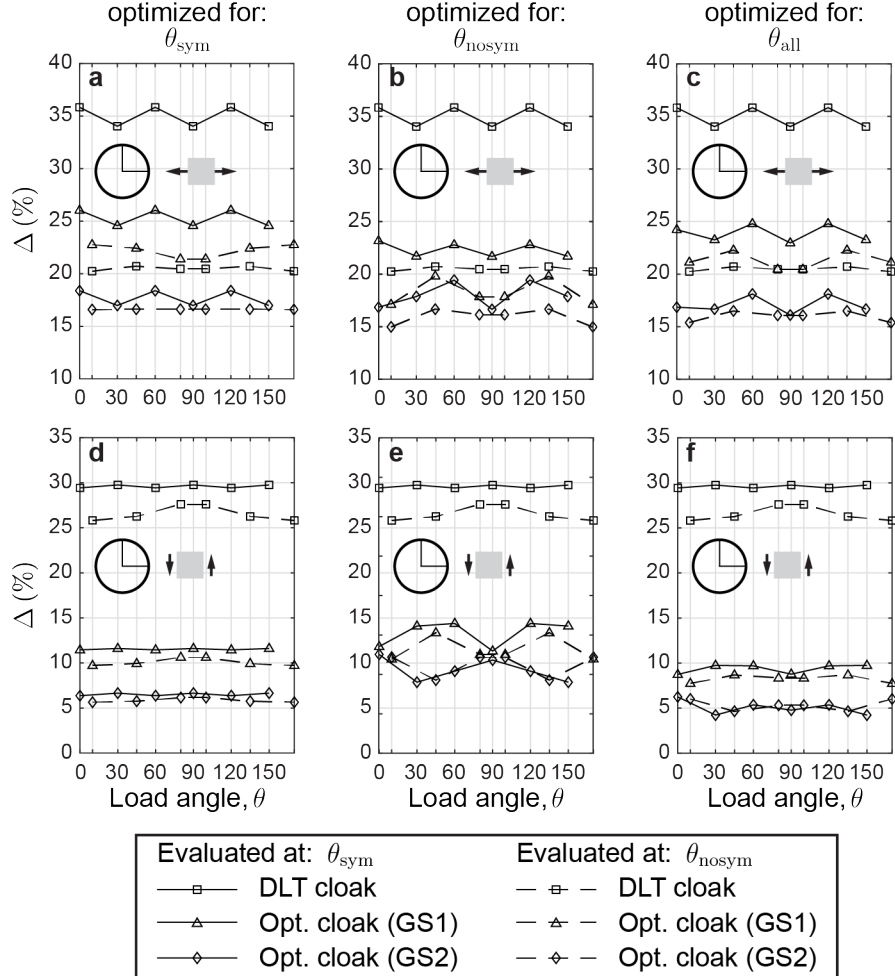


Figure 3.14: Cloaking metric, Δ , versus load angle, θ , for the DLT cloak and optimization-based cloaks (GS1 and GS2) designed for a circular hole ($k_x = 1$): In (a-c), the cloaks are evaluated in tension; in (d-e), the cloaks are evaluated in shear. In the left, middle, and right columns, Δ is reported for the cloaks designed considering the six tensile load cases aligned along the reference lattice's lines of symmetry (θ_{sym}), the six tensile load cases aligned away from the reference lattice's lines of symmetry (θ_{nosym}), and all twelve tensile load cases (θ_{all}), respectively. Solid and dashed lines indicate Δ evaluated for load directions aligned, respectively, along (i.e., $\theta_{\text{sym}} = [0^\circ, 30^\circ, 60^\circ, 90^\circ, 120^\circ, 150^\circ]$) and away from (i.e., $\theta_{\text{nosym}} = [10^\circ, 45^\circ, 80^\circ, 100^\circ, 135^\circ, 170^\circ]$), the reference lattice lines of symmetry.

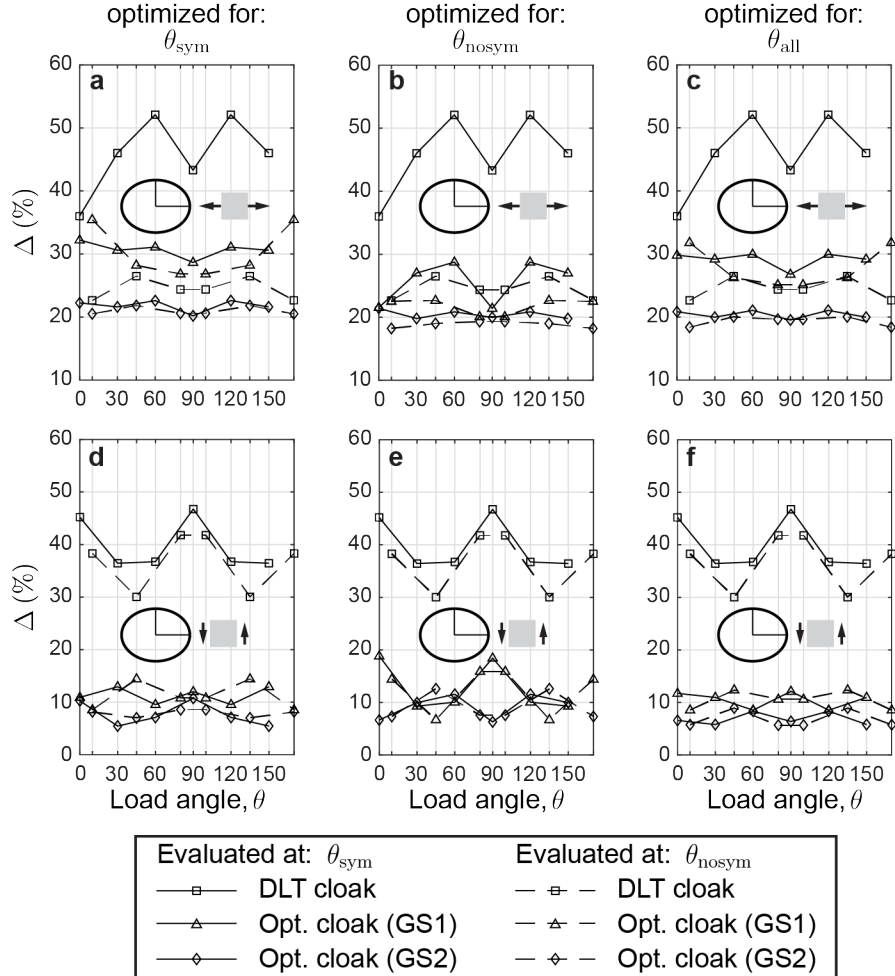


Figure 3.15: Cloaking metric, Δ , versus load angle, θ , for the DLT cloak and optimization-based cloaks (GS1 and GS2) designed for an elliptical hole ($k_x = 1.25$): In (a-c), the cloaks are evaluated in tension; in (d-e), the cloaks are evaluated in shear. In the left, middle, and right columns, Δ is reported for the cloaks designed considering the six tensile load cases aligned along the reference lattice's lines of symmetry (θ_{sym}), the six tensile load cases aligned away from the reference lattice's lines of symmetry (θ_{nosym}), and all twelve tensile load cases (θ_{all}), respectively. Solid and dashed lines indicate Δ evaluated for load directions aligned, respectively, along (i.e., $\theta_{\text{sym}} = [0^\circ, 30^\circ, 60^\circ, 90^\circ, 120^\circ, 150^\circ]$) and away from (i.e., $\theta_{\text{nosym}} = [10^\circ, 45^\circ, 80^\circ, 100^\circ, 135^\circ, 170^\circ]$), the reference lattice lines of symmetry.

CHAPTER 4

MULTI-MATERIAL TOPOLOGY OPTIMIZATION CONSIDERING POROUS, ANISOTROPIC MATERIALS

Many natural structures and materials such as wood, bone, and shells synthesize spatially-varying mechanical properties and/or structural hierarchy to achieve novel functionalities [107]. For example, teeth have a hard and brittle outer layer (enamel) that is prone to cracking, but also a tough inner layer (dentin) that mitigates crack propagation into the tooth's interior [108]; the Mantis Shrimp's dactyl club exhibits functionally graded elasticity and hardness that leads to high impact resistance needed for striking its prey [109]; bamboo's fiber-reinforcement is functionally graded radially along the culm (stalk) cross-section for high bending rigidity per unit mass [110]; and the internal bone structure of the cuttlefish contains a porous, layered architecture that simultaneously resists high pressures experienced in the deep sea while remaining lightweight and enabling the cuttlefish to control its buoyancy [111, 112]. Many of these natural features have been borrowed to enhance the performance of engineered structures and materials. For example, reinforced concrete is a multi-material system with properties exceeding that of the constituent materials; sandwich panels use structural hierarchy to enhance their strength-to-weight ratio; and engineered foams provide energy absorption, thermal insulation, and buoyancy in various engineering applications [113]. Topology optimization provides a rational way to further elicit novel functionalities from such systems (e.g., artificial materials with negative thermal expansion [5, 114], structures exhibiting prescribed deformations [115–117]), but a formulation in the continuum setting that can handle a large number of (possibly porous, anisotropic) materials is needed. In this chapter, the material space in continuum topology optimization is *relaxed* to accommodate an arbitrary number of materials in a general framework that can handle material anisotropy.

4.1 Overview

Inspired by natural systems that harness hierarchical architectures and spatially-varying properties, a formulation is put forth for design of maximally stiff structures and components

that mimic such properties. In this case, Equation 1.1 takes the specific form

$$\begin{aligned}
& \min_{\mathbf{Z} \in [0,1]^{N \times m}} && f(\mathbf{Z}, \mathbf{u}(\mathbf{Z})) = \mathbf{F}^T \mathbf{u}(\mathbf{Z}) \\
& \text{s.t.} && g_j(\mathbf{Z}) = \frac{\sum_{i \in \mathcal{G}_j} \sum_{\ell \in \mathcal{E}_j} A_\ell v_{\ell i}}{\sum_{\ell \in \mathcal{E}_j} A_\ell} - \bar{v}_j \leq 0, \quad j = 1, \dots, K \\
& \text{with} && \mathbf{K}(\mathbf{Z}) \mathbf{u}(\mathbf{Z}) = \mathbf{F},
\end{aligned} \tag{4.1}$$

where the objective function, f , is a linear measure of compliance and the constraint functions, g_j , $j = 1, \dots, K$, are limits on material volume. The formulation in Equation 4.1 differs from the single material formulation in Equation 1.1 in two main ways. First, instead of a single density field, a set of density fields, $\mathbf{Z} = \{z_{\ell 1}, \dots, z_{\ell m}\}_{\ell=1}^N$, are defined, where $z_{\ell i}$ is a density design variable for each of the m candidate materials at each of the N design points in the domain (see Figure 4.1). Additionally, instead of a single volume constraint, $j = 1, \dots, K$ volume constraints are specified to control any subset of the candidate materials, $\{z_{\ell i} : i \in \mathcal{G}_j\}$, in any subregion of the domain, $\{z_{\ell i} : \ell \in \mathcal{E}_j\}$. Furthermore, A_ℓ , is the area/volume of the domain associated with design variables, $\{z_{\ell 1}, \dots, z_{\ell m}\}$, at location, ℓ ; $v_{\ell i}$ is the volume fraction of material associated with design variable $z_{\ell i}$; and \bar{v}_j is the volume fraction limit for constraint j . When the subscript, j , is omitted, it is understood that there is only one volume constraint. Additionally, the discretized state equations of static (linear) elasticity have been derived from the principle of minimum potential energy, where $\mathbf{K}(\mathbf{Z})$, $\mathbf{u}(\mathbf{Z})$, and \mathbf{F} are the stiffness matrix, displacement vector, and external load vector, respectively.

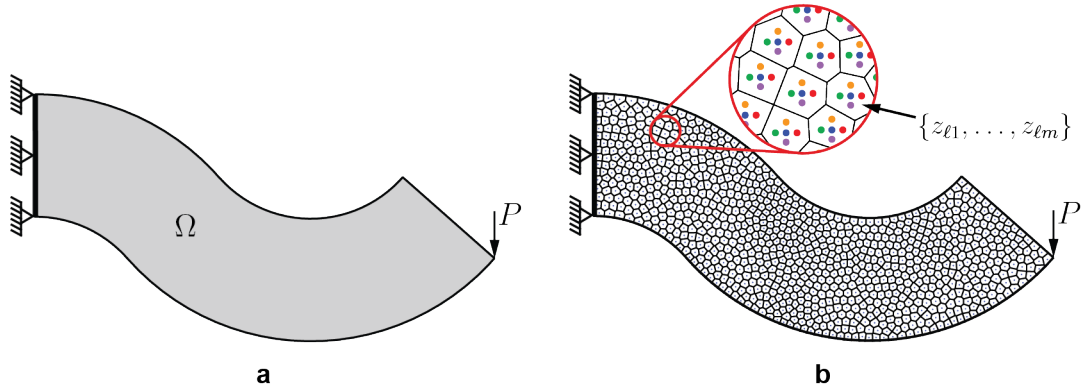


Figure 4.1: Density-based, multi-material topology optimization problem setup: (a) Domain and boundary conditions; (b) design variable discretization considering m candidate materials.

A key component needed to obtain practical multi-material structures and components from Equation 4.1 is an effective material interpolation scheme that both translates the design variable fields into material properties and penalizes undesirable behavior (e.g., intermediate densities, material mixing). Later in this chapter, a material interpolation function that accomplishes these goals is discussed.

4.2 Related work

The great majority of work in density-based topology optimization considering multiple material phases is based on some extension of the Solid Isotropic Material with Penalization (SIMP) interpolation scheme, which uses a power law to penalize intermediate densities and achieve designs with distinct solid and void regions [13, 14]. For two-material (no void) topology optimization of materials with extreme thermal expansion, a single design variable is used to interpolate between two material phases [25]. The approach has also been used to design, multi-physics actuators [118], piezocomposites [119–121], and functionally graded structures with optimal eigenfrequencies [122]. A three-phase extension of SIMP is characterized by a topology design variable that controls the material/void distribution and a second design variable that interpolates between two solid material phases [25]. Although this “three-phase mixing scheme” has been extended further to incorporate up to m candidate materials [28], some authors claim that designs tend to get stuck in local minima when the number of materials exceeds three solid phases [28, 123]. Actually, most results in the literature for multi-material topology optimization using this “ m -phase mixing scheme” have been limited to two [124–126] or three [123] solid phases plus void.

Other material interpolation schemes that are better equipped to handle greater than three solid phases have also been proposed. For example, in the context of composite design via fiber orientation optimization, the Discrete Material Optimization (DMO) technique was proposed to consider an arbitrary number of materials, each characterized by a discrete fiber orientation [28, 127]. The DMO interpolation schemes are typically also an extension of SIMP, but differ from the “ m -phase mixing scheme” discussed above in that they require a density field for each candidate material (as in Equation 4.1). DMO interpolation schemes were found to reach superior designs as compared to those from the “ m -phase mixing scheme,” even in cases considering only two solid phases plus void and a single mass constraint [124].

One variant of DMO sums the stiffness contributions of each design variable at a given point and does not inherently prevent mixing or the sum of material densities at a point from exceeding one as the “ m -phase mixing scheme” does. The first issue can be mitigated with a quadratic constraint that prevents mixing [128] and the second issue can be mitigated by imposing a large system of sparse linear constraints that enforces that the sum of densities must be less than or equal to one at each point [128, 129].

The aforementioned variant of DMO was also coupled with an alternating active-phase (AAP) algorithm in which designs containing up to m material phases are achieved by performing m binary material phase updates in an inner loop of each outer optimization iteration. In this case, the AAP algorithm enforces that the sum of material densities at a point does not exceed one. Implementing the AAP approach essentially amounts to adding a loop over an existing two-phase topology optimization code. Although this approach is seemingly flexible enough to accommodate an arbitrary number of candidate materials, numerical experiments using the downloadable AAP code indicates that it is difficult to obtain converged solutions for more than five materials. Additionally, resulting designs depend on the order that the materials are updated, which may prevent the method from being applied to problems considering materials with more general constitutive behavior (e.g., nonlinear or anisotropic materials). Furthermore, the AAP algorithm leads to an increase in the number of finite element solves by a factor of the number of candidate materials times the number of specified inner iterations, and thus, may not scale to large problems [29]. Despite these drawbacks, a number of authors have adopted the AAP algorithm [130–133].

Another DMO variant inhibits mixing via a penalty term, which is the approach adopted and extended in subsection 4.3.3 of this dissertation. Since mixing is inherently avoided with this approach, the densities are also prevented from summing to greater than one at any given point; however, when coupled with a density filter that causes intermediate densities at the boundaries of each material region, this DMO variant cannot completely remove mixing at the material interfaces. In subsection 4.4.4, two methods to avoid or remove these small regions of mixing are discussed.

A pitfall of the DMO approaches is that the number of design variables scales linearly with the number of candidate materials. A peak function material interpolation keeps the number of design variables constant as the number of candidate materials increases [26]; however, the DMO interpolations have several advantages that potentially outweigh the

disadvantage of increased, but non-prohibitive, computational cost. First, the DMO interpolations lead to linear and variable separable volume constraints. As a result, the update of each design variable is dependent only on the Lagrange multiplier of its associated volume constraint (no cross-term dependency), allowing the design to be updated for each volume constraint independently (i.e., order-independent updates) using the ZPR update scheme [30]. Thus, the ZPR update scheme enables the flexible setting of volume constraints put forth in Equation 4.1, is straightforward to implement, and does not lead to an increased number of finite element solves as the number of constraints increases. Second, the DMO interpolations enable a continuation scheme, put forth in subsection 4.3.3, that biases the solution toward a convex one at the beginning of the optimization and reduces the potential of getting trapped in a poor local minimum, although a global minimum cannot be guaranteed.

Although this literature review is limited to density-based topology optimization, it is noted that multi-material topology optimization has also been explored using discrete methods [30], level-set methods, [27, 134, 135], phase-field methods [136–139], and evolutionary methods [140].

4.3 Problem setting and optimization formulation

In this section, the volume constrained compliance minimization problem is formulated in pursuit of an optimal shape, ω , contained in the extended domain, $\Omega \subseteq \mathbb{R}^2$, where ω is the union of a finite number of non-intersecting partitions composed of m different materials, i.e., $\omega = \cup_i^m \omega_i$ and $\omega_i \cap \omega_j = \emptyset$ for $i \neq j$ (see Figure 4.2). The optimization problem is stated as

$$\begin{aligned} \inf_{\omega \in \mathcal{O}} f(\omega, \mathbf{u}_\omega) \quad & \text{subject to} \\ g_j(\omega, \mathbf{u}_\omega) \leq 0, \quad & j = 1, \dots, K, \end{aligned} \tag{4.2}$$

where \mathcal{O} is the space of admissible shapes and $\mathbf{u}_\omega \in \mathcal{V}_\omega$ satisfies the governing variational problem of linear elasticity

$$\int_{\omega} \mathbf{C}(\mathbf{x}) \nabla \mathbf{u}_\omega : \nabla \mathbf{v} d\mathbf{x} = \int_{\tilde{\Gamma}_N} \mathbf{t} \cdot \mathbf{v} ds, \quad \forall \mathbf{v} \in \mathcal{V}_\omega, \tag{4.3}$$

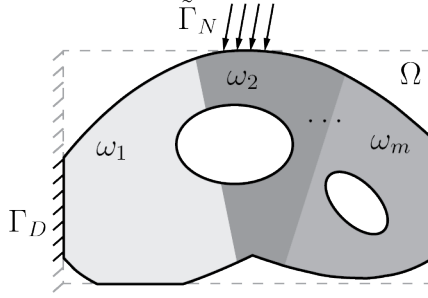


Figure 4.2: Multi-material problem setup: Extended design domain, boundary conditions, and illustration of material partitions making up the optimal shape.

with the space of admissible displacements defined as

$$\mathcal{V}_\omega = \{ \mathbf{v} \in H^1(\omega, \mathbb{R}^2) : \mathbf{v}|_{\partial\omega \cap \Gamma_D} = \mathbf{0} \}. \quad (4.4)$$

In Equation 4.3, $\mathbf{C}(\mathbf{x})$ is the stiffness tensor that varies according to the material from which each partition, $\omega_i \subseteq \omega$, is made, Γ_D is the partition of $\partial\Omega$ on which displacements are prescribed, Γ_N is the complimentary partition of $\partial\Omega$ such that $\overline{\Gamma_D} \cup \overline{\Gamma_N} = \partial\Omega$ and $\Gamma_D \cap \Gamma_N = \emptyset$, and $\tilde{\Gamma}_N \subseteq \Gamma_N$ is the partition of $\partial\Omega$ on which non-zero tractions, \mathbf{t} , are prescribed.

The reader is referred to the discussions by Talischi et al. [17] that develop, in sufficient detail, the notions needed to arrive at the sizing problem from the problem described in Equation 4.2 and Equation 4.3. Here, the key steps are highlighted with emphasis on the additional requirements to incorporate multiple materials into the formulation.

4.3.1 Continuous sizing problem

First, a characteristic function, χ_ω , is introduced to recast the boundary value problem in Equation 4.3 onto Ω such that it becomes

$$\int_{\Omega} \chi_\omega \mathbf{C}(\mathbf{x}) \nabla \mathbf{u} : \nabla \mathbf{v} d\mathbf{x} = \int_{\tilde{\Gamma}_N} \mathbf{t} \cdot \mathbf{v} ds, \quad \forall \mathbf{v} \in \mathcal{V} \quad (4.5)$$

and the space of admissible displacements, \mathcal{V} , is independent of ω . Next, a continuous parameterization, $\rho_i \in [0, 1]$, $i = 1, \dots, m$, is introduced for each candidate material to avoid integer programming, which can be prohibitively expensive. A penalty function, m_W , (e.g., SIMP [13, 14]) recovers the binary nature of the problem and an additional interpolation

function, m_M , (e.g., DMO [28, 127]) enforces selection of a single material at each point in ω .

Further, a restriction setting is adopted in which a regularization map, \mathcal{P} , enforces well-posedness of the optimization problem by introducing a function, η_i , $i = 1, \dots, m$, for each candidate material, such that each ρ_i in the admissible space of designs inherits the smoothness characteristics of the kernel used to define \mathcal{P} , i.e., $\rho_i = \mathcal{P}(\eta_i)$. Here, the regularization map is defined by convolution of the design functions, η_i , $i = 1, \dots, m$, with a smooth kernel (filter), F , [141, 142], i.e.,

$$\mathcal{P}_F(\eta_i) = \int_{\Omega} F(\mathbf{x}, \bar{\mathbf{x}}) \eta_i(\bar{\mathbf{x}}) d\bar{\mathbf{x}}, \quad (4.6)$$

where the filter of radius, R , is defined as

$$F(\mathbf{x}, \bar{\mathbf{x}}) = c(\mathbf{x}) \max \left[0, \left(1 - \frac{\|\mathbf{x} - \bar{\mathbf{x}}\|}{R} \right)^q \right]. \quad (4.7)$$

In Equation 4.7, $c(\mathbf{x})$ is a normalizing coefficient and q defines the order of the filter [21] (e.g., linear filter when $q = 1$).

Now, the volume constrained compliance minimization problem is stated as

$$\begin{aligned} \inf_{\{\rho_i\}_{i=1}^m \in \mathcal{A}} f(\rho_1, \dots, \rho_m, \mathbf{u}) &= \int_{\tilde{\Gamma}_N} \mathbf{t} \cdot \mathbf{u} ds \quad \text{subject to} \\ g_j(\{\rho_i : i \in \mathcal{G}_j\}) &= \sum_{i \in \mathcal{G}_j} \frac{1}{|\Omega_j|} \int_{\Omega_j} m_V(\rho_i) d\mathbf{x} - \bar{v}_j \leq 0, \quad j = 1, \dots, K, \end{aligned} \quad (4.8)$$

where \mathcal{G}_j is the set of material indices associated with constraint j , $\Omega_j \in \Omega$ is the partition of the domain for which constraint j is specified, m_V is the volume interpolation function, \bar{v}_j is a volume fraction limit for constraint j , and the space of admissible designs is

$$\mathcal{A} = \left\{ \mathcal{P}_F(\eta_1) \circ \dots \circ \mathcal{P}_F(\eta_m) : \eta_i \in L^\infty \forall i (\Omega; [\underline{\rho}, \bar{\rho}]) \right\}, \quad (4.9)$$

where \circ indicates composition of functions. In Equation 4.8, $\mathbf{u} \in \mathcal{V}$ satisfies

$$\int_{\Omega} m_E(\rho_1, \dots, \rho_m, \mathbf{C}_1, \dots, \mathbf{C}_m) \nabla \mathbf{u} : \nabla \mathbf{v} d\mathbf{x} = \int_{\tilde{\Gamma}_N} \mathbf{t} \cdot \mathbf{v} ds, \quad \forall \mathbf{v} \in \mathcal{V}, \quad (4.10)$$

where $m_E = m_M \circ m_W$ and the space of admissible displacements is

$$\mathcal{V} = \{\mathbf{v} \in H^1(\Omega, \mathbb{R}^2) : \mathbf{v}|_{\partial\Gamma_D} = \mathbf{0}\}. \quad (4.11)$$

4.3.2 Discrete problem

A final step required to solve the problem numerically is to discretize the displacement field, \mathcal{V} , and design space, \mathcal{A} , on Ω . For convenience, both spaces are discretized using the same fixed partition, $\mathcal{T}_h = \{\Omega_\ell\}_{\ell=1}^N$, for which h represents the characteristic mesh size, $\Omega_\ell \cap \Omega_k = \emptyset$ for $\ell \neq k$, and $\cup_\ell \bar{\Omega}_\ell = \bar{\Omega}$. With this partition, the piecewise constant discretization of \mathcal{A} is defined as

$$\mathcal{A}_h = \{\mathcal{P}_F(\eta_1^h) \circ \dots \circ \mathcal{P}_F(\eta_m^h) : \underline{\rho} \leq \eta_i^h \leq \bar{\rho} \forall i, \eta_i^h|_{\Omega_\ell} = \text{const} \forall \ell\}, \quad (4.12)$$

where $\rho_i^h = \mathcal{P}_F(\eta_i^h)$, $i = 1, \dots, m$, for each candidate material. The matrix of design variables, $\mathbf{Z} = \{z_{\ell i}\}_{\ell=1}^N$, results from discretizing the design functions such that

$$\eta_i^h(\mathbf{x}) = \sum_{\ell=1}^N z_{\ell i} \chi_{\Omega_\ell}(\mathbf{x}), \quad (4.13)$$

where $z_{\ell i}$ is the constant value that η_i^h assumes over Ω_ℓ . The functions, ρ_i^h , are replaced by $\tilde{\rho}_i^h$, $i = 1, \dots, m$, which are constant over each element, i.e.,

$$\tilde{\rho}_i^h(\mathbf{x}) = \sum_{\ell=1}^N y_{\ell i} \chi_{\Omega_\ell}(\mathbf{x}), \quad (4.14)$$

according to an elemental value, $y_{\ell i} = \rho_i^h(\mathbf{x}_\ell^*)$, that is defined using the value of ρ_i^h at the centroid, \mathbf{x}_ℓ^* , of element ℓ . The set of element values can also be organized in matrix form as $\mathbf{Y} = \{y_{\ell i}\}_{\ell=1}^N$.

In the same way that Talischi et al. [17] discretized the mapping \mathcal{P}_F , the elemental values of η_i^h and ρ_i^h are related by

$$\mathbf{y}_i = \mathbf{P}\mathbf{z}_i, \quad i = 1, \dots, m, \quad (4.15)$$

where \mathbf{y}_i and \mathbf{z}_i are the i^{th} columns of \mathbf{Y} and \mathbf{Z} , respectively, and

$$P_{\ell k} = \frac{h_{\ell k} A_k}{\sum_{j=1}^N h_{\ell j} A_j}, \quad h_{\ell k} = \max \left[0, \left(1 - \frac{\|\mathbf{x}_\ell - \mathbf{x}_k\|_2}{R} \right)^q \right]. \quad (4.16)$$

With these definitions, the final discrete problem for volume constrained compliance minimization in Equation 4.1 is obtained, where the stiffness matrix and vector of design-independent applied loads are

$$\mathbf{K} = \sum_{\ell=1}^N m_E (y_{\ell 1}, \dots, y_{\ell m}, \mathbf{k}_{\ell 1}^0, \dots, \mathbf{k}_{\ell m}^0) \quad \text{and} \quad \mathbf{F}_i = \int_{\bar{\Gamma}_N} \mathbf{t} \cdot \mathbf{N}_i ds, \quad (4.17)$$

respectively. In Equation 4.17, $(\mathbf{k}_{\ell i}^0)_{jk} = \int_{\Omega_\ell} \mathbf{B}_j^T \mathbf{D}_i \mathbf{B}_k d\mathbf{x}$ is the constant element stiffness matrix for material i in element ℓ , \mathbf{N} is the vector of interpolation (shape) functions, \mathbf{B} is the strain-displacement matrix of shape function derivatives, and \mathbf{D}_i is the elasticity tensor (in matrix notation) characterizing material i . In Equation 4.17, recall that the stiffness interpolation function, m_E , is the composition of the multi-material interpolation function and the penalty function, i.e., $m_E = m_M \circ m_W$, which are discussed in more detail in the following subsection.

4.3.3 Material interpolation

The material interpolation, $m_E = m_M \circ m_W$, serves to define the constitutive properties of the system and also penalize intermediate densities and material mixing. The material interpolation is accomplished in two steps. First, a penalty function, m_W , is used to push the element densities toward zero and one. In the case of SIMP [13, 14], penalized element densities, $\mathbf{W} = \{w_{\ell 1}, \dots, w_{\ell m}\}_{\ell=1}^m$, are computed as

$$w_{\ell i} = m_W (y_{\ell i}) = y_{\ell i}^p, \quad \ell = 1, \dots, N, \quad i = 1, \dots, m, \quad (4.18)$$

where $p > 1$ is a penalty parameter on intermediate densities. The effect of p is illustrated in Figure 4.3a and b, which will be discussed in more detail after the multi-material interpolation function is defined. In the second step, a multi-material interpolation function, m_M , is used to both determine the stiffness properties in each element and penalize mixing [28].

The multi-material interpolation function takes the form

$$\mathbf{k}_\ell = m_M (\mathbf{w}'_\ell, \mathbf{k}_{\ell 1}^0, \dots, \mathbf{k}_{\ell m}^0) = \sum_{i=1}^m w_{\ell i} \prod_{\substack{j=1 \\ j \neq i}}^m (1 - \gamma w_{\ell j}) \mathbf{k}_{\ell i}^0, \quad \ell = 1, \dots, N. \quad (4.19)$$

In Equation 4.19, \mathbf{w}'_ℓ is the ℓ^{th} row of \mathbf{W} and the penalty parameter, $0 < \gamma < 1$, is introduced to control the amount of allowable mixing. The effect of γ is illustrated in Figure 4.3a and c, and will be discussed in more detail in the context of a scalar interpolation at the end of this subsection. In general, solutions without material mixing (i.e., $\gamma = 1$) are desired; however, the problem in Equation 4.1 is convex for $p = 1$ and $\gamma = 0$ and a continuation scheme on these penalty parameters can be used to bias the solution toward the convex one at the beginning of the optimization iterations [9]. Additional discussion on convexity and the continuation scheme is provided in subsection 4.3.6.

Interpolating the elasticity matrices, $\mathbf{D}_i, i = 1, \dots, m$, is conceptually more straightforward than interpolating the element stiffness matrices, $\mathbf{k}_{\ell i}^0, i = 1, \dots, m$, but would require the element stiffness matrices for each candidate material to be recomputed in each optimization iteration, leading to increased computational cost. Thus, the element stiffness matrices are pre-computed and interpolated directly using Equation 4.19 [10]. If all candidate materials are isotropic, the interpolation can often be simplified. Typically Poisson's ratio does not play a major role in the stiffness of the system and can be assumed to take the same value, ν^0 , for each of the m candidate materials. The Young's moduli, $E_i^0, i = 1, \dots, m$, are the key material parameters contributing to stiffness of the system and the elasticity matrix of material i can be written with the Young's modulus factored out, i.e., $\mathbf{D}_i = E_i^0 \mathbf{D}^0(\nu^0)$. In this case, it is simpler and more computationally efficient to interpolate the scalar-valued Young's modulus in element ℓ as

$$E_\ell = m_M (\mathbf{w}'_\ell, E_1^0, \dots, E_m^0) = \sum_{i=1}^m w_{\ell i} \prod_{\substack{j=1 \\ j \neq i}}^m (1 - \gamma w_{\ell j}) E_i^0, \quad \ell = 1, \dots, N. \quad (4.20)$$

Then a constant element stiffness matrix, $(\mathbf{k}_\ell^0)_{jk} = \int_{\Omega_\ell} \mathbf{B}_j^T \mathbf{D}^0(\nu^0) \mathbf{B}_k d\mathbf{x}$, of element ℓ can be precomputed and, in each iteration, scaled by the interpolated Young's modulus to find

the element stiffness matrix of element ℓ as $\mathbf{k}_\ell = E_\ell \mathbf{k}_\ell^0$, in place of Equation 4.19.

The penalty function in Equation 4.18 and the scalar version of the multi-material interpolation in Equation 4.20 are plotted in Figure 4.3 to illustrate how they serve to penalize intermediate densities and material mixing. A single element with two candidate materials ($E_1^0 = 0.5$, $E_2^0 = 1$) is considered. First note that if $p = 1$, then $w_{\ell i} = y_{\ell i}$ (see Equation 4.18) and if $\gamma = 0$, then $E_\ell = \sum_{i=1}^m w_{\ell i} E_i^0$ (see Equation 4.19). This choice of penalty parameters is plotted in Figure 4.3a, where the interpolation is linear and neither intermediate densities nor material mixing are penalized. In Figure 4.3b, the penalty on material mixing is suppressed by setting $\gamma = 0$ and the penalty on intermediate densities is illustrated by setting $p = 4$. The resulting curvature makes intermediate densities inefficient; however, this choice of penalty parameters likely promotes the case in which $y_{\ell 1} = y_{\ell 2} = 1$ (full material mixing). In Figure 4.3c, the penalty on intermediate densities is suppressed with $p = 1$ and the penalty on material mixing is illustrated by setting $\gamma = 1$. Notice that $E_\ell = 0$ when $y_{\ell 1} = y_{\ell 2} = 1$; thus, it becomes inefficient for both materials to be fully dense simultaneously, but mixing at intermediate densities may still occur. Finally, in Figure 4.3d, both intermediate densities and material mixing are penalized by choosing $p = 4$ and $\gamma = 1$.

4.3.4 Volume interpolation

Element volume fractions, $\mathbf{V} = \{v_{\ell 1}, \dots, v_{\ell m}\}_{\ell=1}^N$, are defined using a volume interpolation function of the form

$$v_{\ell i} = m_V(y_{\ell i}) = y_{\ell i} \hat{v}_i, \quad (4.21)$$

where $0 < \hat{v}_i < 1$ is the solid fraction of material i , which takes a value of one for a solid material.

4.3.5 Sensitivity analysis

Sensitivities of the objective function, f , and constraint functions, $g_j, j = 1, \dots, K$, with respect to the design variables, $\mathbf{z}_i, i = 1, \dots, m$, can be expressed via the chain rule as

$$\frac{\partial f}{\partial \mathbf{z}_i} = \frac{\partial \mathbf{y}_i}{\partial \mathbf{z}_i} \frac{\partial \mathbf{w}_i}{\partial \mathbf{y}_i} \frac{\partial f}{\partial \mathbf{w}_i} \quad (4.22)$$

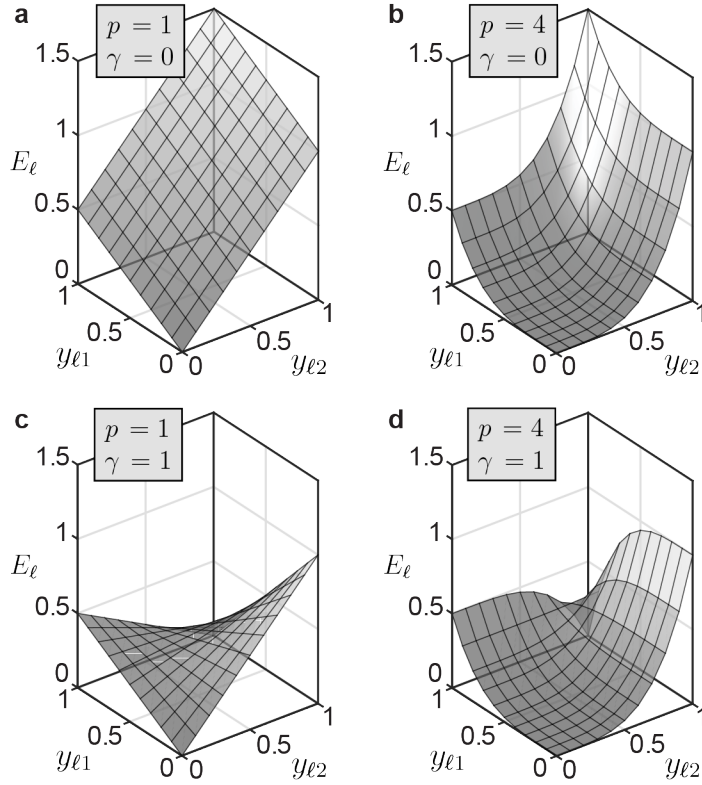


Figure 4.3: Material interpolation functions for two candidate materials ($E_1^0 = 0.5$, $E_2^0 = 1$) in element ℓ with varying choice of penalty parameters, p and γ : (a) For $p = 1$ and $\gamma = 0$, neither intermediate densities nor material mixing are penalized; (b) for $p = 4$ and $\gamma = 0$, the curvature penalizes intermediate densities, but mixing at full densities is likely promoted; (c) for $p = 1$ and $\gamma = 1$, material mixing is penalized at full densities, but not necessarily at intermediate densities; (d) for $p = 4$ and $\gamma = 1$, both intermediate densities and material mixing are penalized.

and

$$\frac{\partial g_j}{\partial \mathbf{z}_i} = \frac{\partial \mathbf{y}_i}{\partial \mathbf{z}_i} \frac{\partial \mathbf{V}_i}{\partial \mathbf{y}_i} \frac{\partial g_j}{\partial \mathbf{V}_i}, \quad (4.23)$$

where $\partial \mathbf{y}_i / \partial \mathbf{z}_i = \mathbf{P}^T$ and the other components are

$$\frac{\partial f}{\partial w_{\ell i}} = -\mathbf{u}^T \frac{\partial \mathbf{K}}{\partial w_{\ell i}} \mathbf{u}, \quad (4.24)$$

$$\frac{\partial w_{kj}}{\partial y_{\ell i}} = \begin{cases} p y_{\ell i}^{p-1}, & \text{if } \ell = k \text{ and } j = i \\ 0, & \text{otherwise,} \end{cases} \quad (4.25)$$

$$\frac{\partial g_j}{\partial v_{\ell i}} = \frac{A_{\ell}}{\sum_{\ell \in \mathcal{E}_j} A_{\ell}}, \quad (4.26)$$

and

$$\frac{\partial v_{kj}}{\partial y_{\ell i}} = \begin{cases} \hat{v}_i, & \text{if } \ell = k \text{ and } j = i \\ 0, & \text{otherwise.} \end{cases} \quad (4.27)$$

To compute Equation 4.24, the derivative of the element stiffness matrices with respect to the penalized element densities are also needed. In the case of anisotropic materials, this derivative involves the pre-computed, constant element stiffness matrix of each material, and is stated as (refer to Equation 4.19)

$$\frac{\partial \mathbf{k}_k}{\partial w_{\ell i}} = \begin{cases} \prod_{\substack{j=1 \\ j \neq i}}^m (1 - \gamma w_{\ell j}) \mathbf{k}_{\ell i}^0 - \sum_{\substack{p=1 \\ p \neq i}}^m \gamma w_{\ell p} \prod_{\substack{r=1 \\ r \neq p \\ r \neq i}}^m (1 - \gamma w_{\ell r}) \mathbf{k}_{\ell p}^0, & \text{if } \ell = k \\ 0, & \text{otherwise.} \end{cases} \quad (4.28)$$

In the case of isotropic materials, where the multi-material interpolation is done on the scalar Young's modulus (refer to Equation 4.20), the derivative in Equation 4.24 can be expressed using the chain rule as

$$\frac{\partial f}{\partial \mathbf{w}_i} = \frac{\partial \mathbf{E}}{\partial \mathbf{w}_i} \frac{\partial f}{\partial \mathbf{E}}, \quad (4.29)$$

where the components are

$$\frac{\partial E_k}{\partial w_{\ell i}} = \begin{cases} \prod_{\substack{j=1 \\ j \neq i}}^m (1 - \gamma w_{\ell j}) E_i^0 - \sum_{\substack{p=1 \\ p \neq i}}^m \gamma w_{\ell p} \prod_{\substack{r=1 \\ r \neq p \\ r \neq i}} (1 - \gamma w_{\ell r}) E_p^0, & \text{if } \ell = k \\ 0, & \text{otherwise} \end{cases} \quad (4.30)$$

and

$$\frac{\partial f}{\partial E_\ell} = -\mathbf{u}^T \frac{\partial \mathbf{K}}{\partial E_\ell} \mathbf{u}, \text{ with } \frac{\partial \mathbf{k}_k}{\partial E_\ell} = \begin{cases} \mathbf{k}_\ell^0, & \text{if } \ell = k \\ 0, & \text{otherwise.} \end{cases} \quad (4.31)$$

4.3.6 Convexity

The volume constraint functions in Equation 4.1 are convex since they are linear in the design variables. The compliance objective function in Equation 4.1 has been shown convex under the assumption that the stiffness matrix is linear in the design variables [106]. Recall from the discussion in subsection 4.3.3 that this assumption holds when $p = 1$ and $\gamma = 0$ since the element stiffness matrix can be written as

$$\mathbf{k}_\ell = \sum_{i=1}^m \left[\sum_{j=1}^N P_{\ell j} z_{ji} \right] \mathbf{k}_{\ell i}^0. \quad (4.32)$$

Although this choice of penalty parameters leads to undesirable solutions with intermediate densities and material mixing, it can be used to bias the solution toward the convex one during the first few iterations. The convex solution is presumably a “good” initial guess and from there, a continuation scheme can be established in which p and γ are gradually increased to arrive at a solution without intermediate densities or material mixing. The stiffness interpolation for a continuation scheme with one intermediate (continuation) step is shown in Figure 4.4 and the role of continuation on the final design is illustrated with an example in subsection 4.5.3.

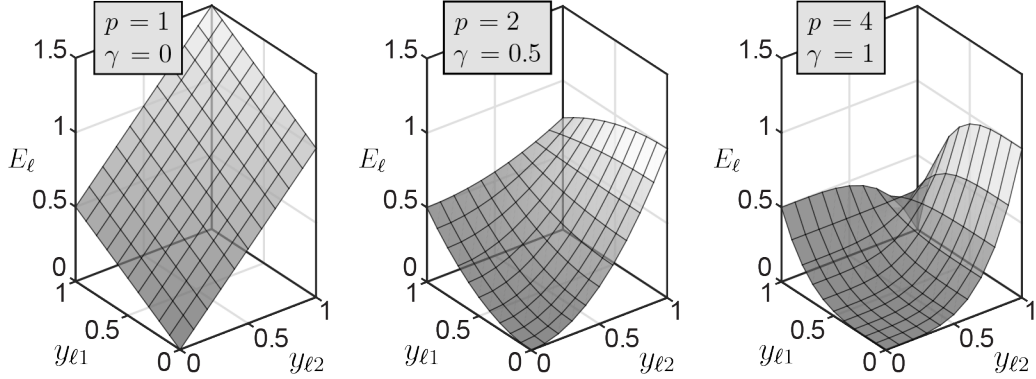


Figure 4.4: Material interpolation functions for two candidate materials ($E_1^0 = 0.5$, $E_2^0 = 1$) in element ℓ with continuation (from left to right) on penalty parameters, p and γ : The optimization problem can be initiated as the convex problem at the left, where both intermediate densities and material mixing are allowed, and gradually transformed into the non-convex problem at the right where both intermediate densities and material mixing are penalized.

4.4 Details of the numerical implementation

In this section, details related to the implementation of the multi-material topology optimization formulation are provided. Specifically addressed are several approaches to handle low density regions, the design variable update scheme, convergence criteria used to stop the iterative optimization algorithm, and post-processing procedures that can be used to remove mixing at material interfaces.

4.4.1 Low density regions

When any of the design variables approach zero, the stiffness matrix, $\mathbf{K}(\mathbf{Z})$, becomes ill-conditioned or singular. To avoid ill-conditioning, it is common to introduce an Ersatz stiffness, $\varepsilon \ll 1$, in those regions. Several approaches can be used to introduce Ersatz stiffness. Three such approaches used in this dissertation are described here, but others are also possible.

For the case of a single, solid, isotropic material in which $E_\ell = y_\ell^p$, the stiffness matrix of element ℓ has been written as $\mathbf{k}_\ell = [\varepsilon + (1 - \varepsilon) E_\ell] \mathbf{k}_\ell^0$ [17]. In this case, since $0 \leq E_\ell \leq 1$, the stiffness pre-multiplying \mathbf{k}_ℓ^0 goes to either ε or 1 as the design variables are pushed toward 0 or 1, respectively. A similar approach can be used for multi-material problems with solid, isotropic candidate materials (i.e., modification of the interpolated stiffness in

Equation 4.20 and the associated sensitivity in Equation 4.30). Note, however, that the value of E_ℓ depends on the Young's moduli of the candidate materials, E_1^0, \dots, E_m^0 . As the design variables are pushed toward 0 or 1, the stiffness pre-multiplying \mathbf{k}_ℓ^0 goes to either 0 or $E_\ell + \varepsilon(1 - E_\ell)$. Thus, \mathbf{k}_ℓ^0 is artificially reduced for $E_\ell > 1$ and artificially increased for $0 < E_\ell < 1$. These effects typically have negligible impact on the system stiffness; however, it is important to tune the value of ε to agree with the magnitude of E_1^0, \dots, E_m^0 . One choice is to normalize E_1^0, \dots, E_m^0 to have a maximum value of 1. In this case, $\varepsilon = 1 \times 10^{-4}$ works well, as long as the smallest material has Young's modulus larger than the Ersatz stiffness. This approach is implemented in `PolyMat` and used for all 2D results in chapter 5 generated using `PolyMat`.

The Ersatz stiffness can be included before the multi-material interpolation (i.e., modification of the penalized element densities, \mathbf{W} , in Equation 4.18 and the associated sensitivities in Equation 4.25). In this case, as the design variables are pushed toward 0 and 1, the penalized element stiffnesses are pushed toward ε or 1, respectively. Note that when the number of candidate materials is large, the Ersatz stiffness of all void materials at a given point is additive and could contribute finite stiffness to the system. Thus, the magnitude of ε should be tuned accordingly. One approach is to choose ε as 1×10^{-4} divided by the number of candidate materials. This approach is implemented in `PolyAnisoMat` and used for the 2D result in chapter 5 generated using `PolyAnisoMat`.

