Structural Grid Shell Design with Islamic Pattern Topologies

by

Noor K. Khouri

S.B. Civil Engineering,
Massachusetts Institute of Technology (2015)

Submitted to the Department of Architecture
in partial fulfillment of the requirements for the degree of

Master of Science in Building Technology

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2017

Author . . . .
Department of Architecture
May 24, 2017

Certified by . . . .
Caitlin T. Mueller
Assistant Professor of Structural Design and
Civil and Environmental Engineering
Thesis Supervisor

Accepted by . . . .
Sheila Kennedy
Professor of Architecture
Chair of the Department Committee on Graduate Students

Structural Grid Shell Design with Islamic Pattern Topologies

by

Noor K. Khouri

Submitted to the Department of Architecture
on May 24, 2017, in partial fulfillment of the
requirements for the degree of
Master of Science in Building Technology

Abstract

Geometric patterns, pioneered centuries ago as a dominant form of ornamentation in Islamic architecture, represent an abundant source of possible topologies and geometries that can be explored in the preliminary design of discrete structures. This diverse design space motivates the coupling between Islamic patterns and the form finding of funicular grid shells for which structural performance is highly affected by topology and geometry. This thesis examines one such pattern through a parametric, performance-driven framework in the context of conceptual design, when many alternatives are being considered. Form finding is conducted via the force density method, which is augmented with the addition of a force density optimization loop to enable grid shell height selection. A further modification allows for force densities to be scaled according to the initial member lengths, introducing sensitivity to pattern geometry in the final form-found structures. The results attest to the viable synergy between architectural and structural objectives through grid shells that perform as well as, or better than, quadrilateral grid shells. Historic and cultural patterns therefore present design opportunities that both expand the conventional grid shell design vocabulary and offer designers an alternative means of referencing vernacular traditions in the modern built environment, through a structural engineering lens.

Key words: grid shell, structural topology, Islamic pattern, parametric design, performance driven design, force density method, form finding

Thesis Supervisor: Caitlin T. Mueller
Title: Assistant Professor of Structural Design and
Civil and Environmental Engineering

Acknowledgments

Over the past six years, I have had the privilege to meet and work with extraordinary members of the MIT community. I am incredibly grateful to each and every person I have encountered during my time here, for challenging my thoughts, shaping my trajectory, and for supporting me throughout my academic endeavors.

Firstly, I would like to extend my sincere gratitude to my thesis advisor, Professor Caitlin Mueller, for her ongoing encouragement and support. The research presented in this thesis is an outcome of our many fruitful discussions, fueled by our mutual appreciation for interdisciplinarity, particularly at the intersection of computation, engineering and design.

My time at MIT would be incomplete without acknowledging the support of my undergraduate mentors. I would like to thank Professor John Ochsendorf, my undergraduate academic advisor, for first welcoming me into his beam testing lab as an eager prospective student and showing me the warmth and energy that make MIT a truly special place. His sheer dedication to his students, approachable demeanor, and thoughtful guidance are some of the many reasons I am thankful to him for.

I am also indebted to Professor Pedro Reis and to Dr. Khalid Jawed who I was lucky to have worked with as a UROP student for a duration longer than my Master's degree. They introduced me to the intellectual rigor of research, teaching and challenging me every day, and ultimately fostered my growth mindset. Their lessons propelled me forward throughout my graduate studies and permeate throughout each page of this thesis.

I would also like to thank my fellow members of the Digital Structures research group for graciously lending a helping hand whenever I needed it, and for cultivating a friendly research environment. On a similar note, I appreciate all of the exchanges that I have had with the Building Technology faculty, staff and students. The Hacking Arts and Creative Arts Competition teams that I am honored to have been a part of also deserve special recognition; in particular, for helping build the MIT arts entrepreneurship ecosystem, but also for the many friendships that stemmed from

our dedication to teamwork and to working toward our common mission.

To my friends and family, who have supported me from near and far, and who continue to bring me so much joy, I look forward to thanking you all in person. The memories and good times that we have shared together are very dear to me, and have energized me when faced with trying times.

Finally, I would like to dedicate this thesis to my parents and sister, to whom I owe everything.

Contents

List of topology and form finding variables

Variable	Description
o_1, o_2	Radial dimensionless pattern offset variable pertaining to the octagonal and diamond pattern units, respectively ($0 \leq o_1 \leq 1$ and $0 \leq o_2 \leq 1$). See Figure 3-3
$o_{dist,1}$, $o_{dist,2}$	Radial physical offset distance within the octagonal and diamond pattern units, respectively
c_1, c_2	Empirically derived constants that are used to remap the dimensionless offsets to physical distances according to Equation 3.1
d, d_{quad}	Pattern density of the Islamic pattern and its corresponding quadrilateral pattern, respectively. Values are expressed as the number of polygonal units along the outer edge of the square plan. See Figure 3-3
m, m_{quad}	Number of members in a given Islamic pattern or its quadrilateral counterpart, respectively
a	Area of the square that fully encloses a pattern, held constant throughout the parametric studies ($a = 100$)
i	Pattern or grid shell member number
BC	Boundary condition applied for form finding: 1- convex hull, 2- barrel, 3- corners
p_{total}	Total external force applied to the grid shell, in the direction of gravity. This value is held constant at $p_{total} = 1$
h_{max}	Maximum z-coordinate within the grid shell during the optimization iterations
h_{target}	Predefined grid shell height, defined as the maximum z-coordinate within the final structure
q_{start}	Starting guess for the member force density optimization loop
q, q'	Force density variables that are minimized in the optimization loop in phase one and two, respectively (Equations 3.2 and 3.3)
q_{opt}	Optimal force density to achieve the prescribed height, h_{target}

q_i Force density of a given member, i. In phase one, $q_i = q_{opt}$ for all members, while in phase two $q_i = \frac{q_{opt}}{l_{i,initial}}$

f_i Internal force of a given member, i, in the form-found grid shell

f_{min}, f_{max} Minimum and maximum internal member force in the form-found grid shell, respectively

l_i Final length of a given member, i, in the form-found grid shell

$l_{i,initial}$ Initial length of given member,i, within the two-dimensional starting pattern

Chapter 1

Introduction

Contemporary architects and engineers often search for innovative and diverse forms during conceptual design iterations in order to counteract the rectilinear monotony that is commonplace in urban areas. In addition, environmental impact is becoming increasingly important in the built environment, necessitating a shift toward efficiency in design. This can be achieved through reductions in both the operational and embodied energy of buildings, of which the latter is most significantly affected by structural form and material distribution [1]. The objective of emphasizing a structure's aesthetic qualities while augmenting its efficiency therefore represents a dichotomy often faced by industry practitioners.