Yet another approach to avoid ill-conditioning is to derive the state equations by introducing Tikhonov regularization [104], such that the total potential energy of the system (in discretized form) is written as

$$\Pi(\mathbf{u}(\mathbf{Z}), \mathbf{Z}) = \frac{1}{2} \mathbf{u}(\mathbf{Z})^T \mathbf{K}(\mathbf{Z}) \mathbf{u}(\mathbf{Z}) - \mathbf{F}^T \mathbf{u}(\mathbf{Z}) + \frac{\eta}{2} \mathbf{u}(\mathbf{Z})^T \mathbf{u}(\mathbf{Z}). \quad (4.33)$$

The last term in Equation 4.33 is the Tikhonov regularization term and η is the Tikhonov parameter defined as $\eta_0 \approx 10^{-12}$ to 10^{-8} multiplied by the mean of the diagonal of $\mathbf{K}(\mathbf{Z})$. Then according to the principal of minimum potential energy, $\partial\Pi/\partial\mathbf{u} = 0$ implies that the discretized state equations become $(\mathbf{K}(\mathbf{Z}) + \eta\mathbf{I}) \mathbf{u}(\mathbf{Z}) = \mathbf{F}$, where \mathbf{I} is the identity matrix. The Tikhonov approach is considered for all 3D problems provided in this dissertation (chapter 4 and chapter 6). Note that in the implementation used here, Tikhonov regularization is treated as a numerical remedy to treat ill-posedness of the linear system of equations

and the design sensitivities are not modified to account for the associated Ersatz stiffness. No convergence issues were found for the problems considered here, but the Ersatz stiffness associated with Tikhonov regularization could be incorporated into the sensitivities if needed.

4.4.2 Design variable update

To solve Equation 4.1, the ZPR design variable update scheme is adopted. It uses Lagrangian duality to solve a series of convex approximate subproblems of Equation 4.1 around the current design, $\mathbf{Z}^0 = \{z_{\ell 1}^0, \dots, z_{\ell m}^0\}$, to iteratively move toward an optimal point [30]. The ZPR update scheme was derived specifically for the multi-material formulation of interest in Equation 4.1, in which each design variable is associated with a single volume constraint. Because the Lagrange multipliers associated with each constraint are independent of one another (i.e., the dual subproblem is separable), the constraints can be updated independently, leading to an update scheme that efficiently handles a large number of volume constraints at a cost on par with that of the OC update scheme [75].

For brevity, the full derivation of the ZPR update scheme is left to Appendix B, where the update is found to be

$$z_{\ell i}^{\text{new}} = \begin{cases} z_{\ell i}^+, & z_{\ell i}^* \geq z_{\ell i}^+ \\ z_{\ell i}^-, & z_{\ell i}^* \leq z_{\ell i}^- \\ z_{\ell i}^*, & \text{otherwise.} \end{cases} \quad (4.34)$$

In Equation 4.34, $z_{\ell i}^{\text{new}}$ is the design at the next iteration and $z_{\ell i}^*$ is the candidate design for the next iteration that is accepted if it is within bounds,

$$z_{\ell i}^- = \max(0, z_{\ell i}^0 - M) \quad \text{and} \quad z_{\ell i}^+ = \min(1, z_{\ell i}^0 + M), \quad (4.35)$$

defined by box constraints, $\bar{\rho} = 0$ and $\underline{\rho} = 1$, and move limit, M . The candidate design, $z_{\ell i}^*$, is obtained from a fixed-point iteration of the form

$$z_{\ell i}^* = (B_{\ell i})^{\frac{1}{1+\alpha}} \left(\sum_{k=1}^N P_{\ell k} z_{k i}^0 \right), \quad \text{where} \quad B_{\ell i} = - \frac{\frac{\partial f}{\partial z_{\ell i}} \Big|_{\mathbf{z}=\mathbf{Z}^0}}{\Lambda_j \frac{\partial g_j}{\partial z_{\ell i}} \Big|_{\mathbf{z}=\mathbf{Z}^0}}, \quad (4.36)$$

α is the exponent of the exponential intervening variables used in the convex approximation of the objective function, and Λ_j is the Lagrange multiplier associated with constraint j . Equation 4.36 departs from the update derived in Equation B.10 because of an additional heuristic filter introduced in this dissertation [8, 9]. The heuristic filter provides a different path toward an optimal design and often leads to a different local minimum (often a slightly better design for volume constrained compliance minimization problems).

Note that the multi-material interpolation function, m_M , defined in Equation 4.19 is a non-monotonous function of \mathbf{W} , allowing components of its derivative to become negative (see Figure 4.3b). Thus, the derivatives, $\partial f / \partial z_{\ell i}$, $\ell = 1, \dots, N$, $i = 1, \dots, m$, may become positive, leading to difficulties in the ZPR’s recursive update term in Equation 4.36 when the updated design variable becomes undefined (or negative) [8, 9]. Note that in the composition, $m_E = m_M \circ m_W$ with $p = 4$ and $\gamma = 1$, as shown in Figure 4.3d, the derivatives become negative in regions in which the densities of both materials are relatively high. Since this situation is penalized, material mixing only occurs in practice due to overlapping “grey” regions arising due to the filter operation. One option is to simply ignore the positive sensitivities in these localized regions by setting them to zero [5]. A more consistent option is to incorporate sensitivity separation [11, 143, 144] into the ZPR update scheme to arrive at a non-monotonic, convex approximation of the objective function. Sensitivity separation is discussed in section B.2 of Appendix B.

4.4.3 Convergence criteria

The optimization problem is considered to have converged when a prescribed maximum number of iterations is reached or when

$$\max_{\ell, i} |z_{\ell i}^{\text{new}} - z_{\ell i}^0| < \text{tol} \quad (4.37)$$

where tol is a prescribed convergence tolerance (whichever is met first).

4.4.4 Handling mixing at material interfaces

Although composition of the penalty function, m_W , and material interpolation function, m_M , penalizes material mixing (see Figure 4.3d), some mixing may occur due to overlapping intermediate densities that exist at the boundaries of each material region due to the density

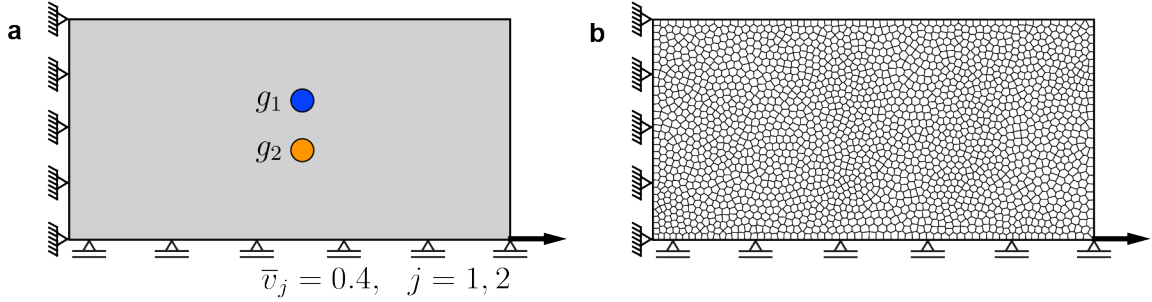


Figure 4.5: Short column: (a) Domain, boundary conditions, and constraint specification, where the blue and orange materials have $E_1^0 = 1$ and $E_2^0 = 0.15$, respectively and are controlled by g_1 ($\bar{v}_1 = 0.4$) and g_2 ($\bar{v}_2 = 0.4$), respectively; (b) finite element mesh.

Table 4.1: Optimization input parameters used for the short column (brackets indicate continuation and $R = -1$ indicates no filter).

ZPR move limit, M	[0.2, 0.2]
ZPR intermediate variable exponent, α	[-1, -1]
convergence tolerance, tol	[0.01, 0.01]
intermediate density penalty parameter, p	[3, 3]
mixing penalty parameter, γ	[1, 1]
filter radius, R ,	[*varies*, -1]
filter exponent, q ,	[1, 1]
max. number iterations	[100, 100]

filter operation. To illustrate how the filter radius influences material mixing, the symmetric short column shown in Figure 4.5 is optimized with varying filter radius, R . The design domain is discretized into 2000 polygonal finite elements using `PolyMesher` [16]. The blue and orange (solid, isotropic) candidate materials have $E_1^0 = 1$ and $E_2^0 = 0.15$, respectively. The optimization parameters used in this study are provided in Table 4.1.

The converged solutions for $R = 0.1$, $R = 0.03$, and $R = -1$ (i.e., no filter) after the first continuation step are shown in the first row of Figure 4.6. Notice that as the filter radius is reduced from left to right, the length scale of the mixing region decreases. When no filter is considered (right column), no mixing is observed, but some voids appear at the interface of the two materials (Figure 4.6c).¹

To achieve crisp interfaces without any mixing and without the appearance of voids, two approaches are pursued. In the first approach, the filter is turned off in a second continuation step (second row of Figure 4.6). Note that the shape of the material regions are influenced

¹ Numerical artifacts like checkerboarding are not observed when no filter is applied because of the polygonal discretization [17, 145].

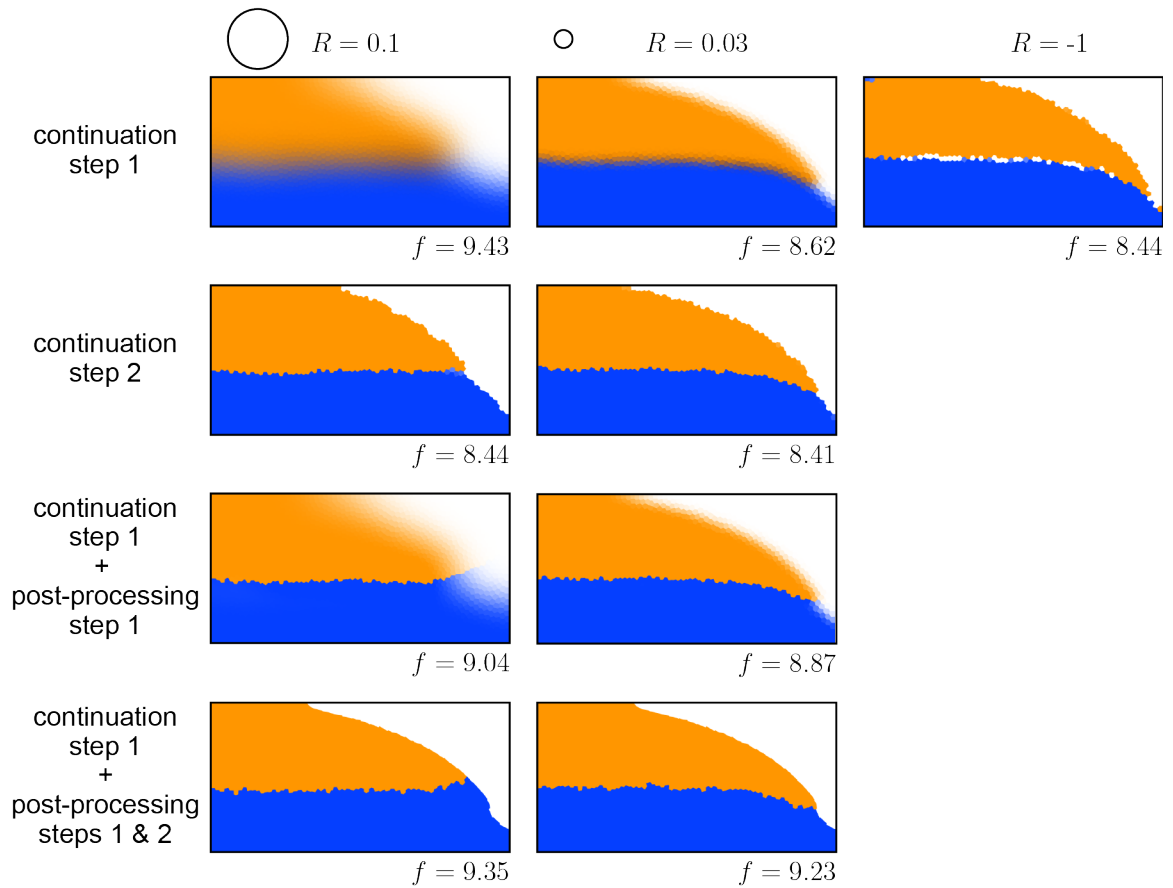


Figure 4.6: Demonstration of two ways to handle mixing at material interfaces using the short column example: The first row shows the converged solutions after the first continuation step for $R = 0.1$, $R = 0.03$, and $R = -1$ (no filter); the second row shows the converged solutions after the second continuation step in which the filter is turned off; the third and fourth rows show the result of applying post-processing step 1 and post-processing steps 1 and 2, respectively, on the converged solution from the first continuation step.

by the filter radius used in the first continuation step and filter reduction simply removes material mixing caused by the filter. Additionally, the objective function tends to decrease without affecting satisfaction of the volume constraints. A potential drawback of filter reduction is that the boundary of the structure inherits the features of the underlying mesh and may not be smooth if the discretization is coarse. Additionally, in some problems small oscillations are observed when the filter is turned off and the maximum number of iterations must be used as the stopping criterion. Gradually reducing the filter radius tends to mitigate the oscillatory behavior.

The second approach for achieving crisp interfaces, is to post-process the result obtained after the first continuation step. The post-processing procedure proposed here is accomplished in two steps. The first step is to remove any mixing present in the design by setting

$$y_{\ell i} = \begin{cases} \min [1, \sum_{k=1}^m y_{\ell k}] & \text{if } y_{\ell i} = \max [y_{\ell 1}, \dots, y_{\ell m}] \\ 0 & \text{otherwise} \end{cases} \quad (4.38)$$

for $\ell = 1, \dots, N, i = 1, \dots, m$. The min operator is needed in Equation 4.38 because in some elements that contain mixing, the total density of material may sum to greater than one [8]. After this first post-processing step, intermediate densities remain at the structure boundaries and at the material interfaces; however, mixing is fully eliminated. In the third row of Figure 4.6, the short column is shown with mixing fully eliminated according to post-processing step 1. Any post-processing is expected to increase the objective function value, as is the case for $R = 0.03$; however, since the material interpolation function (see Equation 4.19 and Figure 4.3d) leads to low stiffness in regions with mixing, the objective function reduces with removal of mixing when the filter radius is large ($R = 0.1$). Note that the volume fraction of each material are slightly reduced with the first post-processing step, but remain within 2% of the expected value.

Intermediate densities can also be eliminated with a second post-processing step in which the materials are projected onto a new mesh that is generated inside an iso-contour of the composite element density field, $y_{\ell}^c = \sum_{i=1}^m y_{\ell i}$, $\ell = 1, \dots, N$.² The iso-contour leads to a smooth boundary, but the interface remains abrupt and inherits features of the underlying discretization. The last row of Figure 4.6 shows the design with mixing eliminated and a smooth boundary obtained using an iso-contour. With proper choice of iso-value (0.5 in this

² Here, `PolyMesher` [16] is used to generate a mesh that conforms to the iso-contour.

case), the volume fraction of each material should not deviate too far from the expected value (within 2% in this case). Note that the objective function value tends to increase with the second post-processing step, as expected.

4.5 3D multi-lattice design examples

In this section, several designs of a 3D cantilever beam considering porous, anisotropic candidate materials (microstructural-materials) are provided. The formulation in Equation 4.1 can handle any material for which the full material tensor is available. The microstructural-materials considered here are characterized by a periodic tessellation of lattice-based unit cells with cubic symmetry, where the lattice elements are cylindrical bars. Effective macroscopic properties for the microstructural-materials are obtained using computational homogenization [146, 147], specifically using an educational MATLAB code [148], and assuming a bulk material with Young’s modulus, $E^0 = 1$, and Poisson’s ratio, $\nu^0 = 0.3$. All examples were run considering sensitivity separation and the heuristic filter implemented in the ZPR design variable update scheme. Additionally, in all cases, the ZPR move limit, $M = 0.15$, and the ZPR intermediate variable exponent, $\alpha = -1$. The plotted results are post-processed according to the approach discussed in subsection 4.4.4 using an isosurface on the composite density field that preserves the volume to within a tolerance of 4%.

4.5.1 Lattice-based material properties

To obtain the effective macroscopic properties for the microstructural-materials from the educational MATLAB code [148], the geometry of the lattice unit cell is inscribed in a hexahedral (hex) mesh that is used in the homogenization computations. Borrowing ideas from the educational polygonal mesh generator, PolyMesher [16], the signed distance of each hex centroid from the boundary of the lattice’s cylindrical bars are computed. Any hex element with a negative signed distance to one of the cylinders’ boundaries is determined to be inside the unit cell structure and is assigned a value of one. All other hex elements are void and are assigned a value of zero. Such implementation facilitates extension to other types of unit cells (e.g., unit cells composed of non-cylindrical bars or plates). The educational code [148] outputs the homogenized stiffness elasticity tensor of microstructural-material i in matrix notation, \mathbf{D}_i^H , and the volume fraction of microstructural-material i ’s

Table 4.2: Optimization input parameters used for the 3D cantilever beam design examples (brackets indicate continuation).

ZPR move limit, M	0.15
ZPR intermediate variable exponent, α	-1
convergence tolerance, tol	0.01
intermediate density penalty parameter, p	[1, 1.5, 2, 2.5, 3]
mixing penalty parameter, γ	[0, 0.2, 0.5, 0.8, 1]
filter radius, R	[0.064, 0.064, 0.064, 0.064, 0.064]
filter exponent, q	[3, 3, 3, 3, 3]
max. number of iterations	[100, 100, 100, 100, 200]

unit cell, \widehat{v}_i , can easily be computed as the sum of the solid hex element volumes. These two properties, $\mathbf{D}_i = \mathbf{D}_i^H$ and \widehat{v}_i , are needed for each candidate microstructural-material defined in Equation 4.1. Computational homogenization for each microstructural-material considered here is performed using a hex mesh with at least $160 \times 160 \times 160$ elements.

Directional tensile and shear moduli, $E'_{11} = 1/S'_{11}$ and $G'_{12} = 1/S'_{66}$, are extracted from the compliance elasticity matrix, $\mathbf{S}_i^H = [\mathbf{D}_i^H]^{-1}$, for any arbitrary orientation of a rotated (prime) coordinate frame and visualized as a 3D surface plot in which the radial coordinate represents the magnitude, E'_{11} or G'_{12} . In the results to follow, in addition to normalizing by the tensile and shear moduli of the bulk material, E^0 and G^0 , E'_{11} and G'_{12} are also normalized by the unit cell volume fraction, \widehat{v}_i , to capture the conflicting volume and stiffness requirements in the volume constrained compliance minimization problem.³

4.5.2 Cantilever beam - effect of varying anisotropic mechanical properties

This example illustrates how porous, anisotropic microstructural-materials affect the geometry, topology, microstructural-material distribution, and structural efficiency of topology optimized parts. A cantilever beam is designed according to the domain and boundary conditions defined in Figure 4.7a, considering free selection from different subsets of seven porous, anisotropic microstructural-materials. The topology optimization problem is solved on half of the domain (symmetry enforced on the $x_1 - x_3$ plane) on a $3 \times 0.625 \times 1$ hex mesh with $192 \times 40 \times 64$ elements, for a total of 491,520 elements. Table 4.2 lists additional topology optimization input parameters for the cantilever problem.

For comparison, a reference beam is designed first, considering a single, solid, isotropic

³ The full derivation of the directional tensile and shear moduli is provided in Appendix F.

material with domain volume fraction limited to $\bar{v} = 0.022$ (Figure 4.7a). Then, using the same volume limit on the bulk material, the beam is designed considering microstructural materials composed of the unit cells indicated to the right of each design in Figure 4.7b-g. The volume constraint is active in all cases and all designs in Figure 4.7a-g have the same volume of bulk material at convergence and within a tolerance of 4% after post-processing. Normalized directional tensile and shear moduli, $E'_{11}/(E^0\hat{v}_i)$ and $G'_{12}/(G^0\hat{v}_i)$, are plotted for the solid, isotropic material and the porous, anisotropic microstructural-materials in Figure 4.7h and i, respectively.

The microstructural-materials in Figure 4.7b-g distribute themselves according to the mechanics of a cantilever beam. That is, microstructural-materials with higher tensile stiffness in the x_3 direction tend toward the tension/compression regions farthest from the neutral axis of the beam and more isotropic microstructural-materials that are also stiffer in shear (in the $x_2 - x_3$ plane) tend toward the inclined members and regions of high shear. Additionally, not all of the available microstructural-materials are used. For example, in Figure 4.7e-g, only the most efficient in tension in the x_3 direction and the most isotropic and efficient in shear in the $x_2 - x_3$ plane are selected.

The beam geometry is also influenced by the available microstructural-materials. For example, the tension/compression members farthest from the neutral axis become increasingly inclined from the supports to the load point as the microstructural-material becomes more isotropic. Notice that the design in Figure 4.7b avoids this inclination as much as possible and the inclination of these members gradually increases as the microstructural-material becomes more isotropic from microstructure 3 in Figure 4.7e and g to microstructure 5 in Figure 4.7f, microstructure 8 in Figure 4.7c, and microstructure 6 in Figure 4.7d.

The normalized objective function values, f/f_0 , in Figure 4.7b-g, indicate how efficient each design is relative to the reference case in Figure 4.7a (in terms of stiffness). The porous structures become more efficient as the microstructural-material freedom is increased (i.e., as the homogenized material properties of the available microstructural-materials become more diverse) because the varying directions and magnitudes of the principal stresses can be better represented. The beam in Figure 4.7g, which has more microstructural-material freedom than the other multi-scale structures, is the most efficient, but still has much higher compliance than the solid, isotropic case because the design space is still limited by the

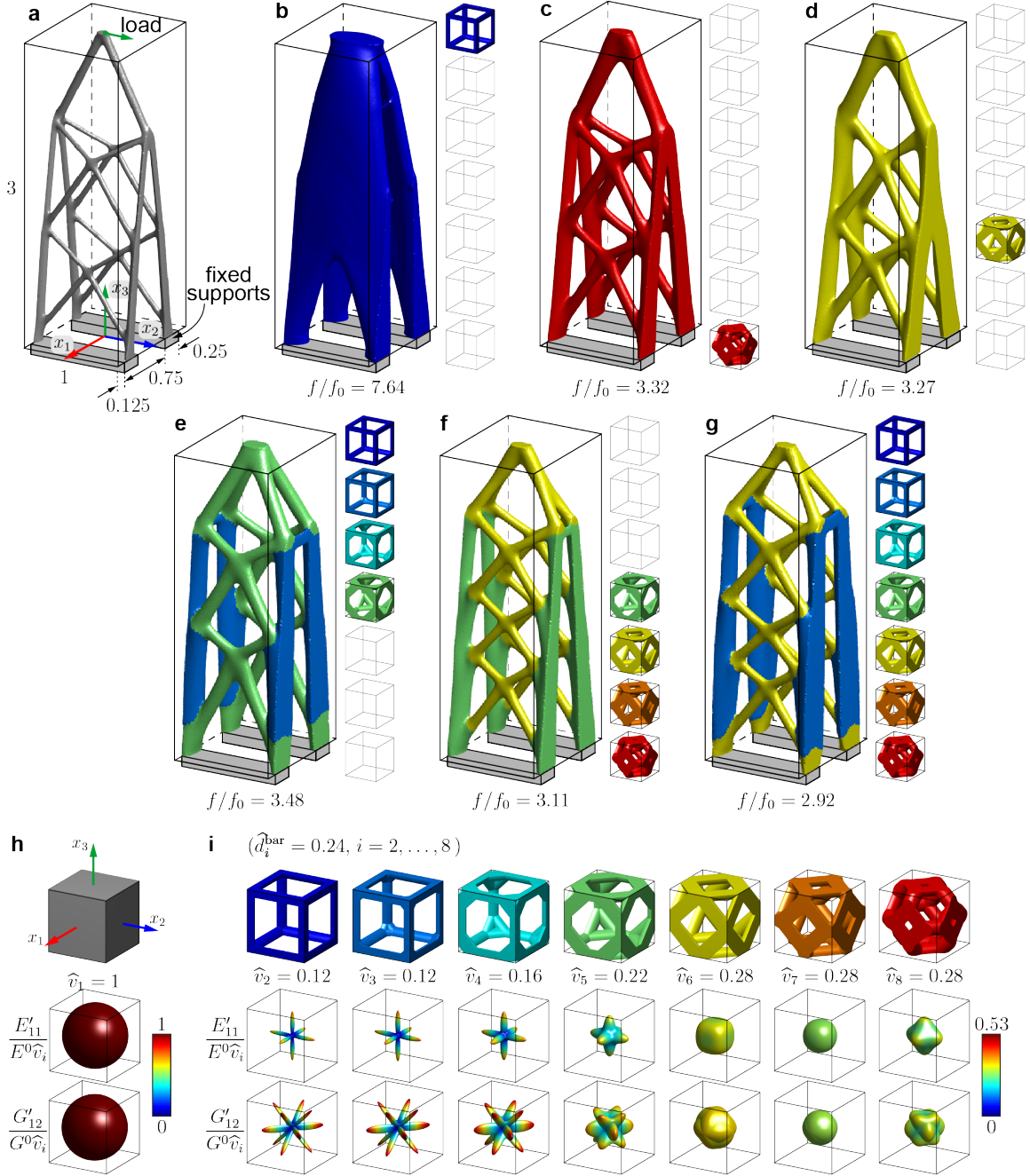


Figure 4.7: Effect of porous, anisotropic microstructural-materials: (a) Design considering a single, solid, isotropic material with objective function value, f_0 ; (b-g) designs considering different subsets, indicated to their right, of seven porous, anisotropic microstructural-materials. A single volume constraint is specified in each case such that the total volume of bulk material occupies, at most, a domain volume fraction of $\bar{v} = 0.022$ (the volume constraint is active in all cases and all designs in (a-g) have the same volume of bulk material). Normalized directional tensile and shear moduli (based on homogenized properties) are provided in (h) for the solid, isotropic material and in (i) for the seven porous, anisotropic microstructural-materials. Variable \hat{d}_i^{bar} is bar diameter and f/f_0 is the objective function normalized to that of the structure in (a).

available microstructural-materials.⁴ Moreover, using low volume fraction lattices as space filling structural elements forces material away from optimal regions and can lead to sub-optimal results. Although the solid, isotropic structure has superior stiffness, multi-scale structures tend to have increased buckling resistance [149] and can provide other biomimetic functionalities (e.g., buoyancy and impact resistance).

4.5.3 Cantilever beam - effect of initial guess and continuation scheme

To demonstrate the effectiveness of the continuation scheme in achieving a “good” local minimum, the 4-microstructural-material example from Figure 4.7f is re-run considering four different initial guesses, without continuation (top row of Figure 4.8) and with continuation (second row of Figure 4.8). In the first column (Figure 4.8a and f), the initial densities are divided evenly among the four candidate microstructural-materials (uniform initial guess). In the last four columns, one of the four candidate microstructural-materials dominates in the initial guess. Specifically, in each case, the initial densities of one of the microstructural-materials are specified at $0.85\bar{v}$ and the other three microstructural-materials’ initial densities are specified at $0.05\bar{v}$ (initial guesses 1-4), as indicated by the bar plots at the bottom of Figure 4.8. In the first row of Figure 4.8, the material interpolation parameters are held constant throughout the optimization ($p = 3$, $\gamma = 1$, and the maximum number of iterations is 600). Without continuation, the solution is significantly biased by the initial guess and the objective function value, f/f_0 (normalized to that of the structure in Figure 4.7a), is compromised. Even in the case of a uniform initial guess (Figure 4.8a), the final design does not perform as well as those obtained using continuation, which are provided in the second row of Figure 4.8. With continuation, three of the four biased initial guesses (Figure 4.8b-d) lead to results very similar to the with a uniform initial guess. One of the four biased initial guesses arrives at a distinctly different local minimum (Figure 4.8j) with different topology and a small region of microstructural-material 8 arising in the design; however, the objective function values of all five designs considering continuation are very similar. Continuation on the material interpolation parameters allows the design to evolve toward the solution of the convex problem, essentially providing an educated initial guess; however, there is still no guarantee that the final solution reaches the global optimum.

⁴ The formulation in Equation 4.1 allows the material space to be *relaxed* by accommodating an arbitrary number of candidate materials; however, depending on the candidate materials available, the material space may actually be *restricted*, as is the case here.

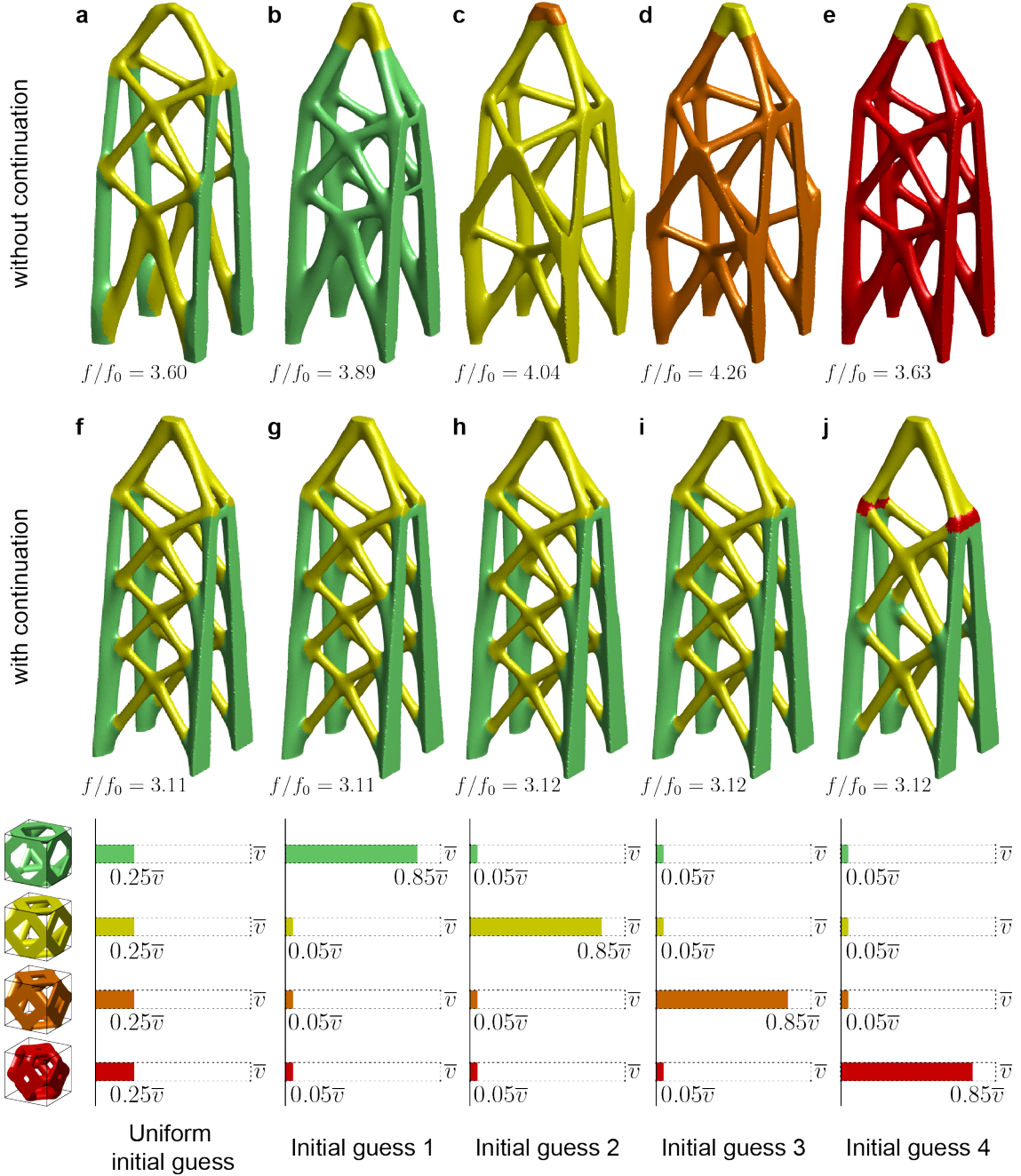


Figure 4.8: Results considering different initial guesses for the example in Figure 4.7f, without and with continuation on the material interpolation parameters, p and γ : In the top row (a-e), the material interpolation parameters are constant ($p = 3$, $\gamma = 1$) and the maximum number of iterations is set to 600. In the second row (f - j), the continuation scheme defined in Table 4.2 is used. In (a) and (f), a uniform initial guess is used. For each case in (b - e) and (g - j), one microstructural-material dominates the initial guess with its densities equal to $0.85\bar{v}$ and all other microstructural-material densities equal to $0.05\bar{v}$. The schematic at the bottom indicates the value of the design variables associated with each candidate material at the initial guess. Variable f_0 refers to the objective function value of the solid, isotropic structure in Figure 4.7a.

CHAPTER 5

AN EDUCATIONAL IMPLEMENTATION OF MULTI-MATERIAL TOPOLOGY OPTIMIZATION (POLYMAT AND POLYANISOMAT)

Two educational codes have been developed that implement the ideas and formulations discussed in chapter 4. These educational codes add to a series of educational MATLAB codes for topology optimization. The first code in the series is `PolyTop`, which performs single-material topology optimization for compliance minimization and can accommodate unstructured meshes [16, 17]. `PolyTop` is written using a modular structure that separates the analysis routines from the optimization formulation, which facilitates extensions to the code. The effectiveness of `PolyTop`'s modular framework is demonstrated by changing only a few lines in the analysis routines and arriving at `PolyFluid`, an implementation for minimization of dissipated power in Stokes flow [150]. `PolyTop` has also been extended to `PolyStress`, which handles single-material, minimum volume structures with local stress constraints [151]. Here, `PolyMat` is built on `PolyTop` with minimal modification to enable solution to the problem in Equation 4.1 with many materials and possibly many local or global volume constraints. `PolyMat` considers 2D problems and is limited to solid, isotropic materials [9]. An extension of `PolyMat`, `PolyAnisoMat`, handles 2D problems with many, possibly porous, anisotropic materials. The general framework of `PolyMat` and `PolyAnisoMat` is the same, but since `PolyAnisoMat` is more general, its implementation is the focus of this chapter. Simplifications to the code for solid, isotropic materials (i.e., the `PolyMat` implementation) are described in subsection 5.2.9.

5.1 Overview

`PolyAnisoMat` leverages the modular framework established in `PolyTop` such that only a few modifications are required to arrive at a general multi-material implementation. In addition to some new required input data for multi-material problems (see subsection 5.2.1), multi-dimensional arrays are needed to store the variables for multiple materials and constraints. MATLAB variables and their dimensions are defined in Table 5.1 for the key quantities and derivatives used in `PolyAnisoMat`. Additionally, a new multi-material inter-

polation function, `MultiMatIntFnc`, works with the existing penalty function, `MatIntFnc`, to interpolate the elemental stiffness properties. For anisotropic materials, the interpolation requires that the constant element stiffness matrices for each candidate material are pre-computed during initialization of the finite element problem. Furthermore, the objective function, `ObjectiveFnc`, is modified to receive `dkdw` instead of the pre-computed `dkdE` used for problems in `PolyTop` with solid, isotropic materials. The constraint function, `ConstraintFnc`, is also modified so that it computes multiple volume fraction constraints. Lastly, the Zhang-Paulino-Ramos Jr. (ZPR, pronounced “zipper”) design variable update scheme [30] is adopted to handle many volume constraints. The ZPR is integrated via a modification to the main function in `PolyAnisoMat` such that the subset of design variables associated with each constraint is passed to `UpdateScheme`, one constraint at a time. The `UpdateScheme` function itself is modified to include the heuristic filter introduced as part of this dissertation. These modifications are detailed in the following section.

5.2 Implementation details

In this section, key aspects of the `PolyAnisoMat` implementation are provided, with an emphasis on changes needed to convert the single material code, `PolyTop`, to accommodate multiple, possibly porous, anisotropic candidate materials.

5.2.1 Input data and PolyScript

A MATLAB script, `PolyScript`, is used to define all input data, which is passed to the `PolyAnisoMat` kernel to perform the analysis and optimization. Three inputs, defined in `PolyScript`, are required in the call to `PolyAnisoMat`: two MATLAB `struct` arrays, `fem` and `opt`, which hold parameters related to the finite element analysis and the topology optimization, respectively; and `Color`, which is an $N_{\text{Mat}} \times 3$ matrix containing an RGB triplet for each candidate material (for visualization of results). The fields stored in `fem` and `opt` are provided in Table 5.2 and are almost identical to those needed for the single material implementation in `PolyTop`. The only new fields are: (1) the cell array, `fem.Mat`, containing the constitutive matrix of each candidate material; (2) the array, `fem.vhat`, containing the solid fractions of each material; (3) the number of candidate materials, `fem.NMat`; (4) the array, `fem.k0`, holding arrays of the local stiffness matrix entries for each candidate material;

Table 5.1: MATLAB variables and their dimensions for the key quantities and derivatives used in PolyAnisoMat.

Quantity	MATLAB variable	Dimension ¹
f	f	1
$\{g_j\}_{j=1}^K$	g	[NConstr \times 1]
$\{z_{\ell 1}, \dots, z_{\ell m}\}_{\ell=1}^N$	z	[NVar \times NMat]
$\{y_{\ell 1}, \dots, y_{\ell m}\}_{\ell=1}^N$	y	[NElem \times NMat]
$\{w_{\ell 1}, \dots, w_{\ell m}\}_{\ell=1}^N$	w	[NElem \times NMat]
$\{v_{\ell 1}, \dots, v_{\ell m}\}_{\ell=1}^N$	v	[NElem \times NMat]
$\{\partial f / \partial z_{\ell 1}, \dots, \partial f / \partial z_{\ell m}\}_{\ell=1}^N$	dfd	[NVar \times NMat]
$\left\{ \left[\frac{\partial(\mathbf{k}_\ell)_{jk}}{\partial w_{\ell 1}} \right]_{j,k=1}^{\text{NDof}_\ell}, \dots, \left[\frac{\partial(\mathbf{k}_\ell)_{jk}}{\partial w_{\ell m}} \right]_{j,k=1}^{\text{NDof}_\ell} \right\}_{\ell=1}^N$	dkdw	[sum(ElemNDof.^2) \times 1]
$\{\partial f / \partial w_{\ell 1}, \dots, \partial f / \partial w_{\ell m}\}_{\ell=1}^N$	dfd	[NElem \times NMat]
$\{\partial w_{\ell 1} / \partial y_{\ell 1}, \dots, \partial w_{\ell m} / \partial y_{\ell m}\}_{\ell=1}^N$	dw	[NElem \times NMat]
$\{\{\partial g_j / \partial z_{\ell 1}\}_{j=1}^K, \dots, \{\partial g_j / \partial z_{\ell m}\}_{j=1}^K\}_{\ell=1}^N$	dgd	[NVar \times NConstr \times NMat]
$\{\{\partial g_j / \partial v_{\ell 1}\}_{j=1}^K, \dots, \{\partial g_j / \partial v_{\ell m}\}_{j=1}^K\}_{\ell=1}^N$	dgd	[NElem \times NConstr \times NMat]
$\{\partial v_{\ell 1} / \partial y_{\ell 1}, \dots, \partial v_{\ell m} / \partial y_{\ell m}\}_{\ell=1}^N$	dv	[NElem \times NMat]

¹ NVar is the number of design variables, NElem is the number of elements, N , in the finite element mesh; NMat is the number of candidate materials, m ; ElemNDof is a vector containing the number of degrees of freedom (DOF) associated with each element in the finite element mesh; and NConstr is the number of constraints, K .

(5) a handle to the multi-material interpolation function, `opt.MultiMatIntFnc`; (6) cell arrays, `opt.ElemInd` and `opt.MatInd`, holding the element indices and material indices, respectively, associated with each constraint; and (8) the number of volume constraints, `opt.NConstr`. Additionally, `opt.VolFrac` is modified such that it is an array containing a volume fraction for each of the volume constraints.

As in `PolyTop`, a call to a mesher (e.g., `PolyMesher` [16]) with a pre-specified domain file can be used to obtain the finite element mesh and boundary condition data for a desired problem. In a similar spirit, for the multi-material implementation with possibly many volume constraints, a call to a pre-specified constraint file can be used to obtain the necessary constraint (and passive region) information. The constraint and passive region specification are detailed in the next two sub-sections.

5.2.2 Constraint specification

The formulation in Equation 4.1 allows for a very general definition of the volume constraints in which a given constraint may control a subset of the candidate materials and/or a subset of the elements (i.e., sub-regions of the domain). The constraints are fully specified to `PolyAnisoMat` by the vector of volume fraction constraints, `opt.VolFrac`, and two cell arrays, `opt.ElemInd` and `opt.MatInd`. In the latter two arrays, each cell entry corresponds to a constraint and contains a vector of element (for `opt.ElemInd`) or material (for `opt.MatInd`) indices associated with that constraint. These vectors of element and material indices correspond to \mathcal{E}_j and \mathcal{G}_j in Equation 4.1, respectively.

To avoid repeated modification of `PolyScript` for different problems, it is convenient to define the constraint information in a separate constraint file that is called from `PolyScript`. In addition to defining the required constraint data, `VolFrac`, `ElemInd`, and `MatInd`, in the constraint file, it is also convenient to specify the constitutive properties, `Mat`, solid fraction properties, `what`, and the array of RGB colors, `Color`, for each of the candidate materials.

In general, it is straightforward to specify the required constraint information; however, if the constraints are defined on many sub-regions or if the sub-regions have complex geometries, it may be difficult to determine the element indices associated with each constraint (i.e., the entries of `ElemInd`). One simple approach is to borrow some of the ideas used in `PolyMesher` for generating complex domains. In the same way that `PolyMesher` uses signed distance functions to implicitly represent the domain geometry, signed distance func-

Table 5.2: List of fields in the PolyAnisoMat input structs. The fields marked with the superscript †, if empty, are populated inside PolyAnisoMat. The fields marked with the superscript ††, are new or modified relative to the single-material implementation in PolyTop.

fem field	
fem.NNode	Number of nodes
fem.NElem	Number of elements
fem.Node	[NNode × 2] array of nodes
fem.Element	[NElem × Var] cell array of elements
fem.Supp	[NSupp × 3] support array
fem.Load	[NLoad × 3] load array
fem.Mat ^{††}	Cell array of constitutive matrices for candidate materials
fem.vhat ^{††}	Array of solid fractions for candidate materials
fem.NMat ^{††}	Number of candidate materials
fem.SElem ^{††}	Elements in passive regions
fem.Reg	Tag for regular meshes
fem.ElemNDof [†]	Array holding number of DOFs of each element
fem.ShapeFnc [†]	Cell array with tabulated shape functions and weights
fem.k0 ^{†, ††}	Arrays of local stiffness matrix entries for candidate materials
fem.k [†]	Array of interpolated local stiffness matrix entries
fem.dkdw ^{†, ††}	Array of interpolated local stiffness matrix derivatives
fem.i [†]	Index array for sparse assembly of fem.k
fem.j [†]	Index array for sparse assembly of fem.k
fem.e [†]	Array of element IDs corresponding to fem.k
fem.ElemArea [†]	Array of element areas
fem.F [†]	Global load vector
fem.FreeDofs [†]	Array of free DOFs
opt field	
opt.zMin	Lower bound for design variables
opt.zMax	Upper bound for design variables
opt.zIni	Initial array of design variables
opt.MatIntFnc	Handle to penalty function
opt.MultiMatIntFnc ^{††}	Handle to material interpolation function
opt.P	Matrix that maps design to element variables
opt.Volfrac ^{††}	Array of specified volume fraction constraints
opt.NConstr ^{††}	Number of volume constraints
opt.ElemInd ^{††}	Cell array of elements associated with each constraint
opt.MatInd ^{††}	Cell array of materials associated with each constraint
opt.Tol	Convergence tolerance on design variables
opt.MaxIter	Max. number of optimization iterations
opt.ZPRMove	Allowable move step in the ZPR update scheme
opt.ZPREta	Exponent used in the ZPR update scheme

tions are used here to implicitly represent the sub-region geometries. For the sub-region associated with each constraint, a vector of distance values can be constructed using the distance functions provided with `PolyMesher` (i.e., `dLine`, `dCircle`, `dRectangle`, `dDiff`, `dIntersect`, `dUnion`). Then, the elements within the sub-region boundary can be determined by evaluating the sign of the distance values. The indices of elements with centroids inside the boundary of constraint j 's sub-region are returned as a vector entry in the j^{th} cell of `ElemInd`.¹

As an illustrative example, the `ElemInd` cell array is constructed for the domain in Figure 5.1a via a number of boolean operations on one rectangle and two circles (c_1 and c_3). To define the constraints shown in Figure 5.1b, two additional circles, c_2 and c_4 , are defined and the signed distance values of the element seeds with respect to these circles are stored in variables `c2` and `c4`, respectively. The signed distance values for the portion of the domain between c_2 and c_4 is stored in variable `int`. With these signed distance values, `ElemInd` can be defined for constraints g_1 to g_5 using the MATLAB code provided in Figure 5.1b.

In addition to the domain files provided with download of `PolyTop`, a number of associated constraint files are provided with download of `PolyMat`. The constraint files can easily be modified for use with `PolyAnisoMat`. A constraint file, `Mbb2AnisoConstraints.m`, is provided in section D.7 of Appendix D as an example.

5.2.3 Passive region specification

A similar approach used to specify the constraints can be used to incorporate passive regions, i.e., regions that are assigned solid or void at the start and do not participate in the optimization. Passive solid regions are described here since void regions can often be achieved in the domain definition; however, if needed, passive void regions can be achieved using similar ideas.

A passive solid region implies that all of the design variables in that region associated with one candidate material are equal to one and the design variables in that region associated with all other candidate materials are equal to zero. Thus, each design variable associated with the passive elements must be set to zero or one during initialization. This information is stored in the `SElemInd` and `SMatInd` cell arrays, which contain a vector of element indices

¹ For “nice” polygonal meshes, the seeds used to construct the mesh are very close to the element centroids and since the seeds are easily accessible from `PolyMesher` [16], the distance values are computed based on the locations of the element seeds.

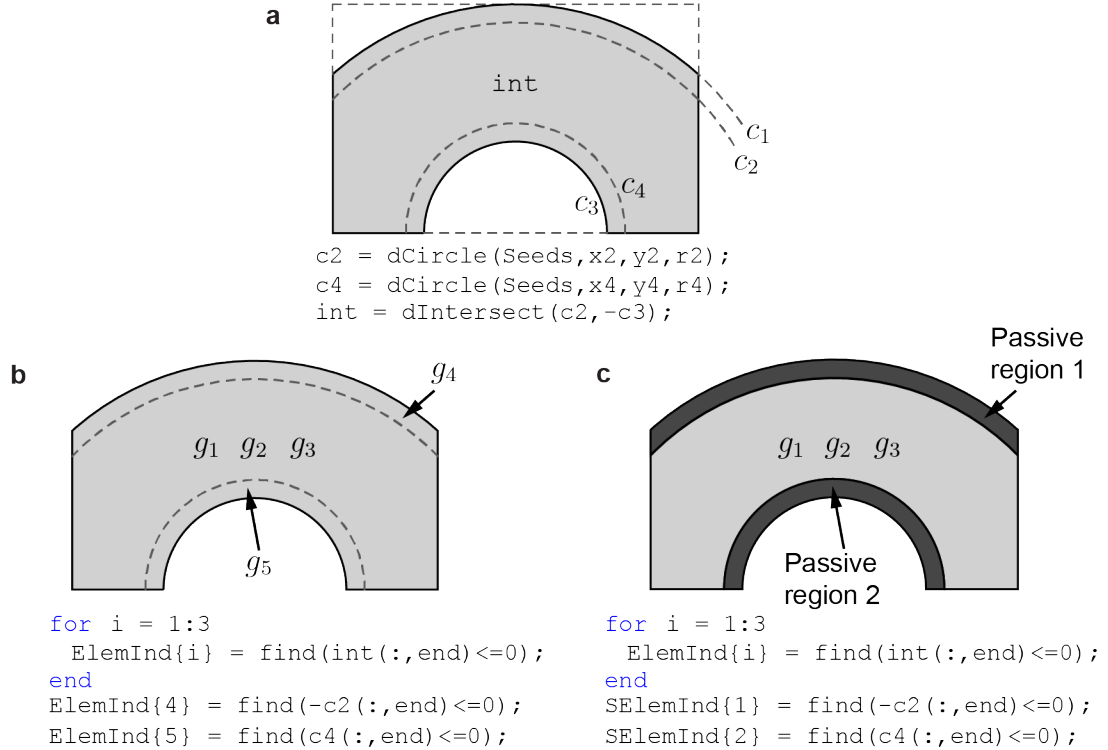


Figure 5.1: Constraint and passive region specification: (a) Curved beam domain and code needed to generate the distance values required to define the constraints and/or passive regions; (b) constraint specification and code needed to assemble `ElemInd`; (c) constraint and passive region specification and code needed to assemble `ElemInd` and `SElemInd`.

for each passive region and the corresponding solid material index for each passive region, respectively. These cell arrays can be defined within the constraint files in a similar way that `ElemInd` and `MatInd` are defined (see an example in Figure 5.1c), but `SElemInd` and `SMatInd` are not passed to the `PolyAnisoMat` kernel function via the `opt struct` since they are only needed during initialization. It is important to ensure that element indices corresponding to the passive regions are left out of the `ElemInd` array so that they are never passed to `UpdateScheme` and will remain as initialized throughout the optimization.

One additional detail is that the matrix associated with the regularization mapping, P , can be modified so that it does not alter the design variables associated with elements in the passive regions. To do so, `fem.SElem`, a vector of all passive element indices, is passed to `PolyFilter`, so that all entries associated with indices of the passive elements are assigned zero during assembly of P (refer to Talischi et al. [17] for additional detail on the `PolyFilter` function).

5.2.4 Initial guess

The $NElem \times NMat$ matrix containing the initial guess, `zIni`, is specified in `PolyScript` by passing `VolFrac`, `ElemInd`, `MatInd`, `SElemInd`, and `SMatInd` to an `InitialGuess` function that sets the initial densities of both the optimizable and non-optimizable regions. For optimizable regions, the `InitialGuess` function evenly distributes the volume fraction specified for each constraint between the materials associated with that constraint, within the appropriate sub-region. For passive regions, the `InitialGuess` function assigns zeros and ones to the design variables in accordance with the values of `SElemInd` and `SMatInd` described in subsection 5.2.3. The code listing for the `InitialGuess` function is included in section D.3 of Appendix D.

5.2.5 Interpolation functions

Function handles to the two interpolation functions, `MatIntFnc` and `MultiMatIntFnc`, are stored in `opt`, allowing the user to specify parameters related to the interpolations in `PolyScript` (e.g., p and γ). Together, `MatIntFnc` and `MultiMatIntFnc`, perform the volume and stiffness interpolations, outputting the volume interpolation, m_V , the stiffness interpolation, $m_E = m_M \circ m_W$, and their derivatives, back to the kernel function, `PolyAnisoMat`. `MatIntFnc` computes quantities explicitly dependent on `y` (i.e., `w` using Equation 4.18, `v` using Equation 4.21, and their derivatives with respect to `y` using Equation 4.25 and Equation 4.27). The only change with respect to the version of `MatIntFnc` used in `PolyTop` is that the solid fractions, `fem.what`, are accounted for in computing the element volume fractions. `MultiMatIntFnc` computes quantities explicitly dependent on `w` (i.e., `fem.k` using Equation 4.19 and its derivatives with respect to `w` using Equation 4.28).

The implementation of Equation 4.19 and Equation 4.28 in `MultiMatIntFnc` takes advantage of the matrix math capabilities in MATLAB and avoids nested for-loops over the elements and candidate materials. For example, an $NElem \times NMat$ matrix, `Prod`, is assembled to contain the product, $w_{\ell i} \prod_{j=1, j \neq i}^m (1 - \gamma w_{\ell j})$, for each (ℓ, i) entry, such that the interpolation in Equation 4.19 can be done with a simple element-wise multiplication and summation:

```
% Compute k
S = 1-gamma*w; S(S==0)=1;
Prod = w.*repmat(prod(S,2),1,NMat)./S;
```

```

Prodtmp = repelem(Prod,ElemNDof.^2,1);
k = sum(Prodtmp.*fem.k0,2);

```

Note that since each component of each element stiffness matrix needs to be interpolated, the elements of the `Prod` matrix are repeated appropriately in `Prodtmp` to match the length of the `fem.k0` matrix. Upon return to the `PolyAnisoMat` kernel function, `k` is assigned to `fem.k`.

Similarly, an $NElem \times NMat$ matrix, `dProd`, is assembled to contain the products needed in Equation 4.28. In this case, a similar element-wise multiplication and summation are done in a loop over the materials:

```

% Compute dkdw
for m = 1:NMat
    M1 = zeros(NElem,NMat);
    M1(:,m) = prod(S,2)./S(:,m);
    M2 = -gamma.*Prod./repmat(S(:,m),1,NMat);
    M2(:,m) = zeros(NElem,1);
    dProd = M2 + M1;
    dProdtmp = repelem(dProd,ElemNDof.^2,1);
    dkdw(:,m) = sum(dProdtmp.*fem.k0,2);
end

```

The full code listing for the `MatIntFnc` and `MultiMatIntFnc` functions are included in section D.4 and section D.5 of Appendix D, respectively.

5.2.6 Analysis functions

Since `PolyAnisoMat` aims to minimize compliance, no change is needed in computing the objective function, `f`, in the `ObjectiveFnc` function used in `PolyTop`; however, the computation of `dfdW` (based on Equation 4.24), is done in a loop over the candidate materials to arrive at an $[NElem \times NMat]$ array. Loops are also added in `ConstraintFnc` to compute the $[NConstr \times 1]$ array of volume fraction constraints, `g`, and the $[NElem \times NConstr \times NMat]$ array of derivatives, `dgdv`. Here, the entires of `opt.ElemInd` and `opt.MatInd` are used to consider the appropriate subset of design variables associated with increment of the loops. The code listings for both the `ObjectiveFnc` and `ConstraintFnc` functions are provided in section D.2 of Appendix D.

5.2.7 Sensitivity analysis

The derivatives on the right-hand-side of Equation 4.22 and Equation 4.23 have all been computed using `MatIntFnc`, `MultiMatIntFnc`, `ObjectiveFnc`, and `ConstraintFnc`. Only the final product needed to obtain `dfdZ` and `dgdZ` is done in the `PolyAnisoMat` function (see lines 19 and 22 in the `PolyAnisoMat` code listing in section D.2 of Appendix D).

5.2.8 Design variable update

As noted in subsection 4.4.2, the ZPR design variable update takes advantage of the fact that the Lagrange multipliers associated with each constraint are independent of one another, and as a result, the set of design variables associated with each constraint can be updated independently [30]. The implementation includes a loop over all volume constraints and only the design variables associated with the current constraint are passed to `UpdateScheme`. The lines of code in `PolyAnisoMat` used to call `UpdateScheme` for each constraint are as follows:

```
%Update design variable and analysis parameters
for c=1:opt.NConstr
    ElemInd = cell2mat(opt.ElemInd(c));
    MatInd = cell2mat(opt.MatInd(c));
    [z(ElemInd,MatInd),Change(c)] = UpdateScheme(...
        dfdz(ElemInd,MatInd),g(c),...
        dgdz(ElemInd,c,MatInd),z(ElemInd,MatInd),opt);
end
```

The structure of the `UpdateScheme` function itself remains unchanged from the original `PolyTop` code, with two exceptions: (1) the heuristic filter is incorporated by passing both `z` and `y` to `UpdateScheme` and using `y` in the recursive update instead of `z` (see line 82 in the `PolyAnisoMat` code listing in section D.2 of Appendix D); and (2) positive sensitivities are ignored by setting them to zero [5] (see line 82 in the `PolyAnisoMat` code listing in section D.2). Sensitivity separation, described in Appendix B, is a more consistent way to handle positive sensitivities [11, 143, 144] and is used for the 3D examples in chapter 4 and chapter 6, but the implementation is not provided with this dissertation. The code listing for the `UpdateScheme` function, without sensitivity separation, is included in section D.2 of Appendix D.

5.2.9 Simplifications for the solid, isotropic case in PolyMat

As described in subsection 4.3.3, the key material parameter contributing to stiffness for a solid, isotropic material is the Young's modulus. Thus, in `PolyMat`, instead of specifying a full constitutive matrix for each candidate material in `fem.Mat`, only the Young's modulus is specified and the Poisson's ratio, `fem.nu0`, is assumed to be the same for each of the candidate materials. Additionally, for fully-dense (solid) materials, the solid fractions are all unity and `fem.what` is not needed.

With the candidate materials distinguished only by their scalar Young's modulus, several simplifications can be made in the implementation. First, the precomputed element stiffness matrices for each candidate material, stored in `fem.k0` in `PolyAnisoMat`, can be replaced by an $[\text{NElem} \times 1]$ vector of interpolated stiffnesses, `E`, computed using Equation 4.20. In `MultiMatIntFnc`, to interpolate a scalar quantity, the $[\text{NElem} \times \text{NMat}]$ matrix, `Prod`, can be used directly without extending it to `Prodtmp`:

```
% Compute E
S = 1-gamma*w; S(S==0)=1;
Prod = w.*repmat(prod(S,2),1,NMat)./S;
E = Prod*E0;
```

The implementation of the derivative in Equation 4.30, `dEdw`, can be simplified in a similar way:

```
% Compute dEdw
for m = 1:NMat
    M1 = zeros(NElem,NMat);
    M1(:,m) = prod(S,2)./S(:,m);
    M2 = -gamma.*Prod./repmat(S(:,m),1,NMat);
    M2(:,m) = zeros(NElem,1);
    dProd = M2 + M1;
    dEdw(:,m) = dProd*E0;
end
dEdy = dwdy.*dEdw;
```

Note that in the last line of the implementation above, the chain rule is used to compute `dEdy`, which is returned back to the kernel function, `PolyMat`, instead of `dkdw` as in `PolyAnisoMat`. The derivative, `dkdE`, needed to compute `dfdE` (see Equation 4.31) in `ObjectiveFnc` has already been precomputed and stored in `fem.k`. The full code listing for `PolyMat` is not provided with this dissertation, but is integrated into an educational paper [9].

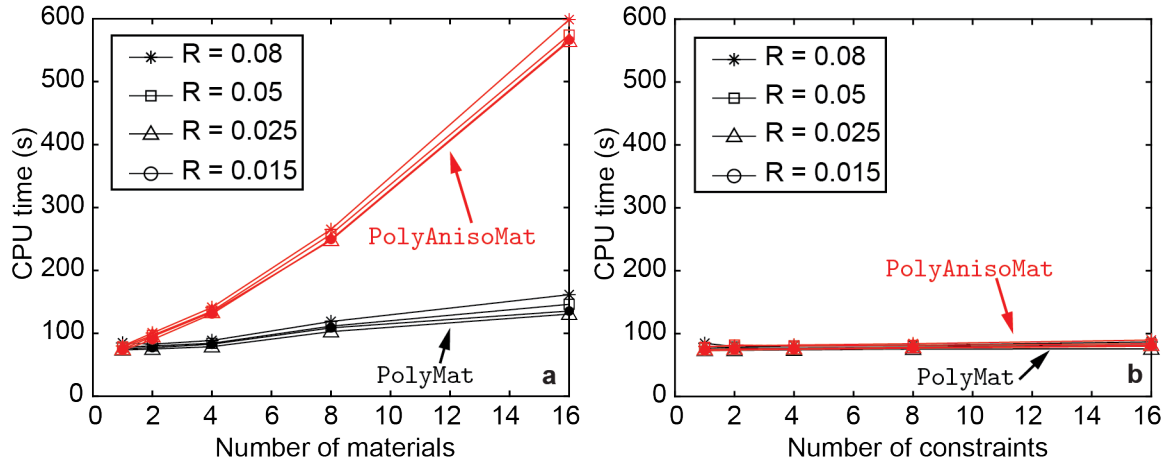


Figure 5.2: CPU time (seconds) for the MBB beam problem with a 300×100 orthogonal mesh run for 200 iterations: (a) A single global volume constraint controlling a varying number of candidate materials; (b) a single candidate material controlled by a varying number of local volume constraints on sub-regions of the domain.

5.3 Efficiency of the 2D implementations

The efficiency of `PolyMat` and `PolyAnisoMat` are studied using the Messerschmitt-Bölkow-Blohm (MBB) beam problem in which the design domain is a rectangle of length 6 and height 1 (see Table E.1). Due to symmetry, only half of the domain is discretized into a 300×100 orthogonal mesh. In each case, the optimization is run for 200 iterations and no continuation of the material interpolation parameters ($p = 3, \gamma = 1$) or filter radius ($R = 0.025$) are considered. All runs in this section are performed using MATLAB R2020b on an Intel(R) Xeon(R) central processing unit (CPU) E5-1660 v3 @ 3.0 GHz with 256-GB random access memory (RAM).

Since `PolyMat` and `PolyAnisoMat` are structured similarly to `PolyTop`, the two codes should perform similarly when the number of materials and constraints are relatively low. In Table 5.3, each code’s runtime breakdown is provided for a single-material design with a single global volume constraint and the performance of the three codes is indeed similar (they are within three seconds of each other). Table 5.3 also breaks down the code runtimes of `PolyMat` and `PolyAnisoMat` for the same MBB beam problem with five candidate materials and two volume constraints. It is shown that the increased CPU time for a typical multi-isotropic-material problem in `PolyMat` is small relative to `PolyTop`, but the CPU time increases more dramatically for a multi-anisotropic-material problem in `PolyAnisoMat`. The

main reason for the increased runtime in `PolyAnisoMat` is due to repeated operations on arrays with size $[\text{sum}(\text{ElemNDof}.\wedge 2) \times \text{NMat}]$ in `MultiMatIntFnc` (refer to subsection 5.2.5).

Additionally, the efficiency degradation of `PolyMat` and `PolyAnisoMat` is studied as the number of materials and the number of constraints increases. In Figure 5.2a, the total CPU time is reported for the MBB beam problem, considering a single global volume constraint controlling 1, 2, 4, 8, and 16 candidate materials, that is, 30k, 60k, 120k, 240k, and 480k design variables, respectively. As the number of materials increases, the sensitivity computations in `MultiMatIntFnc` become increasingly expensive because the size of the arrays increase by a factor of the number of materials and these computations are in a loop over the number of materials. As illustrated in Figure 5.2b, the computational cost increases from just over 1 minute for a single candidate material to about 2-3 minutes in the case of `PolyMat` and about 10 minutes in the case of `PolyAnisoMat` for sixteen candidate materials.

In Figure 5.2b, the total CPU time is reported for the MBB beam problem, considering a single candidate material and local volume constraints controlling each of 1, 2, 4, 8, and 16 sub-regions, that is, 1×1 , 2×1 , 2×2 , 4×2 , and 4×4 sub-regions, respectively. Increased cost due to an increased number of constraints manifests itself in additional time for the ZPR update, since the `UpdateScheme` function must be accessed a number of times in each iteration. However, the ZPR update accounts for a relatively small percentage of the total runtime, and thus, as indicated in Figure 5.2b, the computational cost does not significantly increase as the number of constraints increases. In fact, it takes at most 10 additional seconds to run the problem with 16 constraints as compared to the problem with a single constraint.

Fig. 7 also shows that an increase in the filter radius leads to increased computational time for a given number of constraints and materials. The increase is more pronounced as the number of materials increases; however, based on numerical experimentation, the increased CPU time due to an increased filter radius is negligible. As a side note, although only the stiffest material arises in all of the designs obtained in this efficiency study (aside from the 5-material, 1-constraint example in Table 5.3), the designs are similar, but not identical, since the initial guess is different in each case.

Table 5.3: Code runtime breakdown comparison of PolyTop (1 material, 1 constraint), PolyMat (1 material, 1 constraint), PolyMat (5 materials, 2 constraints), PolyAnisoMat (1 material, 1 constraint), and PolyAnisoMat (5 materials, 2 constraints) for the MBB beam design run for 200 iterations (times are in seconds with percentage of total runtime of PolyScript in parentheses)

	PolyTop (1 mat., 1 constr.)	PolyMat (1 mat., 1 constr.)	PolyMat (5 mat., 2 constr.)	PolyAnisoMat (1 mat., 1 constr.)	PolyAnisoMat (5 mat., 2 constr.)
Computing \mathbf{P}	3.56 (4.5%)	3.72 (4.9%)	3.40 (4.0%)	3.67 (4.8%)	3.72 (2.3%)
Assembling \mathbf{K}	14.36 (18.3%)	14.37 (19.0%)	14.42 (16.8%)	11.42 (14.9%)	12.28 (7.4%)
Solving $\mathbf{KU} = \mathbf{F}$	38.96 (49.7%)	38.76 (51.4%)	41.31 (48.1%)	37.13 (48.5%)	42.13 (25.5%)
Interpolating volume and materials	0.11 (0.1%)	0.16 (0.2%)	0.58 (0.7%)	0.82 (1.1%)	5.66 (3.4%)
Computing constraints	0.02 (0.0%)	0.11 (0.1%)	0.25 (0.3%)	0.11 (0.1%)	0.27 (0.2%)
Computing constraint sensitivities	0.05 (0.1%)	0.16 (0.2%)	1.90 (2.2%)	0.15 (0.2%)	1.80 (1.1%)
Computing compliance sensitivities	5.75 (7.3%)	5.86 (7.8%)	8.47 (9.9%)	8.77 (11.5%)	78.00 (47.1%)
Updating the design	7.01 (8.9%)	2.28 (3.0%)	4.77 (5.5%)	2.32 (3.0%)	5.33 (3.2%)
Plotting the solutions	6.51 (8.3%)	6.86 (9.1%)	7.19 (8.4%)	6.92 (9.0%)	7.34 (4.4%)
Total time of PolyScript	78.44	75.44	85.93	76.53	165.51

5.4 2D multi-material design examples

In this section, several 2D design examples are provided to demonstrate the capabilities of the educational codes. The first three examples use `PolyMat` with solid, isotropic candidate materials (Poisson’s ratio, $\nu^0 = 0.3$). The last example considers `PolyAnisoMat` with porous, anisotropic candidate materials. Unless otherwise noted, the optimization input parameters used for all examples in this section are those provided in Table 5.4. Note that continuation on the material interpolation parameters (p and γ) is considered, each continuation step is allowed to run for a maximum of 200 iterations, and the filter radius, R , which is different for each problem, is turned off in the final continuation step.

Table 5.4: Optimization input parameters used for 2D multi-material design examples (brackets indicate continuation and $R = -1$ indicates no filter).

ZPR move limit, M	0.2
ZPR intermediate variable exponent, α	-1
convergence tolerance, tol	0.01
intermediate density penalty parameter, p	[1, 1.5, 2, 3, 4]
mixing penalty parameter, γ	[0, 0.3, 0.5, 1, 1]
filter radius, R	[R , R , R , R , -1]
filter exponent, q	[1, 1, 1, 1, 1]
max. number of iterations	[200, 200, 200, 200, 200]

5.4.1 Michell cantilever (`PolyMat`) - Volume constraint specification

In Figure 5.3, a “Michell” cantilever with circular support [152] is used to illustrate three main types of volume constraints that can be specified with the formulation in Equation 4.1 and how to implement each case: 1) global constraints that control many materials (Figure 5.3a); 2) global constraints that control a single material (Figure 5.3b); and 3) local constraints that control one or more materials (Figure 5.3c). The domain and boundary conditions for the Michell cantilever are provided in Figure 5.3a, where the dimensional parameters are defined as $H = 4$, $L = 5$, and $r = 1$, and the load $P = 1$. A 90,000-element polygonal finite element mesh is generated using `PolyMesher` with a horizontal line of symmetry (symmetry is on the mesh only, i.e., symmetry is not enforced on the design variables). In all three cases, fifteen solid, isotropic candidate materials are available - the stiffest material is shown in dark red and the least stiff is shown in dark blue. The sub-regions and materials associated

with each constraint for cases 1, 2, and 3 are provided at the left of Figure 5.3a, b, and c, respectively. The filter radius used for all three cases is $R = 0.05$.

In case 1, a single global volume constraint ($\bar{v} = 0.45$) that controls all elements in the domain and all candidate materials is considered. The MATLAB code used to generate the cell arrays, `ElemInd` and `MatInd`, for case 1 is as follows:

```
%Case 1: Single global constraint
NMat = 15;
VolFrac = 0.45;
mat = linspace(NMat,1,NMat)'./NMat;
matorder = randperm(NMat);
ElemInd{1} = linspace(1,NElem,NElem);
MatInd{1} = matorder;
```

In case 2, fifteen global volume constraints are considered ($\bar{v}_j = 0.45/15$, $j = 1, \dots, 15$): one for each of the fifteen candidate materials. The MATLAB code used to generate the cell arrays, `ElemInd` and `MatInd`, for case 2 is as follows:

```
% Case 2: Fifteen global constraints (one for each material)
NMat = 15;
NConstr = NMat;
VolFrac = 0.45/NMat*ones(NMat,1);
mat = linspace(NMat,1,NMat)'./NMat;
matorder = randperm(NMat);
for c = 1:opt.NConstr
    ElemInd{c} = linspace(1,NElem,NElem);
    MatInd{c} = matorder(c);
end
```

Finally, in case 3, fifteen local volume constraints on each side of the domain's horizontal line of symmetry (i.e., thirty local constraints, $\bar{v}_j = 0.45$, $j = 1, \dots, 30$) are considered, such that a single material is available in each subregion. The MATLAB code used to generate the cell arrays, `ElemInd` and `MatInd`, for case 3 is as follows:

```
% Case 3: Thirty local constraints
NMat = 15;
NConstr = 2*NMat;
VolFrac = 0.45*ones(NConstr,1);
mat = linspace(NMat,1,NMat)'./NMat;
matorder = [randperm(NMat) randperm(NMat)];
for c = 1:NConstr
    ElemInd{c} = find(Dist{c} <= 0);
    MatInd{c} = matorder(c);
end
```

Note that in the code listing for case 3, `Dist` is a cell array in which each cell represents a constraint and contains the distance values of each element for the sub-region associated with that constraint. In particular, `PolyMesher` is used to generate the sub-regions for case 3, which are defined by the “nodes” and “elements” of a polygonal mesh with 15 “elements” on each side of the horizontal line of symmetry of the Michell domain intersected with a circle of radius H that is tangent to the point of load application.

The designs obtained for cases 1, 2, and 3 are provided on the right of Figure 5.3. As expected, when one global constraint controls all fifteen candidate materials (Figure 5.3a), the final design is composed only of the stiffest candidate material. When each material is controlled by a separate global constraint (Figure 5.3b), all fifteen candidate materials appear in the final design and they are distributed such that the stiffer materials are located where strain is expected to be high and more compliant materials are located where strain is expected to be low. In the case of local volume constraints (Figure 5.3c), all of the constraint boundaries are respected. Also note that the design in Figure 5.3b is not perfectly symmetric about the horizontal centerline.

5.4.2 Curved beam (`PolyMat`) - Passive regions

Here, the curved beam problem described in Figure 5.4a is used to demonstrate the ease with which passive regions (i.e., regions that do not participate in the optimization) can be specified using the `PolyMat` framework. The dimensional parameters defining the domain are $H = 1.25$, $L = 2$, $c_1 = (x_{c1}, y_{c1}, r_{c1}) = (0, 0, 1.5)$, $c_2 = (0, 0, 1.4)$, $c_3 = (0, 0.25, 0.5)$, and $c_4 = (0, 0.25, 0.6)$, and the load parameter $P = 1$. The curved beam is discretized into 100,000 polygonal finite elements via `PolyMesher` and is designed considering three candidate materials with properties given in Figure 5.4b. The top and bottom curved surfaces are considered passive regions and are assigned material 1 and 2, respectively, in the constraint file. Note that the elements with seeds inside of the passive regions are assigned to `SElemInd` and not to `ElemInd`; thus, they are not updated after initialization. Three constraints each control one of the three candidate materials in the optimizable region such that each of the three candidate materials is limited to occupy no more than 10% of the optimizable region volume, i.e., $\bar{v}_j = 0.1, j = 1, \dots, 3$. The filter radius used in design is $R = 0.016$. Figure 5.4c shows the resulting design, where it is clear that the passive regions have been respected.

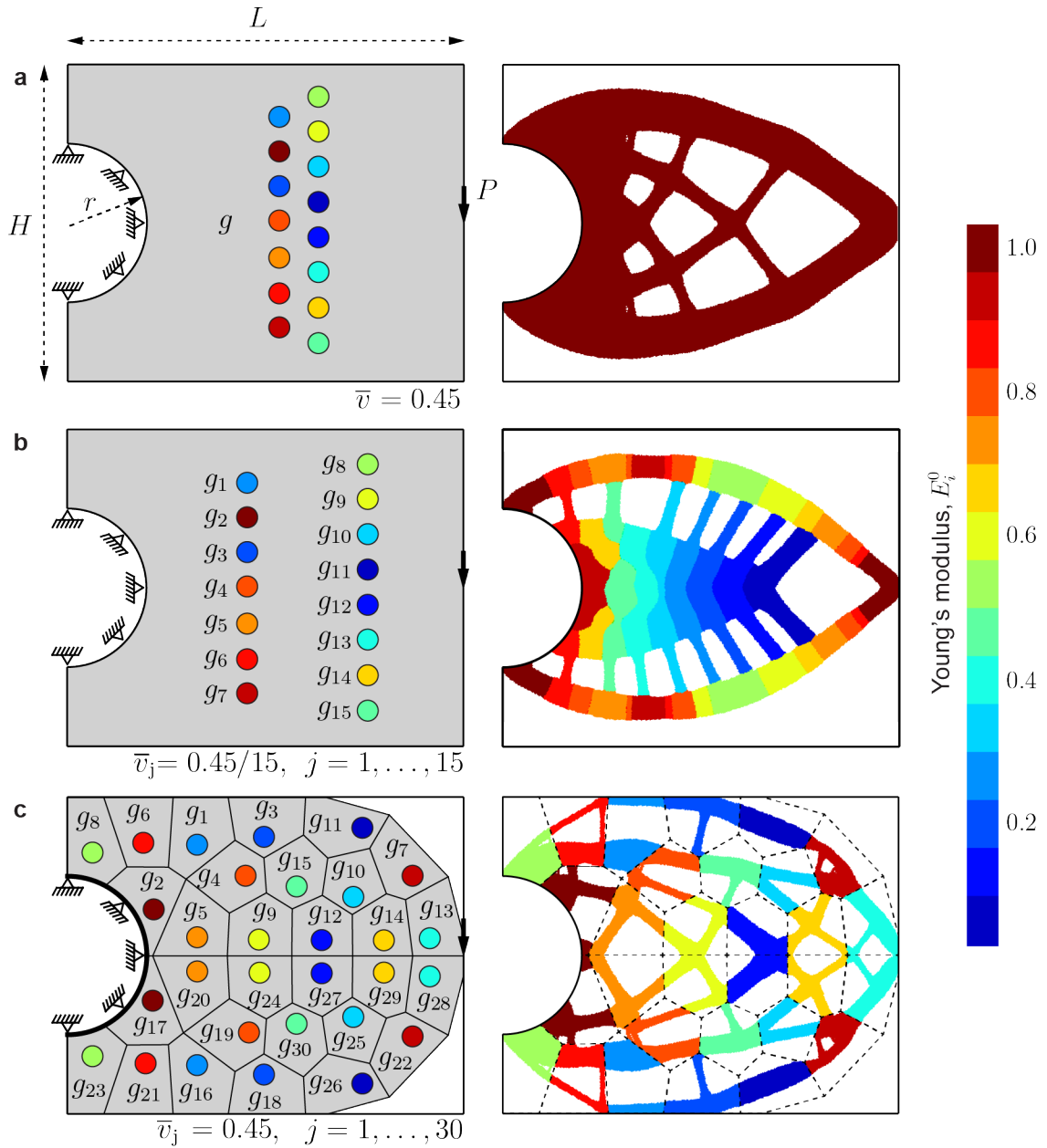


Figure 5.3: Michell cantilever with circular support to demonstrate various ways the volume constraints can be specified: (a) In case 1, a single global constraint ($\bar{v} = 0.45$) controls all fifteen candidate materials; (b) in case 2, fifteen global constraints ($\bar{v}_j = 0.45/15$, $j = 1, \dots, 15$) each control a single candidate material; (c) in case 3, thirty local constraints ($\bar{v}_j = 0.45$, $j = 1, \dots, 30$) that are symmetric about the horizontal center line each control a single candidate material.

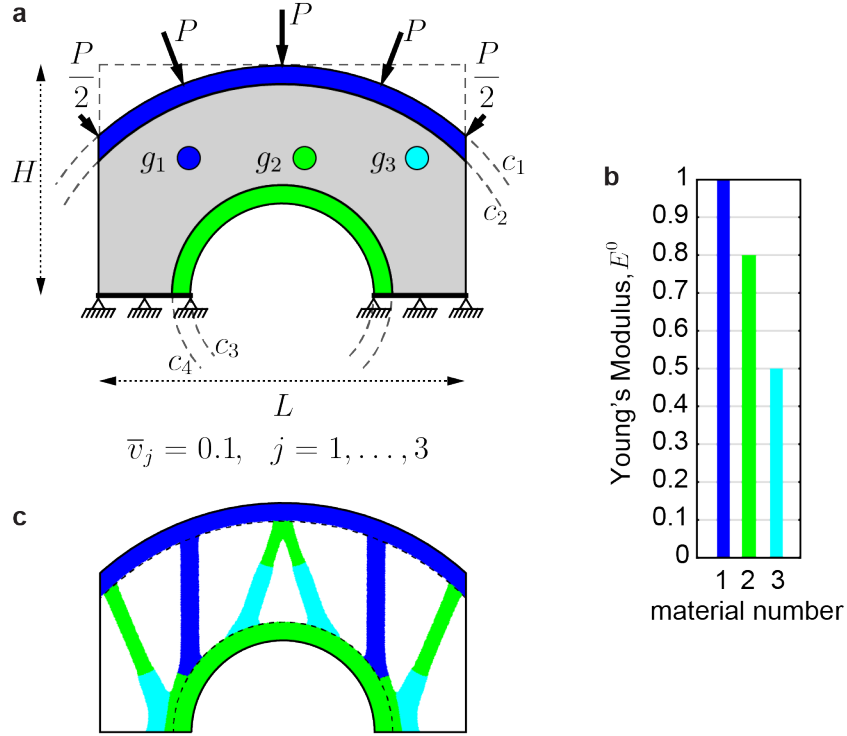


Figure 5.4: Curved beam problem: (a) Domain, boundary conditions, constraint specification, and passive region specification; (b) Young's modulus of the three candidate materials available; (c) optimized design.

5.4.3 Flower and serpentine (PolyMat) - Local volume constraints

In this example, constraint sub-region and material property specification are used to control geometric features. First a single candidate material is considered in the serpentine domain shown with boundary conditions in Figure 5.5a, where the centers of the circles are $O_1 = (0, -2.6458)$ and $O_2 = (9, 5.2915)$, circles c_1 and c_{14} each have radius 8, circles c_9 and c_{10} each have radius 4, and the load $P = 1$. The domain is discretized into 75,000 polygonal finite elements using `PolyMesher`. Figure 5.5b shows the result of optimization considering a single global constraint that limits the material to occupy no more than 50% of the total domain volume, i.e., $\bar{v} = 0.5$. Seeking a structure with finer details at the left end, the domain is divided into the 84 sub-regions shown in the top portion of Figure 5.5c. The single candidate material is limited to occupy no more than 50% of each sub-region volume, i.e., $\bar{v}_j = 0.5, j = 1, \dots, 84$, and the resulting design is provided in the bottom portion of Figure 5.5c. As expected, the design with more constraints has a higher objective value. Continuation on the material interpolation parameters (p and γ) is not considered for the

Table 5.5: Optimization input parameters used for the serpentine problem (brackets indicate continuation and $R = -1$ indicates no filter).

ZPR move limit, M	0.2
ZPR intermediate variable exponent, α	-1
convergence tolerance, tol	0.01
intermediate density penalty parameter, p	[3, 3]
mixing penalty parameter, γ	[1, 1]
filter radius, R	[0.1, -1]
filter exponent, q	[1, 1]
max. number of iterations	[500, 500]

serpentine problem; Table 5.5 lists the optimization input parameters used for both of the serpentine designs.

Next, the donut-shaped domain with torsion boundary conditions shown in Figure 5.6a [15] is considered, where the diameters of the inner and outer circles are 0.25 and 2, respectively, and the load parameter $P = 1$. The domain is discretized into 100,000 polygonal finite elements using `PolyMesher` and a filter radius of $R = 0.01$ is used in design. Five constraints are defined, each in a different sub-region of the domain, and each controlling one of the five candidate materials defined in Figure 5.6b. The resulting design in Figure 5.6c has no lines of symmetry, in contrast to the single material design provided in Figure 5.6d, which has five lines of symmetry and resembles a flower (perfect symmetry is not achieved due to the polygonal mesh). Notice that the complexity of features in the five-material design is correlated with the stiffness of the candidate material. For example, the portion of the structure made of the most compliant material, shown in cyan, has the least complex geometry, and complexity gradually increases clockwise around the domain as the materials become stiffer. Additionally, the members tend to become thinner as the materials increase in stiffness. Finally, “flower pedals” can be identified in the multi-material design as each “Michell-like” structure touching each load point. Notice that the pedals tend to become smaller moving clockwise around the domain from the most compliant material (cyan) to the stiffest material (red).

5.4.4 MBB beam (`PolyAnisoMat`) - Porous, anisotropic material specification

In the final example, the MBB beam defined in Figure 5.7a is used to demonstrate how porous, anisotropic candidate materials can be specified for use with `PolyAnisoMat` and

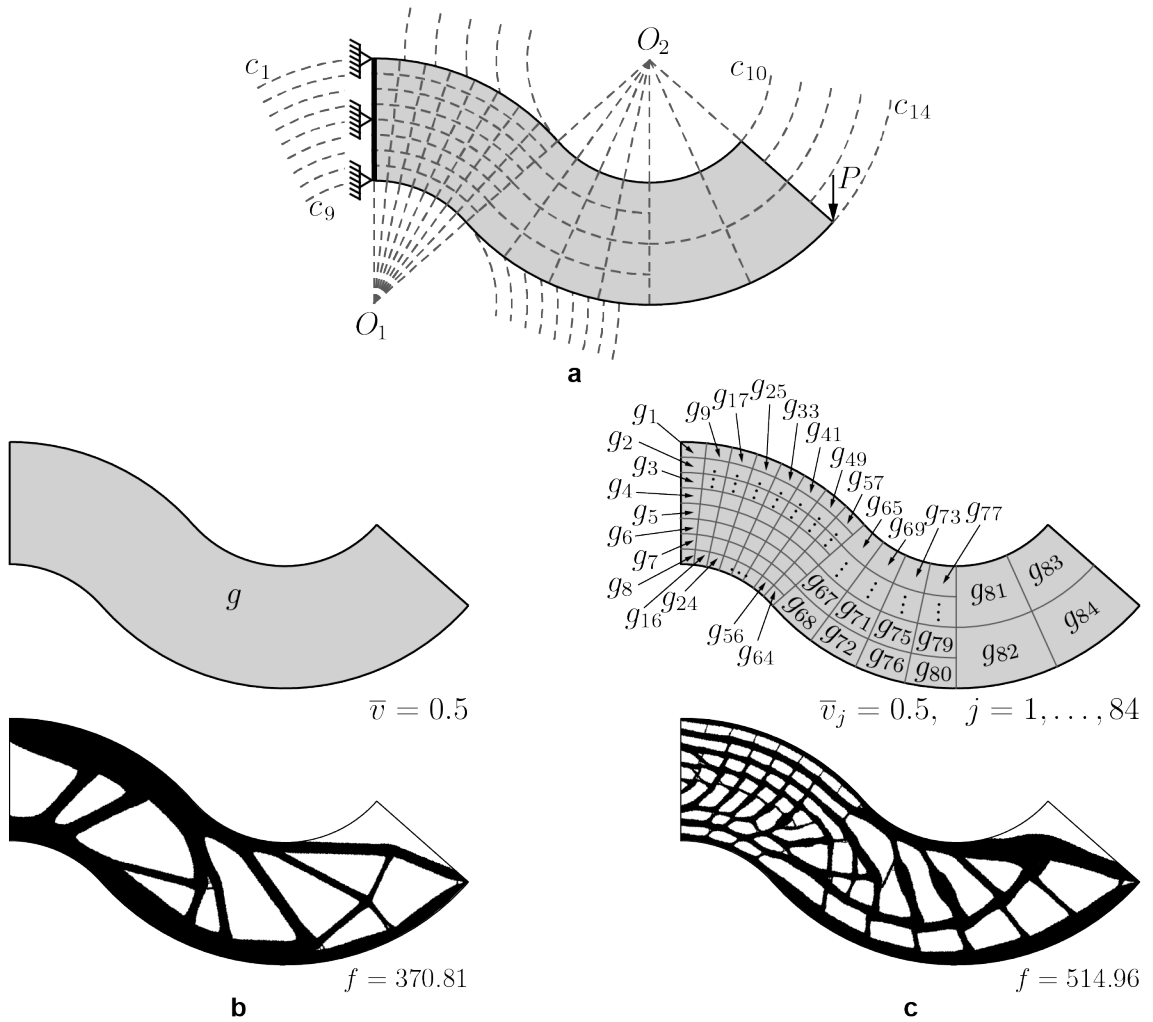


Figure 5.5: Serpentine problem: (a) Domain and boundary conditions; (b) the single region used to define a single global volume constraint with volume fraction, $\bar{v} = 0.5$, (top) and resulting optimized design (bottom); (c) the 84 sub-regions used to define 84 local volume constraints each with volume fraction, $\bar{v}_j = 0.5, j = 1, \dots, 84$, (top) and resulting optimized design (bottom).

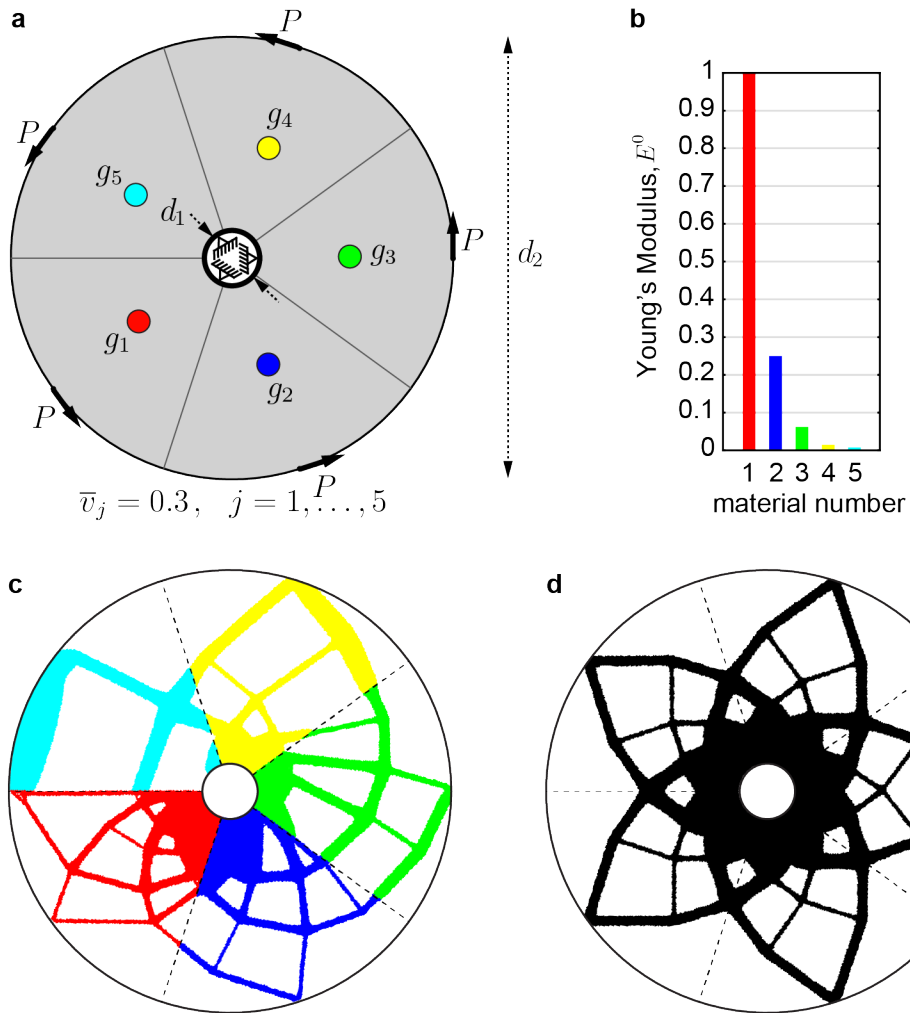


Figure 5.6: Flower problem: (a) Domain, boundary conditions, and constraint specification for the multi-material design; (b) Young's modulus of the five candidate materials available in the multi-material design; (c) multi-material design; (d) corresponding single-material design.

how materials of this nature (i.e., those with directional stiffness) can lead to multi-material designs even when a single, global volume constraint is specified (in contrast to the solid, isotropic case in Figure 5.3a). Dimensional parameters of the MBB beam are $L = 3$ and $H = 1$. The load, $P = 0.5$, is divided evenly among nodes within a distance of $L/16$ from the left edge of the domain. The domain is discretized into 80,000 polygonal finite elements via `PolyMesher` and a filter radius of $R = 0.025$ is considered in design.

Two porous, anisotropic microstructural-materials are controlled by a single, global volume constraint with volume fraction limit of the bulk material, $\bar{v} = 0.3$. The bulk materials of candidate microstructural-materials 1 and 2 have Young's modulus, $E_1^0 = 0.75$ and $E_2^0 = 1$, respectively, and Poisson's ratio, $\nu_1^0 = \nu_2^0 = 0.3$. The stiffness elasticity tensors of each candidate microstructural-material, \mathbf{D}_1^H and \mathbf{D}_2^H , are obtained via computational homogenization [146, 147]. The MATLAB code used to define the candidate microstructural-materials is as follows (see the `Mbb2AnisoConstraints` constraint file in section D.7 of Appendix D):

```
%Cross with horizontals top and bottom, E=0.75, nu=0.3, t = 0.2
Mat{1,1} = [0.283994135147905,0.0991736450570570,0;
           0.0991736450570570,0.160516511103827,0;
           0,0,0.104359003867811];
%Cross with verticals left and right, E=1, nu=0.3, t = 0.2
Mat{2,1} = [0.214897989423356,0.132122022785080,0;
           0.132122022785080,0.377073505855695,0;
           0,0,0.139172263121906];
vhat = [0.609;0.609];
Color = [5, 63, 255; 255, 150, 0]./255;
MatInd{1} = linspace(1,NMat,NMat);
```

Microstructural-materials 1 and 2 have the same geometry except that one is rotated 90° with respect to the other. Since the bulk material of microstructural-material 1 has Young's modulus 75% of that composing microstructural-material 2, the directional tensile modulus contour of microstructural-material 1 is 75% the size of that of microstructural-material 2 and rotated by 90° (refer to Figure 5.7b²). As a result, notice that the tensile modulus of microstructural-material 1 exceeds that of microstructural-material 2 for tension oriented in the horizontal direction and up to about $\pm 34^\circ$ from horizontal. Indeed, the final design in Figure 5.7c makes use of microstructural-material 1 at the top and bottom of the beam where horizontal tension and compression are induced by bending and in any members

² The 2D directional tensile and shear moduli plots in Figure 5.7b are constructed using the same theory described in Appendix F for 3D.

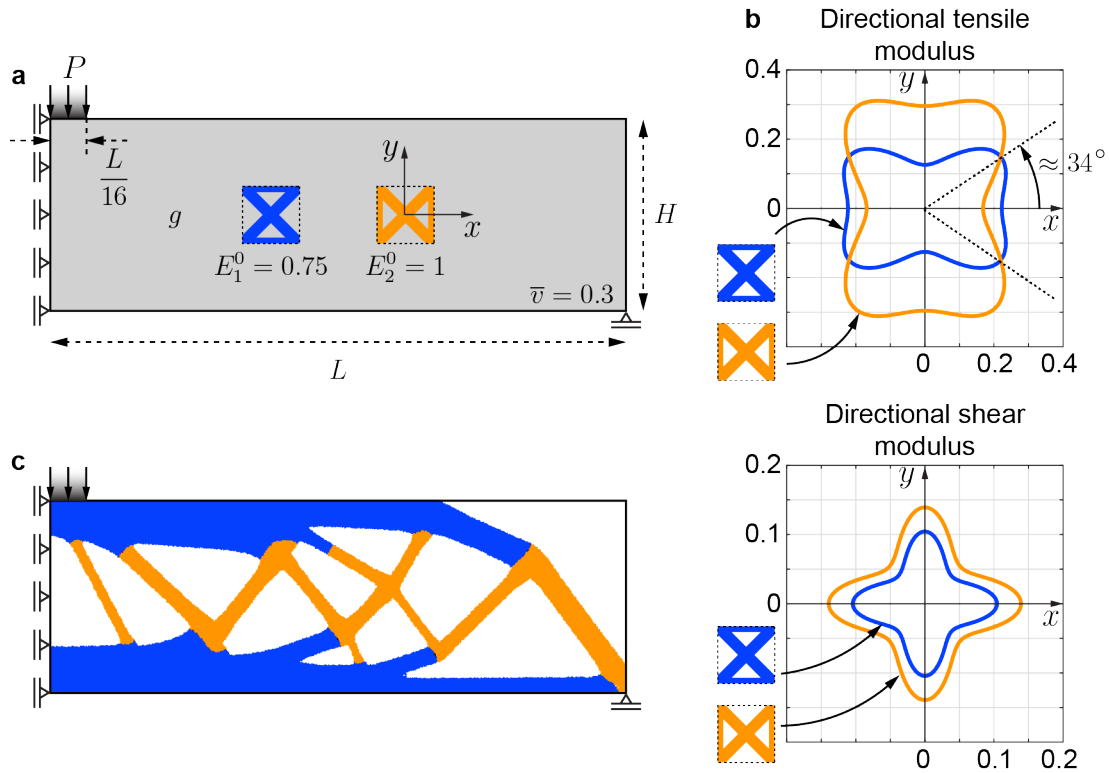


Figure 5.7: MBB problem with porous, anisotropic candidate microstructural-materials: (a) Domain, boundary conditions, and material/constraint specification; (b) directional tensile (top) and shear (bottom) modulus plots of the candidate microstructural-materials; (c) design.

oriented less than 34° from the horizontal. The placement of microstructural-material 2 toward the neutral axis of the beam is likely most heavily influenced by the need for higher tensile stiffness along the length of the truss-like members; nevertheless, it is worth noting that microstructural-material 2 has higher shear stiffness for all possible orientations and its placement may also be influenced by the higher shears expected toward the neutral axis.

CHAPTER 6

OPTIMAL AND CONTINUOUS MULTI-LATTICE EMBEDDING

Due to increased geometric freedom at a widening range of length scales and access to a growing material space, additive manufacturing has spurred renewed interest in topology optimization of parts with spatially-varying material properties and structural hierarchy [153]. Simultaneously, additive manufacturing is pushing architected materials to new limits. For example, two-photon lithography enables unit cells with 300 nm minimum feature size [154, 155] and projection micro-stereolithography enables hierarchical structures spanning length scales from tens of nanometers to tens of centimeters [156, 157]. Nevertheless, topology optimization considering multiple architected materials has yet to be sufficiently integrated with micro-/nano-scale additive manufacturing to achieve freeform, multi-scale, biomimetic structures. In this chapter, a continuous multi-lattice embedding scheme is discussed that enables manufacturing of topology optimized parts designed considering a *relaxed* (or *restricted*) material space consisting of multiple lattice-based microstructural-materials.