One possible solution that seeks to address this question of combining architectural and structural intents is form finding. Form finding was first manifested through the physical hanging models that were central to the works of great twentieth century engineers [2]: Heinz Isler [3], Antoni Gaudí [4], and Frei Otto [5]. The technique produces funicular shapes, or forms, acting in pure tension or compression, and are determined as a direct response to the applied forces. With bending action mitigated, all internal forces are transferred axially, which makes the most efficient use of the member cross-section and allocated material strength. This intentional focus on structural efficiency during the very foundation of the design process stands in direct contrast to the highly inefficient approach of designing free-form structures that are rationalized *a posteriori*.

While free-form design was popularized via the advent of computational design tools, the development of a multitude of numerical methods [6] has made form finding equally viable. As a result, form finding has been used to design a variety of typologies ranging from membrane structures to shell structures. Within the shell structures category, grid shells are particularly conducive to formalized form finding methods for two-dimensional discrete networks, or patterns, for which the final structural form is not known. A grid shell is a surface structure that achieves its strength and stiffness through double curvature which is articulated through a network of discrete, linear members. This structural typology is favored for its ability to combine aesthetic, performance, and daylighting objectives, and as such, has been used as an elegant solution for long span roof structures. Noteworthy examples of grid shells include the Great Court at the British Museum in London (2000), as well as the National Maritime Museum courtyard in Amsterdam (2011), which is particularly intriguing for deriving a modified funicular form from a complex, historically significant patterned map (Figure 1-1) [2].

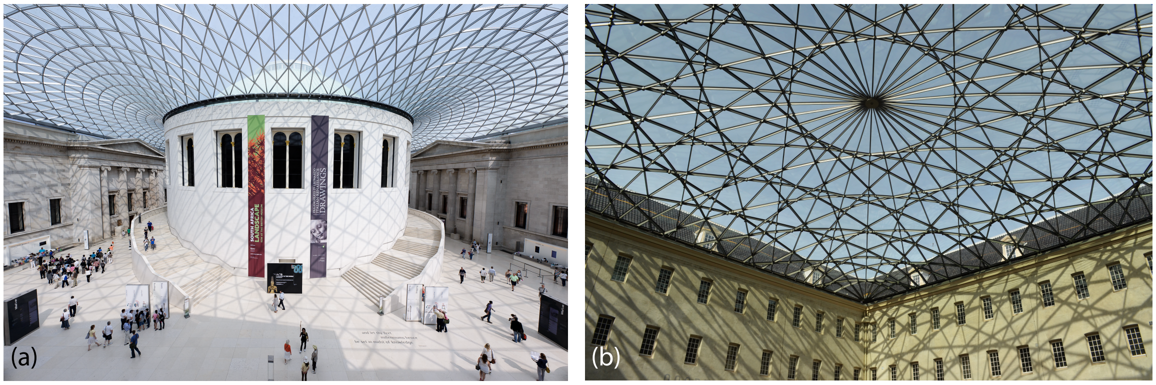

Figure 1-1: (a) Grid shell roof structure of the Great Court at the British Museum in London, UK, designed by Foster + Partners (photograph taken by Andrew Stawarz, reprinted from Ref. [7]) (b) Grid shell glass roof of the National Maritime Museum in Amsterdam, Netherlands, designed by Ney + Partners (photograph taken by Flickr user: kaysgeog, reprinted from Ref. [8])

The National Maritime Museum therefore represents a departure from the confines of quadrilateral or polygonal patterns, and instead makes a case for a broadening of the grid shell design vocabulary through an investigation of alternative pattern topologies [9–11], as opposed to global forms, which have been well studied [12,13]. The topology, or connectivity, of a two-dimensional pattern plays a particularly important role, not only in influencing the overall form of the resulting grid shell, and hence its visual appearance, but also in determining the force distribution [14]. Extending that further, the geometry pertaining to a particular topology, defined as the coordinates of the nodes comprising a pattern, also directly impacts the flow of forces. As a result, form finding with a variety of pattern topologies and geometries simultaneously provides designers with creative freedom and control over structural performance in early stage grid shell design.

1.1 Definition of the problem

This thesis aims to further investigate the effect of pattern topology and geometry within the context of funicular grid shells, but through an alternative design lens that looks to historic precedents in the field of geometry. Islamic geometric patterns are perhaps the most widely recognized ornamental feature of Islamic architecture and are revered for the breadth of design possibilities that stem from their inherent mathematical nature. For this reason, the presented research draws upon Islamic geometric tradition to generate diverse topologies, which are at the confluence of culture, architecture and design, and instead, reorients their function toward a structural one. Allowing for both architectural and structural objectives to be prioritized through topological and geometric considerations, the proposed methodology for the form finding of grid shells integrates the necessity for rapid design explorations within a parametric, performance-driven computational framework. Form finding is implemented via the well-established and commonly used force density method [15], with the addition of an optimization loop that determines the necessary member force densities to achieve the grid shell height prescribed by the designer. Possible design outcomes derived from Islamic patterns are demonstrated through a grid shell design case study comprising two phases that can be performed independently, but are shown sequentially. The first phase conducts a comprehensive parametric study of the grid shell design space with a particular emphasis on topology, as per the variables defined in Figure 3-3. The second phase examines the effects of pattern geometry by presenting a strategy for scaling member force densities, and results in efficient, yet diverse grid shells that have not yet been realized in the built environment.

1.2 Outline of the thesis

This chapter outlined the underlying motivation behind the proposed two-phase grid shell study, which is addressed through the lens of geometric and topological design opportunities brought forth by culturally significant patterns. Chapter 2 provides background information on Islamic geometric design, as well as the use of the patterns in both historic and modern architectural contexts. Moreover, a brief introduction to the force density method is included. Building upon this, Chapter 3 presents the computational methodology for the design investigations in three sections: pattern parameterization (3.2), form finding and optimization via the force density method (3.3), and methods to evaluate grid shell designs (3.4). Design outcomes and an analysis of the performance trends observed in both phases are discussed in Chapter 4. Also, a summary of the research contributions and the possible implications on architectural and structural design practices are summarized in Chapter 5, along with suggested avenues for future work. Finally, an appendix that documents the MATLAB code comprising the computational framework explained in Chapter 3 is included.

Chapter 2

Literature Review

This chapter introduces the historical and cultural background that is pertinent to the patterns used to generate grid shell designs. Contemporary uses and interpretations of these patterns in architectural design are also discussed, noting a particular emphasis on the extension of the metaphor of traditional shading lattice screens to intelligent facades; Islamic patterns have yet to be used as core components of structural systems. Furthermore, a general overview of the force density method implemented for form finding is outlined.

2.1 Islamic geometric design

Prior to sampling grid shell designs using Islamic patterns, it is important to establish the cultural context and significance of this geometric innovation. The history and development of Islamic architecture is one characterized by a continuous exploration of personal, communal and religious spaces, as well as material, structural and ornamental advancements. The impressive complexity that was achieved centuries ago, without the aid of computational tools, stands as a testament to the synergy between architects, engineers and artisans [16]. Of the many innovations that stemmed from this relationship, geometric patterns are perhaps the most universal, becoming the dominant decorative repertoire in Islamic architecture from the eleventh century onward, and adopting various styles across the Islamic dominion [17]. The plurality of pattern types [18] observed in extant historic structures is in part due to cultural ties and regional craftwork, but is primarily due to the deeply rooted mathematical rules and order that create an intrinsically diverse design space. By their very nature, Islamic patterns result in remarkable global pattern variations with only slight modifications in local geometries, thus enabling a wide variety of design possibilities, and creative agency [19] (Figure 2-1). Moreover, symmetry and modularity are key design principles that define these patterns, where motifs are periodically repeated across a surface, allowing for geometric complexity to be achieved with ease.

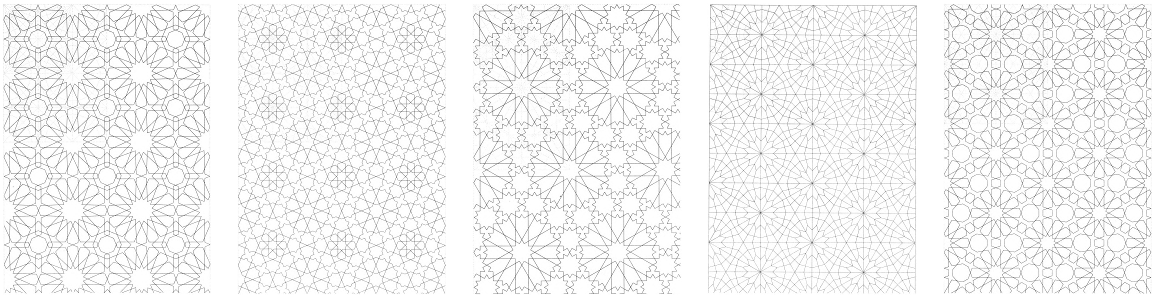

Figure 2-1: Examples of patterns, documented by Jules Bourgoin, that illustrate the abundant variety of Islamic patterns. This thesis reinterprets the patterns as grid shell topologies and geometries. The images are reprinted from Ref. [18].

Precisely how Islamic geometric patterns were designed by craftsmen many centuries ago remains shrouded with mystery despite continued art historical and mathematical research [20]. This is because there are very few surviving manuscripts that document geometric compositions and the numerous techniques with which they were created. The primary source of evidence can be found on the historic sites themselves; however, the patterns belie the design procedures that led to their existence. Nonetheless, nineteenth and twentieth century Western architects and scholars, most notably Owen Jones [21], Emile Prisse d'Avennes [22], Jules Bourgoin [23], and Ernest Hanbury Hankin [24], were fascinated with the vibrancy of the designs of the Islamic world, and have collectively written several books that catalog the impressive variety of patterns that have been achieved. More recently, Eric Broug examined the patterns through the viewpoint of the artisans, demonstrating that Islamic patterns can be created with only a compass and ruler in hand [20]. In what is likely one possible traditional construction method, a pattern motif begins as a primary circle from which any geometric shape can be determined in accordance with the chosen roots and proportions of the design [19]. The primary circle is interlaced with a given number of secondary circles of various sizes, then intersections are connected with straight lines [16]. The way in which the circles and lines intersect dictates the geometry of the resulting motif once it is extracted from the construction lines. Star patterns, which represent only a subset of possible geometric patterns, can not only be produced by following the aforementioned procedure, but also via computational means as a result of the research developed within the field of computer graphics.

It is important to note that no single physical or computational approach to Islamic pattern design can achieve universality, thus numerous computational methodologies have emerged in attempts to reproduce existing patterns and uncover new ones. Group-theoretic approaches are employed by Grünbaum and Shephard [25], and Abas and Salman [26], who develop algorithms around pattern symmetries. On the other hand, Kaplan proposes a tiling-based construction method wherein template tiles with embedded motifs are used to guide the layout of the final design; star patterns emerge once the tiling is assembled [27]. These bodies of work, to name a few, pave the way for extending the generation of Islamic patterns into parametric design tools that are common in architectural design practices. One such example is developed by Yazar [28] in Grasshopper for Rhinoceros 5, and builds upon Kaplan's method comprising tile units and motifs. Yazar's script [29] forms the basis for the grid shell form finding process discussed in Chapter 3.

2.2 Islamic patterns for structural design

Evidently, Islamic patterns are widely recognized for their aesthetic qualities, and their use has been prolific in ornamentation, but not in structural design. Historically, these patterns were predominantly used in shading lattice screens (*Mashrabiya*) and other ornamental elements, often for both functional and aesthetic purposes (Figure 2-2 (a)). Referencing these traditional screens, both the Al Bahr Towers in Abu

Dhabi (2012), and l'Institut du Monde Arabe in Paris (1987), designed in the twenty-first century, employ patterned facades as key distinctive architectural features in addition to serving as intelligent shading systems that are responsive to daylighting conditions (Figure 2-3). Examining historic precedents in Islamic architecture reveals that these patterns were rarely, if ever, incorporated as load-bearing structural elements; the closest parallel that can be drawn is an intricate decorative vaulting system, *Muqarnas*, often used to adorn the interior surface of domes and squinches [30] (Figure 2-2 (b)). Such designs have generally not been translated into modern architecture to serve a structural purpose either. Even Jean Nouvel's design for the dome of the Louvre in Abu Dhabi (scheduled to open in 2017), which most closely aligns with the structural intents presented in this thesis, consists of what appears to be several layers of non-structural interlacing geometric patterns that are supported by an underlying polygonal grid shell (Figure 2-4). By extension, there is also a limited availability of literature on the use of Islamic patterns as key components of structural systems. Emami et al. [31] study the effects of geometric parameters on the daylighting and structural performance of shading screens designed using the principles underlying Islamic patterns; however, the parameters and performance metrics of interest are more conducive to daylighting analysis. Nonetheless, the ever-growing interest in these patterns exhibited by leading global architecture firms reinforces the premise that there are many avenues that have yet to be explored in the realm of structural systems developed from Islamic patterns. As a result, this thesis aims to begin those investigations and attest to the viability of the use of Islamic patterns as core components of structural systems. Hence, the historic function of these patterns is reinterpreted through an engineering lens by deeply engaging the mathematical principles underlying the design space in order to transform a delicate ornamental feature to an architectural scale.