6.1 Overview

A limiting factor in manufacturing parts with spatially-varying, hierarchical architectures is effectively representing the 3D geometry in a way that is meaningful (to a 3D printer). In fact, multi-material and multi-scale additive manufacturing are shifting from impractical surface representations to procedural [158] or voxel representations [159–161] that are less memory intensive. Accordingly, here, the microstructural-materials are embedded directly into the 3D printer slices, avoiding prohibitively expensive surface representations (i.e., STLs) of the microstructure-embedded parts. The slice-based approach is also scalable in that it allows the macroscale geometry to be scaled arbitrarily, while the micro-scale features are maintained at the resolution of the 3D printer (i.e., a large separation of length scales can be pursued). Another limiting factor in manufacturing spatially-varying, hierarchical structures is handling connectivity between microstructural-materials with dissimilar micro-scale geometries. This challenge is overcome here by introducing functionally graded transitions over which the microstructure unit cells are interpolated, ensuring well-connected interfaces

with smoothly-varying micro-scale geometries. The transition region is shown, numerically, to have little impact on the global structural behavior of the part (i.e., the compliance function for which it is designed). Capabilities of the continuous multi-lattice embedding scheme are demonstrated using an in-house MATLAB slicing and embedding code and a commercially available masked-stereolithography (m-SLA) 3D printer from Prusa Research.

6.2 Related work

Fundamental to modern topology optimization is the homogenization-based method proposed by Bendsoe and Kikuchi in 1988 [19, 162, 163]. In this method, an optimized material distribution is determined using homogenized microstructural-material properties, interpolated as a function of the microstructure’s porosity and orientation. A key limitation that sidelined this approach in favor of other methods (e.g., SIMP [13, 14]) was manufacturability, due to geometric complexity, small length scales, and connectivity of the microstructures. With recent advances in additive manufacturing, which easily handles complex geometries and is becoming more capable at a wide range of length scales, several attempts to revive and extend the homogenization-based approach have been pursued. Here, several of these attempts are reviewed with an emphasis on the efforts to manufacture the resulting designs or to address key manufacturing challenges (e.g., connectivity).

Bendsoe and Kikuchi’s method has been simplified by considering a unit cell of fixed orientation and allowing only the density or geometry to vary in such a way that connectivity is always guaranteed [164–170]. This approach has led to a number of additively manufactured, microstructure-embedded, rectangular beams (fixed macrostructure), often extruded from 2D results [164, 166, 168, 170–172]. In these cases, which rely on laser-SLA or photopolymer jetting, macrostructure lengths range from 120-300 mm and unit cell edge-lengths range from 2.5-8 mm.

A library of unit cells covering a wider range of properties can also be fed to a homogenization-based topology optimization formulation. In this case, connectivity is typically addressed post-optimization by relying on non-uniqueness of unit cell geometry for a given material property and strategically selecting compatible unit cells. Structures designed in this way have been manufactured using photopolymer jetting with multi-material unit cells embedded in the topology optimized macrostructures [117]. Other unit cell databases have also been generated and used to tile fixed-geometry macrostructures for targeted local properties.

As a result, box-like macrostructures on the order of 50-180 mm with 5 mm unit cells were fabricated by digital light processing (DLP)-SLA [116] and cartoon-like 3D geometries on the order of 150-300 mm with 8 mm unit cells were fabricated by selective laser sintering (SLS) [115].

Others have attempted to “de-homogenize” the results of homogenization-based topology optimization by projecting them onto a distorted grid according to the spatial variation of the unit cell design parameters and orientations [173–180]. Parts designed in this way have been manufactured using SLA in 2D [180] and 3D [175] with relatively low resolution. Higher resolution designs have been obtained from the homogenization-based results by using field-aligned parameterizations, typical in meshing algorithms, to extract globally-connected lattices that are conformal to the 3D stress field. Using laser-SLA, microstructures with 0.5 mm minimum feature size (unit cells on the order of 2-2.4 mm before scaling and orienting) were printed within complex macrostructures on the order of 100-150 mm [181].

Rather than projecting the results after performing homogenization-based topology optimization, a mapping function that determines the spatial variation of a unit cell scaling, rotation, and distortion can be optimized directly. Although manufacturing has yet to be demonstrated, this method handles connectivity and seems amenable to additive manufacturing. It has been pursued in 2D considering a single, fixed unit cell [182, 183] and in 3D considering a continuous “menu” of unit cells [184], leading to functionally graded and oriented microstructures embedded in a fixed macrostructure.

A different approach to multi-scale topology optimization is to design the microstructural unit cell(s) concurrently with the macrostructure geometry [185–191]. Concurrent topology optimization provides significant design freedom, but tends to be computationally expensive and connectivity is only guaranteed when a single microstructure is used throughout the entire design [187, 192–195]. For cases in which multiple microstructures are designed concurrently with the macrostructure, connectivity has been enforced by adding passive solid regions on the boundaries of the unit cell domains [196–201], considering a pseudo load or nonlinear diffusion across the unit cell boundaries [196], adding a connectivity index to the objective function [202], or introducing design variable linking [203]. A continuous function that interpolates the unit cell geometry has also been used in concurrent approaches to limit the design space to unit cells with guaranteed connectivity [171, 204]. Concurrent designs have been extruded to 2D for manufacturing [203, 205].

The SIMP method has also been adapted for multi-scale design and manufacturing. For example, a single, periodic material, was interpolated in the spirit of SIMP to design structures composed of a fixed microstructure. A resulting cantilever beam on the order of $300\ \mu\text{m}$ was manufactured by multi-photon lithography with microstructure conformal to the boundary [206, 207]. Additionally, SIMP, together with a series of projections [208] or local volume fractions [209], were proposed to optimize the topology of porous infill structures. Resulting 2D infill structures were manufactured by fused filament fabrication (FFF) after being extruded to 3D [149] and 3D infill structures were manufactured using SLS and FFF [209].

Here, multi-scale design is achieved through a multi-material topology optimization formulation, where each candidate material is characterized by a periodic tessellation of a unit cell. Such candidate microstructural-materials can be constructed from truss elements using concepts from crystallography and geometry [210, 211] (here), taken from existing databases [115–117], or designed to exhibit desired behavior via topology optimization (e.g., extreme bulk or shear modulus [212–214], negative Poisson’s ratio [215–218], prescribed constitutive tensor [116, 219, 220]). Connectivity is handled by functionally grading the material interfaces of the topology optimization result and mapping transitional unit cells in those regions. The transitional unit cells are found via shape interpolation, which can be done using known geometrical relationships between the unit cells (here), using shape metamorphosis techniques from computer graphics [221], or by designing the candidate material unit cells to be functionally graded [222, 223].

6.3 Stereolithography 3D printing

SLA 3D printing or vat photopolymerization, is a technique in which sequential layers of liquid photopolymer resin are selectively exposed to radiation (typically ultraviolet (UV) light) to initiate photopolymerization and build a part layer-by-layer. Several variations of SLA have been explored since its inception in the early 1980’s. The early systems use a bottom-up approach (Figure 6.1a), where the UV light source is directed downward onto the top of the resin vat and the build plate is submerged deeper into the resin vat with each new layer, i.e., the part is built from the bottom up). Many newer systems use a top-down approach (Figure 6.1b and c) in which the UV light source is directed upward onto the bottom of the resin vat through a transparent window and the build plate is moved upward

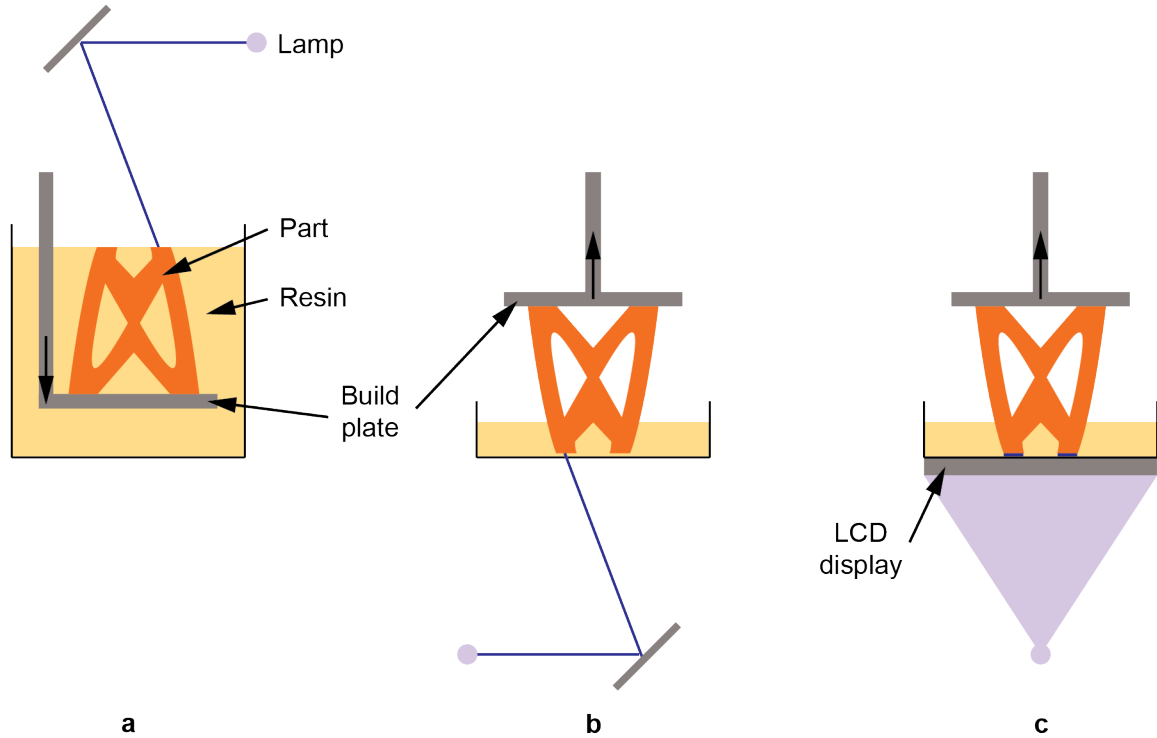


Figure 6.1: Stereolithography 3D printing process: (a) Bottom-up approach with pointwise UV laser radiation; (b) top-down approach with pointwise UV laser radiation; (c) top-down approach with layerwise UV radiation masked by LCD display. Images are adapted from [224].

to pull the part out of the resin vat with each new layer (i.e., the part is built from the top down). There are also several methods of exposing the part to the UV light. Vector scan approaches use laser beams to photopolymerize the part point-wise (Figure 6.1a and b) while mask projection techniques pattern the UV light over an entire layer using a liquid crystal display (LCD) (Figure 6.1c) or Digital Micromirror DeviceTM. Laser printers require toolpath information so that the laser can trace out the contours and infill of each layer. Mask projection printers require bitmap information to display a pixelated representation of each layer [224].

SLA is chosen here because it can simultaneously achieve a relatively large build volume and relatively high resolution. Mask projection is specifically selected because the bitmap-based slice representation provides a convenient platform for performing the slice-based, multi-lattice embedding described next, although the bitmaps generated here could be converted to a toolpath for use with laser systems or other toolpath-based additive manufacturing techniques (e.g., FFF, SLS).

All physical models to follow are fabricated using the Original Prusa SL1 m-SLA 3D printer (Prusa Research, Czech Republic), which uses a top-down approach with a 2560×1440 pixel LCD display to pattern the UV light. The pixel edge length and layerheight are $47.25 \mu\text{m}$ and $50 \mu\text{m}$, respectively. The build volume is $120.96 \times 68.04 \times 150$ mm. All models are built using Prusa’s Transparent Red Tough resin (an acrylate-based photopolymer resin) with 6 second exposure time per layer, followed by 3 minutes of UV post-cure. The bitmap (png) images for each layer of the microstructure-embedded topology optimized parts are generated using a MATLAB code written as part of this dissertation work.

6.4 Manufacturing multi-lattice topology optimized parts

Consider re-designing the cantilever beam discussed in subsection 4.5.2 for the three different sets of two microstructural-materials shown in Figure 6.2. In each case, the bulk material associated with microstructural-materials 1 and 2 are limited to a domain volume fraction of $\bar{v}_1 = 0.07\hat{v}_1$ and $\bar{v}_2 = 0.03\hat{v}_2$, respectively, and half of the domain is designed considering symmetry boundary conditions along the $x_2 - x_3$ plane. Refer to Table 4.2 for the additional topology optimization parameters used to design these beams.

The three beams in Figure 6.2 are used in this section to discuss the key challenges associated with manufacturing topology optimized parts composed of spatially-varying, microstructural-materials, and the associated solutions put forth in this dissertation. Specific challenges include interpreting the multi-material topology optimization data for additive manufacturing, transitioning between different microstructural-materials, and communicating the microstructure-embedded parts to the 3D printer without a prohibitively expensive surface representation (STL file). The entire process is summarized in a flowchart in Figure G.1.

6.4.1 Assumptions and limitations

Although the proposed multi-material topology optimization formulation can handle a general class of porous, anisotropic microstructural-materials (i.e., those for which an elasticity tensor can be provided), the proposed multi-microstructure-embedding scheme needed for manufacturing requires a few simplifying assumptions:

1. each microstructural-material must be periodic;

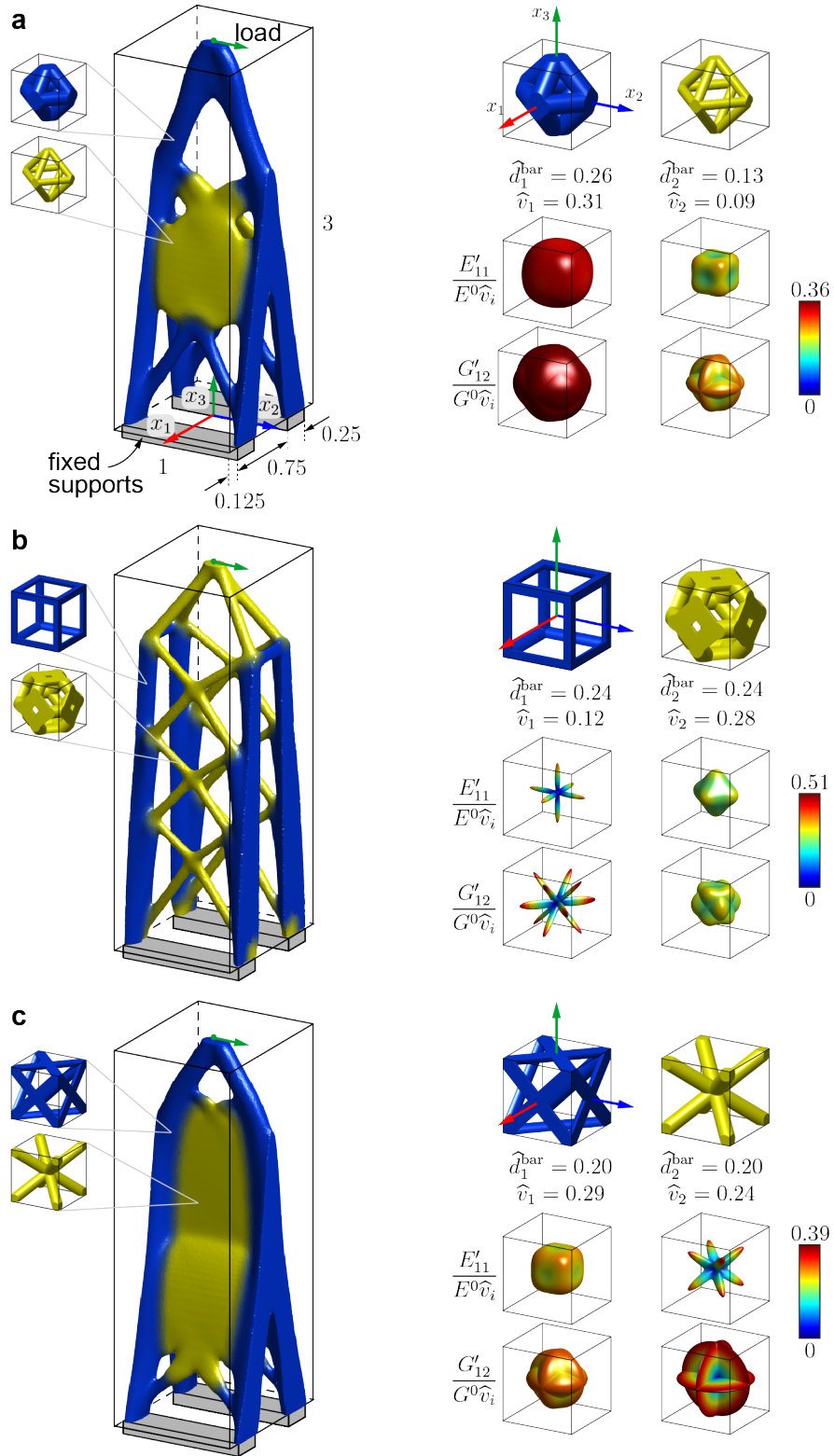


Figure 6.2: Two-microstructural-material topology optimized cantilever beams considering three different combinations of two microstructural-materials: In all cases, the bulk materials are limited to a domain volume fraction of $\bar{v}_1 = 0.07\hat{v}_1$ and $\bar{v}_2 = 0.03\hat{v}_2$ for microstructural-materials 1 and 2, respectively. Variable \hat{d}_i^{bar} is the unit cell bar diameter.

2. the same translation operations must be used to create each periodic microstructural-material from its associated unit cell; and
3. it must be possible to define a set of transitional unit cells that ensure a smooth, well-connected transition between the different unit cell geometries.

In addition to the lattice-like unit cells considered here, unit cells consisting of plate elements or triply periodic minimal surfaces [225], can be directly integrated with the embedding approach. In addition to cubic materials considered here, periodic materials without cubic symmetry (see examples by Zok et al. [226]) can be handled.¹ Non-periodic materials (e.g., spinodal architectures [227–230]) require a different embedding approach.

In characterizing microstructural-materials using homogenized properties, infinite periodicity and separation of length scales [146] are assumed. Neither of these assumptions can be verified in the manufactured parts that follow. Two factors prevent infinite periodicity: the presence of multiple microstructural-materials and truncation of the periodic microstructure tessellations at the structure boundaries. The first factor can be mitigated by avoiding abrupt transitions between the microstructural-materials, which is facilitated by the proposed continuous multi-microstructure-embedding scheme discussed next. The boundary remains a challenge. The manufactured parts obtained from the proposed embedding scheme contain truncated unit cells at the boundaries that likely influence the mechanical behavior. These edge effects cannot be completely removed. Although not considered here, making the unit cells conform to the boundary [159, 206] can alleviate edge effects.

Inadequate separation of length scales is due to the fact that the microstructures cannot be printed at an infinitely small scale nor can the macrostructure be printed at an infinitely large scale. However, the voxel-based approach pursued here is scalable; that is, the maximum macrostructure size and minimum microstructure feature size are dictated by the 3D printer and not the data representation. Thus, the approach provides a means to obtain a practical separation of length scales, for example, by using large area projection microstereolithography [157], high-area rapid printing [231], or by assembling the part from a number of components manufactured at a practical scale. In fact, the latter approach, in combination with the current multi-microstructure-embedding scheme, could make optimized,

¹ Additional thought may be needed to select or tailor an appropriate additive manufacturing technology for different types of unit cells. For example, unit cells consisting of plates typically have enclosed voids that may prevent excess resin from being drained from the part.

multi-scale, architectural engineering-scale, structures possible.

6.4.2 Interpreting the multi-material topology optimization data

The topology optimized parts in Figure 6.2 are obtained from Equation 4.1 and although they are composed of multiple microstructural-materials, the data is first processed without knowledge of the particular microstructural geometries (i.e., they are first considered to be composed of multiple solid materials).

The first step is to post-process by removing any mixing present in the design as discussed in subsection 4.4.4 (see Equation 4.38). After post-processing, intermediate densities, which are also an artifact of the density filter, remain at the structure boundaries and at the material interfaces; however, mixing is fully eliminated.

With mixing eliminated, the multi-material density data can be communicated to a 3D printer in various ways. Recall that the topology optimization density fields coincide spatially with the (hex) mesh used for the finite element (FE) analysis. The data can be communicated using this representation directly by neglecting intermediate densities at the boundaries that are less than a cutoff value (≈ 0.5); however, the resolution of the density data is typically much coarser than the resolution of the printer’s pixel grid and this approach leads to visible stair-stepping features on the boundaries. A portion of the 3D structure represented directly on the underlying hex mesh is shown at the top of Figure 6.3a and a portion of one of its slices is shown at the bottom.

As is common in single-material topology optimization, a smooth isosurface of the topology optimization data can be generated and sent to the 3D printer as an STL file [21]; however, a surface representation cannot capture material variations through the volume of the part. A separate STL can be generated for each material individually as shown in Figure 6.3b, but due to intermediate densities resulting from the density filter, multiple STLs result in disjointed material interfaces. To achieve smooth boundaries, promote well-connected material interfaces, and capture material variations through the volume of the part, a volume (tetrahedral (tet)) mesh can be generated inside an isosurface of the composite density field, $y_\ell^c = \sum_{i=1}^m y_{\ell i}$, $\ell = 1, \dots, N$, with the multi-material density data projected onto it (Figure 6.3c).²

² The iso2mesh toolbox [232], which relies on TetGen [233], is used to generate the tet meshes in this dissertation.

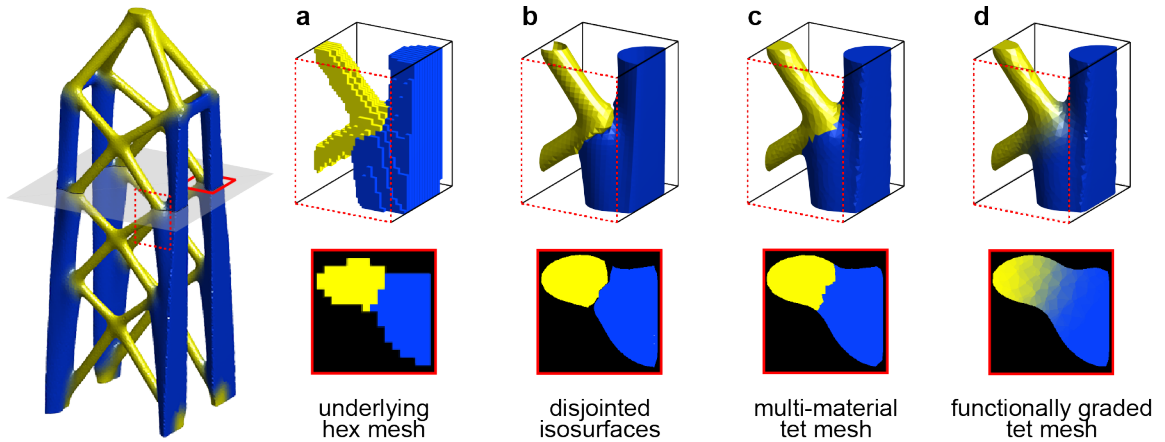


Figure 6.3: Various ways to represent multi-material topology optimization data: Each top image shows a portion of the 3D part and each bottom image shows a portion of one slice for the density data represented (a) directly on the underlying FE (hex) mesh; (b) with separate isosurfaces for each material; on a volume (tet) mesh generated inside an isosurface of the composite density data with (c) abrupt material interfaces and (d) functionally graded material interfaces.

The resulting abrupt material interfaces can lead to abrupt changes in microstructural geometry and disconnected regions, both of which can compromise structural integrity. To promote the continuous multi-lattice embedding discussed in subsection 6.4.3, a functionally graded transition region can be generated by applying an additional convolution filter to the multi-material tet data (refer to Equation 4.16), where the filter radius, R , controls the length scale of the functionally graded region.

Both the abrupt-transition and functionally graded multi-material tet data can be directly processed for multi-material 3D printing (e.g., inkjet [160, 234], grayscale DLP [33, 34], multi-vat DLP [31, 32]); however, the functionally graded, multi-material tet data are essential for continuously embedding the microstructural-materials into the part as will be described next. To illustrate the importance of functional grading, both the abrupt-transition and functionally graded multi-material tet data are considered next in the multi-microstructure embedding step.

6.4.3 Microstructural-material transitions

The key idea in embedding the microstructural-materials is to perform a boolean intersection of the multi-material tet data with tessellations of each of the microstructural-materials such that each material region in the topology optimized result is replaced by the associ-

ated microstructural-material. Note that the material interfaces in Figure 6.2 have arbitrary geometry (i.e., they can cut through the unit cells at any arbitrary location, in any orientation). Thus, upon embedding, lattices with dissimilar geometries may need to be connected at any arbitrary cross-section.

The three combinations of two microstructural-materials used to design the cantilever beams in Figure 6.2 illustrate three different types of connectivity handled here. In the top row of Figure 6.4a, a well-connected interface with abrupt transition is shown between two octahedron unit cells that have the same geometry, but different bar diameter; in the top row of Figure 6.4b, a discontinuous interface is shown between the simple cubic and truncated octahedron unit cell for which connectivity at any arbitrary interface is, in general, not guaranteed; and in the top row of Figure 6.4c, a disconnected interface is shown between the face-x and center-x unit cell for which connectivity is only guaranteed at the unit cell boundaries. In the bottom row, smooth and continuous transitions between the unit cells are achieved with a set of transitional unit cells generated by interpolating the bar diameter in Figure 6.4a, interpolating the unit cell geometry in Figure 6.4b, and composing the two unit cells into a series of hybrid unit cells in Figure 6.4c.³ The transitional unit cells can be mapped to the functionally graded regions of the multi-material tet mesh.

The transitional unit cells are chosen to achieve smooth and continuous geometry transitions and do not directly enforce any requirements on how the microstructural-material properties vary over the transition region. Without being enforced, the transition regions used here demonstrate monotonically decreasing stiffness over the transition in Figure 6.5a; monotonically increasing stiffness followed by some oscillations in Figure 6.5b; and a stiffening effect, much like standard connections used in engineering, in Figure 6.5c. Other transitioning techniques can be used to achieve desired homogenized material property transitions [235, 236].

6.4.4 Communicating the microstructure-embedded geometry to the 3D printer

The typical workflow for 3D printing a part is to generate a surface representation of the 3D geometry and send it (as an STL file) to a slicing software that generates layer-wise instructions for the 3D printer. An STL file of a microstructure-embedded part with sufficient

³ The transitions in Figure 6.4 are shown in 1D, but in the manufactured parts, they can occur in any arbitrary direction in 3D space.

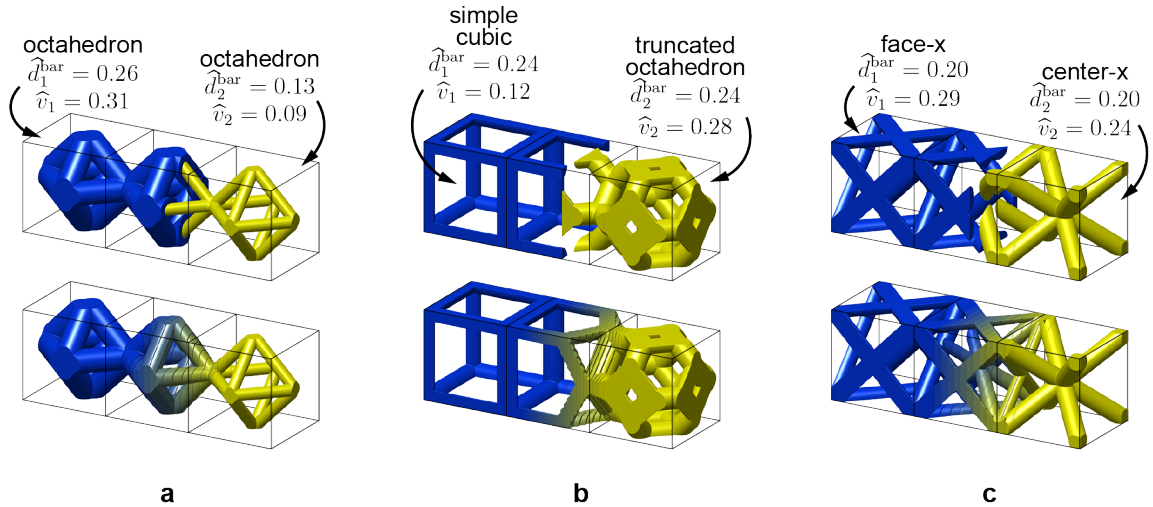


Figure 6.4: Smooth and continuous microstructure connectivity: (a) Two octahedron unit cells with different bar diameter for which connectivity is always guaranteed and a smooth transition is achieved by interpolating the bar diameter; (b) a simple cubic and a truncated octahedron unit cell for which connectivity is, in general, not guaranteed and a smooth and continuous transition is achieved by interpolating the unit cell geometry; (c) a face-x and center-x unit cell for which connectivity is only guaranteed at the unit cell boundaries and a smooth and continuous transition is achieved by using hybrid unit cells composed of the two basic unit cells. Variable \hat{d}_i^{bar} is the unit cell bar diameter.

separation of length scales between the macro- and micro-scales requires a huge number of triangles to sufficiently represent the microscale features and is prohibitively expensive to communicate to the slicing software. Thus, rather than embedding the microstructural-materials in 3D and generating a microstructure-embedded STL, the embedding is done at the slice level. For the m-SLA technique adopted here, the microstructural-materials are embedded in the bitmap (png) images that the printer projects to the underside of the resin vat.

The first step is to generate slices for each layer of the multi-material tet data (macro-slices) that indicate how the microstructural-materials vary spatially (here, a color-code is used). A portion of one macro-slice is shown for the abrupt-transition and functionally graded tet meshes in Figure 6.6a and b for the cantilever composed of simple cubic and truncated octahedron unit cells. The next step is to generate slices, using a layerheight consistent with that of the macro-slices, for each unit cell and tile those slices over the entire bitmap to generate a one-unit-cell-high tessellation of each unit cell (micro-slices). In Figure 6.6c, a portion of the bottom slice of these tessellations are shown for the simple cubic, truncated octahedron, and a subset of the transitional unit cells considered here. Note

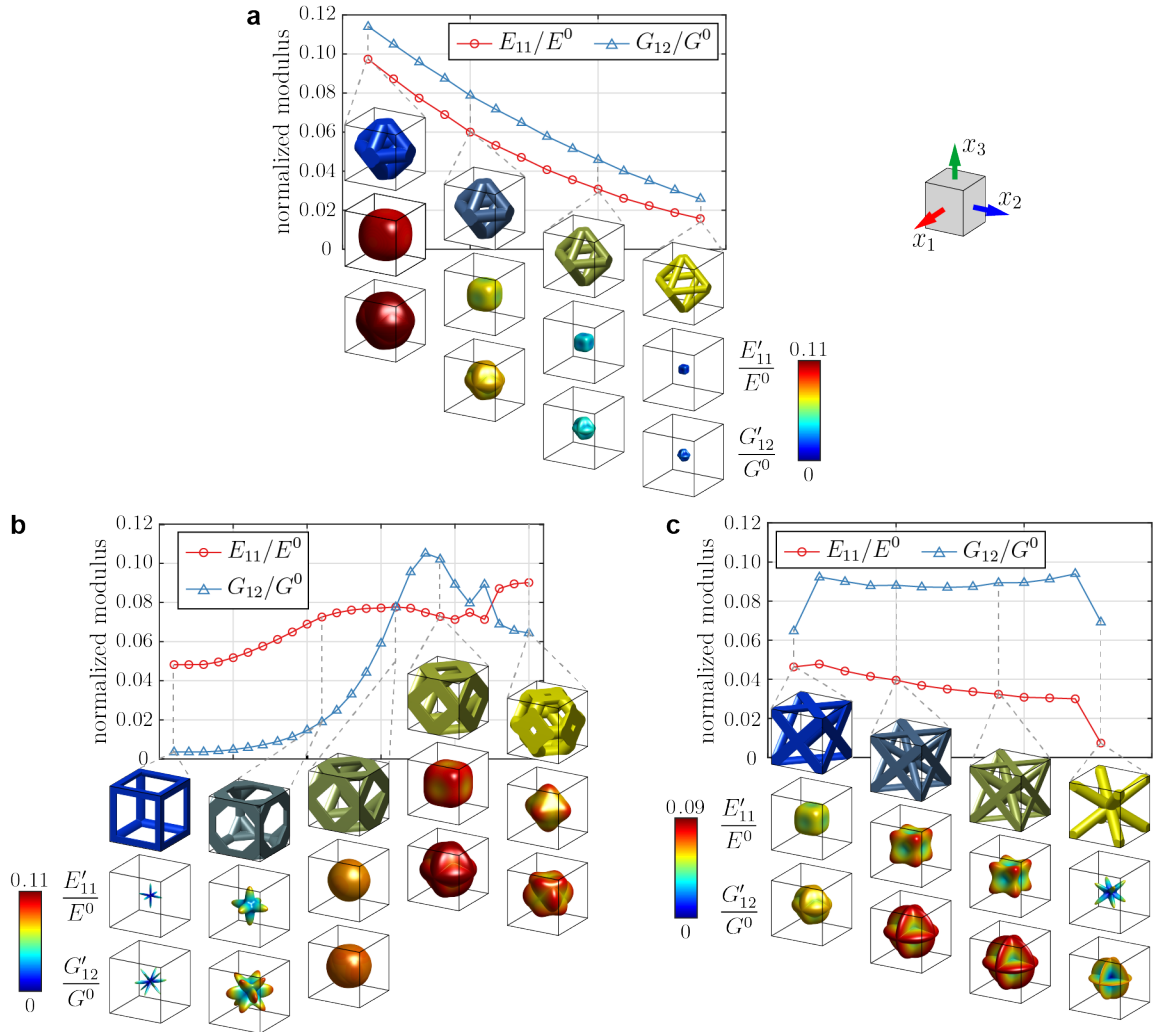


Figure 6.5: Line plots showing how the normalized tensile and shear moduli, E_{11}/E^0 and G_{12}/G^0 (in the reference frame and based on homogenized properties), vary over the transitional unit cells between (a) two octahedron unit cells with different bar diameter; (b) a simple cubic and a truncated octahedron unit cell; and (c) a face-x and center-x unit cell.

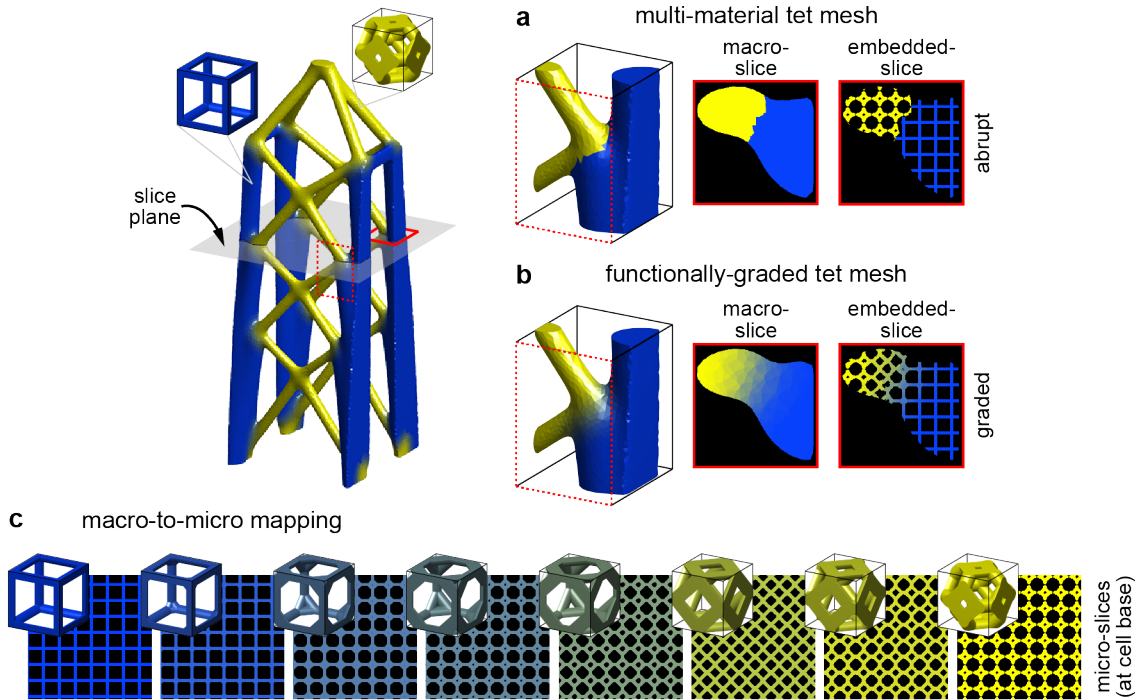


Figure 6.6: Multi-microstructure-embedding: In (a) and (b), a portion of one macro-slice and one embedded slice is shown for the abrupt-transition and functionally graded tet meshes, respectively; in (c), the simple cubic, truncated octahedron, and 6 of the 23 transitional unit cells are shown with a portion of the bottom slice of their tessellations (micro-slices).

that the length scale of the unit cells is dictated by the 3D printer’s resolution and has no dependence on the resolution of the density fields or FE mesh used in topology optimization.

The color-coded macro-slices can now be used as a mapping to the micro-slices. Each pixel in the macro-slice is replaced by the corresponding pixel of the appropriate micro-slice to obtain microstructure-embedded-slices as shown at the right of Figure 6.6a and b. By comparing the embedded-slices of the multi-material tet mesh and the functionally graded tet mesh in Figure 6.6a and b, it is clear that functional grading is critical to ensure connectivity of the microstructural-materials. Macro-slices, micro-slices, embedded-slices, and the macro-to-micro mapping for the other two, two-microstructural-material cantilever beams are provided in Figure 6.7.

6.4.5 Manufactured parts

Using the approach described above, the three cantilever beams from Figure 6.2 were manufactured using the Original Prusa SL1 m-SLA 3D printer. Each beam, shown in Figure 6.8- Figure 6.10, has an (expected) height of 14.5 cm and the embedded microstructures are

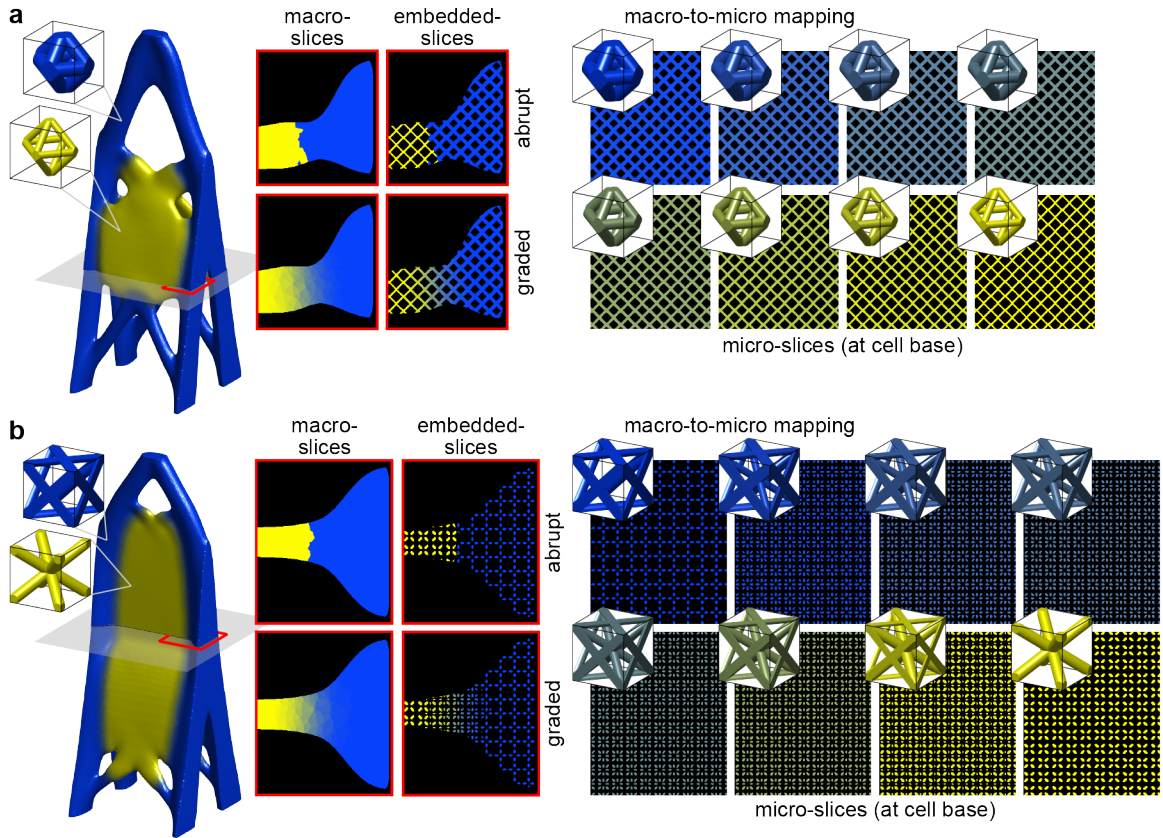


Figure 6.7: Continuous multi-microstructure-embedding for two-microstructural-material cantilevers: The cantilevers are composed of (a) two octahedron unit cells with different bar diameter and (b) a face-x and a center-x unit cell. The transitional unit cells making up the macro-to-micro mapping in (a) are obtained by interpolating the bar diameter. Those in (b) are obtained by composing the two unit cells into a set of hybrid unit cells, where the face-x unit cell gradually disappears from one end and the center-x unit cell gradually disappears from the other (with minimum bar diameter limited to 0.065 of the unit cell edge length for manufacturability). In (a), 6 of the 12 transitional unit cells are shown and in (b), 6 of the 11 transitional unit cells are shown.

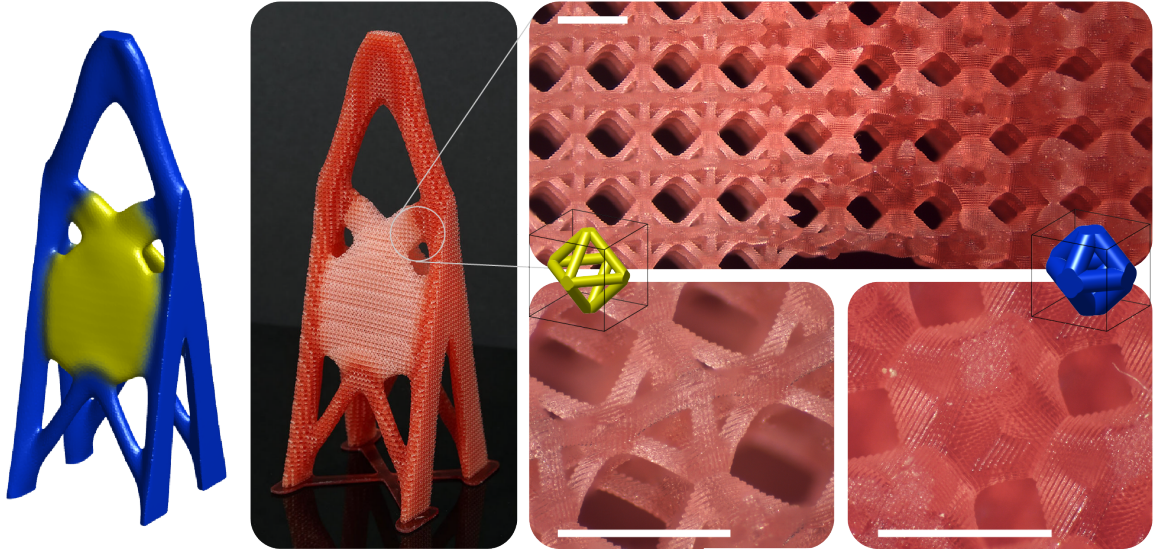


Figure 6.8: Cantilever beam designed and manufactured with two microstructural-materials composed of octahedron unit cells with two different bar diameters: The manufactured beam is 14.5 cm tall, the embedded microstructures have unit cells with edge length scaled to 1.5 mm, and the higher and lower volume unit cells have (expected) bar diameter $360 \mu\text{m}$ and $195 \mu\text{m}$, respectively. Scale bars are 1.5 mm.

Table 6.1: Bounding box dimensions of computer and physical models for each cantilever design (cm)

	computer model	physical model
octahedron, octahedron	$4.84 \times 4.96 \times 14.52$	$4.80 \times 5.05 \times 14.45$
simple cubic, truncated octahedron	$4.84 \times 4.95 \times 14.52$	$4.80 \times 4.95 \times 14.45$
face-x, center-x	$4.84 \times 5.07 \times 14.52$	$4.75 \times 5.05 \times 14.45$

scaled such that the unit cells have an (expected) edge length of 1.5 mm. Close-up images of the transition and each of the two microstructural-materials are also shown in Figure 6.8-Figure 6.10, where the minimum and maximum (expected) microstructure bar diameters are $195 \mu\text{m}$ and $390 \mu\text{m}$, respectively. Actual dimensions of the printed parts (accounting for printing inaccuracies and potential shrinkage due to UV curing) are provided in Table 6.1.

6.5 Effect of transition region length scale

The filter power and radius used to generate the functionally graded tet mesh are $q = 1$ and $R = 0.10$, respectively, for the cantilevers discussed in section 6.4, where the magnitude of R is with respect to the domain dimensions used during design and provided in

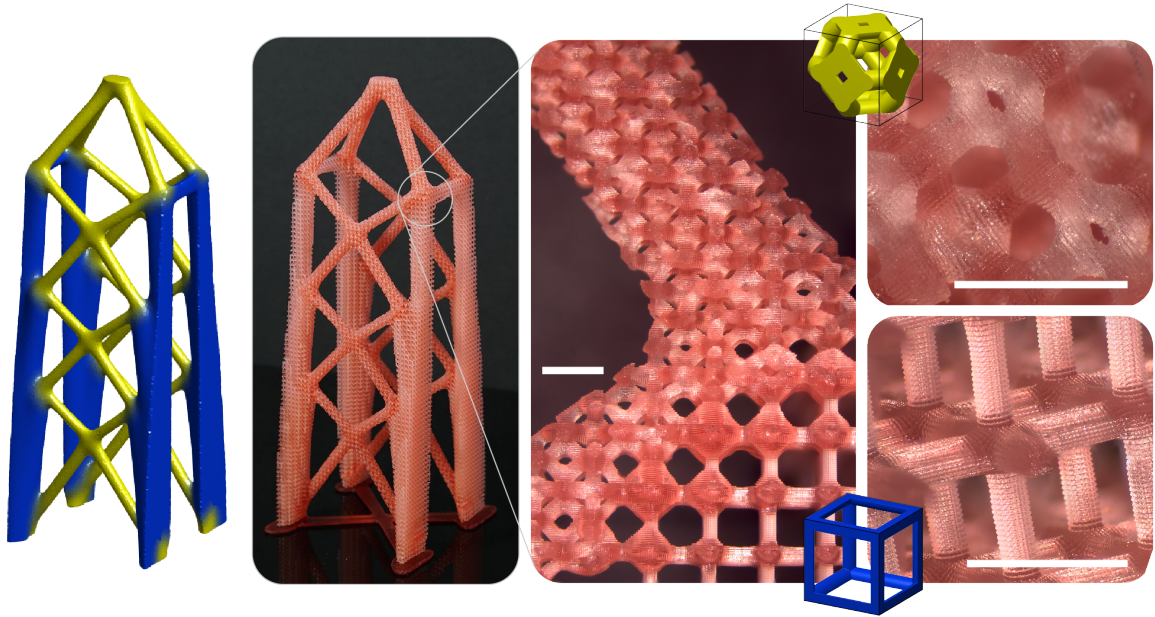


Figure 6.9: Cantilever beam designed and manufactured with two microstructural-materials composed of simple cubic and truncated octahedron unit cells: The manufactured beam is 14.5 cm tall, the embedded microstructures have unit cells with edge length scaled to 1.5 mm, and the bars of both unit cell types have (expected) bar diameter $360 \mu\text{m}$. Scale bars are 1.5 mm.