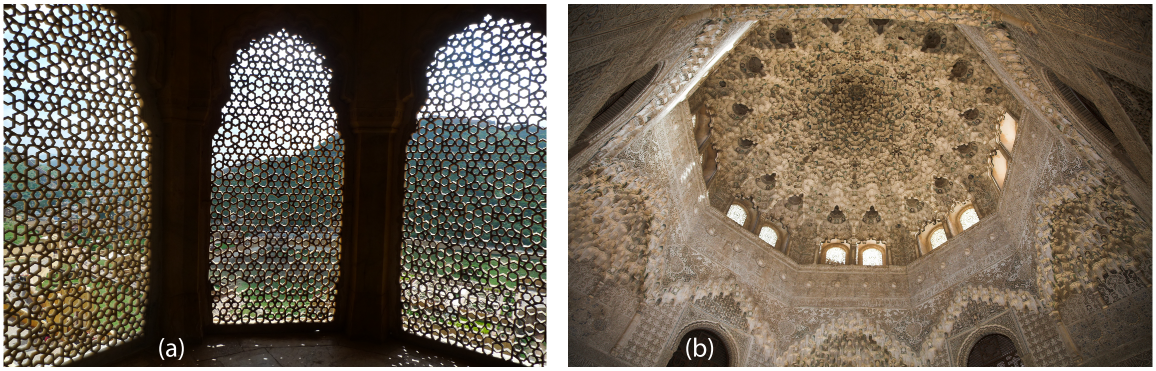

Figure 2-2: (a) Decorative lattice screen at the Amber Fort in Jaipur, India (c. sixteenth century). (photograph taken by Matthew Lewinski, reprinted from Ref. [32]) (b) *Muqarnas* dome of the Hall of the Two Sisters in the Alhambra in Granada, Spain (c. fourteenth century) (photograph taken by Flickr user: Funky Chickens, reprinted from Ref. [33]).

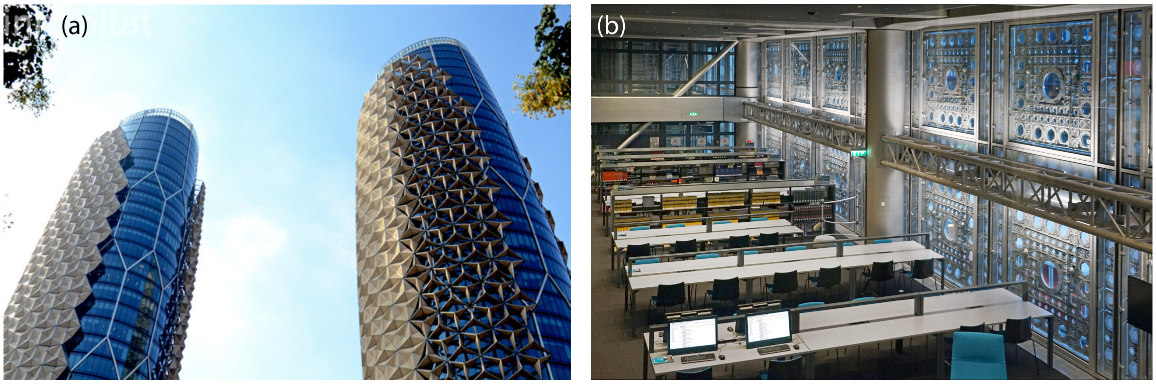

Figure 2-3: (a) Responsive shading facade system of the Al Bahr Towers in Abu Dhabi, UAE, designed by Aedas Architects (2012). The design concept is inspired by traditional *Mashrabiya* screens (photograph taken by Flickr user: Inhabitat, reprinted from Ref. [34]) (b) Responsive shading facade system of l'Institut du Monde Arabe in Paris, France, designed by Ateliers Jean Nouvel (1987). This structure also modernizes the *Mashrabiya* shading system, evoking a strong cultural reference that is central to the building's program, while also fulfilling both functional and aesthetic objectives (photograph taken by Jean-Pierre Dalbéra, reprinted from Ref. [35]).

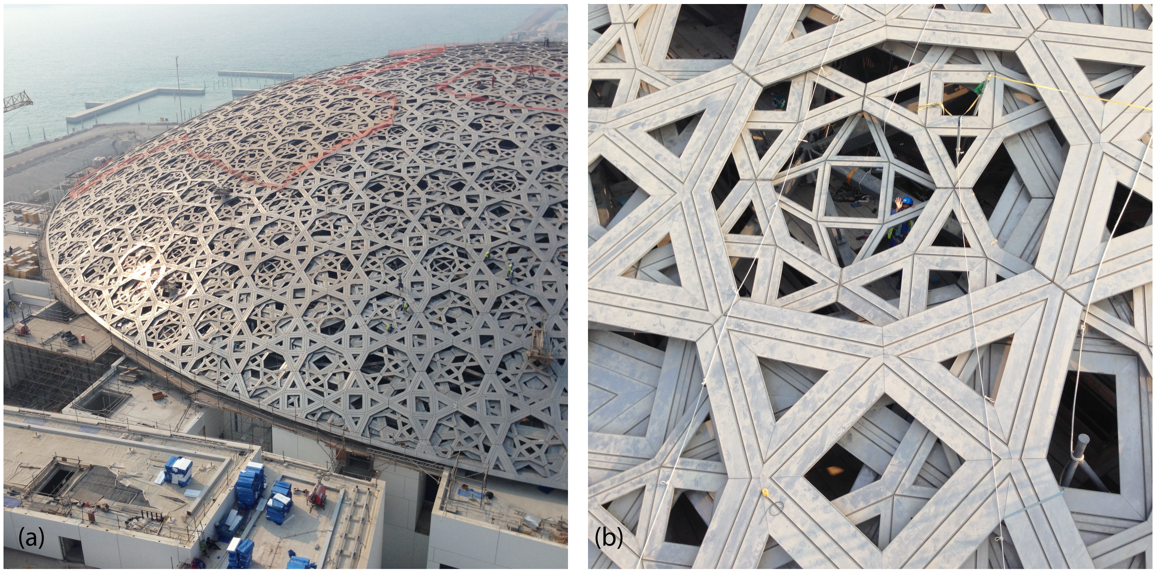

Figure 2-4: (a) Aerial view of the Louvre museum dome, designed by Ateliers Jean Nouvel, shown here under construction in 2015 in Abu Dhabi, UAE. (b) Detail of the layered interlacing geometric patterns comprising the dome (photographs courtesy of Nick Leech, `writingtoinform.com`, reprinted with permission).

2.3 Form finding using the force density method

The proposed methodology for incorporating Islamic patterns for a structural, rather than ornamental, function involves using form finding techniques to transform two-dimensional patterns into funicular grid shells. As noted in Chapter 1, form finding is the practice of determining the optimal structural form that acts in pure tension or compression, subject to particular loading and support conditions. This principle is of particular interest to structural engineers and architects since internal bending moments are minimized while the majority of the applied load is transferred axially, thus increasing the efficiency of the structure. Many form finding methods have been developed [6], but perhaps one of the most established and widely used methods is the force density method developed by Schek in 1974 [15]. One advantage of applying this theory is that form finding can be performed independently from material properties and units of measurement, which makes it an ideal candidate for conceptual design.

The formulation of the force density method enforces equilibrium at each node in an arbitrary pattern of nodes and branches, or members, as per the applied loads. Assuming a load application in the direction of gravity, which could be analogous to structural self-weight and area loads, the result would be a form-found structure that acts in pure compression once inverted. The true value of applying this theory manifests itself when considering the fact that the equilibrium equations can be expressed as a series of linear equations that are conducive to linear algebra solvers in MATLAB and can therefore be solved both quickly and efficiently [36]. This linearization is made possible via the introduction of force densities for each of the members, which is the ratio between the internal axial force in the member and its length ($q_i = \frac{f_i}{l_i}$). For this reason, the force density method is an efficient theory that enables the rapid generation of feasible funicular grid shell forms from two-dimensional patterns subject to external loading, and is thus suitable for conceptual design iterations. Additional details regarding the input parameters and the steps that a designer must take in order to initiate the form finding sequence are explained in Section 3.3.

Chapter 3

Methodology

Generating grid shells that draw upon the Islamic geometric tradition entails coupling pattern generation with existing form finding methods. The methodology outlined in this chapter was devised to achieve a synergistic, performance-driven approach to the conceptual architectural and structural design of funicular grid shells. Namely, the developed computational framework enables rapid geometry generation via form finding iterations, along with preliminary structural performance analyses that serve as a basis for comparing various topologies and geometries; this informed evaluation of several potential design options is a fundamental process of early stage design. The key steps of the embedded design process include two investigative phases that may be implemented independently, but are presented sequentially:

1. Phase one: a comprehensive parametric study of the effect of pattern topology on the form of the grid shells, and on the relative structural performance. The objective of this phase is to generate several forms, while revealing performance trends and benchmarks that allow the designer to make informed design decisions.

2. Phase two: a parametric study of pattern geometry for a grid shell prescribed by the designer. Here, force densities are scaled to further enhance the diversity of the achieved forms.

Design outcomes are demonstrated for one Islamic pattern case study, which is an eightfold star pattern that is modified from the Grasshopper script implemented by Yazar [29] (Figure 3-3).

3.1 Overview of the computational framework

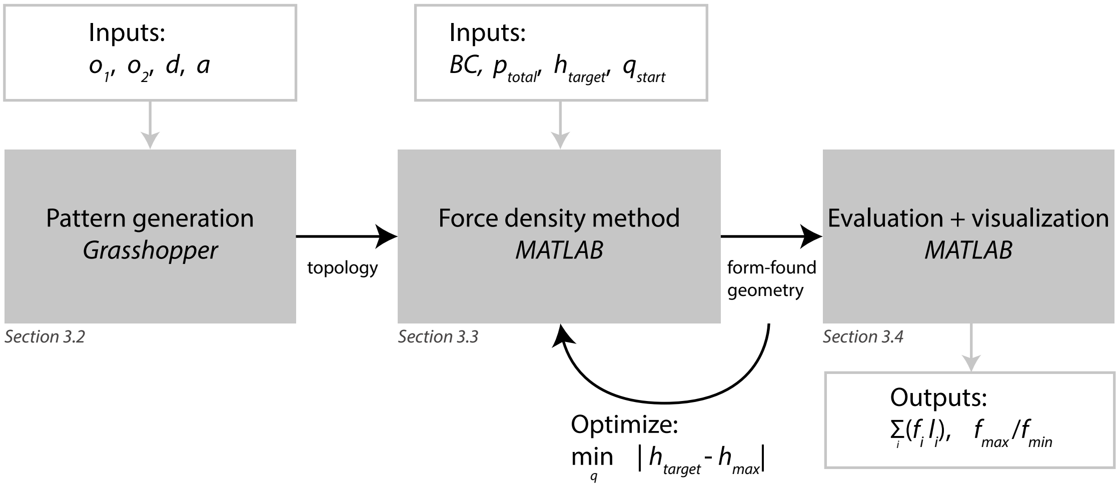

Figure 3-1: Overview of the key stages comprising the computational framework that conducts parametric studies for the conceptual structural design of grid shells with Islamic pattern topologies.

The computational framework that enables grid shell design explorations is summarized in Figure 3-1. The three core stages of the framework are as follows: 1- pattern generation and sampling in Grasshopper [29], 2- form finding via the force density method with a height-selection optimization loop, conducted in MATLAB, and 3- structural performance evaluation and form visualization, also in MATLAB. Obtaining a single funicular grid shell for a given pattern constitutes going through the entire sequence once. However, to conduct parametric studies for multiple patterns in phases one and two, which more directly correlate with generative conceptual design, the process is repeated several times in a loop: the patterns are first sampled [37] independently, and then the grid shells are automatically generated by taking advantage of parallel processing, which capitalizes on the available computational power.

3.2 Pattern parameterization

The first crucial step in the implementation of the framework for form finding is the generation of a structural pattern from which a grid shell can be derived. While the framework is intentionally designed with the flexibility to take any pattern as an input, all subsequent investigations make reference to the parameterization of an eightfold star pattern [29] shown in Figure 3-3, and the corresponding summary of design variables included at the beginning of this thesis.

An intrinsic property of Islamic vernacular star patterns is that several nodes have only two connecting edges, meaning that these nodes would have to be fixed in order to ensure structural stability during the form finding procedure. For this reason, rather than limiting possible design outcomes to a reduced set, the original Islamic pattern is first modified by reintroducing the construction lines that were used to generate the pattern in the first place. Here, these construction lines are referred to as the structural grid (Figure 3-3 (a)), which requires that each node is connected to at least three edges. In this way, the designer is able to freely select the appropriate boundary conditions per their design specifications.