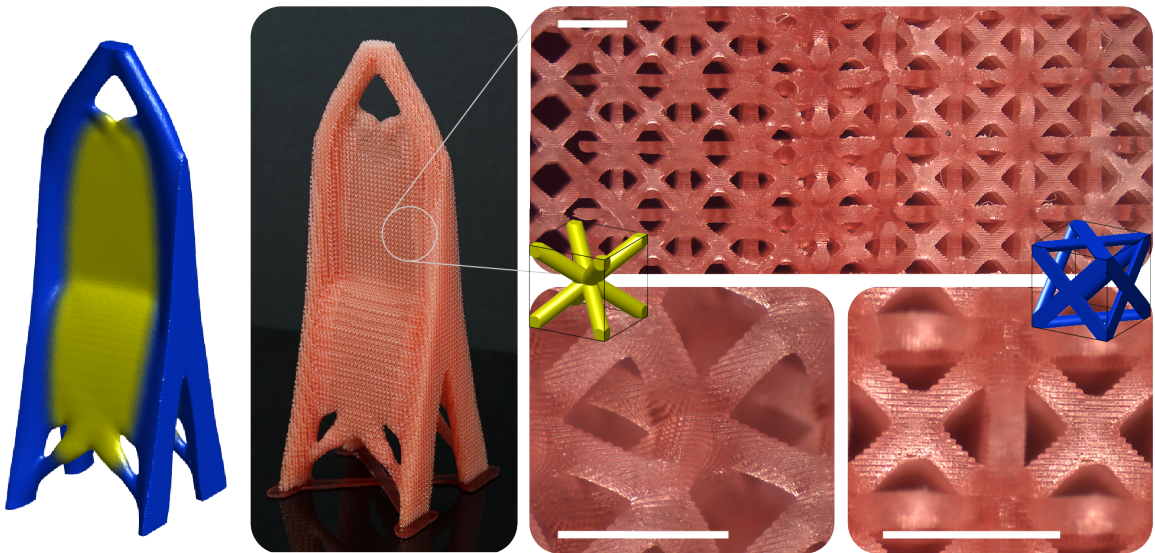


Figure 6.10: Cantilever beam designed and manufactured with two microstructural-materials composed of face-x and center-x unit cells: The manufactured beam is 14.5 cm tall, the embedded microstructures have unit cells with edge length scaled to 1.5 mm, and the bars of both unit cell types have (expected) diameter $300 \mu\text{m}$. Scale bars are 1.5 mm.

Figure 6.2. To demonstrate how the length scale of the transition region, connectivity of the microstructures, and objective function value are affected by the filter radius, R , used in functional grading, the functionally graded tet mesh for the two-microstructural-material beam composed of simple cubic and truncated octahedron unit cells is re-generated using $R = 0, 0.010, 0.025, 0.050, 0.100,$ and 0.200 . To ensure that a sufficient number of tet elements are encompassed by the radius during filtering, the tet mesh is refined near the microstructural-material interfaces for the cases of $R \leq 0.050$. Note that $R = 0$ corresponds to the abrupt transition shown in Figure 6.6c and $R = 0.100$ corresponds to the functionally graded transition shown in Figure 6.6d.

To achieve a well-connected interface between the microstructural-materials, the transition region must have a finite length. A rule of thumb is that the transition region should be at least as long as the edge length of the unit cells. In the printed part shown in Figure 6.9, the edge length of the unit cells is 1.5 mm, which corresponds to an edge length of 0.030 relative to the domain dimensions used during design and provided in Figure 6.2 (the part was scaled up for manufacturing). When the filter radius is larger than 0.030 (i.e., $R = 0.050, 0.100, 0.200$), the microstructures are well-connected at the interfaces. As shown in Figure 6.11, when the filter radius is smaller than 0.030 (i.e., $R = 0.010, 0.0250$), the interfaces are still well-connected, but the interface may not be as robust. The objective function values, f , provided in Figure 6.11 are normalized to that of the case with $R = 0$, which has objective function value, f_0 . The normalized objective function values, f/f_0 , indicate that the length scale of the transition regions does not significantly affect the global elastic properties for which the structure was designed.⁴ Note that the objective function, f , is computed here using the tet meshes with homogenized microstructural-material properties mapped appropriately.

6.6 Scaling to larger build volumes

To demonstrate scalability of the proposed multi-microstructure-embedding scheme, two additional structures are designed and manufactured at a larger scale: a hyperbolic paraboloid canopy structure and an Eiffel Tower-inspired structure. The optimization design space for both problems is described in Figure 6.12 and the optimization input parameters are pro-

⁴ Local properties (e.g., stress concentrations) are not investigated here, but it is expected that the transitions may redistribute stress concentrations at the microstructural-material interfaces.

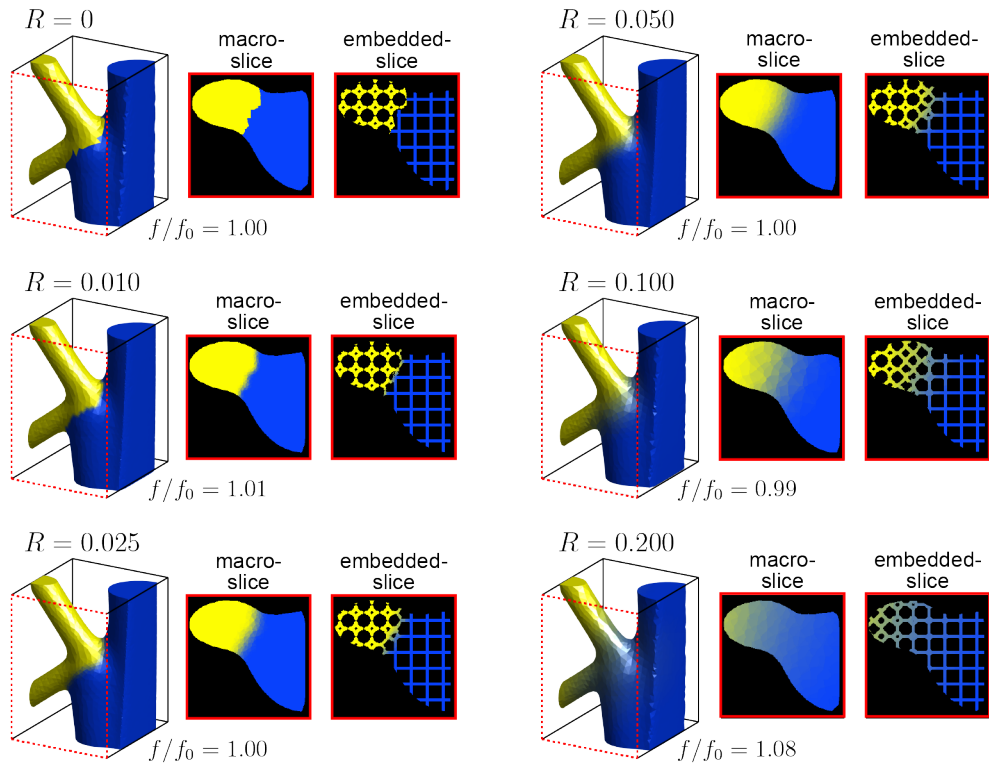


Figure 6.11: Illustration of how the transition region length scale, microstructure connectivity, and objective function value are affected by the radius, R , used in functional grading: The figure shows a closeup of a portion of the functionally graded tet mesh and associated macro-slices and embedded-slices for $R = 0$ (abrupt interfaces), $R = 0.010$, $R = 0.025$, $R = 0.050$, $R = 0.100$, and $R = 0.200$, where the magnitude of R is with respect to the domain dimensions used during design (see Figure 6.2). The objective function values, f/f_0 , which are normalized to that of the case with $R = 0$, indicate that the length scale of the transition regions does not significantly affect the global elastic properties for which the structure is designed.

Table 6.2: Optimization input parameters used for the hyperbolic paraboloid canopy and Eiffel tower-inspired structure design examples (brackets indicate continuation).

ZPR move limit, M	0.15
ZPR intermediate variable exponent, α	-1
convergence tolerance, tol	0.01
intermediate density penalty parameter, p	[1, 1.5, 2, 2.5, 3]
mixing penalty parameter, γ	[0, 0.2, 0.5, 0.8, 1]
filter radius, R	[R, R, R, R, R]
filter exponent, q	[3, 3, 3, 3, 3]
max. number of iterations	[100, 100, 100, 100, 200]

vided in Table 6.2. Although the printer’s display area (2560×1440 pixels) and build height (15 cm) are limited, the parts are sliced for a larger build volume and then the slices are divided into smaller images that the printer can handle (i.e., the structures are printed in pieces and assembled). To slice for a larger build volume, the number of pixels per slice are increased (for increased length and width) and a larger number of slices are generated (for increased height). Since small subsets of the pixels are processed at a time (in parallel), the increase in number of pixels does not lead to increased memory requirements, but does lead to increased slice time.

6.6.1 Canopy

A hyperbolic paraboloid canopy is designed considering the microstructural-materials in Figure 6.12a. The domain and boundary conditions are provided in Figure 6.12b, where the top surface of the canopy, which is defined on the domain $x_1, x_2 \in [-0.5, 0.5]$ by the equation $x_3 = x_1^2 - x_2^2 + 1$, is subjected to a uniformly-distributed, vertical load. The canopy itself is a passive region occupied by a face-x microstructural-material (shown in blue in Figure 6.12a), i.e., it does not participate in the optimization. A short tube just above the structure’s fixed support is another passive region occupied by a solid, isotropic material (shown in red in Figure 6.12a). No material can occupy the inner volume of the tube or the space above the canopy. In the remainder of the domain, the multi-material topology optimization formulation is used to design a structure composed of octet and truncated octahedron microstructural-materials (shown in cyan and yellow in Figure 6.12a) to transfer the loads from the canopy to the fixed support. The total optimizable domain volume fraction is limited to $\bar{v} = 0.0096$ and the problem is solved on $1 \times 1 \times 1.5$ hex mesh with

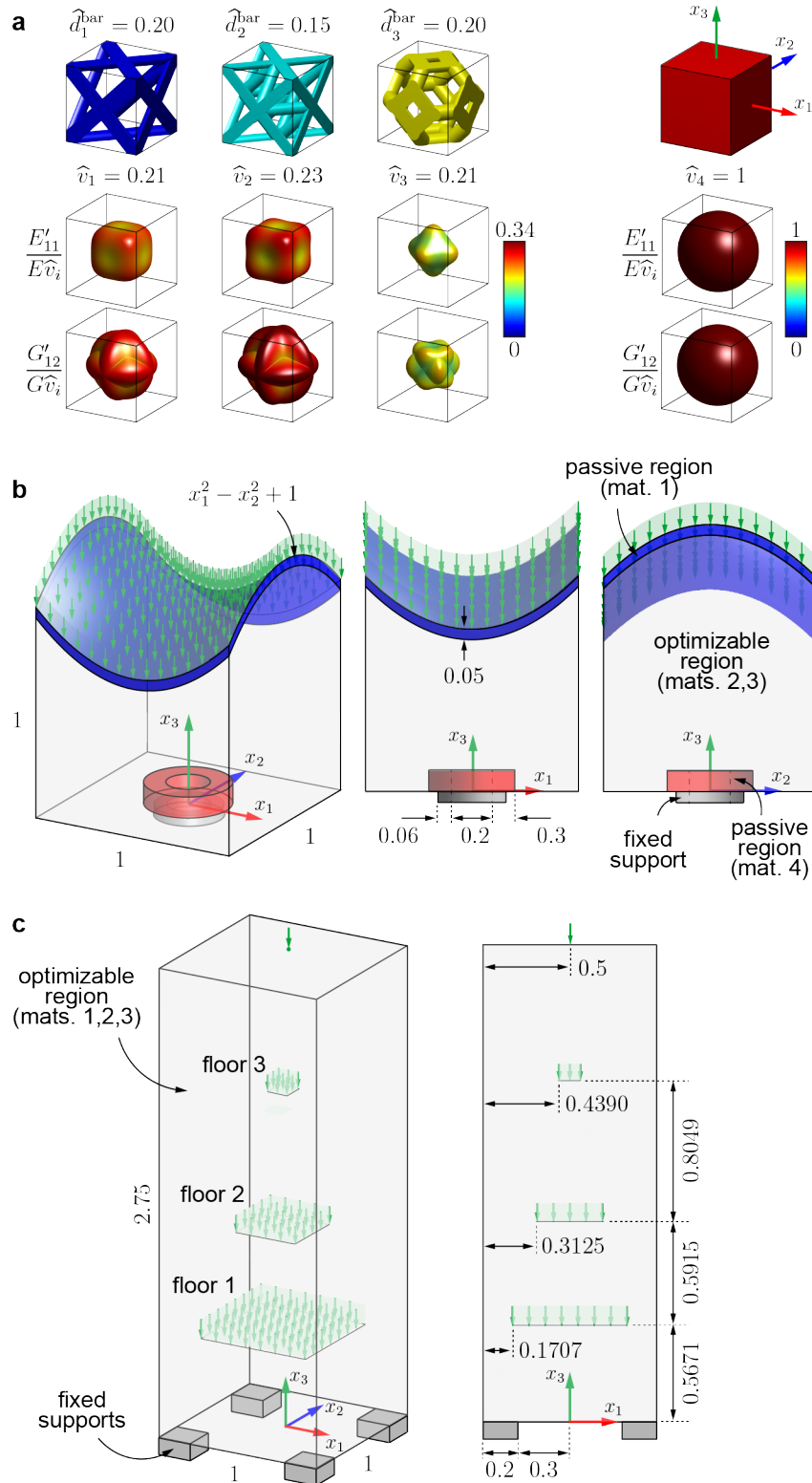


Figure 6.12: Problem description for hyperbolic paraboloid canopy and Eiffel tower-inspired structures: (a) Candidate microstructural materials and associated normalized, directional tensile and shear moduli plots; (b) canopy domain and boundary conditions; (c) Eiffel tower domain and boundary conditions.

$80 \times 80 \times 120$ elements; however, elements with centroid above the hyperbolic paraboloid surface and interior to the tube are removed, for a total of 510,411 elements. The filter radius used in design is $R = 0.032$.

The final numerical design, considering $q = 1$ and $R = 0.04$ (with respect to the design dimensions in Figure 6.12b) for functional grading, and multi-microstructure-embedded, manufactured canopy structure are shown in Figure 6.13. Details of the transition regions between microstructural-materials are provided in Figure 6.14. The manufactured part fits inside a bounding box of dimensions $11.6 \times 11.6 \times 14.4$ cm and the embedded unit cells are scaled to have an (expected) edge length of 2 mm (at this scale, the (expected) bar diameters are $300 \mu\text{m}$ for the octet unit cells and $400 \mu\text{m}$ for the face-x and truncated octahedron unit cells).

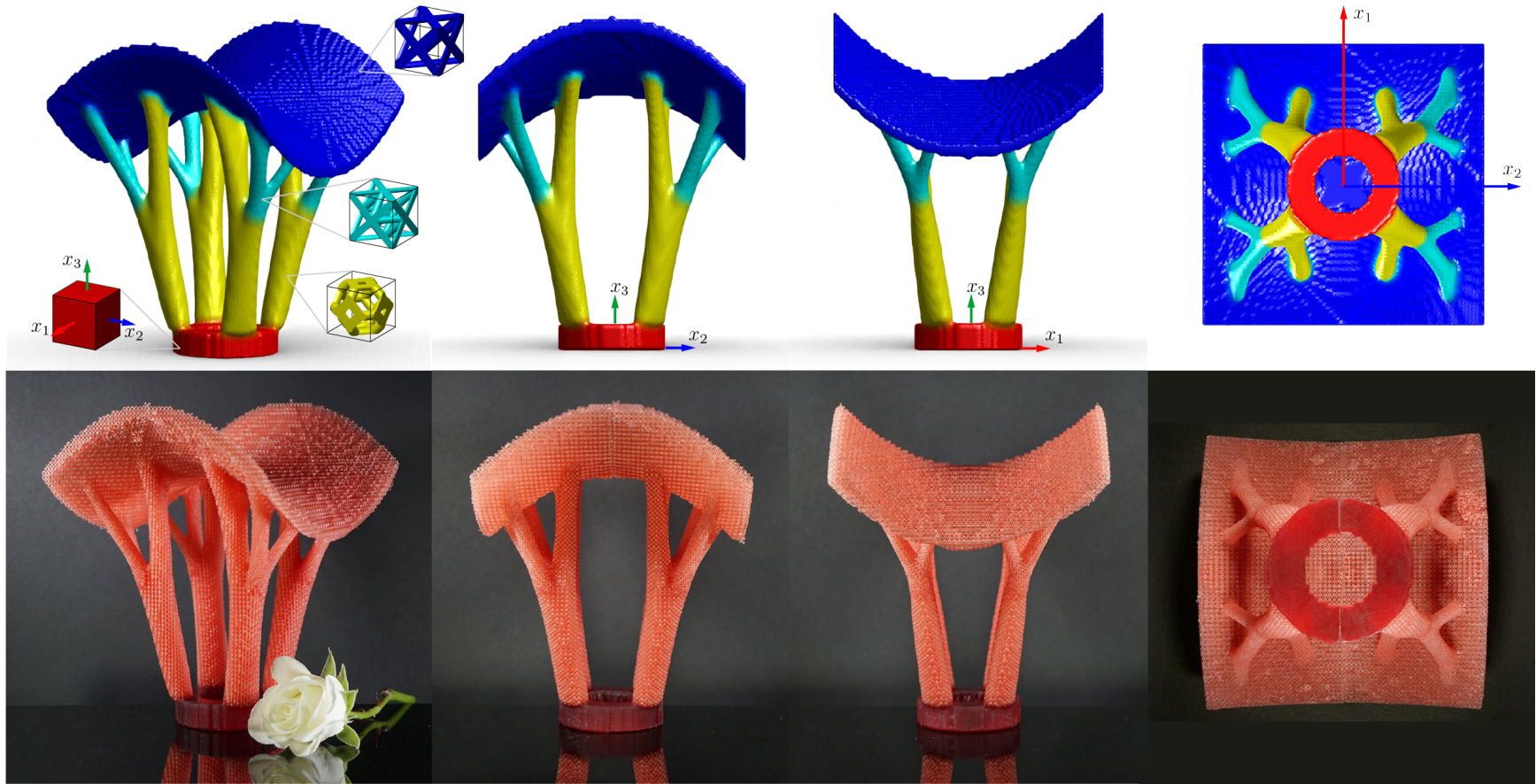


Figure 6.13: Canopy structure: Topology optimized design (top) and manufactured part (bottom) with a (printed) height of 14.4 cm and an (expected) unit cell edge length of 2 mm.

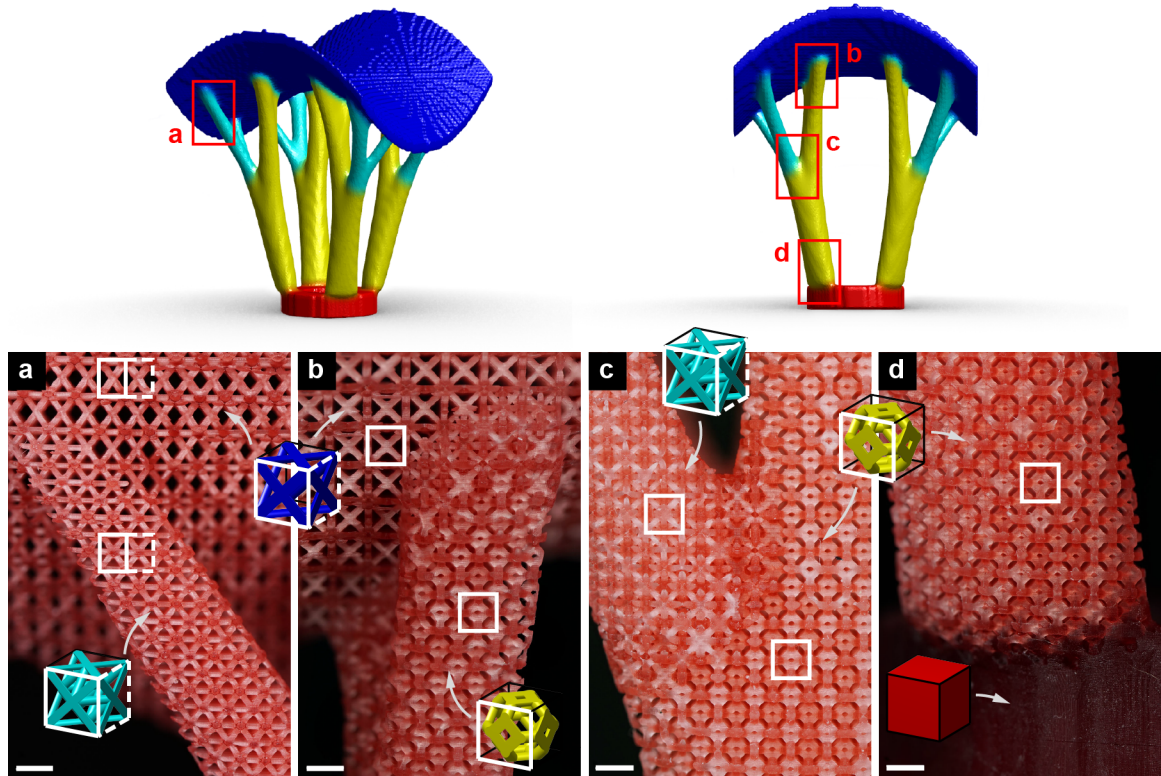


Figure 6.14: Canopy structure transition regions: (a) Octet unit cells with $300\ \mu\text{m}$ bar diameter to face-x unit cells with $400\ \mu\text{m}$ bar diameter; (b) truncated octahedron unit cells with $400\ \mu\text{m}$ bar diameter to face-x unit cells with $400\ \mu\text{m}$ bar diameter; (c) truncated octahedron unit cells with $400\ \mu\text{m}$ bar diameter to octet unit cells with $300\ \mu\text{m}$ bar diameter; (d) solid to truncated octahedron unit cells with $400\ \mu\text{m}$ bar diameter. Scale bars are 2 mm.

The canopy required a support structure during manufacturing. The support structure, shown in Figure 6.15a, was designed using Rhino[®] software. Additionally, because the part exceeds the printer's build volume, 2560×2560 pixel slices were generated and half of the structure was printed at a time, as shown in Figure 6.15b. The two parts were bonded together using Krazy Glue[™].

6.6.2 Eiffel Tower-inspired structure

Inspired by Gustave Eiffel's open-lattice, multi-scale Eiffel Tower, the multi-material topology optimization formulation is used to design an Eiffel Tower-inspired structure considering microstructural-materials 1-3 in Figure 6.12a. The domain and boundary conditions provided in Figure 6.12c are roughly based on those of the actual Eiffel Tower. Three floors are defined between the base and top of the tower. The width of each floor reduces to imitate the shape of the actual tower, which was chosen by Eiffel to efficiently resist wind loading

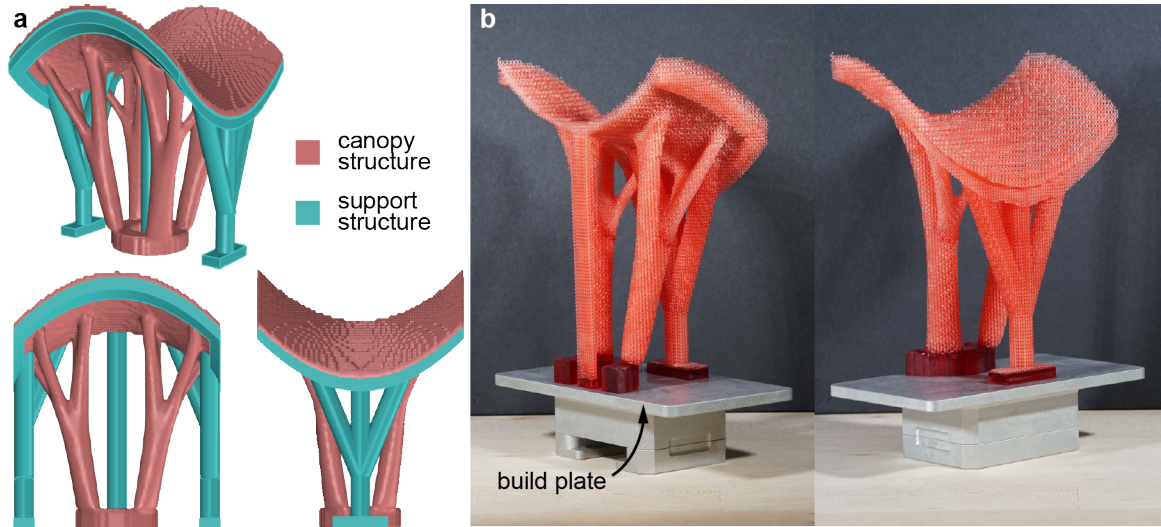


Figure 6.15: Manufacturing details for canopy structure: (a) Support structures required for printing the canopy structure; (b) one half of the canopy structure before removing the support structure or removing it from the build plate; (c) canopy structure after removing the support and before gluing the pieces together.

[237, 238]. Here, the shape of the original tower is adopted and the remaining form is designed considering only vertical (gravity) loads. The structure is fully fixed at the corner regions of its base. Uniformly distributed, vertical loads with total force equal to 1, 0.766, and 0.3 are applied at floors 1, 2, and 3, respectively, and a point load of magnitude 0.01 is applied at the top of the tower. The tower is designed considering face-x, octet, and truncated octahedron microstructural-materials with total domain volume fraction limited to $\bar{v} = 0.008$. The topology optimization problem is solved on a quarter of the domain (symmetry enforced on the $x_1 - x_3$ and $x_2 - x_3$ planes) on a $0.5 \times 0.5 \times 2.75$ hex mesh with $46 \times 46 \times 253$ elements, for a total of 535,381 elements. The filter radius used in design is $R = 0.025$.

The final numerical design, considering $q = 1$ and $R = 0.08$ (with respect to the design dimensions in Figure 6.12c) for functional grading, and multi-microstructure-embedded, manufactured Eiffel Tower-inspired structure are shown in Figure 6.16. The manufactured part fits inside a bounding box of dimensions $8.2 \times 8.2 \times 26.0$ cm and the embedded unit cells are scaled to have an (expected) edge length of 2 mm (at this scale, the (expected) bar diameters are $300 \mu\text{m}$ for the octet unit cells and $400 \mu\text{m}$ for the face-x and truncated octahedron unit cells).

Because the part exceeds the printer's build volume, 2560×2560 pixel slices were gener-

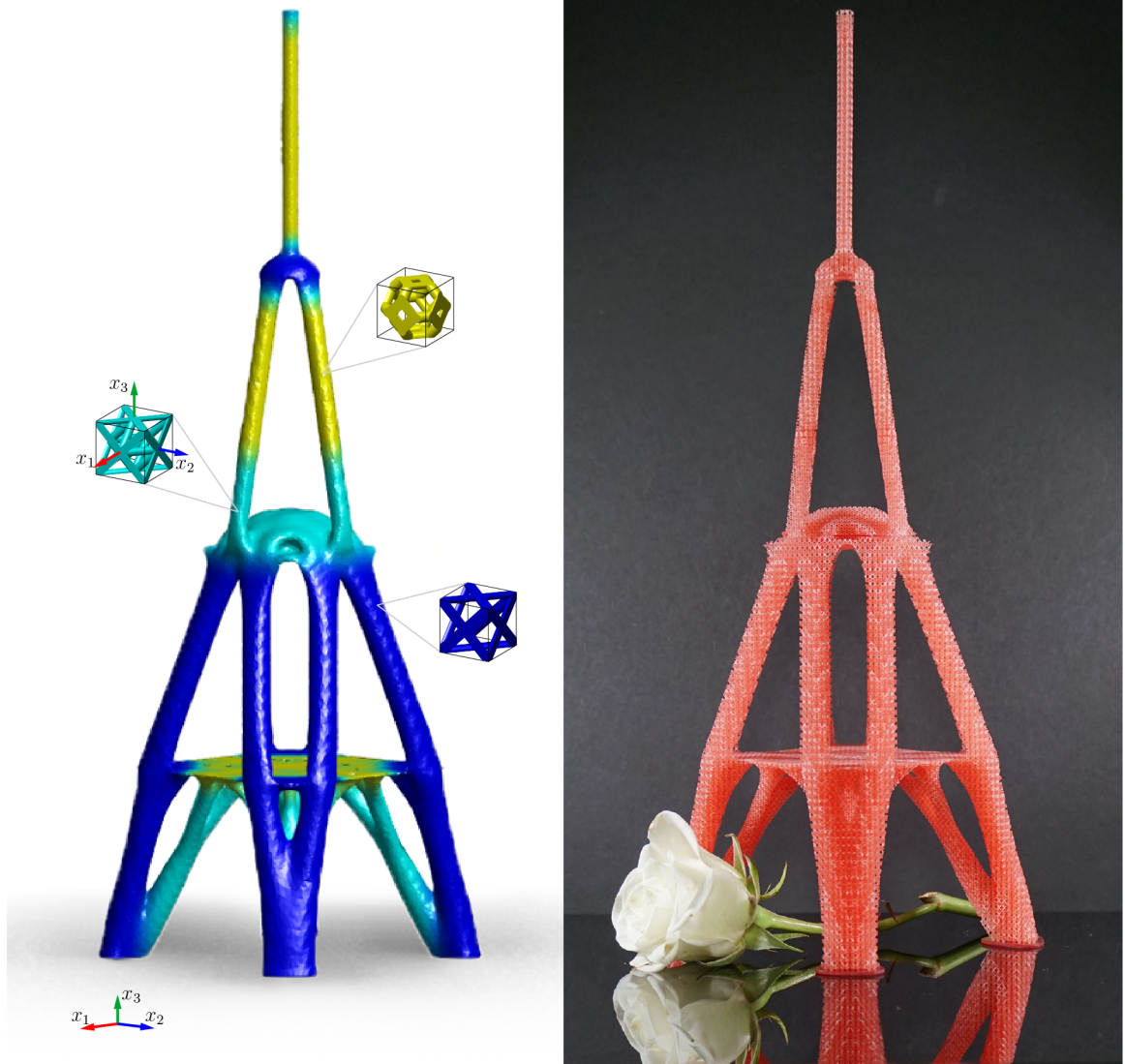


Figure 6.16: Eiffel Tower-inspired structure: Topology optimized design (left) and manufactured part (right). The assembled (printed) structure is 26 cm tall.



Figure 6.17: Eiffel Tower-inspired structure before assembly.

ated for the bottom portion of the tower, which was printed in two pieces. The top portion of the tower was printed as a third piece. The structure, before assembly by Krazy Glue™, is shown in Figure 6.17. Additionally, in Figure 6.18, two cross-sections of the Eiffel Tower-inspired structure highlight some interesting macrostructural details: macroscale voids are present on the interior of the dome structure at floor 2 and the supports at the base branch several times to provide relatively uniform support at the first floor.

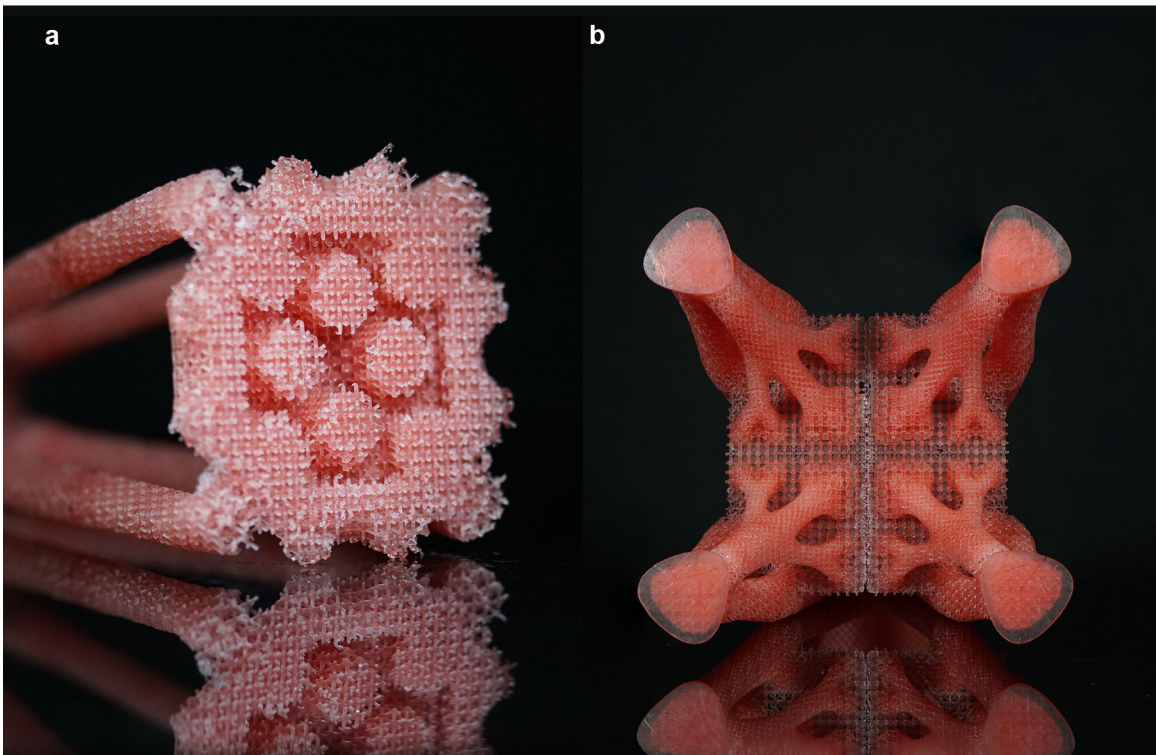
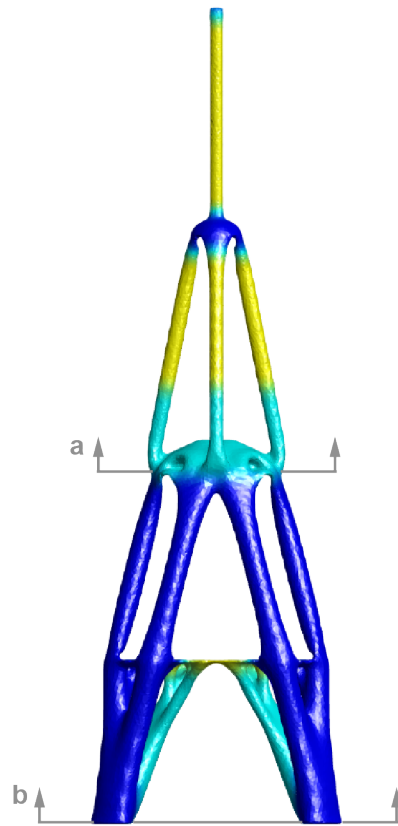


Figure 6.18: Additional details of the Eiffel Tower-inspired structure: (a) Cross-section through floor 2; (b) cross-section through base.

CHAPTER 7

CONCLUSIONS AND FUTURE WORK

This dissertation explored how *restricting* or *relaxing* the material space in topology optimization can be used as a tool to target desired mechanical functionality or to simply expand the design space. It was shown in the discrete topology optimization setting that optimized tension-only cable nets can be obtained by *restricting* the material space and optimized elastostatic cloaking devices can be obtained by *relaxing* the material space. In continuum topology optimization, a broad design space was accommodated by *relaxing* the material space and establishing a general multi-material formulation capable of handling porous, anisotropic candidate materials. In this context, practical considerations related to manufacturing complex macrostructure geometries embedded with microscale features were also considered.

For each of these problems, the theoretical and computational implications associated with *relaxing* or *restricting* the material space were addressed. Key contributions put forth in this dissertation are summarized below and several future directions are highlighted.

Topology optimization of tension-only cable nets under finite deformations The key contributions associated with topology optimization-based design in the *restricted* setting of tension-only materials included: 1) incorporating finite deformation kinematics that enable equilibrated solutions to be found via large configurational changes relative to the initial ground structure; 2) choosing a suitable measure of compliance in the finite deformation setting that does not require an additional adjoint solve during the sensitivity analysis and is convex under certain assumptions on the mechanics; and 3) determining relevant assumptions that simultaneously capture the nonlinear mechanics of the system and preserve desirable features of the objective function.

The tension-only examples provided in this dissertation are relatively simple and are only meant to illustrate the key features of the proposed formulation; however, the formulation may have practical implications in more complex problems. For example, form-finding is an approach that historically relied on experimental, tension-only models for design of compression-only grid shells (e.g., hanging chains by Antoni Gaudí and soap films by Frei

Otto). More recently, form-finding has been pursued using computational methods, most commonly force density methods [239]. The cable topology optimization formulation used here is somewhat of a hybrid version of the potential energy and ground structure methods proposed by Jiang et al. [240] in that it uses nonlinear mechanics (similar to the former) and solves an optimal sizing problem (similar to the latter). Unlike the potential energy method, the current cable formulation does not require any special treatment of compression members. In fact, in the context of large deformation kinematics, a tension-only material model avoids many of the challenges associated with handling compression in topology optimization. For example, by allowing members to also take compression, limit points in the equilibrium path would no longer be avoided and a more general nonlinear solution scheme would be needed. Additionally, the possibility of large negative internal forces in the members, for example, in elements assigned low stiffness during topology optimization, could lead to an indefinite or negative definite tangent stiffness matrix, both making the topology optimization problem non-convex and causing numerical issues in solving the nonlinear state equations. Thus, it would be interesting to integrate self-weight capabilities [11, 241] with the cable topology optimization formulation and explore it in greater detail in the context of form-finding of grid shells.

Optimized lattice-based metamaterials for elastostatic cloaking The key contributions associated with topology optimization-based design of multi-directional elastostatic cloaking devices included: 1) formulating a least squares objective function weighted by the stiffness matrix coefficients such that unimportant terms are nullified, an additional adjoint solve during the sensitivity analysis is avoided, and the objective function is convex under certain assumptions on the design variables; 2) recognizing that the global stiffness characteristics of the cloaked system are critical to mimicking the response of the reference system and pursuing such global stiffness matching by considering uniaxial tension in multiple directions during design; and 3) demonstrating approximate elastostatic cloaking of circular and elliptical holes, not only when subjected to the boundary conditions used in design, but also for uniaxial tension in directions not used in design and for shear loading.

A next step could be to experimentally validate the performance of physical realizations of the GS1 cloaks, which were specifically designed to avoid crossing members and promote manufacturing. In this dissertation, the design variables are defined as linear scale factors

on the element stiffness matrix, which can be interpreted as Young’s modulus varying in a continuous range. An additive manufacturing technique such as grayscale DLP [33, 34] that can achieve spatially-varying stiffness by locally controlling light intensity during UV curing would enable the necessary material stiffness variations. A projection method may need to be introduced to the optimization formulation such that the design variables can assume zero value (to allow for member disappearance), but do not fall within the range between zero and the lower bound of achievable Young’s modulus. It is noted that the design variables can also be interpreted as scale factors on a beam cross-sectional property such that most additive manufacturing technologies that allow for freedom in geometry could be considered. For example, if rectangular cross-sections are chosen, the design variables could scale the thickness without affecting convexity of the optimization problem. The connections between variable thickness members would need careful consideration.

In terms of theoretical developments, the ideas put forth in this dissertation for design of elastostatic cloaking devices leave room for several additional studies. For example, how do the nodal positions in the design region, which are currently determined by the selected coordinate transformation and remain fixed during the optimization, influence cloak effectiveness? Perhaps improved cloaking could be achieved with a different coordinate transformation (e.g., a cubic [103] or optimized coordinate transformation), or by allowing more design freedom in choosing the number of and location of nodal positions available in the design region, either *a priori* or as part of the optimization problem. Convexity of the current formulation relies on the fact that no additional degrees of freedom are added to the design lattice. One way to increase the number of degrees of freedom in the design region is to add those degrees of freedom to the reference lattice before the coordinate transformation and connect them by zero-stiffness elements that would not affect the reference lattice’s response. Other ideas to explore include the size and shape of the design region, the nodal positions within the design region, and how to best choose the load cases considered in design (e.g., load directions, number of load cases, type of loading conditions, type of support conditions).

The weighted least-squares objective function proposed here could also be explored beyond elastostatic cloaking in 2D lattices. For example, since the global stiffness characteristics of the cloaks designed here seem to mimic those of the reference, do these cloaks designed for static boundary conditions also perform well in elastodynamics? For such problems, can

the distribution of mass be incorporated into the formulation? Could the approach be useful for designing cloaks in 3D continuum problems, perhaps by using multiple, anisotropic candidate materials to enrich the material space? Is the weighted least squares objective function also effective for other physics-based inverse problems (e.g., parameter estimation)?

Multi-material topology optimization considering porous, anisotropic materials The key contributions associated with topology optimization-based design of material systems composed of multiple, porous, anisotropic materials included: 1) adopting a multi-material interpolation function that handles an arbitrary number of candidate materials; 2) endowing the material interpolation function with a mixing parameter that can be chosen to preserve convexity of the formulation at the start of the iterations and then continued toward increased penalty on material mixing in such a way that the influence of the initial guess on the final design is mitigated and (presumably) improved performance is achieved; 3) generalizing the material interpolation function to handle any material for which a stiffness elasticity tensor can be provided; and 4) providing two educational codes to facilitate adoption and extension of the formulation.

Although the proposed formulation significantly broadens the design space, mechanical performance could be further enhanced by allowing the porous, anisotropic materials to vary continuously in density and orientation. Such extension would essentially merge the homogenization-based topology optimization approach proposed by Bendsøe and Kikuchi [19] with the multi-material formulation proposed here, such that multiple classes of porous, anisotropic architectures can vary freely in the design. This extension would likely reduce the number of candidate materials needed to span the desired material space; however, the computational cost savings related to fewer material candidates comes at a tradeoff with the increased computational cost required to orient the stiffness elasticity tensors and compute the element stiffness matrices in each iteration during optimization. Furthermore, connectivity of the lattice-based materials pursued here may be challenging when orientation is free to vary; however a new class of spinodal metamaterials with tunable anisotropy [227–230] could help resolve that issue.

The formulation could also be extended to other problems. For example, architected materials likely have greater potential for problems beyond volume-constrained compliance minimization and it would be interesting to explore the formulation for problems related

to, for example, buckling resistance, damage tolerance, thermal control, and cloaking. Such problems are not as well-behaved as volume-constrained compliance minimization and will likely require further development to handle nonlinear mechanics or multiple physics. Additionally, material interfaces are a key challenge in any multi-material engineered part. The formulation proposed here does not give any special treatment to the interfaces. In fact, the material interpolation scheme, which is only concerned with penalizing mixing, leads to reduced stiffness at the interfaces. A modified multi-material interpolation function that accounts for realistic properties at the interfaces and handles potential stress concentrations is needed. Strength criteria can be considered in topology optimization by incorporating local stress constraints [242, 243], which could be integrated with the multi-material formulation pursued here. A failure criterion for each candidate material as well as interfacial failure criteria could be explored experimentally and integrated with the topology optimization problem. In the case of architected candidate materials, design of functional grading during the optimization could also be explored.

Optimal and continuous multi-lattice embedding The key contributions associated with manufacturing multi-lattice-embedded topology optimized parts included: 1) recognizing the potential of a voxel-based embedding scheme at the slice level over generating a surface representation from a 3D embedding; 2) establishing a means to functionally grade the transition between microstructural-materials and guarantee connectivity over arbitrarily-oriented microstructural-material interfaces; and 3) manufacturing 3D, multi-lattice-embedded topology optimized parts with a larger separation of length scales than previously demonstrated.

Flexibility of the multi-lattice-embedding scheme would be significantly enhanced by allowing for microstructural-materials beyond the lattice-based ones considered here. Periodic microstructural-materials characterized by plate-like unit cells or triply periodic minimal surface unit cells [225] can be directly integrated with the current embedding approach, but may require a different manufacturing approach since enclosed voids may prevent resin from draining during printing. In addition to cubic materials, periodic materials without cubic symmetry (see examples by Zok et al. [226]) can be handled directly. Non-periodic architected materials would allow for even more freedom. Those with functional representation (e.g., voronoi foams [158], spinodal metamaterials [227–230]) facilitate a voxel-based

approach and could easily be integrated with the current embedding scheme.

The intuition-based approach used here to design the microstructural-geometries in the transition regions can be limiting because it does not facilitate control over material properties in the transitions and each transition between each combination of microstructural-materials needs to be designed on a case-by-case basis. Methods with better control over material property transitions [235, 236] could be integrated with the current approach. Furthermore, the aforementioned non-periodic materials with functional representation enable transitions to be generated through a simple interpolation.

An important next step is to experimentally-validate the multi-lattice-embedded parts. Due to printing inaccuracies and defects that are exacerbated when the minimum feature size approaches the resolution of the 3D printer, it is unlikely that the printed microstructural-materials will behave as predicted numerically [244, 245]. Thus, the first step in experimental validation will be to characterize the microstructural material properties, which may have directional dependence not only because of the anisotropies associated with the expected geometry, but also because of anisotropies introduced by the printing process. Quantifying the full anisotropic behavior experimentally is not practical, but perhaps a combination of experimental and numerical testing can capture the important aspects of the microstructural-material behavior. The UV curing (during and post printing) may also introduce variability that needs to be quantified. At the structure level, in addition to quantifying the displacement behavior of the multi-lattice-embedded parts (i.e., validate the compliance), it will be interesting to investigate potential local microstructural-material failures and the behavior of the transition regions.

Lastly, scalability was only demonstrated here considering a maximum scale factor of two, relative to the printer’s build volume, which is still relatively small. Although the embedding scheme scales arbitrarily, printing multi-meter-scale macrostructures with millimeter-scale microstructures will be an extreme challenge in practice due to for example, the adhesive forces and excessive heat generation that occur in large-scale SLA 3D printing. Progress is being made in this area [231] that may make large-scale multi-lattice-embedded parts possible. Scaling could also be explored using other rapidly advancing technologies in the construction industry (e.g., concrete additive manufacturing [246]).

Appendices

APPENDIX A
NONLINEAR EQUILIBRIUM SOLUTION SCHEME

The damped Newton algorithm used to solve the nonlinear equilibrium equations associated with the cable topology optimization problem in chapter 2 is provided in Algorithm 1 [43, 65, 72]. In line 8 of the algorithm, line search is used to compute the step length for the solution update. The line search algorithm is provided in Algorithm 2, where the Armijo condition is stated on line 4 and the backtracking parameter is computed using a quadratic interpolant on line 5 [71]. This line search algorithm has been used for topology optimization considering various nonlinear mechanics models by [63, 247].