The resulting structural pattern now consists of the original eightfold star pattern enclosed within a semiregular tessellation. The topology is parameterized by two radially-oriented, dimensionless offset variables: o_1, which controls the offset of the octagonal unit, and o_2, which corresponds to the diamond unit. The possible offset node paths are shown in Figure 3-3 (a) for each pattern unit, wherein the movable nodes are indicated by a circle, and the extremities, where the offset variables are equal to either 0 or 1, are also shown. While o_1 and o_2 are dimensionless to standardize the pattern definition, their values need to be remapped to physical offset distances, $o_{dist,1}$ and $o_{dist,2}$, when generating patterns. This procedure is achieved according to Equation 3.1, where c_1 and c_2 are empirically derived constants specific to the pattern in question, and the shape and dimensions of its plan (Figure 3-2).

$$o_{dist,1} = o_1 \times \frac{c_1}{d} \text{ and } o_{dist,2} = o_2 \times \frac{c_2}{d} \tag{3.1}$$

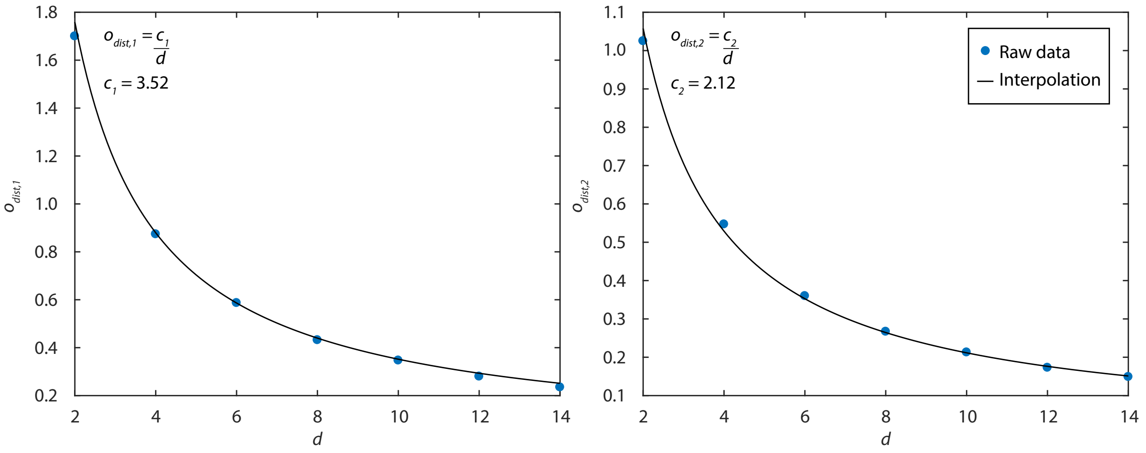

Figure 3-2: Plots of the empirically determined upper limits of the physical offset distances, $o_{dist,1}$ and $o_{dist,2}$ versus the pattern density, d, respectively. The raw data is represented as points, while the interpolated functions relating the axis parameters are shown as solid lines. The equations (Equation 3.1) and the values of the corresponding constants, c_1 and c_2, are also shown for the pattern constraints applied in the parametric studies.

The nature of this offset parameterization results in four possible topologies within which a myriad of pattern geometries can be generated (Figure 3-3 (b)). It is important to note that the force density method is agnostic to the geometry of a particular pattern topology, or the coordinates of the nodes within the pattern, in the case where member force densities are equal. This means that the first design phase is able to isolate the effects of pattern topology from its geometry, allowing for four form-found structures, corresponding to the four topology categories, to be derived for a single pattern density and choice of boundary conditions. On the other hand, in the second phase, where a force density scaling method is proposed, geometric effects are reintroduced and the full gradient of o_1 and o_2 values illustrated in Figure 3-3 (b) can be achieved.

The third and final variable that defines the pattern is d, which is defined as the number of polygonal units along a single outer edge of the pattern (Figure 3-3 (c)). The investigations discussed in this thesis constrain the patterns to square plans, where the area enclosing the pattern, a, is held constant to allow for comparisons to be made across topologies. As a result, d serves as a proxy for the pattern density.

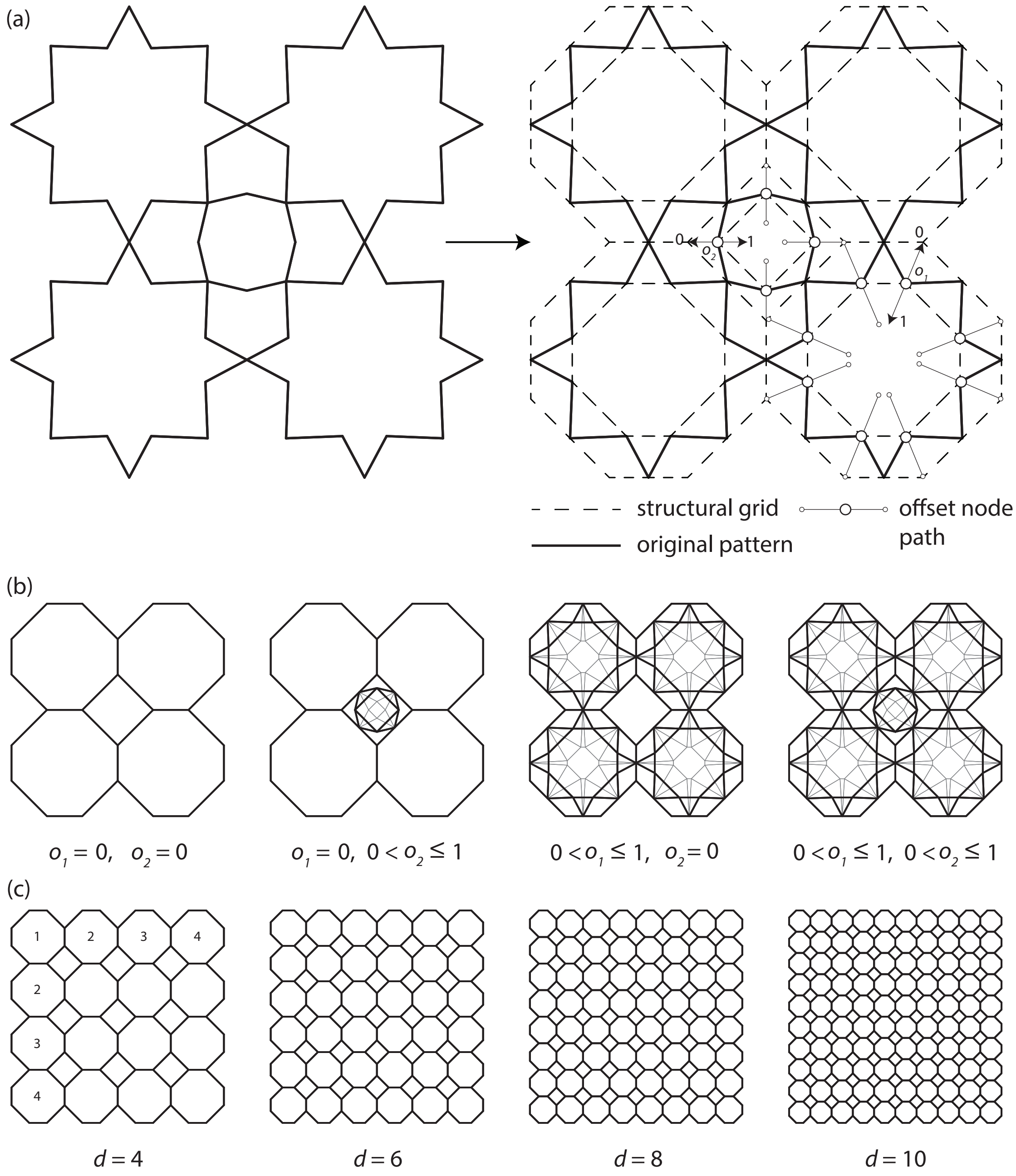

Figure 3-3: Summary of the pattern parameterization, demonstrated for the simplest case where the pattern density, d, is set to 2. (a) The transformation of the Islamic geometric pattern to a structural pattern that can be used in the form finding procedures. The dashed lines represent the structural grid that is added to the original pattern [29] to ensure that each node has at least three connecting edges. Movable nodes are indicated with a circle, and the corresponding offset node paths are shown within the octagonal and diamond units of the pattern, where o_1 and o_2 extremities are also marked. (b) The four topology categories that can be achieved via the parametric definition of the pattern. These topologies form the basis of the possible structures that can be obtained in phase one. The fine lines represent the upper limits of the offset variables within each topology category; this geometric gradient can only be taken advantage of in phase two. (c) An illustration of the pattern density, d, which is defined as the number of exterior octagonal units along the edge of the pattern.

3.3 Form finding and optimization via the force density method

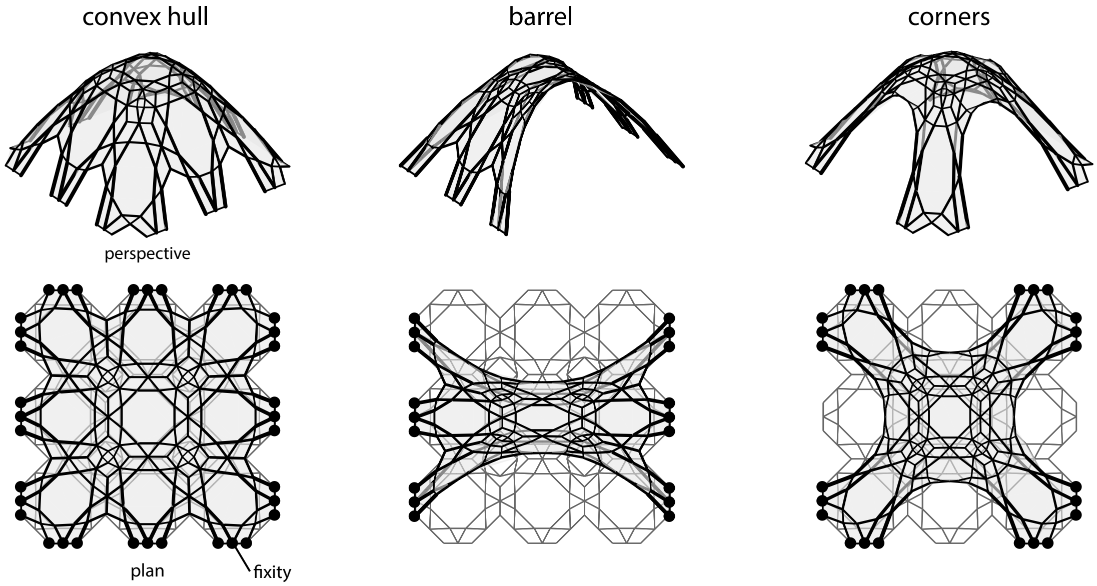

Figure 3-4: Illustration of the three boundary conditions that are automated within the computational framework and are subsequently investigated through the parametric studies. The convex hull boundary condition fixes all nodes along the outer edge of the square plan, while the barrel vault fixes the nodes along two parallel edges, and the corners case fixes the exterior corner nodes. The sample form-found structures are shown for the following topology parameterization in phase one: $d = 3$, $o_1 = 0.5$, and $o_2 = 0.5$. Fixities are marked with a filled circle in the plan views. The plan views of the form-found structures are overlain on top of the original pattern from which the structures are derived, demonstrating the range of global forms that can stem from a single pattern.

Once the patterns are selected and sampled, the boundary conditions need to be specified prior to the form finding procedure. Three boundary conditions are considered and automated within the computational framework: 1- convex hull, which results in a dome structure with no free edges, 2- barrel, which defines a barrel vault, and 3- corners, which can be thought of as a pavilion with four inclined arches. An example of each of these three forms is shown in Figure 3-4; the illustration demonstrates both the prescription of the fixity locations along the outer edges of the square plan, in addition to the range of structural geometries that can be generated from the same starting pattern. The precise definition of the boundary conditions can be found in Appendix A.3.

Next, to initiate the form finding process, the designer prescribes the the desired target height, h_{target}, in addition to the loading condition, here defined as the total external force applied to the grid shell, p_{total}, in the direction of gravity. The total load is held constant throughout the parametric studies, and is applied as nodal point forces once p_{total} is normalized by the number of free nodes in the pattern. Since the tributary area of each node is not equal, this loading condition is a simplification of the physical system; however, it is considered acceptable for preliminary design space studies. This part of the methodology could be refined in future work to encompass a secondary set of iterations in the form finding procedure, where the loads are updated to account for changes in geometry until a satisfactory form of the desired height is reached. Also, it is important to reiterate that since form finding via the force density method for conceptual design can be conducted independently from material properties, the material is not specified for the grid shell case study, and physical units are not selected since the structural analysis is performed in relative terms.