Algorithm 1 Damped Newton Algorithm

```

1: assume  $\mathbf{u}_0$  is the displacement at the previous optimization iteration
2: set maxIter = 30
3: for  $k = 0$  to maxIter do
4:   compute  $\mathbf{T}(\mathbf{u}_k)$ 
5:   compute  $\mathbf{R}(\mathbf{u}_k) = \mathbf{T}(\mathbf{u}_k) - \mathbf{F}$ 
6:   compute  $\mathbf{K}^{t,\eta}(\mathbf{u}_k)$ 
7:   compute  $\Delta\mathbf{u}_k = -\mathbf{K}^{t,\eta}(\mathbf{u}_k)^{-1} \mathbf{R}(\mathbf{u}_k)$ 
8:   find line search parameter,  $0 < \tau_{min} \leq \tau_k \leq 1$ 
9:   update  $\mathbf{u}_{k+1} = \mathbf{u}_k + \tau_k \Delta\mathbf{u}_k$ 
10:  if  $\|\mathbf{R}(\mathbf{u}_k)\|/\|\mathbf{F}\| < \text{tol}$  or  $\|\Delta\mathbf{u}_k\|/(1 + \|\mathbf{u}_{k+1}\|) < \text{tol}$  then
11:    break
12:  end if
13: end for

```

Algorithm 2 Line Search Algorithm

```
1: input  $\mathbf{u}_k, \Delta\mathbf{u}_k$ 
2: set  $j = 0, s = 10^{-4}, \tau_{\min} = 10^{-6}, \tau_{\max} = 1, \tau_k^0 = \tau_{\max}$ 
3: compute  $\mathbf{u}_{k+1} = \mathbf{u}_k + \tau_k^0 \Delta\mathbf{u}_k$ 
4: while  $\Pi(\mathbf{u}_{k+1}) > \Pi(\mathbf{u}_k) + s\tau_k^j \nabla\Pi(\mathbf{u}_k)^T \Delta\mathbf{u}_k$  do
5:   compute  $\mu = -0.5 \nabla\Pi(\mathbf{u}_k)^T \Delta\mathbf{u}_k \tau_k^j / (\Pi(\mathbf{u}_{k+1}) - \Pi(\mathbf{u}_k) - \nabla\Pi(\mathbf{u}_k)^T \Delta\mathbf{u}_k \tau_k^j)$ 
6:   if  $\mu < 0.1$  or  $\nabla\Pi(\mathbf{u}_k)^T \Delta\mathbf{u}_k \geq 0$  then
7:      $\mu = 0.5$ 
8:   end if
9:   update  $\tau_k^{j+1} = \mu \tau_k^j$ 
10:  update  $\mathbf{u}_{k+1} = \mathbf{u}_k + \tau_k^{j+1} \Delta\mathbf{u}_k$ 
11:  update  $j = j + 1$ 
12: end while
13: Return  $\tau_k = \tau_k^j$ 
```

APPENDIX B

DESIGN VARIABLE UPDATE SCHEMES

The volume constrained stiffness optimization problems in this dissertation (chapter 2 and chapter 4) are solved using sequential approximate optimization (SAO) algorithms. The basic idea is to generate and solve a series of convex approximations of the original problem in order to iteratively move toward an optimal solution. Two particular SAO algorithms are considered: the Optimality Criteria (OC) method (chapter 2) [19, 74] and the ZPR update scheme (chapter 4) [30]. In this appendix, the ZPR update scheme is derived in detail, but it is noted that the ZPR update scheme reduces to the OC method used in chapter 2 when only one volume constraint is considered.

B.1 ZPR design variable update scheme derivation

In each iteration, the problem in Equation 4.1 is linearized about the current design, \mathbf{Z}^0 , to arrive at the primal subproblem,

$$\begin{aligned}
 \min_{\mathbf{z} \in [\underline{\rho}, \bar{\rho}]^{N \times m}} \quad & \tilde{f}(\mathbf{Z}) = f(\mathbf{Z}^0) + \sum_{i=1}^m \left(\left. \frac{\partial f}{\partial \xi_i} \right|_{\mathbf{z}=\mathbf{z}^0} \right)^T (\xi_i(\mathbf{z}_i) - \xi_i(\mathbf{z}_i^0)) \\
 \text{s.t.} \quad & \tilde{g}_j(\mathbf{Z}) = g_j(\mathbf{Z}^0) + \sum_{i=1}^m \left(\left. \frac{\partial g_j}{\partial \mathbf{z}_i} \right|_{\mathbf{z}=\mathbf{z}^0} \right)^T (\mathbf{z}_i - \mathbf{z}_i^0) \leq 0, \quad j = 1, \dots, K,
 \end{aligned} \tag{B.1}$$

where \tilde{f} and $\tilde{g}_1, \dots, \tilde{g}_K$ denote the linearized objective and constraint functions, respectively, and exponential intervening variables, $\xi_{\ell i}(z_{\ell i}) = z_{\ell i}^\alpha$ with $\alpha < 0$, are introduced to provide curvature to the approximation of the objective function without introducing cumbersome higher-order derivatives [75, 248]. Solution of the primal subproblem leads to the design update,

$$z_{\ell i}^{\text{new}} = \begin{cases} z_{\ell i}^+, & z_{\ell i}^* \geq z_{\ell i}^+ \\ z_{\ell i}^-, & z_{\ell i}^* \leq z_{\ell i}^- \\ z_{\ell i}^*, & \text{otherwise,} \end{cases} \tag{B.2}$$

where $z_{\ell_i}^{\text{new}}$ is the design at the next iteration and $z_{\ell_i}^*$ is the candidate design for the next iteration (solution to the primal problem) that is accepted if it is within bounds

$$z_{\ell_i}^- = \max(\underline{\rho}, z_{\ell_i}^0 - M) \quad \text{and} \quad z_{\ell_i}^+ = \min(\bar{\rho}, z_{\ell_i}^0 + M) \quad (\text{B.3})$$

defined by box constraints, $\bar{\rho}$ and $\underline{\rho}$, and move limit, M . The move limits are imposed since the convex approximation in Equation B.1 is only accurate in the vicinity of \mathbf{Z}^0 .

The solution, $\mathbf{Z}^* = \{z_{\ell_1}^*, \dots, z_{\ell_m}^*\}_\ell^N$, to the primal subproblem in Equation B.1, is found using Lagrangian duality. The Lagrangian function of the primal subproblem is formed by augmenting the linearized objective function by a weighted sum of the constraint functions,

$$\begin{aligned} \mathcal{L}(\mathbf{Z}, \Lambda_1, \dots, \Lambda_K) &= \sum_{i=1}^m \mathbf{b}_i^T \boldsymbol{\xi}_i(\mathbf{z}_i) + \sum_{j=1}^K \Lambda_j \tilde{g}_j(\mathbf{z}_i) \\ &= \sum_{j=1}^K \left[\sum_{i \in \mathcal{G}_j} \mathbf{b}_i^T \boldsymbol{\xi}_i(\mathbf{z}_i) + \Lambda_j \tilde{g}_j(\mathbf{z}_i) \right], \end{aligned} \quad (\text{B.4})$$

where $\Lambda_1, \dots, \Lambda_K$ are the Lagrange multipliers (dual variables) associated with constraints, g_1, \dots, g_K , respectively. In Equation B.4, constant terms in the linearized objective function are neglected and the derivative of the objective function, $\partial f / \partial \boldsymbol{\xi}_i$, is denoted by \mathbf{b}_i .

The dual function, $d(\Lambda_1, \dots, \Lambda_K)$, is defined as the minimum of the Lagrangian function,

$$\begin{aligned} d(\Lambda_1, \dots, \Lambda_K) &= \min_{\substack{z_{\ell_i}^- \leq z_{\ell_i} \leq z_{\ell_i}^+ \\ \ell=1, \dots, N, i=1, \dots, m}} \mathcal{L}(\mathbf{Z}, \Lambda_1, \dots, \Lambda_K) \\ &= \sum_{j=1}^K \left[\min_{\substack{z_{\ell_i}^- \leq z_{\ell_i} \leq z_{\ell_i}^+ \\ \ell=1, \dots, N, i=1, \dots, m}} \sum_{i \in \mathcal{G}_j} \mathbf{b}_i^T \boldsymbol{\xi}_i(\mathbf{z}_i) + \Lambda_j \tilde{g}_j(\mathbf{z}_i) \right] \\ &= \sum_{j=1}^K d_j(\Lambda_j). \end{aligned} \quad (\text{B.5})$$

Note that the dual function can be expressed as the summation of K separate minimization problems. Additionally, the dual function is concave and for $\Lambda_j \geq 0, j = 1, \dots, K$, it can be shown that the dual function provides a lower bound to the solution of the primal subproblem, i.e., $d(\Lambda_1, \dots, \Lambda_K) \leq \tilde{f}^*$. The best lower bound can be obtained by solving the

convex dual subproblem,

$$\begin{aligned} & \max_{\Lambda_1, \dots, \Lambda_K} d(\Lambda_1, \dots, \Lambda_K) \\ \text{s.t.} \quad & \Lambda_j \geq 0, j = 1, \dots, K, \end{aligned} \quad \text{which can be written as } \sum_{j=1}^K \max_{\Lambda_j \geq 0} d_j(\Lambda_j). \quad (\text{B.6})$$

Since the primal subproblem is convex and under the assumption that it is also feasible, strong duality holds, i.e., $d^*(\Lambda_1, \dots, \Lambda_K) = \tilde{f}^*$. Thus, the solution to the dual subproblem leads to the solution of the primal subproblem [249].

The stationary conditions of the dual are stated as

$$\frac{\partial d}{\partial \Lambda_j} = 0 = \frac{\partial d}{\partial z_{\ell i}} \frac{\partial z_{\ell i}}{\partial \Lambda_j} + \frac{\partial d}{\partial \Lambda_j}, \quad j = 1, \dots, K, \quad (\text{B.7})$$

where the first term vanishes due to the Karush-Kuhn-Tucker (KKT) optimality conditions of the primal subproblem (i.e., stationary conditions of the Lagrangian function). Thus, the stationary conditions of the dual function imply that

$$0 = g_j(\mathbf{z}^0) + \sum_{i=1}^m \left(\frac{\partial g_j}{\partial z_i} \Big|_{\mathbf{z}=\mathbf{z}^0} \right)^T (\mathbf{z}_i(\Lambda_j) - \mathbf{z}_i^0), \quad (\text{B.8})$$

which is a monotonic function in Λ_j such that Λ_j can be solved for using an interval reducing method (e.g., bisection).

With $\Lambda_j, j = 1, \dots, K$ known, the solution, $z_{\ell i}^*$, to the primal subproblem can be obtained from the stationary conditions of the Lagrangian function (strong duality is used here),

$$\frac{\partial \mathcal{L}}{\partial z_{\ell i}} = 0 = \frac{\partial}{\partial z_{\ell i}} \left(\sum_{i \in \mathcal{G}_j} \mathbf{b}_i^T \xi(\mathbf{z}_i) \right) + \Lambda_j \frac{\partial \tilde{g}_j}{\partial z_{\ell i}}. \quad (\text{B.9})$$

Upon solving Equation B.9, the following solution to the primal problem is found to be

$$z_{\ell i}^* = z_{\ell i}^0 (B_{\ell i})^{\frac{1}{1-\alpha}}, \quad \text{where } B_{\ell i} = - \frac{\frac{\partial f}{\partial z_{\ell i}} \Big|_{\mathbf{z}=\mathbf{z}_0}}{\Lambda_j \frac{\partial \tilde{g}_j}{\partial z_{\ell i}} \Big|_{\mathbf{z}=\mathbf{z}_0}}. \quad (\text{B.10})$$

Note that due to separability of the dual function, the update of each design variable is dependent only on the Lagrange multiplier of the constraint it is associated with. As a result, the update can be performed for each volume constraint independently.

B.2 ZPR design variable update scheme with sensitivity separation

The original ZPR update scheme was derived for a monotonically decreasing objective function. Since the derivatives of the objective function in Equation 4.1 may become positive in regions of material mixing [8, 9], sensitivity separation is integrated into the ZPR update scheme [11, 143, 144] by decomposing the objective function gradient into positive and negative components to arrive at the following non-monotonic, convex approximation of the objective function:

$$\tilde{f}(\mathbf{Z}) = f(\mathbf{Z}^0) + \sum_{i=1}^m \left(\frac{\partial f^-}{\partial \xi_i} \Big|_{\mathbf{z}_i = \mathbf{z}_i^0} \right)^T (\xi_i(\mathbf{z}_i) - \xi_i(\mathbf{z}_i^0)) + \left(\frac{\partial f^+}{\partial \mathbf{z}_i} \Big|_{\mathbf{z}_i = \mathbf{z}_i^0} \right)^T (\mathbf{z}_i - \mathbf{z}_i^0). \quad (\text{B.11})$$

The approximation in Equation B.11 is the sum of a constant term that can be neglected for optimization, a monotonically decreasing convex function in exponential intervening variables, $\xi_{\ell i}(z_{\ell i}) = z_{\ell i}^\alpha$, $\alpha < 0$, and a monotonically (linearly) increasing function. For decomposition of the objective gradient, approximate second order information is used to account for the curvature of the objective function such that the negative and positive components, respectively, are

$$\frac{\partial f^-}{\partial z_{\ell i}} = \min \left(-\frac{|h_{\ell i}^0| z_{\ell i}^0}{1 + \alpha}, \frac{\partial f}{\partial z_{\ell i}} \right) \quad \text{and} \quad \frac{\partial f^+}{\partial z_{\ell i}} = \frac{\partial f}{\partial z_{\ell i}} - \frac{\partial f^-}{\partial z_{\ell i}}, \quad (\text{B.12})$$

where $h_{\ell i}^0$ is a Broyden-Fletcher-Goldfarb-Shanno (BFGS) [250] approximation of the diagonal terms of the objective function's Hessian matrix [143].

By replacing the monotonic, convex approximation of the objective function in Equation B.1 with the non-monotonic, convex approximation of the objective function in Equation B.11 and using a similar strategy as that used in section B.1, the following solution to the primal problem is found to be

$$z_{\ell i}^* = z_{\ell i}^0 (B_{\ell i})^{\frac{1}{1 - \alpha}}, \quad \text{where} \quad B_{\ell i} = - \frac{\frac{\partial f^-}{\partial z_{\ell i}} \Big|_{\mathbf{z} = \mathbf{z}^0}}{\frac{\partial f^+}{\partial z_{\ell i}} \Big|_{\mathbf{z} = \mathbf{z}^0} + \Lambda_j \frac{\partial g_j}{\partial z_{\ell i}} \Big|_{\mathbf{z} = \mathbf{z}^0}}. \quad (\text{B.13})$$

APPENDIX C

ELASTOSTATIC CLOAKING SUPPLEMENTARY INFORMATION

In this appendix, additional information related to the elastostatic cloaking formulation from chapter 3 is provided. Specifically, the DLT approach used for comparison in chapter 3 is benchmarked against the DLT approach used by Bückmann et al. [98], a single-load-case study is performed with symmetry enforced on the design variables, and convergence plots for all of the optimization-based GS1 cloak designs from chapter 3 are provided.

C.1 Direct lattice transformation benchmark

Since the lattice systems are modeled differently here than the ones by Bückmann et al. [98] and the stiffness transformation is also accomplished differently (see section 3.5), a benchmark is provided to justify comparison between the DLT and optimization-based approaches provided in section 3.5. The benchmark is performed using the target problem described in Figure 3.3 with lattice parameters selected to mimic those in one of the examples by Bückmann et al. [98].

The unitless lattice parameters used in the benchmark are outlined in Table C.1. The model used by Bückmann et al. [98] seems to be a 2D, continuum finite element model. Thus, several discrepancies between it and the beam model used here are noted in Table C.1. First, the lattice elements in the model by Bückmann et al. [98] have variable width: the width is w at the nodes and W at midspan. Additionally, since they presumably use a 2D model, the elements have no thickness. The beam cross-sectional dimensions used here were chosen to be $t = 0.2$ and $h = 0.8$ such that the maximum displacement in the lattice without any cloak most closely matches that observed by Bückmann et al. [98]. It is also noted that the Euler-Bernoulli beam equations used here neglect the Poisson effect, which is accounted for by Bückmann et al. [98] in their continuum model.

To evaluate the effectiveness of the DLT design obtained here and compare it with the results provided by Bückmann et al. [98], the $\theta = 0$ lattice is subjected to a uniform, horizontal compressive pressure of $p = 3.3 \times 10^{-5}$, applied as nodal loads on the left and

Table C.1: Lattice parameters used for direct lattice transformation benchmark

	Continuum model [98]	Beam model
lattice width, L_x	152.04	152.04
lattice height, L_y	139.65	139.65
reference hexagon edge length, L	4	4
number of hexagons along lattice width, N_x	22	22
number of hexagons along lattice height, N_y	12	12
hole radius, r_1	30	30
hole semi-major axis parameter, k_x	1	1
hole semi-minor axis parameter, k_y	1	1
design region outer radius, R_2	60	60
Young's modulus, E	3	3
Poisson's ratio, ν	0.4	–
reference lattice element thickness	–	$t = 0.2$
reference lattice element width	$w = 0.8, W = 1$	$h = 0.8$

right edges of the lattice, divided equally between the nodes on each edge.¹ Sliding supports are also considered at the top and bottom edges of the lattice. The left side of Figure C.1 shows the horizontal and vertical strain fields of the reference lattice, lattice with no cloak, and lattice with DLT cloak obtained by Bückmann et al. [98] and the right side shows the corresponding fields of the three lattice systems obtained here. Strain in the x and y are computed as the displacement in the x or y , divided by half of the lattice width, L_x , or height, L_y . The values of Δ are also reported for each lattice system and demonstrate that the approach pursued here is comparable to that used by Bückmann et al. [98].

C.2 Effect of enforcing symmetry in single-load-case design

The load directions used to obtain each of the multi-load-case designs in subsection 3.5.1 were selected specifically to preserve the symmetries of the reference lattice. It is worth investigating whether enforcing symmetry directly on the design variables and considering a single load case in design is another effective way to achieve multi-directional elastostatic cloaks. The study provided in Figure 3.11 is revisited, with symmetry enforced on the design variables in all cases that consider only a single load direction. To enforce symmetry, all lattice elements in the design region with centroid falling within 30° of the x axis are taken as

¹ Dividing the pressure equally between the nodes differs from the approach used in chapter 3, where the pressure is distributed according to tributary width such that the nodes at the corners receive less load than those away from the corners.

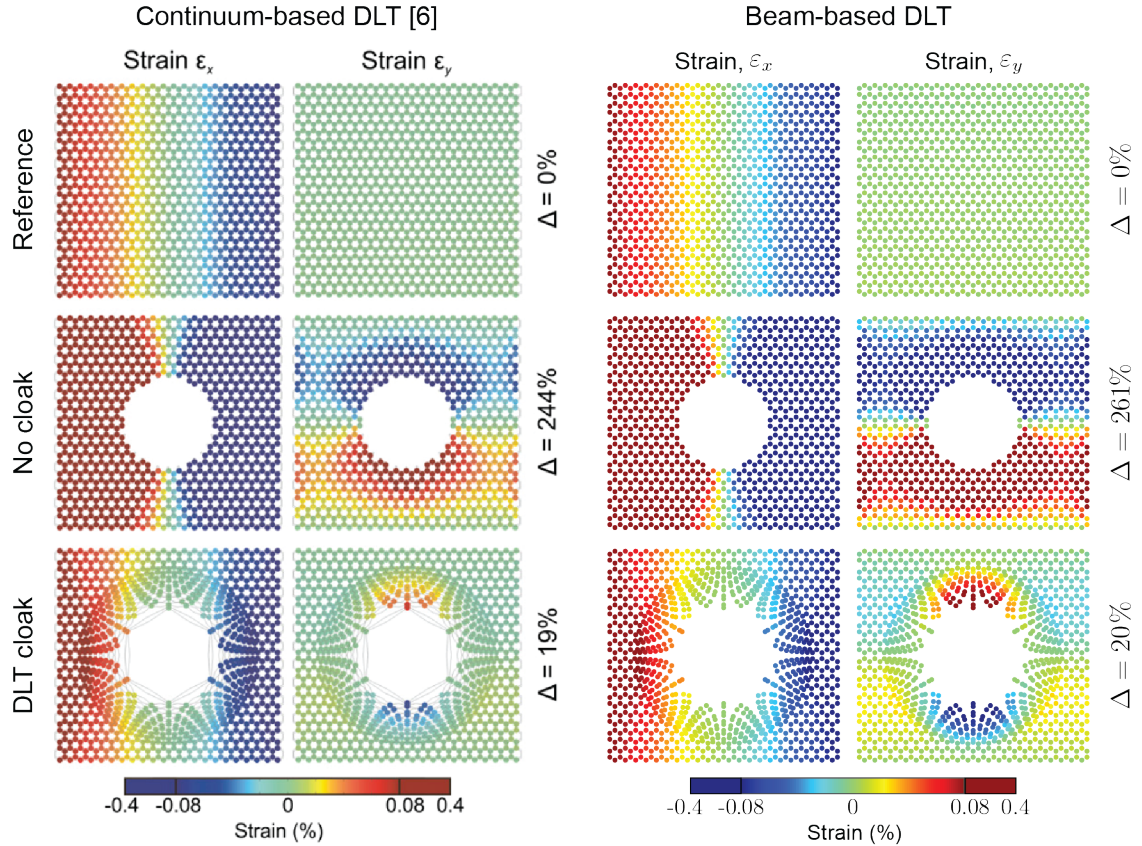


Figure C.1: Direct lattice transformation benchmark (no optimization): The left side shows results taken from Bückmann et al. [98] and the right side shows the results obtained using the Euler-Bernoulli beam elements and the simplified DLT axial stiffness transformation used here. Strain fields in the horizontal (x) and vertical (y) directions are provided for the reference lattice, lattice without any cloak, and lattice with cloak designed by DLT. The colorbar on the right side is stretched to match that used by Bückmann et al. [98] on the left side; that is, all values greater than 20% of the maximum (or less than 20% of the minimum) are mapped to the maximum value (or minimum value) and all other values are scaled by a factor of 5.

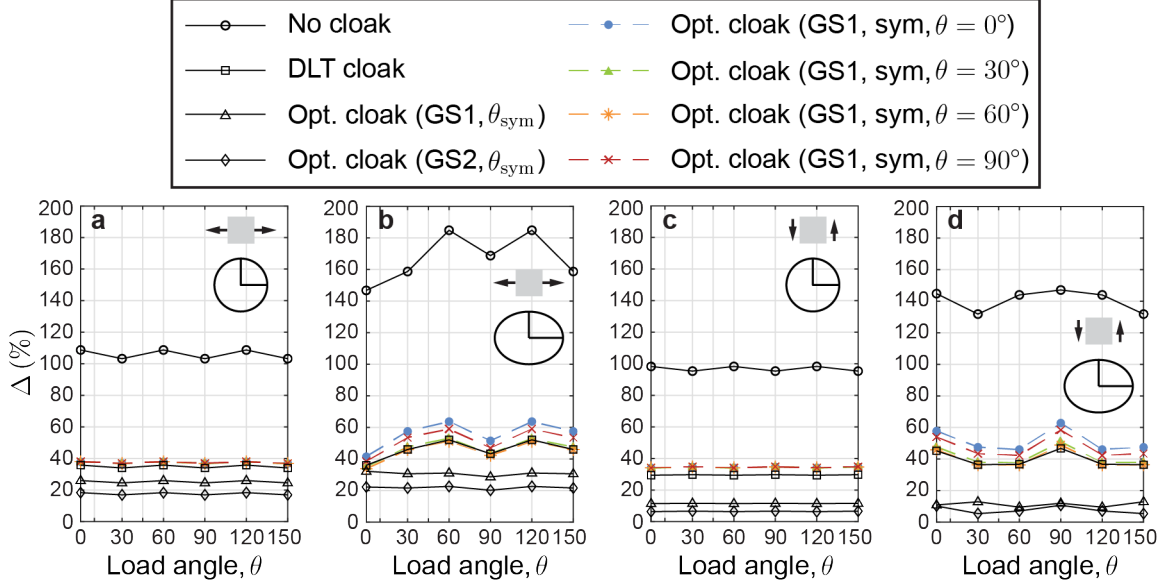


Figure C.2: Cloaking metric, Δ , versus load angle, θ , for the no cloak, DLT cloak, and optimization-based cloaks: In (a) and (b), the cloaks are evaluated under tensile loading for a circular ($k_x = 1$) and an elliptical ($k_x = 1.25$) hole, respectively; In (c) and (d), the cloaks are evaluated under shear loading for a circular ($k_x = 1$) and an elliptical ($k_x = 1.25$) hole, respectively. The single-load-case, optimized cloaks were designed with symmetry enforced on the design variables such that the six rotational and mirror symmetries (with respect to the reference lattice nodal positions) are preserved.

design variables and their values are mapped to the remaining lattice elements such that the six rotational and mirror symmetries (with respect to the reference lattice nodal positions) are preserved. The general approach for enforcing symmetry in topology optimization is taken from Almeida et al. [251]. The resulting values of Δ are provided in Figure C.2, where it is demonstrated that enforcing symmetry on the design variables has a negative impact on cloaking performance and causes the single-load-case, optimized cloaks to perform worse than the DLT cloak in all cases.

C.3 Convergence of the elastostatic cloaking formulation

Convergence of the objective function, f , is shown for the GS1 cloaks with a circular ($k_x = 1$) and an elliptical ($k_x = 1.25$) hole in Figure C.3a and b, respectively, considering single load directions, $\theta = 0^\circ$, $\theta = 30^\circ$, $\theta = 60^\circ$, and $\theta = 90^\circ$, and the three different sets of multiple load directions studied in subsection 3.5.2. The objective function decreases monotonically over the optimization iterations for all of the single- and multi-load-case problems studied here. The objective function value of the single-load-case designs tend to reduce at a faster

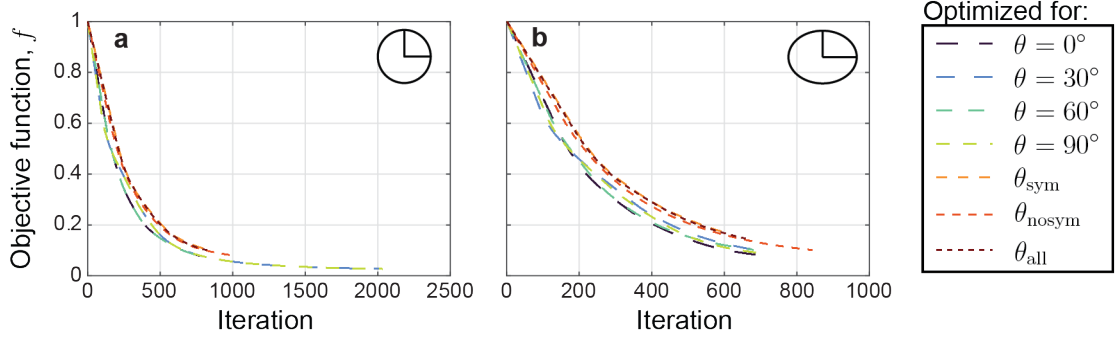


Figure C.3: Convergence of the objective function, f , for optimization-based elastostatic cloak designs based on ground structure GS1 and considering single load directions, $\theta = 0^\circ$, $\theta = 30^\circ$, $\theta = 60^\circ$, and $\theta = 90^\circ$; and multiple load directions, $\theta_{\text{sym}} = [0^\circ, 30^\circ, 60^\circ, 90^\circ, 120^\circ, 150^\circ]$, $\theta_{\text{nosym}} = [10^\circ, 45^\circ, 80^\circ, 100^\circ, 135^\circ, 170^\circ]$, $\theta_{\text{all}} = [0^\circ, 10^\circ, 30^\circ, 45^\circ, 60^\circ, 80^\circ, 90^\circ, 100^\circ, 120^\circ, 135^\circ, 150^\circ, 170^\circ]$ for the (a) circular hole ($k_x = 1$) and (b) elliptical hole ($k_x = 1.25$). In each case, the objective function value is normalized to its value at the initial design.

rate than those of the multi-load-case designs.

In Figure C.4, convergence of the cloaking metric, Δ , for the GS1 cloaks evaluated in tension is reported for the single-load-case designs. Results for the circular hole ($k_x = 1$) are shown in the left column and results for the elliptical hole ($k_x = 1.25$) are shown in the right column. The cloaking metric, Δ , tends to decrease monotonically over the optimization iterations when evaluated for the load direction for which the cloak was designed; however, Δ may increase over the optimization iterations when considering load directions for which the cloak was not designed.

In Figure C.5, convergence of the cloaking metric, Δ , for the lattices evaluated in tension is reported for the GS1 multi-load-case designs. Results for the circular hole ($k_x = 1$) are shown in the left column and results for the elliptical hole ($k_x = 1.25$) are shown in the right column. The cloaking metric, Δ , decreases monotonically over the optimization iterations in almost all cases. Small increases in Δ in late iterations are observed for the tensile load cases aligned along the reference lattice's lines of symmetry (θ_{sym}) when the cloak is not designed for these load cases (middle row of Figure C.5).

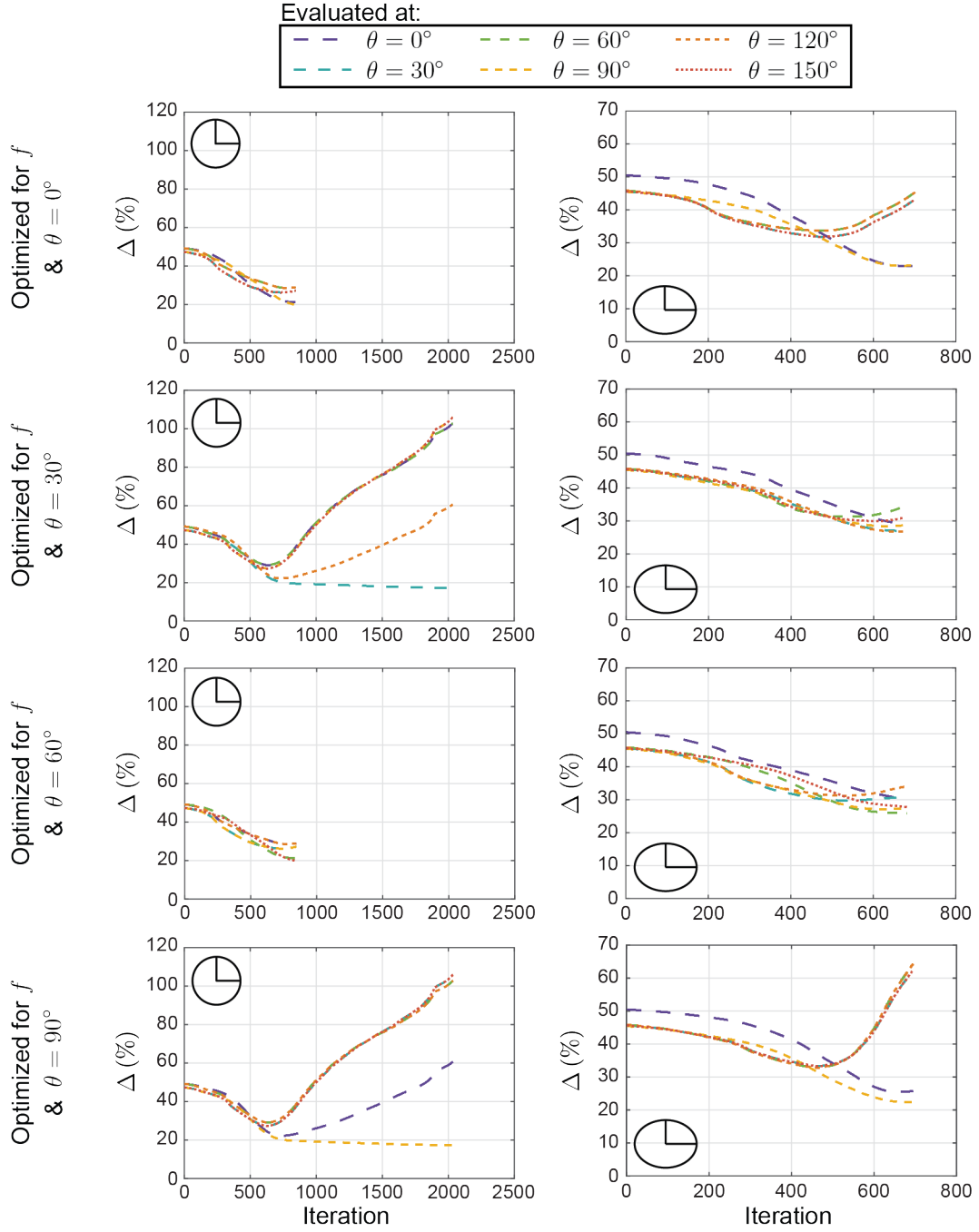


Figure C.4: Convergence of the cloaking metric, Δ , for the four different single-load-case, GS1 cloak designs evaluated in directions aligned along the reference lattice's lines of symmetry. The left and right columns show results for the circular ($k_x = 1$) and elliptical ($k_x = 1.25$) holes, respectively.

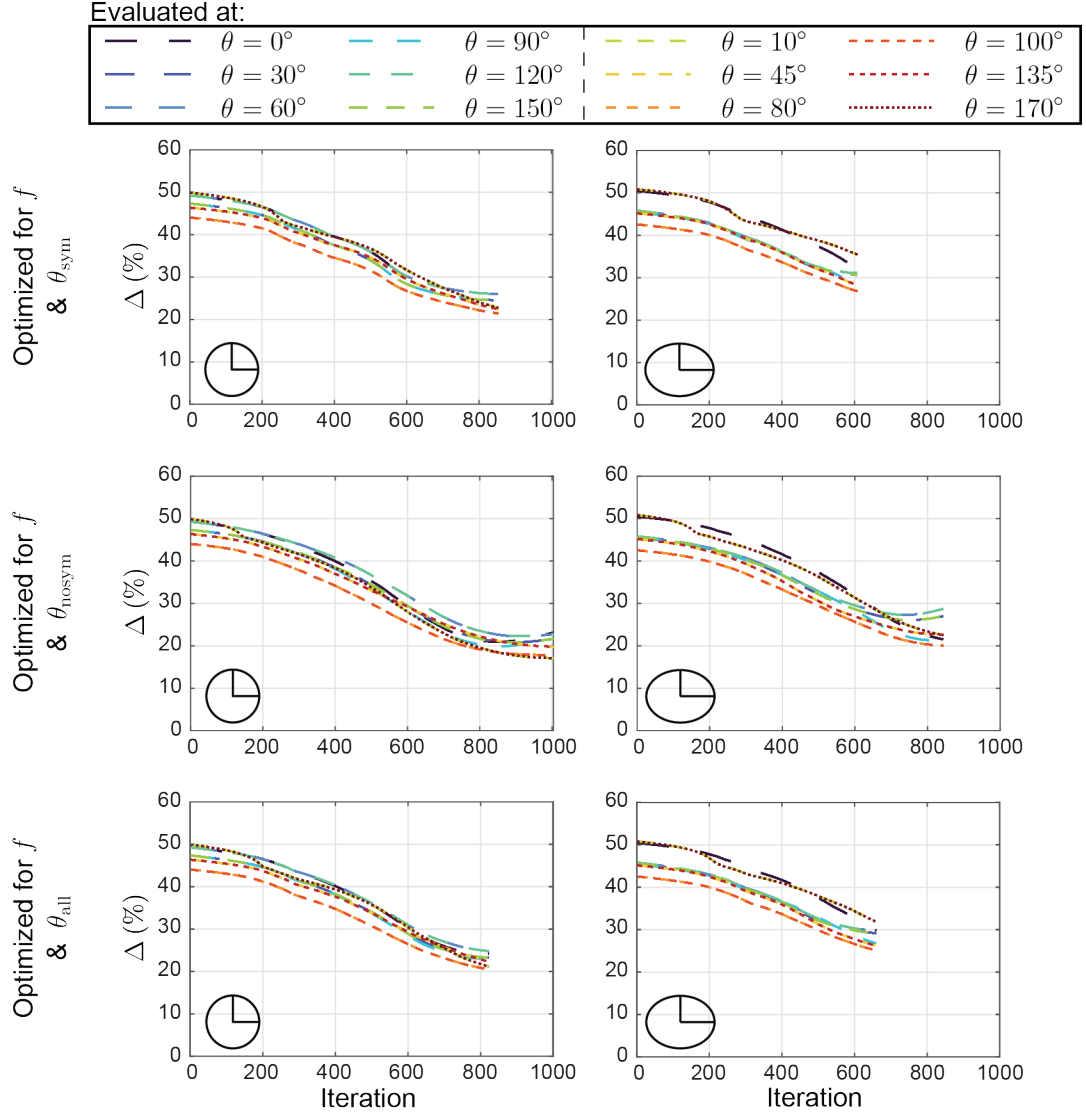


Figure C.5: Convergence of the cloaking metric, Δ , for the three different multi-load-case, GS1 cloak designs evaluated in all twelve load directions. The left and right columns show results for the circular ($k_x = 1$) and elliptical ($k_x = 1.25$) holes, respectively. Convergence for cloaks designed considering $\theta_{\text{sym}} = [0^\circ, 30^\circ, 60^\circ, 90^\circ, 120^\circ, 150^\circ]$, $\theta_{\text{nosym}} = [10^\circ, 45^\circ, 80^\circ, 100^\circ, 135^\circ, 170^\circ]$, and $\theta_{\text{all}} = [0^\circ, 10^\circ, 30^\circ, 45^\circ, 60^\circ, 80^\circ, 90^\circ, 100^\circ, 120^\circ, 135^\circ, 150^\circ, 170^\circ]$ are provided in the top, middle, and bottom rows, respectively.

APPENDIX D

POLYANISOMAT SOURCE CODE

In this appendix, the source code is provided for the PolyAnisoMat educational implementation of multi-material topology optimization that can accommodate many porous, anisotropic candidate materials.

D.1 PolyScript

```

1  %----- PolyAnisoMat -----%
2  % Modified by: Emily D. Sanders, PhD Dissertation (2021) %
3  % Advisor: Glaucio H. Paulino %
4  % Ref: ED Sanders, A Pereira, MA Aguilo, GH Paulino, "PolyMat: an %
5  %     efficient Matlab code for multi-material topology optimization," %
6  %     Struct Multidisc Optim, DOI 10.1007/s00158-018-2094-0, 2018 %
7  %-----%
8  %% ----- CREATE 'fem' STRUCT
9  clear; close all; clc;
10 addpath(genpath('./PolyMesher'),genpath('./PolyConstraints'));
11 [Node,Element,Supp,Load,Seeds] = PolyMesher(@MbbDomain,5001,30);
12 [VolFrac,ElemInd,MatInd,SElemInd,SMatInd,Mat,vhat,Color] =
13     Mbb2AnisoConstraints(Seeds);
14 rmpath('./PolyMesher','./PolyConstraints');
15 fem = struct(...
16     'NNode',size(Node,1),... % Number of nodes
17     'NElem',size(Element,1),... % Number of elements
18     'Node',Node,... % [NNode x 2] array of nodes
19     'Element',{Element},... % [NElement x Var] cell array of elements
20     'Supp',Supp,... % Array of supports
21     'Load',Load,... % Array of loads
22     'NMat',size(Mat,1),... % Number of candidate materials
23     'Mat',{Mat},... % Material properties
24     'vhat',vhat,... % Solid fractions
25     'Reg',0 ... % Tag for regular meshes
26 );
27 %% ----- CREATE 'opt' STRUCT
28 R = 0.04;
29 P = PolyFilter(fem,R);
30 zIni = zeros(fem.NElem,fem.NMat);
31 zIni = InitialGuess(VolFrac,ElemInd,MatInd,SElemInd,SMatInd,zIni);
32 opt = struct(...
33     'zMin',0.0,... % Lower bound for design variables
34     'zMax',1.0,... % Upper bound for design variables
35     'zIni',zIni,... % Initial design variables
36     'P',P,... % Matrix that maps design to element vars.
37     'VolFrac',VolFrac,... % Specified volume fraction constraint
38     'NConstr',size(VolFrac,1),... % Number of volume constraints
39     'ElemInd',{ElemInd},... % Element indices assoc. with each constr.

```

```

39     'MatInd',{MatInd},...           % Material indices assoc. with each constr.
40     'Tol',0.01,...                 % Convergence tolerance on design vars.
41     'MaxIter',100,...              % Max. number of optimization iterations
42     'ZPRMove',0.2,...              % Allowable move step in ZPR update scheme
43     'ZPREta',1/2 ...               % Exponent used in ZPR update scheme
44 );
45 %% ----- RUN 'PolyMat'
46 figure;
47 Param = [1 1.5 2 3 4;
48          0 0.3 0.5 1 1]; %continuation on material interpolation parameters
49 for i = 1:size(Param,2)
50     penal = Param(1,i); gamma = Param(2,i);
51     disp(['current p: ', num2str(penal), ...
52          ', current gamma: ', num2str(gamma)]);
53     opt.MatIntFnc = @(y,fem)MatIntFnc(y,fem,'SIMP',penal);
54     opt.MultiMatIntFnc = @(y,fem)MultiMatIntFnc(y,fem,opt.MatIntFnc,gamma);
55     [opt.zIni,v,fem] = PolyAnisoMat(fem,opt,Color);
56 end
57 %% -----

```

D.2 PolyAnisoMat

```

1  %----- PolyAnisoMat -----%
2  % Modified by: Emily D. Sanders, PhD Dissertation (2021) %
3  % Advisor: Glaucio H. Paulino %
4  % Ref: ED Sanders, A Pereira, MA Aguilo, GH Paulino, "PolyMat: an %
5  %       efficient Matlab code for multi-material topology optimization," %
6  %       Struct Multidisc Optim, DOI 10.1007/s00158-018-2094-0, 2018 %
7  %-----%
8  function [z,v,fem] = PolyAnisoMat(fem,opt,Color)
9  Iter=0; Tol=opt.Tol*(opt.zMax-opt.zMin); Change=2*Tol*ones(1,opt.NConstr); z
    =opt.zIni; P=opt.P;
10 [fem] = InitializeFEAnalysis(fem);
11 y=P*z;[fem.k,fem.dkdw,dwdy,v,dvdy] = opt.MultiMatIntFnc(y,fem);
12 [FigHandle,FigData,Color,RBM] = InitialPlot(fem,y,Color);
13 while (Iter<opt.MaxIter) && (max(Change)>Tol)
14     Iter = Iter + 1;
15     %Compute cost functionals and analysis sensitivities
16     [f,dfd,dfd,fer] = ObjectiveFnc(fem,v);
17     [g,~,dgdv,fer] = ConstraintFnc(fem,opt,v);
18     %Compute design sensitivities
19     dfdz = P'*(dfd.*dwdy + dvdy.*dfd);
20     dgdz = zeros(fem.NElem,opt.NConstr,fem.NMat);
21     for c=1:opt.NConstr
22         dgdz(:,c,:) = P'*(dvdy.*squeeze(dgdv(:,c,:)));
23         %Update design variable and analysis parameters
24         ElemInd = cell2mat(opt.ElemInd(c));
25         MatInd = cell2mat(opt.MatInd(c));
26         [z(ElemInd,MatInd),Change(c)] = UpdateScheme(...
27             dfdz(ElemInd,MatInd),g(c),...
28             dgdz(ElemInd,c,MatInd),z(ElemInd,MatInd),y(ElemInd,MatInd),opt);
29     end
30 y=P*z;[fem.k,fem.dkdw,dwdy,v,dvdy] = opt.MultiMatIntFnc(y,fem);
31 %Output results
32 fprintf('It: %i \t Objective: %1.3f \t Max. Constraint: %1.3f \t Change:
    %1.3f\n',Iter,f,max(g),max(Change));

```

```

33     rgb=zeros(fem.NElem,3),ones(size(y,1),1)];
34     for i=1:fem.NMat, rgb(:,1:3) = rgb(:,1:3) + y(:,i)*Color(i,1:3); end
35     rgbT = RBM*rgb'; rgb = rgbT';
36     I = reshape(rgb(:,1:3),fem.NElem,1,3);
37     set(FigHandle, 'FaceColor', 'flat', 'CData', I(FigData, :, :)); drawnow
38 end
39 %----- OBJECTIVE FUNCTION
40 function [f,dfd,dfdv,fem] = ObjectiveFnc(fem,v)
41 [U,fem] = FEAnalysis(fem);
42 f = dot(fem.F,U);
43 dfd = zeros(size(v));
44 for i = 1:fem.NMat
45     temp = cumsum(-U(fem.i).*fem.dkd(:,i).*U(fem.j));
46     temp = temp(cumsum(fem.ElemNDof.^2));
47     dfd(:,i) = [temp(1);temp(2:end)-temp(1:end-1)];
48 end
49 dfdv = zeros(size(v));
50 %----- CONSTRAINTS FUNCTION
51 function [g,dgdk,dgdv,fem] = ConstraintFnc(fem,opt,v)
52 if ~isfield(fem, 'ElemArea')
53     fem.ElemArea = zeros(fem.NElem,1);
54     for el=1:fem.NElem
55         vx=fem.Node(fem.Element{el},1); vy=fem.Node(fem.Element{el},2);
56         fem.ElemArea(el) = 0.5*sum(vx.*vy([2:end 1])-vy.*vx([2:end 1]));
57     end
58 end
59 g = zeros(opt.NConstr,1);
60 dgdv = zeros(fem.NElem,opt.NConstr,fem.NMat);
61 for c=1:opt.NConstr
62     ElemInd = cell2mat(opt.ElemInd(c));
63     MatInd = cell2mat(opt.MatInd(c));
64     for m = 1:size(MatInd,2)
65         g(c) = g(c) + sum(fem.ElemArea(ElemInd).*v(ElemInd,MatInd(m)))./sum(fem.
66             ElemArea(ElemInd));
67         dgdv(ElemInd,c,MatInd(m)) = fem.ElemArea(ElemInd)/sum(fem.ElemArea(
68             ElemInd));
69     end
70     g(c) = g(c) - opt.VolFrac(c,1);
71 end
72 dgdk = zeros(size(fem.k,1),opt.NConstr);
73 %----- ZPR UPDATE
74 function [zNew,Change] = UpdateScheme(dfdz,g,dgdz,z0,y0,opt)
75 nelelem = size(dfdz,1); nmat = size(dfdz,2);
76 dfdz = reshape(dfdz,nelelem*nmat,1); dgdz = reshape(dgdz,nelelem*nmat,1);
77 z0 = reshape(z0,nelelem*nmat,1); y0 = reshape(y0,nelelem*nmat,1);
78 zMin=opt.zMin; zMax=opt.zMax;
79 move=opt.ZPRMove*(zMax-zMin); eta=opt.ZPREta;
80 Bm = -(dfd./dgdz); l1=0; l2=1.4*max(Bm);
81 while l2-l1 > 1e-4
82     lmid = 0.5*(l1+l2);
83     B = -(dfd./dgdz)/lmid;
84     zCnd = zMin+(y0-zMin).*max(0,B).^eta; %MultiMatIntFnc may cause non-
85         negative sensitivities
86     zNew = max(max(min(min(zCnd,z0+move),zMax),z0-move),zMin);
87     if (g+dgdz.*(zNew-z0)>0), l1=lmid;
88     else l2=lmid; end
89 end
90 Change = max(abs(zNew-z0))/(zMax-zMin);
91 zNew = reshape(zNew,nelelem,nmat,1);

```



```

89 %----- INITIALIZE FE-ANALYSIS
90 function [fem] = InitializeFEAnalysis(fem)
91 if ~isfield(fem,'k')
92     fem.ElemNDof = 2*cellfun(@length,fem.Element); % # of DOFs per element
93     fem.i = zeros(sum(fem.ElemNDof.^2),1);
94     fem.j=fem.i; fem.e=fem.i;
95     fem.k0=zeros(sum(fem.ElemNDof.^2),fem.NMat);
96     index = 0;
97     if ~isfield(fem,'ShapeFnc'), fem=TabShapeFnc(fem); end
98     for el = 1:fem.NElem
99         Ke = cell(1,fem.NMat);
100         if ~fem.Reg || el==1,
101             for i = 1:fem.NMat
102                 Ke{i}=LocalK(fem,fem.Element{el},fem.Mat{i});
103             end
104         end
105         NDof = fem.ElemNDof(el);
106         eDof = reshape([2*fem.Element{el}-1;2*fem.Element{el}],NDof,1);
107         I=repmat(eDof ,1,NDof); J=I';
108         fem.i(index+1:index+NDof^2) = I(:);
109         fem.j(index+1:index+NDof^2) = J(:);
110         fem.k0(index+1:index+NDof^2,:) = [Ke{:}];
111         index = index + NDof^2;
112     end
113     NLoad = size(fem.Load,1);
114     fem.F = zeros(2*fem.NNode,1); %external load vector
115     fem.F(2*fem.Load(1:NLoad,1)-1) = fem.Load(1:NLoad,2); %x-crdnt
116     fem.F(2*fem.Load(1:NLoad,1)) = fem.Load(1:NLoad,3); %y-crdnt
117     NSupp = size(fem.Supp,1);
118     FixedDofs = [fem.Supp(1:NSupp,2).*(2*fem.Supp(1:NSupp,1)-1);
119                 fem.Supp(1:NSupp,3).*(2*fem.Supp(1:NSupp,1))];
120     FixedDofs = FixedDofs(FixedDofs>0);
121     AllDofs = [1:2*fem.NNode];
122     fem.FreeDofs = setdiff(AllDofs,FixedDofs);
123 end
124 %----- FE-ANALYSIS
125 function [U,fem] = FEAnalysis(fem)
126 K = sparse(fem.i,fem.j,fem.k);
127 K = (K+K')/2;
128 U = zeros(2*fem.NNode,1);
129 U(fem.FreeDofs,:) = K(fem.FreeDofs,fem.FreeDofs)\fem.F(fem.FreeDofs,:);
130 %----- ELEMENT STIFFNESS MATRIX
131 function [Ke] = LocalK(fem,eNode,D)
132 nn=length(eNode); Ke=zeros(2*nn,2*nn);
133 W = fem.ShapeFnc{nn}.W;
134 for q = 1:length(W) %quadrature loop
135     dNdx_i = fem.ShapeFnc{nn}.dNdx_i(:,:,q);
136     J0 = fem.Node(eNode,:) '*dNdx_i;
137     dNdx = dNdx_i/J0;
138     B = zeros(3,2*nn);
139     B(1,1:2:2*nn) = dNdx(:,1)';
140     B(2,2:2:2*nn) = dNdx(:,2)';
141     B(3,1:2:2*nn) = dNdx(:,2)';
142     B(3,2:2:2*nn) = dNdx(:,1)';
143     Ke = Ke+B'*D*B*W(q)*det(J0);
144 end
145 Ke = Ke(:);
146 %----- TABULATE SHAPE FUNCTIONS
147 function fem = TabShapeFnc(fem)

```

```

148 ElemNNode = cellfun(@length,fem.Element); % number of nodes per element
149 fem.ShapeFnc = cell(max(ElemNNode),1);
150 for nn = min(ElemNNode):max(ElemNNode)
151     [W,Q] = PolyQuad(nn);
152     fem.ShapeFnc{nn}.W = W;
153     fem.ShapeFnc{nn}.N = zeros(nn,1,size(W,1));
154     fem.ShapeFnc{nn}.dNdx1 = zeros(nn,2,size(W,1));
155     for q = 1:size(W,1)
156         [N,dNdx1] = PolyShapeFnc(nn,Q(q,:));
157         fem.ShapeFnc{nn}.N(:, :, q) = N;
158         fem.ShapeFnc{nn}.dNdx1(:, :, q) = dNdx1;
159     end
160 end
161 %----- POLYGONAL SHAPE FUNCTIONS
162 function [N,dNdx1] = PolyShapeFnc(nn,xi)
163 N=zeros(nn,1); alpha=zeros(nn,1); dNdx1=zeros(nn,2); dalpha=zeros(nn,2);
164 sum_alpha=0.0; sum_dalpha=zeros(1,2); A=zeros(nn,1); dA=zeros(nn,2);
165 [p,Tri] = PolyTrnglt(nn,xi);
166 for i=1:nn
167     sctr = Tri(i,:); pT = p(sctr,:);
168     A(i) = 1/2*det([pT,ones(3,1)]);
169     dA(i,1) = 1/2*(pT(3,2)-pT(2,2));
170     dA(i,2) = 1/2*(pT(2,1)-pT(3,1));
171 end
172 A=[A(nn,:);A]; dA=[dA(nn,:);dA];
173 for i=1:nn
174     alpha(i) = 1/(A(i)*A(i+1));
175     dalpha(i,1) = -alpha(i)*(dA(i,1)/A(i)+dA(i+1,1)/A(i+1));
176     dalpha(i,2) = -alpha(i)*(dA(i,2)/A(i)+dA(i+1,2)/A(i+1));
177     sum_alpha = sum_alpha + alpha(i);
178     sum_dalpha(1:2) = sum_dalpha(1:2)+dalpha(i,1:2);
179 end
180 for i=1:nn
181     N(i) = alpha(i)/sum_alpha;
182     dNdx1(i,1:2) = (dalpha(i,1:2)-N(i)*sum_dalpha(1:2))/sum_alpha;
183 end
184 %----- POLYGON TRIANGULATION
185 function [p,Tri] = PolyTrnglt(nn,xi)
186 p = [cos(2*pi*((1:nn))/nn); sin(2*pi*((1:nn))/nn)]';
187 p = [p; xi];
188 Tri = zeros(nn,3); Tri(1:nn,1)=nn+1;
189 Tri(1:nn,2)=1:nn; Tri(1:nn,3)=2:nn+1; Tri(nn,3)=1;
190 %----- POLYGONAL QUADRATURE
191 function [weight,point] = PolyQuad(nn)
192 [W,Q]= TriQuad; %integration pnts & wgts for ref. triangle
193 [p,Tri] = PolyTrnglt(nn,[0 0]); %triangulate from origin
194 point=zeros(nn*length(W),2); weight=zeros(nn*length(W),1);
195 for k=1:nn
196     sctr = Tri(k,:);
197     for q=1:length(W)
198         [N,dNds] = TriShapeFnc(Q(q,:)); %compute shape functions
199         J0 = p(sctr,:)'*dNds;
200         l = (k-1)*length(W) + q;
201         point(l,:) = N'*p(sctr,:);
202         weight(l) = det(J0)*W(q);
203     end
204 end
205 %----- TRIANGULAR QUADRATURE
206 function [weight,point] = TriQuad

```

```

207 point=[1/6,1/6;2/3,1/6;1/6,2/3]; weight=[1/6,1/6,1/6];
208 %----- TRIANGULAR SHAPE FUNCTIONS
209 function [N,dNds] = TriShapeFnc(s)
210 N=[1-s(1)-s(2);s(1);s(2)]; dNds=[-1,-1;1,0;0,1];
211 %----- INITIAL PLOT
212 function [handle,map,Color,RBM] = InitialPlot(fem,z0,Color)
213 [Color,RBM] = ConvertColors(Color);
214 Tri = zeros(length([fem.Element{:}])-2*fem.NElem,3);
215 map = zeros(size(Tri,1),1); index=0;
216 for el = 1:fem.NElem
217     for enode = 1:length(fem.Element{el})-2
218         map(index+1) = el;
219         Tri(index+1,:) = fem.Element{el}([1,enode+1,enode+2]);
220         index = index + 1;
221     end
222 end
223 I=[zeros(fem.NElem,3),ones(size(z0,1),1)];
224 for i=1:fem.NMat
225     I(:,1:3) = I(:,1:3) + z0(:,i)*Color(i,1:3);
226 end
227 IT = RBM*I';
228 I = IT';
229 handle = patch('Faces',Tri,'Vertices',fem.Node,'FaceVertexCData',...
230             I(map,1:3),'FaceColor','flat','EdgeColor','none');
231 axis equal; axis off; axis tight; caxis([0 1]);
232 %-----%
233 %----- PolyAnisoMat - History -----%
234 % version: 1.0 (Apr21)
235 %
236 % history: Created:      1-Apr-21      Emily Sanders
237 %             Supervised by:          Glaucio Paulino
238 %-----%

```

D.3 InitialGuess

The InitialGuess function is taken directly from PolyMat [9].

```

1 %----- PolyMat -----%
2 % Ref: ED Sanders, A Pereira, MA Aguiló, GH Paulino, "PolyMat: an      %
3 %     efficient Matlab code for multi-material topology optimization,"  %
4 %     Struct Multidisc Optim, DOI 10.1007/s00158-018-2094-0, 2018    %
5 %-----%
6 function [zIni] = InitialGuess(VolFrac,ElemInd,MatInd,SElemInd,SMatInd,zIni)
7 NConstr = size(VolFrac,1);
8 for i = 1:NConstr
9     MatIndices = cell2mat(MatInd(i));
10    ElemIndices = cell2mat(ElemInd(i));
11    for m = 1:size(MatIndices,2)
12        zIni(ElemIndices,MatIndices(m)) = VolFrac(i)./size(MatIndices,2);
13    end
14 end
15 NFixed = size(SElemInd,1);
16 for i = 1:NFixed
17    MatIndices = cell2mat(SMatInd(i));
18    ElemIndices = cell2mat(SElemInd(i));
19    for m = 1:size(MatIndices,2)
20        zIni(ElemIndices,MatIndices(m)) = 1;

```

```

21     end
22 end
23 %-----%

```

D.4 MatIntFnc

The `MatIntFnc` function is identical to that used in `PolyTop`, except that the solid fraction of the candidate materials, stored in `vhat`, are accounted for in computing the element volume fractions.

```

1  %----- PolyAnisoMat -----%
2  % Modified by: Emily D. Sanders, PhD Dissertation (2021) %
3  % Advisor: Glaucio H. Paulino %
4  % Ref: ED Sanders, A Pereira, MA Aguiló, GH Paulino, "PolyMat: an %
5  %     efficient Matlab code for multi-material topology optimization," %
6  %     Struct Multidisc Optim, DOI 10.1007/s00158-018-2094-0, 2018 %
7  %-----%
8  function [w,dwdy,v,dvdy] = MatIntFnc(y,fem,type,param)
9  eps = 1e-4/size(y,2);
10 switch(type)
11     case('SIMP')
12         penal = param(1);
13         w = eps+(1-eps)*y.^penal;
14         v = repmat(fem.vhat',[size(y,1),1]).*y;
15         dwdy = (1-eps)*penal*y.^(penal-1);
16         dvdy = repmat(fem.vhat',[size(y,1),1]);
17     case('SIMP-H')
18         penal = param{1};
19         beta = param{2};
20         h = 1-exp(-beta*y)+y*exp(-beta);
21         w = eps+(1-eps)*h.^penal;
22         v = repmat(fem.vhat',[size(y,1),1]).*h;
23         dhdy = beta*exp(-beta*y)+exp(-beta);
24         dwdy = (1-eps)*penal*h.^(penal-1).*dhdy;
25         dvdy = repmat(fem.vhat',[size(y,1),1]).*dhdy;
26     case('RAMP')
27         q = param(1);
28         w = eps+(1-eps)*y./(1+q*(1-y));
29         v = repmat(fem.vhat',[size(y,1),1]).*y;
30         dwdy = ((1-eps)*(q+1))./(q-q*y+1).^2;
31         dvdy = repmat(fem.vhat',[size(y,1),1]);
32     case('RAMP-H')
33         q = param{1};
34         beta = param(2);
35         h = 1-exp(-beta*y)+y*exp(-beta);
36         w = eps+(1-eps)*h./(1+q*(1-h));
37         v = repmat(fem.vhat',[size(y,1),1]).*h;
38         dhdy = beta*exp(-beta*y)+exp(-beta);
39         dwdy = ((1-eps)*(q+1))./(q-q*h+1).^2.*dhdy;
40         dvdy = repmat(fem.vhat',[size(y,1),1]).*dhdy;
41 end
42 %-----%

```

D.5 MultiMatIntFnc

```
1 %----- PolyAnisoMat -----%
2 % Modified by: Emily D. Sanders, PhD Dissertation (2021) %
3 % Advisor: Glaucio H. Paulino %
4 % Ref: ED Sanders, A Pereira, MA Aguilo, GH Paulino, "PolyMat: an %
5 % efficient Matlab code for multi-material topology optimization," %
6 % Struct Multidisc Optim, DOI 10.1007/s00158-018-2094-0, 2018 %
7 %-----%
8 function [k,dkdw,dwdy,v,dvdy] = MultiMatIntFnc(y,fem,MatIntFnc,param)
9 gamma = param;
10 [w,dwdy,v,dvdy] = MatIntFnc(y,fem);
11 NElem = size(y,1); NMat = size(y,2);
12 ElemNDof = fem.ElemNDof;
13 dkdw = zeros(sum(ElemNDof.^2),NMat);
14 % Compute k
15 S = 1-gamma*w; S(S==0)=1;
16 Prod = w.*repmat(prod(S,2),1,NMat)./S;
17 Prodtmp = repelem(Prod,ElemNDof.^2,1);
18 k = sum(Prodtmp.*fem.k0,2);
19 % Compute dkdw
20 for m = 1:NMat
21     M1 = zeros(NElem,NMat);
22     M1(:,m) = prod(S,2)./S(:,m);
23     M2 = -gamma.*Prod./repmat(S(:,m),1,NMat); M2(:,m) = zeros(NElem,1);
24     dProd = M2 + M1;
25     dProdtmp = repelem(dProd,ElemNDof.^2,1);
26     dkdw(:,m) = sum(dProdtmp.*fem.k0,2);
27 end
28 %-----%
```

D.6 ConvertColors

The `ConvertColors` function is used to enable a result visualization in which white represents void. It is taken directly from `PolyMat` [9].

To plot the multi-material density in a given element, the RGB Colors associated with each material in the element are summed and scaled by the corresponding material densities, to find a point on the unit RGB color cube representing the mixing in that element. In general, the unit RGB color cube is defined such that black represents void (0,0,0) and white represents full mixing (1,1,1). Thus, `ConvertColors` is called before the optimization loop in `PolyMat`/`PolyAnisoMat` to compute a rigid body motion matrix, RBM, that can be applied to the RGB color cube so that white is at the origin. Using RBM, the RGB Colors specified by the user are converted to the corresponding colors on the rotated unit RGB color cube (e.g., red (1,0,0) goes to cyan (0,1,1), green (0,1,0) goes to yellow (1,1,0), and blue (0,0,1) goes to magenta (1,0,1)). These converted colors are scaled by the density

of each material and summed in line 30 of PolyMat or line 34 of PolyAnisoMat to find a point on the rotated unit RGB color cube representing the mixing in a given element, where void elements have RGB value (0,0,0). When rotated back to the original unit RGB color cube, the specified color scheme is re-gained and the void elements are now at (1,1,1), which represents white. The code used to compute the RBM matrix and convert the user-specified colors is provided here.

```

1 %----- PolyMat -----%
2 % Ref: ED Sanders, A Pereira, MA Aguilo, GH Paulino, "PolyMat: an      %
3 %     efficient Matlab code for multi-material topology optimization,"  %
4 %     Struct Multidisc Optim, DOI 10.1007/s00158-018-2094-0, 2018    %
5 %-----%
6 function [Color,RBM] = ConvertColors(Color)
7 %This function applies a rigid body motion to the RGB cube such that white
8 % is at [0 0 0] and black is at [1 1 1]. That way, we plot the
9 % multi-material results on a white background.
10 %Translation Matrix
11 T1 = eye(4,4); T1(1:3,4) = [-0.5;-0.5;-0.5];
12 %Rotation Matrix
13 thz = pi; thx = pi/2;
14 Rz = [cos(thz) -sin(thz) 0 0;sin(thz) cos(thz) 0 0;0 0 1 0;0 0 0 1];
15 Rx = [1 0 0 0;0 cos(thx) -sin(thx) 0;0 sin(thx) cos(thx) 0;0 0 0 1];
16 %Translation Matrix
17 T2 = eye(4,4); T2(1:3,4) = [0.5;0.5;0.5];
18 %Full rigid body motion matrix
19 RBM = T2*Rx*Rz*T1;
20 %Converted colors
21 Color = [Color, ones(size(Color,1),1)];
22 ColorT = RBM*Color';
23 Color = ColorT';
24 %-----%

```

D.7 Mbb2AnisoConstraints

The constraint file for the MBB beam problem designed considering two porous, anisotropic candidate materials in subsection 5.4.4 is provided here. Note that `Mat` is a cell array containing the stiffness elasticity tensors of the two candidate materials and `vhat` is specified to account for their solid fractions.

```

1 %----- PolyAnisoMat -----%
2 % Modified by: Emily D. Sanders, PhD Dissertation (2021)              %
3 % Advisor: Glaucio H. Paulino                                         %
4 % Ref: ED Sanders, A Pereira, MA Aguilo, GH Paulino, "PolyMat: an      %
5 %     efficient Matlab code for multi-material topology optimization,"  %
6 %     Struct Multidisc Optim, DOI 10.1007/s00158-018-2094-0, 2018    %
7 %-----%
8 function [VolFrac,ElemInd,MatInd,SElemInd,SMatInd,Mat,vhat,Color] =
    Mbb2AnisoConstraints(Seeds)

```

```

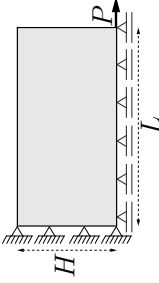
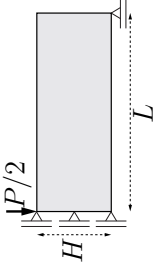
9   % Parameters
10  NElem = size(Seeds,1);
11  NMat = 2;
12  VolFrac = 0.3;
13  % Materials
14  [MatInd,Mat,vhat,Color] = MaterialsPerConstraint(NMat);
15  % Constrained regions
16  ElemInd{1} = linspace(1,NElem,NElem);
17  % Passive regions
18  SElemInd = []; SMatInd = [];
19  %----- GET MATERIALS PER CONSTRAINT
20  function [MatInd,Mat,vhat,Color] = MaterialsPerConstraint(NMat)
21  %Cross with horizontals top and bottom, E=0.75, nu=0.3, t = 0.2
22  Mat{1,1} = [0.283994135147905,0.0991736450570570,0;
23             0.0991736450570570,0.160516511103827,0;
24             0,0,0.104359003867811];
25  %Cross with verticals left and right, E=1, nu=0.3, t = 0.2
26  Mat{2,1} = [0.214897989423356,0.132122022785080,0;
27             0.132122022785080,0.377073505855695,0;
28             0,0,0.139172263121906];
29  vhat = [0.609;0.609];
30  Color = [5, 63, 255; 255, 150, 0]./255;
31  MatInd{1} = linspace(1,NMat,NMat);

```

APPENDIX E
POLYMAT AND POLYANISOMAT LIBRARY OF EXAMPLES

The MATLAB files needed to run each of the problems available with download of `PolyMat` are summarized in Table E.1. The files that need to be called from `PolyScript` to run each problem are also indicated. Note that each of the constraint files provided with download of `PolyMat` can be modified for use with `PolyAnisoMat` by changing `Mat` to be a cell array of constitutive matrices and by including an array of solid fractions, `vhat`. One example, `Mbb2AnisoConstraints`, is provided in section D.7 of Appendix D.

Table E.1: Examples provided with PolyMat

Domain	PolyMesher domain file	PolyMat constraint file	Description
	@ShortColumnDomain	ShortColumnConstraints	<ul style="list-style-type: none"> • Defaults: $H = 0.5, L = 1, P = 1$ • 2 global volume constraints, each controlling 1 of 2 candidate materials
	@MbbDomain	Case 1: MbbConstraints Case 2: MbbAnisoConstraints	<ul style="list-style-type: none"> • Defaults: $H = 1, L = 3, P = 1$ • Case 1: 1 global volume constraint controlling 1 candidate material • Case 2: 1 global volume constraint controlling 2 anisotropic candidate materials (only available with PolyAnisoMat)¹

¹ The result in subsection 5.4.4 considers a modified version of the MbbDomain file in which the load is distributed over a small horizontal length to prevent oscillations in the values of the design variables near the load.

Table E.1: (continued)

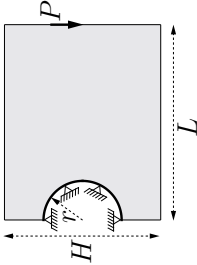
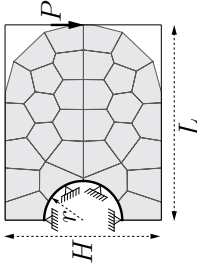
Domain	PolyMesher domain file	PolyMat constraint file	Description
 <p style="text-align: center;">@MichellDomain</p>	Case 1: Michell1Constraints Case 2: Michell2Constraints	<ul style="list-style-type: none"> • Defaults: $H = 4, L = 5, r = 1, P = 1$ • Case 1: 1 global volume constraint controlling 15 candidate materials • Case 2: 15 global volume constraints, each controlling 1 of 15 candidate materials 	
 <p style="text-align: center;">@MichellDomain</p>	Michell3Constraints	<ul style="list-style-type: none"> • Defaults: $H = 4, L = 5, r = 1, P = 1$ • Case 3: 30 local volume constraints, each controlling a sub-region of the domain and 1 of 15 candidate materials 	

Table E.1: (continued)

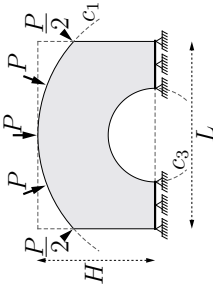
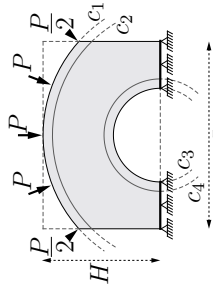
Domain	PolyMesher domain file	PolyMat constraint file	Description
	@CurvedBeamDomain	CurvedBeamConstraints	<ul style="list-style-type: none"> • Defaults: $H = 1.25$, $L = 2$, $c_1 = (x_{c1}, y_{c1}, r_{c1}) = (0, 0, 1.5)$, $c_3 = (0, 0.25, 0.5)$, $P = 1$ • 5 volume constraints, each controlling 1 of 5 candidate materials
	@CurvedBeamDomain	CurvedBeamPassive-Constraints	<ul style="list-style-type: none"> • Defaults: $H = 1.25$, $L = 2$, $c_1 = (x_{c1}, y_{c1}, r_{c1}) = (0, 0, 1.5)$, $c_2 = (0, 0, 1.4)$, $c_3 = (0, 0.25, 0.5)$, $c_4 = (0, 0.25, 0.6)$, $P = 1$ • 2 passive regions • 3 volume constraints, each controlling 1 of 3 candidate materials in the optimizable region

Table E.1: (continued)

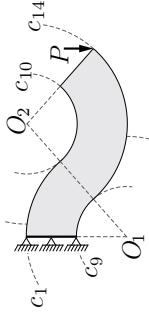
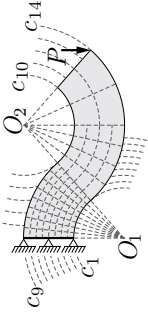
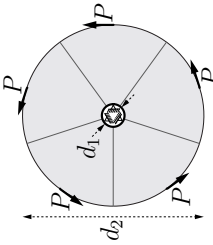
Domain	PolyMesher domain file	PolyMat constraint file	Description
	@SerpentineDomain	Serpentine1Constraints	<ul style="list-style-type: none"> • Defaults: $O_1 = (0, -2.6458)$, $O_2 = (9, 5.2915)$, $r_{c_1} = 8$, $r_{c_9} = 4$, $r_{c_{10}} = 4$, $r_{c_{14}} = 8$, $P = 1$ • 1 global volume constraint controlling 1 candidate material
	@SerpentineDomain	Serpentine84Constraints	<ul style="list-style-type: none"> • Defaults: $O_1 = (0, -2.6458)$, $O_2 = (9, 5.2915)$, $r_{c_1} = 8$, $r_{c_9} = 4$, $r_{c_{10}} = 4$, $r_{c_{14}} = 8$, $P = 1$ • 84 local volume constraints, each controlling 1 of 1 candidate material in each of 84 sub-regions

Table E.1: (continued)

Domain	PolyMesher domain file	PolyMat constraint file	Description
 <p style="text-align: center;">@FlowerDomain</p>	Case 1: Flower1MatConstraints	Case 1: Flower1MatConstraints	<ul style="list-style-type: none"> • Defaults: $d_1 = 0.25$, $d_2 = 2$, $P = 1$ • Case 1: 5 local volume constraints, each controlling 1 of 5 candidate materials in each of 5 sub-regions
	Case 2: Flower5MatConstraints	Case 2: Flower5MatConstraints	<ul style="list-style-type: none"> • Case 2: 5 local volume constraints, each controlling the same candidate material in 5 sub-regions

APPENDIX F

DIRECTIONAL TENSILE AND SHEAR MODULI

This appendix describes how directional tensile and shear moduli are computed for anisotropic materials and how to construct the directional tensile and shear moduli plots used in chapter 4 and chapter 6.

Hooke's Law is stated as $\sigma_{ij} = C_{ijkl}\varepsilon_{kl}$ or $\varepsilon_{ij} = Z_{ijkl}\sigma_{kl}$, $i, j, k, \ell = 1, 2, 3$, with Cauchy stress, σ_{ij} , linearized strain, ε_{kl} , homogenized stiffness elasticity tensor, C_{ijkl} , and homogenized compliance elasticity tensor, $Z_{ijkl} = C_{ijkl}^{-1}$. To gain insight about the stiffness of the microstructural-materials, two mechanical constants are defined as follows: 1) the tensile modulus is obtained by imposing $\sigma_{11} \neq 0, \sigma_{ij} = 0, \forall ij \neq 11$ and using Hooke's Law to compute $E_{11} = \sigma_{11}/\varepsilon_{11} = 1/Z_{1111}$; and 2) the shear modulus is obtained by imposing $\sigma_{12} \neq 0, \sigma_{ij} = 0, \forall ij \neq 12, ji \neq 21$ and using Hooke's Law to compute $G_{12} = \sigma_{12}/(2\varepsilon_{12}) = 1/(2Z_{1212})$. These mechanical constants indicate the tensile stiffness of the microstructural-material in the x_1 direction and the shear stiffness of the microstructural-material in the $x_1 - x_2$ plane, respectively, with respect to a given reference (unprime) frame.

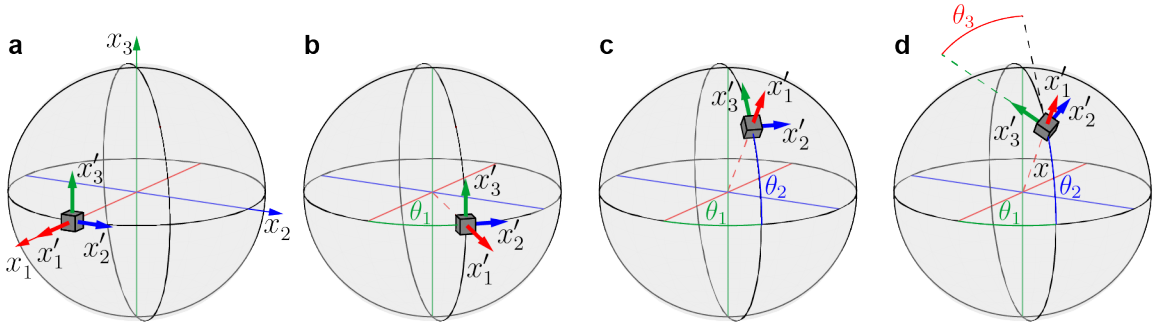


Figure F.1: Frame rotation conventions: (a) Rotated (prime) frame initially aligned with reference (unprime) frame; (b) rotation about the x'_3 axis by θ_1 ; (c) rotation about the x'_2 axis by θ_2 ; (d) rotation about the x'_1 axis by θ_3 .

To understand how these constants vary for different loading directions, tensor transformation laws are used to obtain the compliance elasticity tensor, Z'_{ijkl} , in a rotated (prime) frame. Let A_{ij} be a direction cosine matrix that transforms vectors from the reference to the rotated frame via a general rotation, i.e., $x'_i = A_{ij}x_j$. To construct A_{ij} , consider that

the rotated frame is originally oriented with the reference frame (Figure F.1a), and orient it by first rotating about the x'_3 axis by θ_1 , then about the x'_2 axis by θ_2 , and finally about the x'_1 axis by θ_3 as illustrated in Figure F.1b-d. These rotations can be expressed in terms of proper orthogonal matrices,

$$R_1(\theta_3) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c_{\theta_3} & -s_{\theta_3} \\ 0 & s_{\theta_3} & c_{\theta_3} \end{bmatrix}, \quad R_2(\theta_2) = \begin{bmatrix} c_{\theta_2} & 0 & s_{\theta_2} \\ 0 & 1 & 0 \\ -s_{\theta_2} & 0 & c_{\theta_2} \end{bmatrix}, \quad R_3(\theta_1) = \begin{bmatrix} c_{\theta_1} & -s_{\theta_1} & 0 \\ s_{\theta_1} & c_{\theta_1} & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad (\text{F.1})$$

where $s_{(\cdot)}$ and $c_{(\cdot)}$ denote the sine and cosine of angle (\cdot) , respectively.

Let x be the position vector of the x'_1 axis in the reference frame after rotating (see Figure F.1d). Then the coordinate transformation can be expressed as $x = R_3(\theta_1) R_2^T(\theta_2) R_1(\theta_3) x'_1$, where the transpose of $R_2(\theta_2)$ is considered since θ_2 is defined as a negative rotation (see Figure F.1c). This coordinate transformation is inverted to find

$$A_{ij} = R_1^T(\theta_3) R_2(\theta_2) R_3^T(\theta_1) = \begin{bmatrix} c_{\theta_2} c_{\theta_1} & c_{\theta_2} s_{\theta_1} & s_{\theta_2} \\ -s_{\theta_3} s_{\theta_2} c_{\theta_1} - c_{\theta_3} s_{\theta_1} & -s_{\theta_1} s_{\theta_3} s_{\theta_2} + c_{\theta_3} c_{\theta_1} & s_{\theta_3} c_{\theta_2} \\ -c_{\theta_1} c_{\theta_3} s_{\theta_2} + s_{\theta_1} s_{\theta_3} & -s_{\theta_1} c_{\theta_3} s_{\theta_2} - s_{\theta_3} c_{\theta_1} & c_{\theta_3} c_{\theta_2} \end{bmatrix}. \quad (\text{F.2})$$

Recall that if vectors transform as $x'_i = A_{ij} x_j$, then second and fourth-order tensors transform as $\varepsilon'_{ij} = A_{ik} A_{jl} \varepsilon_{kl}$ and $Z'_{ijkl} = A_{im} A_{jn} A_{ko} A_{lp} Z_{mnop}$, respectively (similar expressions hold for transforming stress, σ_{ij} , and the stiffness elasticity tensor, C_{ijkl}). For convenience of computation, the transformations are converted to matrix notation by defining

$$\{\sigma\} = \begin{bmatrix} \sigma_1 = \sigma_{11} \\ \sigma_2 = \sigma_{22} \\ \sigma_3 = \sigma_{33} \\ \sigma_4 = \sigma_{23} \\ \sigma_5 = \sigma_{31} \\ \sigma_6 = \sigma_{12} \end{bmatrix}, \quad \{\varepsilon\} = \begin{bmatrix} \varepsilon_1 = \varepsilon_{11} \\ \varepsilon_2 = \varepsilon_{22} \\ \varepsilon_3 = \varepsilon_{33} \\ \varepsilon_4 = 2\varepsilon_{23} \\ \varepsilon_5 = 2\varepsilon_{31} \\ \varepsilon_6 = 2\varepsilon_{12} \end{bmatrix}, \quad (\text{F.3})$$

$$[D] = \left[\begin{array}{c|c} C_{ppqq} & C_{pprs} \\ \hline C_{pprs} & C_{pqrs} \end{array} \right], \text{ and } [S] = \left[\begin{array}{c|c} Z_{ppqq} & 2Z_{pprs} \\ \hline 2Z_{pprs} & 4Z_{pqrs} \end{array} \right]. \quad (\text{F.4})$$

From the stress and strain transformation laws in tensor notation, matrices, $[M]$ and $[N]$, which perform the transformations in matrix notation can be constructed, where

$$[M] = \left[\begin{array}{ccc|ccc} A_{11}^2 & A_{12}^2 & A_{13}^2 & 2A_{12}A_{13} & 2A_{13}A_{11} & 2A_{11}A_{12} \\ A_{21}^2 & A_{22}^2 & A_{23}^2 & 2A_{22}A_{23} & 2A_{23}A_{21} & 2A_{21}A_{22} \\ A_{31}^2 & A_{32}^2 & A_{33}^2 & 2A_{32}A_{33} & 2A_{33}A_{31} & 2A_{31}A_{32} \\ \hline A_{21}A_{31} & A_{22}A_{32} & A_{23}A_{33} & A_{22}A_{33}+A_{23}A_{32} & A_{21}A_{33}+A_{23}A_{31} & A_{22}A_{31}+A_{21}A_{32} \\ A_{31}A_{11} & A_{32}A_{12} & A_{33}A_{13} & A_{12}A_{33}+A_{13}A_{32} & A_{13}A_{31}+A_{11}A_{33} & A_{11}A_{32}+A_{12}A_{31} \\ A_{11}A_{21} & A_{12}A_{22} & A_{13}A_{23} & A_{12}A_{23}+A_{13}A_{22} & A_{13}A_{21}+A_{11}A_{23} & A_{11}A_{22}+A_{12}A_{21} \end{array} \right] \quad (\text{F.5})$$

and

$$[N] = \left[\begin{array}{ccc|ccc} A_{11}^2 & A_{12}^2 & A_{13}^2 & A_{12}A_{13} & A_{13}A_{11} & A_{11}A_{12} \\ A_{21}^2 & A_{22}^2 & A_{23}^2 & A_{22}A_{23} & A_{23}A_{21} & A_{21}A_{22} \\ A_{31}^2 & A_{32}^2 & A_{33}^2 & A_{32}A_{33} & A_{33}A_{31} & A_{31}A_{32} \\ \hline 2A_{21}A_{31} & 2A_{22}A_{32} & 2A_{23}A_{33} & A_{22}A_{33}+A_{23}A_{32} & A_{21}A_{33}+A_{23}A_{31} & A_{22}A_{31}+A_{21}A_{32} \\ 2A_{31}A_{11} & 2A_{32}A_{12} & 2A_{33}A_{13} & A_{12}A_{33}+A_{13}A_{32} & A_{13}A_{31}+A_{11}A_{33} & A_{11}A_{32}+A_{12}A_{31} \\ 2A_{11}A_{21} & 2A_{12}A_{22} & 2A_{13}A_{23} & A_{12}A_{23}+A_{13}A_{22} & A_{13}A_{21}+A_{11}A_{23} & A_{11}A_{22}+A_{12}A_{21} \end{array} \right]. \quad (\text{F.6})$$

Then, Hooke's Law, $\{\varepsilon\} = [S]\{\sigma\}$, and the stress transformation law, $\{\sigma\} = [M]^{-1}\{\sigma\}'$, can be substituted into the strain transformation law, $\{\varepsilon\}' = [N]\{\varepsilon\}$, to find

$$[S]' = [N][S][M]^{-1} = [N][S][N]^T, \quad (\text{F.7})$$

where the last expression comes from the fact that $A_{ij} = A_{ij}^T$ since it is a product of proper orthogonal matrices. Now the tensile and shear moduli, $E'_{11} = 1/S'_{11}$ and $G'_{12} = 1/S'_{66}$, can be computed for any arbitrary orientation of the rotated coordinate frame by θ_1 , θ_2 , and θ_3 . The above derivations are taken from Auld, 1973 [252] and Turley and Sines, 1971 [253].

To visualize the directional tensile and shear moduli of a given microstructural-material,

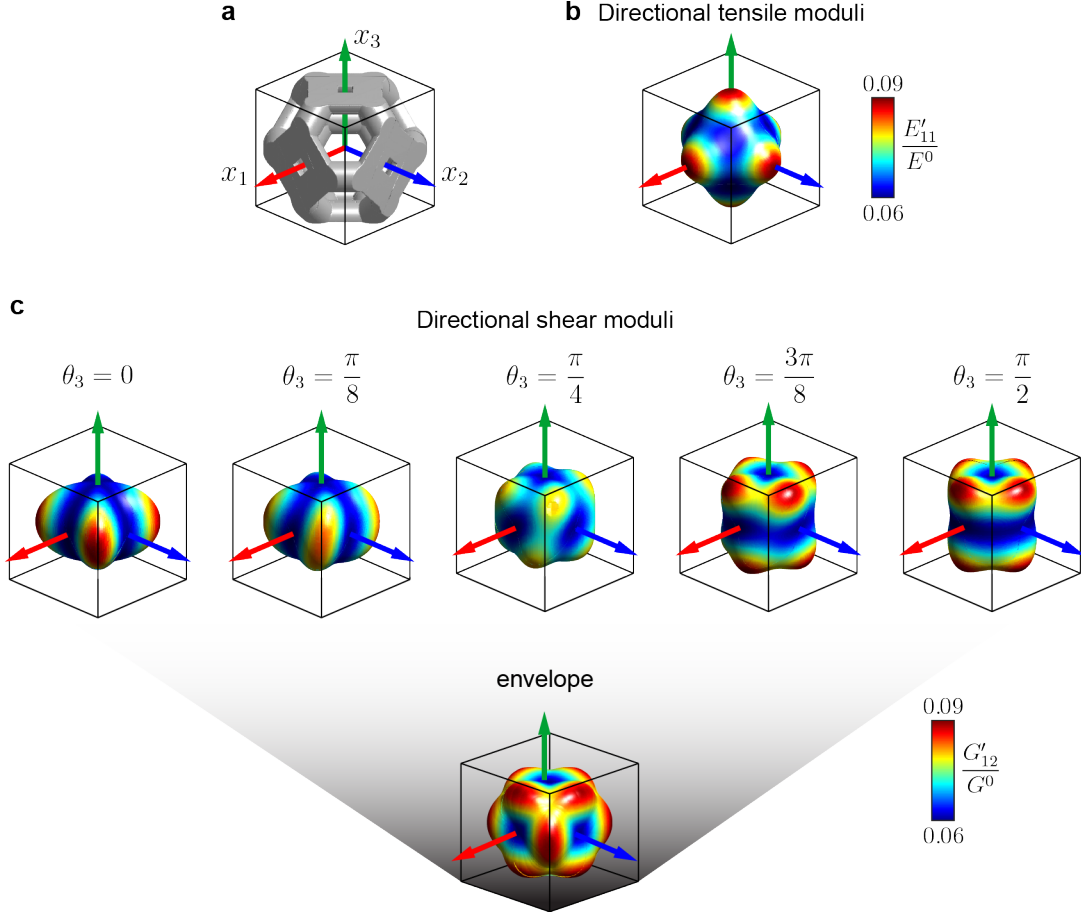


Figure F.2: Directional tensile and shear moduli plots: (a) Truncated octahedron unit cell geometry and corresponding (b) directional tensile moduli; (c) directional shear moduli for various θ_3 rotations of the rotated frame; (d) enveloped directional shear moduli for $\theta_3 = 0, \pi/16, \dots, \pi$.

a 3D surface plot is generated where, for all possible rotations of the prime coordinate frame, a point is plotted along the x vector, which locates the rotated frame with respect to the reference frame, with radial coordinate equal to E'_{11} or G'_{12} . It is noted that there is only one way to represent the directional tensile modulus (with the x'_1 axis oriented in the radial direction of the reference frame), but there are infinite ways to represent the directional shear modulus depending on the orientation of the x'_2 and x'_3 axes about the x'_1 axis. An illustrative example is provided in Figure F.2 for a periodic material composed of a truncated octahedron unit cell. Unless otherwise noted, the shear modulus plots reported in the main text represent an envelope of critical orientations of shear ($\theta_3 = 0$ and $\theta_3 = \pi/2$ for the cubic materials considered here).

APPENDIX G

MULTI-LATTICE EMBEDDING FLOWCHART

A flowchart describing the proposed process for manufacturing the microstructure-embedded topology optimized parts is provided in Figure G.1. The flowchart highlights the theoretical and physical relevance of the multi-lattice topology optimization and manufacturing approach, which bridges the entire process from design to manufacturing.

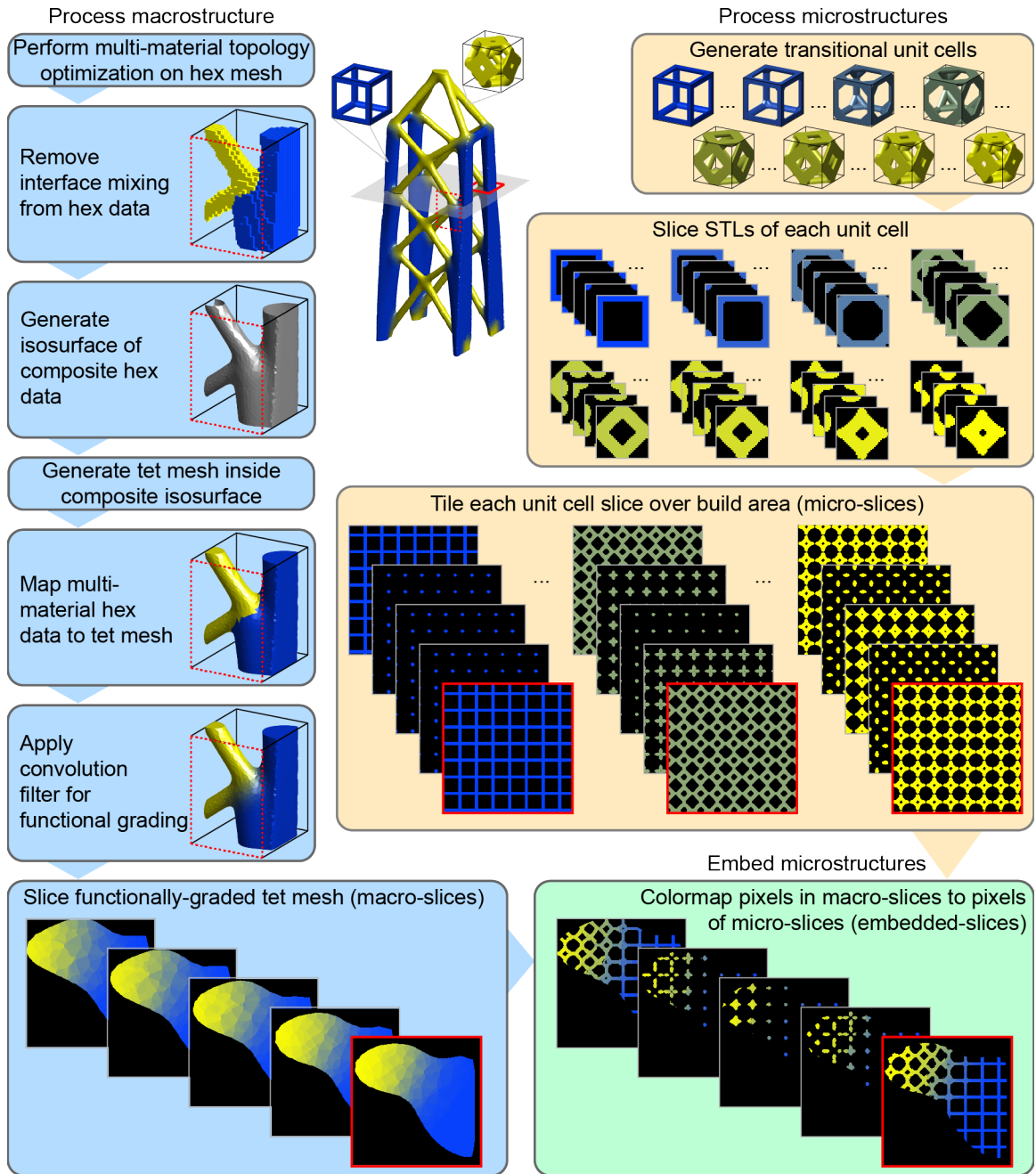


Figure G.1: Flowchart summarizing the overall process from design to manufacturing: On the left side, multi-material topology optimization is performed, the multi-material density data is processed, and the macrostructure is sliced. On the right side, the microstructures associated with the candidate microstructural materials used in topology optimization are sliced and tiled. Finally, the micro-slices are embedded into the macro-slices and sent to the 3D printer.

REFERENCES

- [1] D. Plowden. *Bridges: The Spans of North America*. W.W. Norton & Company, 1974.
- [2] J. C. Williams and E. A. Starke Jr. Progress in structural materials for aerospace systems. *Acta Materialia*, 51(19):5775–5799, 2003.
- [3] A. Taub, E. De Moor, A. Luo, D. K. Matlock, J. G. Speer, and U. Vaidya. Materials for automotive lightweighting. *Annual Review of Materials Research*, 49:327–359, 2019.
- [4] P.-Y. Chen, J. McKittrick, and M. A. Meyers. Biological materials: functional adaptations and bioinspired designs. *Progress in Materials Science*, 57(8):1492–1704, 2012.
- [5] M. P. Bendsøe and O. Sigmund. *Topology optimization: Theory, Methods, and Applications*. Springer, 2003.
- [6] E. D. Sanders, A. S. Ramos, and G. H. Paulino. Topology optimization of tension-only cable nets under finite deformations. *Structural and Multidisciplinary Optimization*, 62(2):559–579, 2020.
- [7] E. D. Sanders, M. A. Aguiló, and G. H. Paulino. Optimized lattice-based metamaterials for elastostatic cloaking. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, accepted, 2021.
- [8] E. D. Sanders, M. A. Aguiló, and G. H. Paulino. Multi-material continuum topology optimization with arbitrary volume and mass constraints. *Computer Methods in Applied Mechanics and Engineering*, 340:798–823, 2018.
- [9] E. D. Sanders, A. Pereira, M. A. Aguiló, and G. H. Paulino. PolyMat: an efficient Matlab code for multi-material topology optimization. *Structural and Multidisciplinary Optimization*, 58(6):2727–2759, 2018.
- [10] E. D. Sanders, A. Pereira, and G. H. Paulino. Optimal and continuous multilattice embedding. *Science Advances*, 7(16):eabf4838, 2021.
- [11] Y. Jiang, A. S. Ramos Jr., and G. H. Paulino. Topology optimization with design-dependent loading: An adaptive sensitivity-separation design variable update scheme. *Structural and Multidisciplinary Optimization*, in press, 2021.
- [12] P. W. Christensen and A. Klarbring. *An Introduction to Structural Optimization*, volume 153. Springer Science & Business Media, 2008.
- [13] M. Zhou and G. Rozvany. The COC algorithm, Part II: Topological, geometrical and generalized shape optimization. *Computer Methods in Applied Mechanics and Engineering*, 89(1-3):309–336, 1991.
- [14] M. P. Bendsøe. Optimal shape design as a material distribution problem. *Structural Optimization*, 1(4):193–202, 1989.

- [15] T. Zegard and G. H. Paulino. GRAND – Ground structure based topology optimization for arbitrary 2D domains using MATLAB. *Structural and Multidisciplinary Optimization*, 50(5):861–882, 2014.
- [16] C. Talischi, G. H. Paulino, A. Pereira, and I. F. M. Menezes. PolyMesher: a general-purpose mesh generator for polygonal elements written in Matlab. *Structural and Multidisciplinary Optimization*, 45(3):309–328, 2012.
- [17] C. Talischi, G. H. Paulino, A. Pereira, and I. F. M. Menezes. PolyTop: a Matlab implementation of a general topology optimization framework using unstructured polygonal finite element meshes. *Structural and Multidisciplinary Optimization*, 45(3):329–357, 2012.
- [18] D. P. Bertsekas. *Nonlinear Programming*. Athena Scientific, 1999.
- [19] M. P. Bendsøe and N. Kikuchi. Generating optimal topologies in structural design using a homogenization method. *Computer Methods in Applied Mechanics and Engineering*, 71(2):197–224, 1988.
- [20] J.-P. Kruth, M.-C. Leu, and T. Nakagawa. Progress in additive manufacturing and rapid prototyping. *Cirp Annals*, 47(2):525–540, 1998.
- [21] T. Zegard and G. H. Paulino. Bridging topology optimization and additive manufacturing. *Structural and Multidisciplinary Optimization*, 53(1):175–192, 2016.
- [22] L. Meng, W. Zhang, D. Quan, G. Shi, L. Tang, Y. Hou, P. Breitkopf, J. Zhu, and T. Gao. From topology optimization design to additive manufacturing: Today’s success and tomorrow’s roadmap. *Archives of Computational Methods in Engineering*, pages 1–26, 2019.
- [23] J. Liu, A. T. Gaynor, S. Chen, Z. Kang, K. Suresh, A. Takezawa, L. Li, J. Kato, J. Tang, C. C. Wang, L. Cheng, X. Liang, and A. C. To. Current and future trends in topology optimization for additive manufacturing. *Structural and Multidisciplinary Optimization*, 57(6):2457–2483, 2018.
- [24] A. T. Gaynor and J. K. Guest. Topology optimization considering overhang constraints: Eliminating sacrificial support material in additive manufacturing through design. *Structural and Multidisciplinary Optimization*, 54(5):1157–1172, 2016.
- [25] O. Sigmund and S. Torquato. Design of materials with extreme thermal expansion using a three-phase topology optimization method. *Journal of the Mechanics and Physics of Solids*, 45(6):1037–1067, 1997.
- [26] L. Yin and G. Ananthasuresh. Topology optimization of compliant mechanisms with multiple materials using a peak function material interpolation scheme. *Structural and Multidisciplinary Optimization*, 23(1):49–62, 2001.
- [27] M. Y. Wang and X. Wang. “Color” level sets: a multi-phase method for structural topology optimization with multiple materials. *Computer Methods in Applied Mechanics and Engineering*, 193(6-8):469–496, 2004.

- [28] J. Stegmann and E. Lund. Discrete material optimization of general composite shell structures. *International Journal for Numerical Methods in Engineering*, 62(14):2009–2027, 2005.
- [29] R. Tavakoli and S. M. Mohseni. Alternating active-phase algorithm for multimaterial topology optimization problems: a 115-line MATLAB implementation. *Structural and Multidisciplinary Optimization*, 49(4):621–642, 2014.
- [30] X. S. Zhang, G. H. Paulino, and A. S. Ramos. Multi-material topology optimization with multiple volume constraints: a general approach applied to ground structures with material nonlinearity. *Structural and Multidisciplinary Optimization*, 57(1):161–182, 2018.
- [31] C. Zhou, Y. Chen, Z. Yang, and B. Khoshnevis. Digital material fabrication using mask-image-projection-based stereolithography. *Rapid Prototyping Journal*, 2013.
- [32] K. Kowsari, S. Akbari, D. Wang, N. X. Fang, and Q. Ge. High-efficiency high-resolution multimaterial fabrication for digital light processing-based three-dimensional printing. *3D Printing and Additive Manufacturing*, 5(3):185–193, 2018.
- [33] G. I. Peterson, J. J. Schwartz, D. Zhang, B. M. Weiss, M. A. Ganter, D. W. Storti, and A. J. Boydston. Production of materials with spatially-controlled cross-link density via vat photopolymerization. *ACS Applied Materials & Interfaces*, 8(42):29037–29043, 2016.
- [34] X. Kuang, J. Wu, K. Chen, Z. Zhao, Z. Ding, F. Hu, D. Fang, and H. J. Qi. Grayscale digital light processing 3d printing for highly functionally graded materials. *Science Advances*, 5(5):eaav5790, 2019.
- [35] F. Otto and R. Trostel. *Tensile Structures: Design, Structure, and Calculation of Buildings, Cables, Nets, and Membranes*, volume 1. The M.I.T. Press, 1967.
- [36] F. Otto and F.-K. Schleyer. *Tensile Structures: Design, Structure, and Calculation of Buildings, Cables, Nets, and Membranes*, volume 2. The M.I.T. Press, 1969.
- [37] L. Glaeser. *The Work of Frei Otto*. The Museum of Modern Art, New York, 1972.
- [38] F. Otto and B. Rasch. *Finding Form: Towards an Architecture of the Minimal*. Axel Menges, Stuttgart, 1995.
- [39] W. Nerdinger, editor. *Frei Otto Complete Works: Lightweight Construction, Natural Design*. Switzerland: Birkhauser, 2005.
- [40] F. Magalhães, E. Caetano, Á. Cunha, O. Flamand, and G. Grillaud. Ambient and free vibration tests of the Millau Viaduct: Evaluation of alternative processing strategies. *Engineering Structures*, 45:372–384, 2012.
- [41] D. V. Herlihy. *Bicycle: The History*. Yale University Press, 2004.
- [42] L. Eisoldt, A. Smith, and T. Scheibel. Decoding the secrets of spider silk. *Materials Today*, 14(3):80–86, 2011.
- [43] K. Madsen and H. B. Nielsen. *Introduction to Optimization and Data Fitting*. Informatics and Mathematical Modelling, Technical University of Denmark, 2010.

- [44] M. M. Neves, H. Rodrigues, and J. M. Guedes. Generalized topology design of structures with a buckling load criterion. *Structural and Multidisciplinary Optimization*, 10(2):71–78, 1995.
- [45] C. Jog. Distributed-parameter optimization and topology design for non-linear thermoelasticity. *Computer Methods in Applied Mechanics and Engineering*, 132(1-2):117–134, 1996.
- [46] T. Buhl, C. B. W. Pedersen, and O. Sigmund. Stiffness design of geometrically nonlinear structures using topology optimization. *Structural and Multidisciplinary Optimization*, 19(2):93–104, 2000.
- [47] T. Sekimoto and H. Noguchi. Homologous topology optimization in large displacement and buckling problems. *JSME International Journal Series A Solid Mechanics and Material Engineering*, 44(4):616–622, 2001.
- [48] H. C. Gea and J. Luo. Topology optimization of structures with geometrical nonlinearities. *Computers & Structures*, 79(20):1977–1985, 2001.
- [49] D. Jung and H. C. Gea. Topology optimization of nonlinear structures. *Finite Elements in Analysis and Design*, 40(11):1417–1427, 2004.
- [50] G. H. Yoon and Y. Y. Kim. Element connectivity parameterization for topology optimization of geometrically nonlinear structures. *International Journal of Solids and Structures*, 42(7):1983–2009, 2005.
- [51] R. Kemmler, A. Lipka, and E. Ramm. Large deformations and stability in topology optimization. *Structural and Multidisciplinary Optimization*, 30(6):459–476, 2005.
- [52] A. Kawamoto. Stabilization of geometrically nonlinear topology optimization by the Levenberg–Marquardt method. *Structural and Multidisciplinary Optimization*, 37(4):429–433, 2009.
- [53] A. Klarbring and N. Strömberg. Topology optimization of hyperelastic bodies including non-zero prescribed displacements. *Structural and Multidisciplinary Optimization*, 47(1):37–48, 2013.
- [54] F. Wang, B. S. Lazarov, O. Sigmund, and J. S. Jensen. Interpolation scheme for fictitious domain techniques and topology optimization of finite strain elastic problems. *Computer Methods in Applied Mechanics and Engineering*, 276:453–472, 2014.
- [55] F. A. Gomes and T. A. Senne. An algorithm for the topology optimization of geometrically nonlinear structures. *International Journal for Numerical Methods in Engineering*, 99(6):391–409, 2014.
- [56] N. P. van Dijk, M. Langelaar, and F. van Keulen. Element deformation scaling for robust geometrically nonlinear analyses in topology optimization. *Structural and Multidisciplinary Optimization*, 50(4):537–560, 2014.
- [57] Y. Luo, M. Y. Wang, and Z. Kang. Topology optimization of geometrically nonlinear structures based on an additive hyperelasticity technique. *Computer Methods in Applied Mechanics and Engineering*, 286:422–441, 2015.

- [58] Q. Luo and L. Tong. An algorithm for eradicating the effects of void elements on structural topology optimization for nonlinear compliance. *Structural and Multidisciplinary Optimization*, 53(4):695–714, 2016.
- [59] X. S. Zhang, H. Chi, and G. H. Paulino. Adaptive multi-material topology optimization with hyperelastic materials under large deformations: A virtual element approach. *Computer Methods in Applied Mechanics and Engineering*, 370:112976, 2020.
- [60] W. Li, F. Wang, O. Sigmund, and X. S. Zhang. Design of composite structures with programmable elastic responses under finite deformations. *Journal of the Mechanics and Physics of Solids*, 151:104356, 2021.
- [61] A. S. Ramos Jr. and G. H. Paulino. Convex topology optimization for hyperelastic trusses based on the ground-structure approach. *Structural Multidisciplinary Optimization*, 51(2):287–304, 2015.
- [62] Y. Kanno and M. Ohsaki. Minimum principle of complementary energy of cable networks by using second-order cone programming. *International Journal of Solids and Structures*, 40(17):4437–4460, 2003.
- [63] X. Zhang, A. S. Ramos Jr., and G. H. Paulino. Material nonlinear topology optimization using the ground structure method with a discrete filtering scheme. *Structural and Multidisciplinary Optimization*, 55(6):2045–2072, 2017.
- [64] S. E. Leon, G. H. Paulino, A. Pereira, I. F. Menezes, and E. N. Lages. A unified library of nonlinear solution schemes. *Applied Mechanics Reviews*, 64(4):040803, 2011.
- [65] J. Bonet and R. D. Wood. *Nonlinear Continuum Mechanics for Finite Element Analysis*. Cambridge University Press, 2nd edition, 2008.
- [66] A. Klarbring and N. Strömberg. A note on the min-max formulation of stiffness optimization including non-zero prescribed displacements. *Structural and Multidisciplinary Optimization*, 45(1):147–149, 2012.
- [67] E. Kreindler and A. Jameson. Conditions for nonnegativeness of partitioned matrices. *IEEE Transactions on Automatic Control*, 17(1):147–148, 1972.
- [68] G. Strang. *Linear Algebra and Its Applications*. Thomson, Brooks/Cole, 2006.
- [69] A. S. Ramos Jr and G. H. Paulino. Filtering structures out of ground structures – a discrete filtering tool for structural design optimization. *Structural and Multidisciplinary Optimization*, 54(1):95–116, 2016.
- [70] E. D. Sanders, A. S. Ramos Jr, and G. H. Paulino. A maximum filter for the ground structure method: An optimization tool to harness multiple structural designs. *Engineering Structures*, 151:235–252, 2017.
- [71] U. M. Ascher and C. Greif. *A First Course on Numerical Methods*. SIAM, 2011.
- [72] P. Wriggers. *Nonlinear Finite Element Methods*. Springer Science & Business Media, 2008.
- [73] S. Wright and J. Nocedal. *Numerical Optimization*. Springer-Verlag, 1999.

- [74] N. Olhoff and J. Taylor. On structural optimization. *Journal of Applied Mechanics*, 50:1139–1151, 1983.
- [75] A. A. Groenwold and L. Etman. On the equivalence of optimality criterion and sequential approximate optimization methods in the classical topology layout problem. *International Journal for Numerical Methods in Engineering*, 73(3):297–316, 2008.
- [76] G. M. Fadel, M. F. Riley, and J. M. Barthelemy. Two point exponential approximation method for structural optimization. *Structural and Multidisciplinary Optimization*, 2(2):117–124, 1990.
- [77] F. Vollrath and W. Mohren. Spiral geometry in the garden spider’s orb web. *Naturwissenschaften*, 72(12):666–667, 1985.
- [78] X. Guo, Z. Du, G. Cheng, and C. Ni. Symmetry properties in structural optimization: some extensions. *Structural and Multidisciplinary Optimization*, 47(6):783–794, 2013.
- [79] Z. Du and X. Guo. Symmetry analysis for structural optimization problems involving reliability measure and bi-modulus materials. *Structural and Multidisciplinary Optimization*, 53(5):973–984, 2016.
- [80] S. W. Cranford, A. Tarakanova, N. M. Pugno, and M. J. Buehler. Nonlinear material behaviour of spider silk yields robust webs. *Nature*, 482(7383):72, 2012.
- [81] W. D. Pilkey, D. F. Pilkey, and Z. Bi. *Peterson’s Stress Concentration Factors*. John Wiley & Sons, 2020.
- [82] J. B. Pendry, D. Schurig, and D. R. Smith. Controlling electromagnetic fields. *Science*, 312(5781):1780–1782, 2006.
- [83] U. Leonhardt. Optical conformal mapping. *Science*, 312(5781):1777–1780, 2006.
- [84] D. Schurig, J. Mock, B. Justice, S. A. Cummer, J. B. Pendry, A. Starr, and D. Smith. Metamaterial electromagnetic cloak at microwave frequencies. *Science*, 314(5801):977–980, 2006.
- [85] M. Farhat, S. Enoch, S. Guenneau, and A. Movchan. Broadband cylindrical acoustic cloak for linear surface waves in a fluid. *Physical Review Letters*, 101(13):134501, 2008.
- [86] S. Zhang, C. Xia, and N. Fang. Broadband acoustic cloak for ultrasound waves. *Physical Review Letters*, 106(2):024301, 2011.
- [87] S. Zhang, D. A. Genov, C. Sun, and X. Zhang. Cloaking of matter waves. *Physical Review Letters*, 100(12):123002, 2008.
- [88] A. Greenleaf, Y. Kurylev, M. Lassas, and G. Uhlmann. Approximate quantum cloaking and almost-trapped states. *Physical Review Letters*, 101(22):220404, 2008.
- [89] S. Guenneau, C. Amra, and D. Veynante. Transformation thermodynamics: cloaking and concentrating heat flux. *Optics Express*, 20(7):8207–8218, 2012.
- [90] R. Schittny, M. Kadic, S. Guenneau, and M. Wegener. Experiments on transformation thermodynamics: molding the flow of heat. *Physical Review Letters*, 110(19):195901, 2013.

- [91] S. Guenneau and T. M. Puvirajesinghe. Fick’s second law transformed: one path to cloaking in mass diffusion. *Journal of The Royal Society Interface*, 10(83):20130106, 2013.
- [92] G. W. Milton, M. Briane, and J. R. Willis. On cloaking for elasticity and physical equations with a transformation invariant form. *New Journal of Physics*, 8(10):248, 2006.
- [93] M. Brun, S. Guenneau, and A. B. Movchan. Achieving control of in-plane elastic waves. *Applied Physics Letters*, 94(6):061903, 2009.
- [94] N. Stenger, M. Wilhelm, and M. Wegener. Experiments on elastic cloaking in thin plates. *Physical Review Letters*, 108(1):014301, 2012.
- [95] D. J. Colquitt, M. Brun, M. Gei, A. B. Movchan, N. V. Movchan, and I. S. Jones. Transformation elastodynamics and cloaking for flexural waves. *Journal of the Mechanics and Physics of Solids*, 72:131–143, 2014.
- [96] D. Misseroni, D. J. Colquitt, A. B. Movchan, N. V. Movchan, and I. S. Jones. Cymatics for the cloaking of flexural vibrations in a structured plate. *Scientific Reports*, 6(1): 1–11, 2016.
- [97] T. Bückmann, M. Thiel, M. Kadic, R. Schittny, and M. Wegener. An elasto-mechanical unfeelability cloak made of pentamode metamaterials. *Nature Communications*, 5: 4130, 2014.
- [98] T. Bückmann, M. Kadic, R. Schittny, and M. Wegener. Mechanical cloak design by direct lattice transformation. *Proceedings of the National Academy of Sciences*, 112 (16):4930–4934, 2015.
- [99] V. D. Fachinotti, I. Peralta, and A. E. Albanesi. Optimization-based design of an elastostatic cloaking device. *Scientific Reports*, 8(1):1–8, 2018.
- [100] J. C. Á. Hostos, V. D. Fachinotti, and I. Peralta. Metamaterial for elastostatic cloaking under thermal gradients. *Scientific Reports*, 9(1):1–9, 2019.
- [101] W. McGuire, R. H. Gallagher, and R. D. Ziemian. *Matrix Structural Analysis*. John Wiley & Sons, Inc, 2000.
- [102] W. X. Jiang, T. J. Cui, G. X. Yu, X. Q. Lin, Q. Cheng, and J. Y. Chin. Arbitrarily elliptical–cylindrical invisible cloaking. *Journal of Physics D: Applied Physics*, 41(8): 085504, 2008.
- [103] M. Kadic, M. Wegener, A. Nicolet, F. Zolla, S. Guenneau, and A. Diatta. Elastodynamic behavior of mechanical cloaks designed by direct lattice transformations. *Wave Motion*, 92:102419, 2020.
- [104] A. Ben-Israel and T. N. Greville. *Generalized Inverses: Theory and Applications*, volume 15. Springer Science & Business Media, 2003.
- [105] Y. Bard. *Nonlinear Parameter Estimation*. Academic press, 1974.

- [106] K. Svanberg. On local and global minima in structural optimization. In E. Atrek, R. H. Gallagher, K. M. Ragsdell, and O. C. Zienkiewicz, editors, *New directions in Optimum Structural Design*. John Wiley & Sons, 1984.
- [107] U. G. Wegst, H. Bai, E. Saiz, A. P. Tomsia, and R. O. Ritchie. Bioinspired structural materials. *Nature Materials*, 14(1):23–36, 2015.
- [108] B. R. Lawn, J. J.-W. Lee, and H. Chai. Teeth: among nature’s most durable biocomposites. *Annual Review of Materials Research*, 40:55–75, 2010.
- [109] J. C. Weaver, G. W. Milliron, A. Miserez, K. Evans-Lutterodt, S. Herrera, I. Gallana, W. J. Mershon, B. Swanson, P. Zavattieri, E. DiMasi, and D. Kisailus. The stomatopod dactyl club: a formidable damage-tolerant biological hammer. *Science*, 336(6086):1275–1280, 2012.
- [110] U. G. Wegst. Bending efficiency through property gradients in bamboo, palm, and wood-based composites. *Journal of the Mechanical Behavior of Biomedical Materials*, 4(5):744–755, 2011.
- [111] J. Birchall and N. Thomas. On the architecture and function of cuttlefish bone. *Journal of Materials Science*, 18(7):2081–2086, 1983.
- [112] T. Yang, Z. Jia, H. Chen, Z. Deng, W. Liu, L. Chen, and L. Li. Mechanical design of the highly porous cuttlebone: A bioceramic hard buoyancy tank for cuttlefish. *Proceedings of the National Academy of Sciences*, 117(38):23450–23459, 2020.
- [113] L. J. Gibson and M. F. Ashby. *Cellular Solids: Structure and Properties*. Cambridge University Press, 1999.
- [114] J. Qi and J. W. Halloran. Negative thermal expansion artificial material from iron-nickel alloys by oxide co-extrusion with reductive sintering. *Journal of Materials Science*, 39(13):4113–4118, 2004.
- [115] C. Schumacher, B. Bickel, J. Rys, S. Marschner, C. Daraio, and M. Gross. Microstructures to control elasticity in 3D printing. *ACM Transactions on Graphics (TOG)*, 34(4):136, 2015.
- [116] J. Panetta, Q. Zhou, L. Malomo, N. Pietroni, P. Cignoni, and D. Zorin. Elastic textures for additive fabrication. *ACM Transactions on Graphics (TOG)*, 34(4):1–12, 2015.
- [117] B. Zhu, M. Skouras, D. Chen, and W. Matusik. Two-scale topology optimization with microstructures. *ACM Transactions on Graphics (TOG)*, 36(4):1, 2017.
- [118] O. Sigmund. Design of multiphysics actuators using topology optimization—Part II: Two-material structures. *Computer Methods in Applied Mechanics and Engineering*, 190(49-50):6605–6627, 2001.
- [119] S. L. Vatanabe, G. H. Paulino, and E. C. N. Silva. Influence of pattern gradation on the design of piezocomposite energy harvesting devices using topology optimization. *Composites Part B: Engineering*, 43(6):2646–2654, 2012.

- [120] S. L. Vatanabe, G. H. Paulino, and E. C. N. Silva. Design of functionally graded piezo-composites using topology optimization and homogenization—toward effective energy harvesting materials. *Computer Methods in Applied Mechanics and Engineering*, 266: 205–218, 2013.
- [121] S. L. Vatanabe, G. H. Paulino, and E. C. N. Silva. Maximizing phononic band gaps in piezocomposite materials by means of topology optimization. *The Journal of the Acoustical Society of America*, 136(2):494–501, 2014.
- [122] A. H. Taheri and B. Hassani. Simultaneous isogeometrical shape and material design of functionally graded structures for optimal eigenfrequencies. *Computer Methods in Applied Mechanics and Engineering*, 277:46–80, 2014.
- [123] A. H. Taheri and K. Suresh. An isogeometric approach to topology optimization of multi-material and functionally graded structures. *International Journal for Numerical Methods in Engineering*, 109(5):668–696, 2017.
- [124] T. Gao and W. Zhang. A mass constraint formulation for structural topology optimization with multiphase materials. *International Journal for Numerical Methods in Engineering*, 88(8):774–796, 2011.
- [125] A. T. Gaynor, N. A. Meisel, C. B. Williams, and J. K. Guest. Multiple-material topology optimization of compliant mechanisms created via polyjet three-dimensional printing. *Journal of Manufacturing Science and Engineering*, 136(6), 2014.
- [126] L. V. Gibiansky and O. Sigmund. Multiphase composites with extremal bulk modulus. *Journal of the Mechanics and Physics of Solids*, 48(3):461–498, 2000.
- [127] E. Lund and J. Stegmann. On structural optimization of composite shell structures using a discrete constitutive parametrization. *Wind Energy: An International Journal for Progress and Applications in Wind Power Conversion Technology*, 8(1):109–124, 2005.
- [128] C. F. Hvejsel, E. Lund, and M. Stolpe. Optimization strategies for discrete multi-material stiffness optimization. *Structural and Multidisciplinary Optimization*, 44(2): 149–163, 2011.
- [129] C. F. Hvejsel and E. Lund. Material interpolation schemes for unified topology and multi-material optimization. *Structural and Multidisciplinary Optimization*, 43(6): 811–825, 2011.
- [130] J. Park and A. Sutradhar. A multi-resolution method for 3d multi-material topology optimization. *Computer Methods in Applied Mechanics and Engineering*, 285:571–586, 2015.
- [131] Q. X. Lieu and J. Lee. A multi-resolution approach for multi-material topology optimization based on isogeometric analysis. *Computer Methods in Applied Mechanics and Engineering*, 2017.
- [132] Q. H. Doan and D. Lee. Optimum topology design of multi-material structures with non-spurious buckling constraints. *Advances in Engineering Software*, 2017.

- [133] K. N. Chau, K. N. Chau, T. Ngo, K. Hackl, and H. Nguyen-Xuan. A polytree-based adaptive polygonal finite element method for multi-material topology optimization. *Computer Methods in Applied Mechanics and Engineering*, 332:712–739, 2018.
- [134] M. Y. Wang, S. Chen, X. Wang, and Y. Mei. Design of multimaterial compliant mechanisms using level-set methods. *Journal of Mechanical Design*, 127(5):941–956, 2005.
- [135] X. Guo, W. Zhang, and W. Zhong. Stress-related topology optimization of continuum structures involving multi-phase materials. *Computer Methods in Applied Mechanics and Engineering*, 268:632–655, 2014.
- [136] M. Y. Wang and S. Zhou. Synthesis of shape and topology of multi-material structures with a phase-field method. *Journal of Computer-Aided Materials Design*, 11(2-3):117–138, 2004.
- [137] S. Zhou and M. Y. Wang. Multimaterial structural topology optimization with a generalized Cahn–Hilliard model of multiphase transition. *Structural and Multidisciplinary Optimization*, 33(2):89–111, 2007.
- [138] R. Tavakoli. Multimaterial topology optimization by volume constrained Allen–Cahn system and regularized projected steepest descent method. *Computer Methods in Applied Mechanics and Engineering*, 276:534–565, 2014.
- [139] M. Wallin, N. Ivarsson, and M. Ristinmaa. Large strain phase-field-based multi-material topology optimization. *International Journal for Numerical Methods in Engineering*, 104(9):887–904, 2015.
- [140] X. Huang and Y. M. Xie. Bi-directional evolutionary topology optimization of continuum structures with one or multiple materials. *Computational Mechanics*, 43(3):393–401, 2009.
- [141] B. Bourdin. Filters in topology optimization. *International Journal for Numerical Methods in Engineering*, 50(9):2143–2158, 2001.
- [142] T. Borrvall and J. Petersson. Topology optimization using regularized intermediate density control. *Computer Methods in Applied Mechanics and Engineering*, 190:4911–4928, 2001.
- [143] O. Giraldo-Londoño, L. Mirabella, L. Dalloro, and G. H. Paulino. Multi-material thermomechanical topology optimization with applications to additive manufacturing: Design of main composite part and its support structure. *Computer Methods in Applied Mechanics and Engineering*, 363:112812, 2020.
- [144] O. Giraldo-Londoño and G. H. Paulino. Fractional topology optimization of periodic multi-material viscoelastic microstructures with tailored energy dissipation. *Computer Methods in Applied Mechanics and Engineering*, 372:113307, 2020.
- [145] C. Talischi, G. H. Paulino, A. Pereira, and I. F. M. Menezes. Polygonal finite elements for topology optimization: A unifying paradigm. *International Journal for Numerical Methods in Engineering*, 82(6):671–698, 2010.

- [146] A. Bensoussan, J.-L. Lions, and G. Papanicolaou. *Asymptotic Analysis for Periodic Structures*, volume 374. American Mathematical Soc., 2011.
- [147] B. Hassani and E. Hinton. *Homogenization and Structural Topology Optimization: Theory, Practice and Software*. Springer, 1999.
- [148] G. Dong, Y. Tang, and Y. F. Zhao. A 149 line homogenization code for three-dimensional cellular materials written in MATLAB. *Journal of Engineering Materials and Technology*, 141(1), 2019.
- [149] A. Clausen, N. Aage, and O. Sigmund. Exploiting additive manufacturing infill in topology optimization for improved buckling load. *Engineering*, 2(2):250–257, 2016.
- [150] A. Pereira, C. Talischi, G. H. Paulino, I. F. Menezes, and M. S. Carvalho. Fluid flow topology optimization in polytop: stability and computational implementation. *Structural and Multidisciplinary Optimization*, 54(5):1345–1364, 2016.
- [151] O. Giraldo-Londoño and G. H. Paulino. PolyStress: a Matlab implementation for local stress-constrained topology optimization using the augmented lagrangian method. *Structural and Multidisciplinary Optimization*, pages 1–33, 2021.
- [152] A. G. Michell. The limits of economy of material in frame structures. *Philosophical Magazine*, 8(6):589–597, 1904.
- [153] J. Wu, O. Sigmund, and J. P. Groen. Topology optimization of multi-scale structures: a review. *Structural and Multidisciplinary Optimization*, pages 1–26, 2021.
- [154] A. Vyatskikh, S. Delalande, A. Kudo, X. Zhang, C. M. Portela, and J. R. Greer. Additive manufacturing of 3D nano-architected metals. *Nature Communications*, 9(1):1–8, 2018.
- [155] D. W. Yee, M. L. Lifson, B. W. Edwards, and J. R. Greer. Additive manufacturing of 3D-architected multifunctional metal oxides. *Advanced Materials*, 31(33):1901345, 2019.
- [156] X. Zheng, H. Lee, T. H. Weisgraber, M. Shusteff, J. DeOtte, E. B. Duoss, J. D. Kuntz, M. M. Biener, Q. Ge, J. A. Jackson, et al. Ultralight, ultrastiff mechanical metamaterials. *Science*, 344(6190):1373–1377, 2014.
- [157] X. Zheng, W. Smith, J. Jackson, B. Moran, H. Cui, D. Chen, J. Ye, N. Fang, N. Rodriguez, T. Weisgraber, et al. Multiscale metallic metamaterials. *Nature Materials*, 15(10):1100–1106, 2016.
- [158] J. Martínez, J. Dumas, and S. Lefebvre. Procedural voronoi foams for additive manufacturing. *ACM Transactions on Graphics (TOG)*, 35(4):1–12, 2016.
- [159] A. O. Aremu, J. Brennan-Craddock, A. Panesar, I. Ashcroft, R. J. Hague, R. D. Wildman, and C. Tuck. A voxel-based method of constructing and skinning conformal and functionally graded lattice structures suitable for additive manufacturing. *Additive Manufacturing*, 13:1–13, 2017.
- [160] C. Bader, D. Kolb, J. C. Weaver, S. Sharma, A. Hosny, J. Costa, and N. Oxman. Making data matter: Voxel printing for the digital fabrication of data across scales and domains. *Science advances*, 4(5):eaas8652, 2018.

- [161] L. Jiang, H. Ye, C. Zhou, and S. Chen. Parametric topology optimization toward rational design and efficient prefabrication for additive manufacturing. *Journal of Manufacturing Science and Engineering*, 141(4), 2019.
- [162] K. Suzuki and N. Kikuchi. A homogenization method for shape and topology optimization. *Computer Methods in Applied Mechanics and Engineering*, 93:291–318, 1991.
- [163] A. Diaz and M. Bendsoe. Shape optimization of structures for multiple loading conditions using a homogenization method. *Structural Optimization*, 4(1):17–22, 1992.
- [164] P. Zhang, J. Toman, Y. Yu, E. Biyikli, M. Kirca, M. Chmielus, and A. C. To. Efficient design-optimization of variable-density hexagonal cellular structure by additive manufacturing: theory and validation. *Journal of Manufacturing Science and Engineering*, 137(2), 2015.
- [165] A. D. Cramer, V. J. Challis, and A. P. Roberts. Microstructure interpolation for macroscopic design. *Structural and Multidisciplinary Optimization*, 53(3):489–500, 2016.
- [166] L. Cheng, P. Zhang, E. Biyikli, J. Bai, J. Robbins, and A. To. Efficient design optimization of variable-density cellular structures for additive manufacturing: theory and experimental validation. *Rapid Prototyping Journal*, 23(4), 2017.
- [167] D. Li, W. Liao, N. Dai, G. Dong, Y. Tang, and Y. M. Xie. Optimal design and modeling of gyroid-based functionally graded cellular structures for additive manufacturing. *Computer-Aided Design*, 104:87–99, 2018.
- [168] C. Wang, J. H. Zhu, W. H. Zhang, S. Y. Li, and J. Kong. Concurrent topology optimization design of structures and non-uniform parameterized lattice microstructures. *Structural and Multidisciplinary Optimization*, 58(1):35–50, 2018.
- [169] D. Li, N. Dai, Y. Tang, G. Dong, and Y. F. Zhao. Design and optimization of graded cellular structures with triply periodic level surface-based topological shapes. *Journal of Mechanical Design*, 141(7), 2019.
- [170] C. Wang, X. Gu, J. Zhu, H. Zhou, S. Li, and W. Zhang. Concurrent design of hierarchical structures with three-dimensional parameterized lattice microstructures for additive manufacturing. *Structural and Multidisciplinary Optimization*, 61(3):869–894, 2020.
- [171] Y. Wang, L. Zhang, S. Daynes, H. Zhang, S. Feih, and M. Y. Wang. Design of graded lattice structure with optimized mesostructures for additive manufacturing. *Materials & Design*, 142:114–123, 2018.
- [172] L. Cheng, J. Bai, and A. C. To. Functionally graded lattice structure topology optimization for the design of additive manufactured components with stress constraints. *Computer Methods in Applied Mechanics and Engineering*, 344:334–359, 2019.
- [173] O. Pantz and K. Trabelsi. A post-treatment of the homogenization method for shape optimization. *SIAM Journal on Control and Optimization*, 47(3):1380–1398, 2008.

- [174] G. Allaire, P. Geoffroy-Donders, and O. Pantz. Topology optimization of modulated and oriented periodic microstructures by the homogenization method. *Computers & Mathematics with Applications*, 78(7):2197–2229, 2019.
- [175] P. Geoffroy-Donders, G. Allaire, and O. Pantz. 3-d topology optimization of modulated and oriented periodic microstructures by the homogenization method. *Journal of Computational Physics*, 401:108994, 2020.
- [176] J. P. Groen and O. Sigmund. Homogenization-based topology optimization for high-resolution manufacturable microstructures. *International Journal for Numerical Methods in Engineering*, 113(8):1148–1163, 2018.
- [177] S. Larsen, O. Sigmund, and J. Groen. Optimal truss and frame design from projected homogenization-based topology optimization. *Structural and Multidisciplinary Optimization*, 57(4):1461–1474, 2018.
- [178] J. P. Groen, J. Wu, and O. Sigmund. Homogenization-based stiffness optimization and projection of 2D coated structures with orthotropic infill. *Computer Methods in Applied Mechanics and Engineering*, 349:722–742, 2019.
- [179] J. P. Groen, F. C. Stutz, N. Aage, J. A. Bærentzen, and O. Sigmund. De-homogenization of optimal multi-scale 3D topologies. *Computer Methods in Applied Mechanics and Engineering*, 364:112979, 2020.
- [180] D. Li, W. Liao, N. Dai, and Y. M. Xie. Anisotropic design and optimization of conformal gradient lattice structures. *Computer-Aided Design*, 119:102787, 2020.
- [181] J. Wu, W. Wang, and X. Gao. Design and optimization of conforming lattice structures. *IEEE Transactions on Visualization and Computer Graphics*, 27(1):43–56, 2019.
- [182] C. Liu, Z. Du, W. Zhang, Y. Zhu, and X. Guo. Additive manufacturing-oriented design of graded lattice structures through explicit topology optimization. *Journal of Applied Mechanics*, 84(8), 2017.
- [183] Y. Zhu, S. Li, Z. Du, C. Liu, X. Guo, and W. Zhang. A novel asymptotic-analysis-based homogenisation approach towards fast design of infill graded microstructures. *Journal of the Mechanics and Physics of Solids*, 124:612–633, 2019.
- [184] D. Xue, Y. Zhu, and X. Guo. Generation of smoothly-varying infill configurations from a continuous menu of cell patterns and the asymptotic analysis of its mechanical behaviour. *Computer Methods in Applied Mechanics and Engineering*, 366:113037, 2020.
- [185] H. Rodrigues, J. M. Guedes, and M. Bendsoe. Hierarchical optimization of material and structure. *Structural and Multidisciplinary Optimization*, 24(1):1–10, 2002.
- [186] P. G. Coelho, P. R. Fernandes, J. M. Guedes, and H. C. Rodrigues. A hierarchical model for concurrent material and topology optimisation of three-dimensional structures. *Structural and Multidisciplinary Optimization*, 35(2):107–115, 2008.
- [187] L. Liu, J. Yan, and G. Cheng. Optimum structure with homogeneous optimum truss-like material. *Computers & Structures*, 86(13-14):1417–1425, 2008.

- [188] L. Xia and P. Breitkopf. Concurrent topology optimization design of material and structure within FE2 nonlinear multiscale analysis framework. *Computer Methods in Applied Mechanics and Engineering*, 278:524–542, 2014.
- [189] L. Xia. *Multiscale Structural Topology Optimization*. Elsevier, 2016.
- [190] L. Xia and P. Breitkopf. Recent advances on topology optimization of multiscale nonlinear structures. *Archives of Computational Methods in Engineering*, 24(2):227–249, 2017.
- [191] J. Gao, Z. Luo, L. Xia, and L. Gao. Concurrent topology optimization of multiscale composite structures in matlab. *Structural and Multidisciplinary Optimization*, 60(6):2621–2651, 2019.
- [192] J. Deng, J. Yan, and G. Cheng. Multi-objective concurrent topology optimization of thermoelastic structures composed of homogeneous porous material. *Structural and Multidisciplinary Optimization*, 47(4):583–597, 2013.
- [193] X. Yan, X. Huang, Y. Zha, and Y. Xie. Concurrent topology optimization of structures and their composite microstructures. *Computers & Structures*, 133:103–110, 2014.
- [194] Y. Wang, M. Y. Wang, and F. Chen. Structure-material integrated design by level sets. *Structural and Multidisciplinary Optimization*, 54(5):1145–1156, 2016.
- [195] W. Chen, L. Tong, and S. Liu. Concurrent topology design of structure and material using a two-scale topology optimization. *Computers & Structures*, 178:119–128, 2017.
- [196] S. Zhou and Q. Li. Design of graded two-phase microstructures for tailored elasticity gradients. *Journal of Materials Science*, 43(15):5157–5167, 2008.
- [197] J. Deng and W. Chen. Concurrent topology optimization of multiscale structures with multiple porous materials under random field loading uncertainty. *Structural and Multidisciplinary Optimization*, 56(1):1–19, 2017.
- [198] Y. Zhang, M. Xiao, H. Li, L. Gao, and S. Chu. Multiscale concurrent topology optimization for cellular structures with multiple microstructures based on ordered simp interpolation. *Computational Materials Science*, 155:74–91, 2018.
- [199] H. Li, Z. Luo, L. Gao, and Q. Qin. Topology optimization for concurrent design of structures with multi-patch microstructures by level sets. *Computer Methods in Applied Mechanics and Engineering*, 331:536–561, 2018.
- [200] H. Li, Z. Luo, L. Gao, and P. Walker. Topology optimization for functionally graded cellular composites with metamaterials by level sets. *Computer Methods in Applied Mechanics and Engineering*, 328:340–364, 2018.
- [201] Y. Wang and Z. Kang. Concurrent two-scale topological design of multiple unit cells and structure using combined velocity field level set and density model. *Computer Methods in Applied Mechanics and Engineering*, 347:340–364, 2019.
- [202] Z. Du, X.-Y. Zhou, R. Picelli, and H. A. Kim. Connecting microstructures for multi-scale topology optimization with connectivity index constraints. *Journal of Mechanical Design*, 140(11), 2018.

- [203] P. Liu, Z. Kang, and Y. Luo. Two-scale concurrent topology optimization of lattice structures with connectable microstructures. *Additive Manufacturing*, 36:101427, 2020.
- [204] Y. Wang, F. Chen, and M. Y. Wang. Concurrent design with connectable graded microstructures. *Computer Methods in Applied Mechanics and Engineering*, 317:84–101, 2017.
- [205] V.-N. Hoang, P. Tran, N.-L. Nguyen, K. Hackl, and H. Nguyen-Xuan. Adaptive concurrent topology optimization of coated structures with nonperiodic infill for additive manufacturing. *Computer-Aided Design*, 129:102918, 2020.
- [206] J. Robbins, S. Owen, B. Clark, and T. Voth. An efficient and scalable approach for generating topologically optimized cellular structures for additive manufacturing. *Additive Manufacturing*, 12:296–304, 2016.
- [207] B. H. Jared, M. A. Aguilo, L. L. Beghini, B. L. Boyce, B. W. Clark, A. Cook, B. J. Kaehr, and J. Robbins. Additive manufacturing: Toward holistic design. *Scripta Materialia*, 135:141–147, 2017.
- [208] J. Wu, A. Clausen, and O. Sigmund. Minimum compliance topology optimization of shell–infill composites for additive manufacturing. *Computer Methods in Applied Mechanics and Engineering*, 326:358–375, 2017.
- [209] J. Wu, N. Aage, R. Westermann, and O. Sigmund. Infill optimization for additive manufacturing—approaching bone-like porous structures. *IEEE Transactions on Visualization and Computer Graphics*, 24(2):1127–1140, 2017.
- [210] F. W. Zok, R. M. Lattice, and M. R. Begley. Periodic truss structures. *Journal of the Mechanics and Physics of Solids*, 96:184–203, 2016.
- [211] R. M. Lattice, M. R. Begley, and F. W. Zok. Design and mechanical properties of elastically isotropic trusses. *Journal of Materials Research*, 33(3):249–263, 2018.
- [212] O. Sigmund. A new class of extremal composites. *Journal of the Mechanics and Physics of Solids*, 48(2):397–428, 2000.
- [213] M. Neves, H. Rodrigues, and J. M. Guedes. Optimal design of periodic linear elastic microstructures. *Computers & Structures*, 76(1-3):421–429, 2000.
- [214] X. Huang, A. Radman, and Y. M. Xie. Topological design of microstructures of cellular materials for maximum bulk or shear modulus. *Computational Materials Science*, 50(6):1861–1870, 2011.
- [215] U. D. Larsen, O. Sigmund, and S. Bouwsta. Design and fabrication of compliant micromechanisms and structures with negative Poisson’s ratio. *Journal of Microelectromechanical Systems*, 6(2):99–106, 1997.
- [216] E. Andreassen, B. S. Lazarov, and O. Sigmund. Design of manufacturable 3D extremal elastic microstructure. *Mechanics of Materials*, 69(1):1–10, 2014.
- [217] A. Clausen, F. Wang, J. S. Jensen, O. Sigmund, and J. A. Lewis. Topology optimized architectures with programmable Poisson’s ratio over large deformations. *Advanced Materials*, 27(37):5523–5527, 2015.

- [218] F. Wang. Systematic design of 3D auxetic lattice materials with programmable poisson's ratio for finite strains. *Journal of the Mechanics and Physics of Solids*, 114: 303–318, 2018.
- [219] O. Sigmund. Materials with prescribed constitutive parameters: an inverse homogenization problem. *International Journal of Solids and Structures*, 31(17):2313–2329, 1994.
- [220] O. Sigmund. Tailoring materials with prescribed elastic properties. *Mechanics of Materials*, 20(4):351–368, 1995.
- [221] D. E. Breen and R. T. Whitaker. A level-set approach for the metamorphosis of solid models. *IEEE Transactions on Visualization and Computer Graphics*, 7(2):173–192, 2001.
- [222] A. Radman, X. Huang, and Y. Xie. Topology optimization of functionally graded cellular materials. *Journal of Materials Science*, 48(4):1503–1510, 2013.
- [223] E. Garner, H. M. Kolken, C. C. Wang, A. A. Zadpoor, and J. Wu. Compatibility in microstructural optimization for additive manufacturing. *Additive Manufacturing*, 26: 65–75, 2019.
- [224] I. Gibson, D. Rosen, B. Stucker, and M. Khorasani. *Additive Manufacturing Technologies*, volume 17. Springer, 2014.
- [225] O. Al-Ketan and R. K. Abu Al-Rub. Multifunctional mechanical metamaterials based on triply periodic minimal surface lattices. *Advanced Engineering Materials*, 21(10): 1900524, 2019.
- [226] F. W. Zok, R. M. Latture, and M. R. Begley. Periodic truss structures. *Journal of the Mechanics and Physics of Solids*, 96:184–203, 2016.
- [227] C. Soyarslan, S. Bargmann, M. Pradas, and J. Weissmüller. 3D stochastic bicontinuous microstructures: Generation, topology and elasticity. *Acta Materialia*, 149:326–340, 2018.
- [228] A. Vidyasagar, S. Krödel, and D. M. Kochmann. Microstructural patterns with tunable mechanical anisotropy obtained by simulating anisotropic spinodal decomposition. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 474(2218):20180535, 2018.
- [229] M.-T. Hsieh, B. Endo, Y. Zhang, J. Bauer, and L. Valdevit. The mechanical response of cellular materials with spinodal topologies. *Journal of the Mechanics and Physics of Solids*, 125:401–419, 2019.
- [230] S. Kumar, S. Tan, L. Zheng, and D. M. Kochmann. Inverse-designed spinodoid metamaterials. *NPJ Computational Materials*, 6(1):1–10, 2020.
- [231] D. A. Walker, J. L. Hedrick, and C. A. Mirkin. Rapid, large-volume, thermally controlled 3D printing using a mobile liquid interface. *Science*, 366(6463):360–364, 2019.
- [232] Q. Fang and D. A. Boas. Tetrahedral mesh generation from volumetric binary and grayscale images. In *2009 IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pages 1142–1145. IEEE, 2009.

- [233] H. Si. Tetgen, a Delaunay-based quality tetrahedral mesh generator. *ACM Transactions on Mathematical Software (TOMS)*, 41(2):1–36, 2015.
- [234] A. Hosny, S. J. Keating, J. D. Dilley, B. Ripley, T. Kelil, S. Pieper, D. Kolb, C. Bader, A.-M. Pobloth, M. Griffin, R. Nezafat, D. Georg, E. A. Chiocca, J. R. Stone, J. S. Michaelson, M. N. Dean, N. Oxman, and J. C. Weaver. From improved diagnostics to presurgical planning: high-resolution functionally graded multimaterial 3d printing of biomedical tomographic data sets. *3D Printing and Additive Manufacturing*, 5(2): 103–113, 2018.
- [235] R. M. Latture, M. R. Begley, and F. W. Zok. Design and mechanical properties of elastically isotropic trusses. *Journal of Materials Research*, 33(3):249–263, 2018.
- [236] A. Radman, X. Huang, and Y. Xie. Topology optimization of functionally graded cellular materials. *Journal of Materials Science*, 48(4):1503–1510, 2013.
- [237] D. P. Billington. *The Tower and the Bridge: The New Art of Structural Engineering*. Princeton University Press, 1985.
- [238] J. Gallant. The shape of the Eiffel Tower. *American Journal of Physics*, 70(2):160–162, 2002.
- [239] H.-J. Schek. The force density method for form finding and computation of general networks. *Computer Methods in Applied Mechanics and Engineering*, 3(1):115–134, 1974.
- [240] Y. Jiang, T. Zegard, W. F. Baker, and G. H. Paulino. Form-finding of grid-shells using the ground structure and potential energy methods: a comparative study and assessment. *Structural and Multidisciplinary Optimization*, 57(3):1187–1211, 2018.
- [241] H. E. Fairclough, M. Gilbert, A. V. Pichugin, A. Tyas, and I. Firth. Theoretically optimal forms for very long-span bridges under gravity loading. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 474(2217): 20170726, 2018.
- [242] F. V. Senhora, O. Giraldo-Londoño, I. F. Menezes, and G. H. Paulino. Topology optimization with local stress constraints: a stress aggregation-free approach. *Structural and Multidisciplinary Optimization*, 62(4):1639–1668, 2020.
- [243] O. Giraldo-Londoño and G. H. Paulino. A unified approach for topology optimization with local stress constraints considering various failure criteria: von mises, drucker–prager, tresca, mohr–coulomb, bresler–pister and willam–warnke. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 476(2238): 20190861, 2020.
- [244] D. Pasini and J. K. Guest. Imperfect architected materials: mechanics and topology optimization. *MRS Bulletin*, 44(10):766–772, 2019.
- [245] A. Moussa, D. Melancon, A. El Elmi, and D. Pasini. Topology optimization of imperfect lattice materials built with process-induced defects via powder bed fusion. *Additive Manufacturing*, page 101608, 2020.

- [246] T. Wangler, N. Roussel, F. P. Bos, T. A. Salet, and R. J. Flatt. Digital concrete: a review. *Cement and Concrete Research*, 123:105780, 2019.
- [247] T. Zhao, A. S. Ramos Jr, and G. H. Paulino. Material nonlinear topology optimization considering the von Mises criterion through an asymptotic approach: Max strain energy and max load factor formulations. *International Journal for Numerical Methods in Engineering*, 118(13):804–828, 2019.
- [248] R. T. Haftka and Z. Gürdal. *Elements of Structural Optimization*, volume 11. Springer Science & Business Media, 2012.
- [249] S. Boyd, S. P. Boyd, and L. Vandenberghe. *Convex Optimization*. Cambridge university press, 2004.
- [250] J. S. Arora. *Introduction to Optimum Design*. Elsevier, 2004.
- [251] S. R. Almeida, G. H. Paulino, and E. C. Silva. Layout and material gradation in topology optimization of functionally graded structures: a global–local approach. *Structural and Multidisciplinary Optimization*, 42(6):855–868, 2010.
- [252] B. A. Auld. *Acoustic Fields and Waves in Solids*, volume I. John Wiley & Sons, 1973.
- [253] J. Turley and G. Sines. The anisotropy of Young’s modulus, shear modulus and Poisson’s ratio in cubic materials. *Journal of Physics D: Applied Physics*, 4(2):264, 1971.