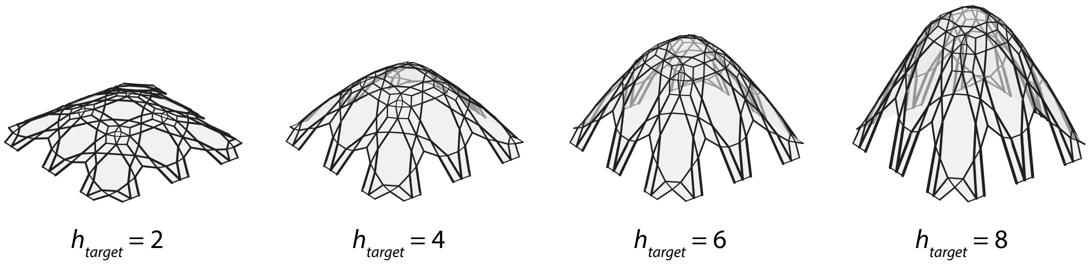

Figure 3-5: Illustration of the different global forms that can be achieved according to the designer's prescribed target height. Shallower grid shells could be appropriate as a courtyard roof structures, while taller grid shells could serve as pavilions, for example. The form-found structures are shown for the following parameters: $d = 3$, $o_1 = 0.5$, and $o_2 = 0.5$.

With all of the input variables set, form finding is then implemented via the standard formulation of the force density method, according to Refs. [15, 36] (Appendix A.4). However, the novelty of the process shown in Figure 3-1 is the addition of the optimization loop, a constrained minimization equation, where the force density, q, is updated in each form finding iteration (Appendices A.6 and A.5). This computation serves the purpose of calculating the optimal force density, q_{opt}, which is applied to each member such that the maximum height of the form-found structure is equivalent to the prescribed target height, h_{target} (Equation 3.2). Thus, the arbitrary value for the member force densities (q_{start}) that is chosen along with the other input parameters is of no true significance in the context of this methodology, since the force density is determined from the optimization loop.

$$\begin{aligned} \min_{q} \quad & \mid h_{target} - h_{max} \mid \\ \text{s.t.} \quad & q \geq 0 \end{aligned} \tag{3.2}$$

This step introduces a constraint into the form finding process that allows the designer to more closely control the global geometry of the final form, while also providing a benchmark for a fair comparison between the performance of various topologies (Figure 3-5). The general optimization method applies to both design phases, however, in order to introduce a sensitivity to geometry within the pattern topologies, and therefore diversity in the final structures, the second phase applies a scaled force density approach. In this case, rather than obtaining a single value for q_{opt}, each member is assigned a different force density, which has been scaled by $\frac{1}{l_{initial}}$. Therefore, member length proportions from the two-dimensional starting pattern are translated to the final three-dimensional form. Note that while this particular scaling method is implemented within the computational framework, many other geometric approaches to scaling could be used instead, at the designer's discretion. The constrained minimization equation for phase two now becomes:

$$\begin{aligned} \min_{q'} \quad & \mid h_{target} - h_{max} \mid \\ \text{s.t.} \quad & q' \geq 0, \text{where } q_i = \frac{q'}{l_{i,initial}}, \forall\ i = 1, \ldots, m. \end{aligned} \tag{3.3}$$

3.4 Methods to evaluate grid shell designs

The final and arguably most important step is to evaluate and compare the form-found grid shells (Appendix A.7). The objective of this stage is not to produce the most comprehensive structural analysis, but rather to present the designer with relative performance metrics which can serve as useful aides in the preliminary form finding process. Further design refinement, structural analysis and member sizing can then be conducted, but are outside of the scope of this body of work. Instead, simplicity and speed are preferred over detailed, time consuming computations, which are more useful in later stages of design.

As a result, performance metrics are carefully chosen to yield preliminary insights into important structural design objectives such as structural weight, member sizing and constructability, while also taking advantage of the parameters that are intrinsic to the force density method. Firstly, the sum of the product of the axial member forces, f_i, by their final lengths in the form found structure, l_i, is determined: $\sum_{i=1}^{m}(\mid f_i \mid l_i)$. Since the generated grid shells are funicular, compression-only structures, the internal forces can be assumed to act in the same direction, and thus the equation can be simplified to: $\sum_{i=1}^{m}(f_i l_i)$. This performance metric is referred to as the load path of the grid shell [38] and is a proxy for the relative weight of the structure; lower values constitute a lower material intensity for constructing the grid shell, which is favorable. In addition, the ratio between the maximum member force in the structure, f_{max}, and the minimum member force, f_{min} is considered as a measure the distribution of member forces. Values of $\frac{f_{max}}{fmin}$ by definition are always ≥ 1; lower values correspond to more a more uniform distribution of forces, which could have positive implications in terms of member sizing requirements and constructability.

For each iteration in the parametric study, the results are presented to the designer in the format shown in Figure 3-6 (Appendix A.8). The form-found structures are displayed from four different viewpoints for each of the three considered boundary conditions, and are accompanied by two graphs [10, 39]: 1- a bar chart that plots the internal force, f_i of each member i, and 2- a histogram of internal forces, f. These graphs serve as visual representations of the internal member force distribution. This information, along with the structural performance metrics, equip the designer with the information they need to either repeat the design explorations with a more refined scope, or to select their preferred designs that they may then analyze in further depth.

Furthermore, as a means of ascertaining the performance of the Islamic patterned grid shells relative to more conventional quadrilateral grid shells, a comparison methodology is also encoded into the evaluation process of the computational framework (Appendix A.9). In order to perform this comparison, form finding is initiated on a near-equivalent quadrilateral pattern. Near-equivalence is achieved by first specifying identical boundary condition types, loading conditions, and global pattern dimensions; quadrilateral patterns have more nodes along the outer edge and will therefore always have a higher number of fixities, which could have positive ramifications on their relative performance. Another key condition is that the number of members in the Islamic and quadrilateral patterns are as comparable as possible. This is achieved by setting the number of members in the quadrilateral pattern, m_{quad}, equal to the number of members in the Islamic pattern, m, and calculating the corresponding quadrilateral pattern density upon which the form finding and evaluation procedures are conducted in MATLAB (Equation 3.4). It is also worth noting that since the chosen pattern (Figure 3-3) yields four possible categories of Islamic pattern topologies, each of which corresponds to a different number of members for a given pattern density, four quadrilateral patterns are investigated for each pattern density value, d. Consequently, the comparison between the Islamic and quadrilateral patterns does not allow for a direct one-to-one correspondence, and is therefore represented as a shaded region within the appropriate performance plots.

$$\begin{aligned} m_{quad} = &\quad 2d_{quad}(d_{quad}+1) = m \\ \therefore &\quad 2d_{quad}^2 + 2d_{quad} - m = 0 \\ d_{quad} = &\quad \text{round}\left(min\left(\left|\frac{-2 \pm \sqrt{4+8m}}{4}\right|\right)\right) \end{aligned} \tag{3.4}$$

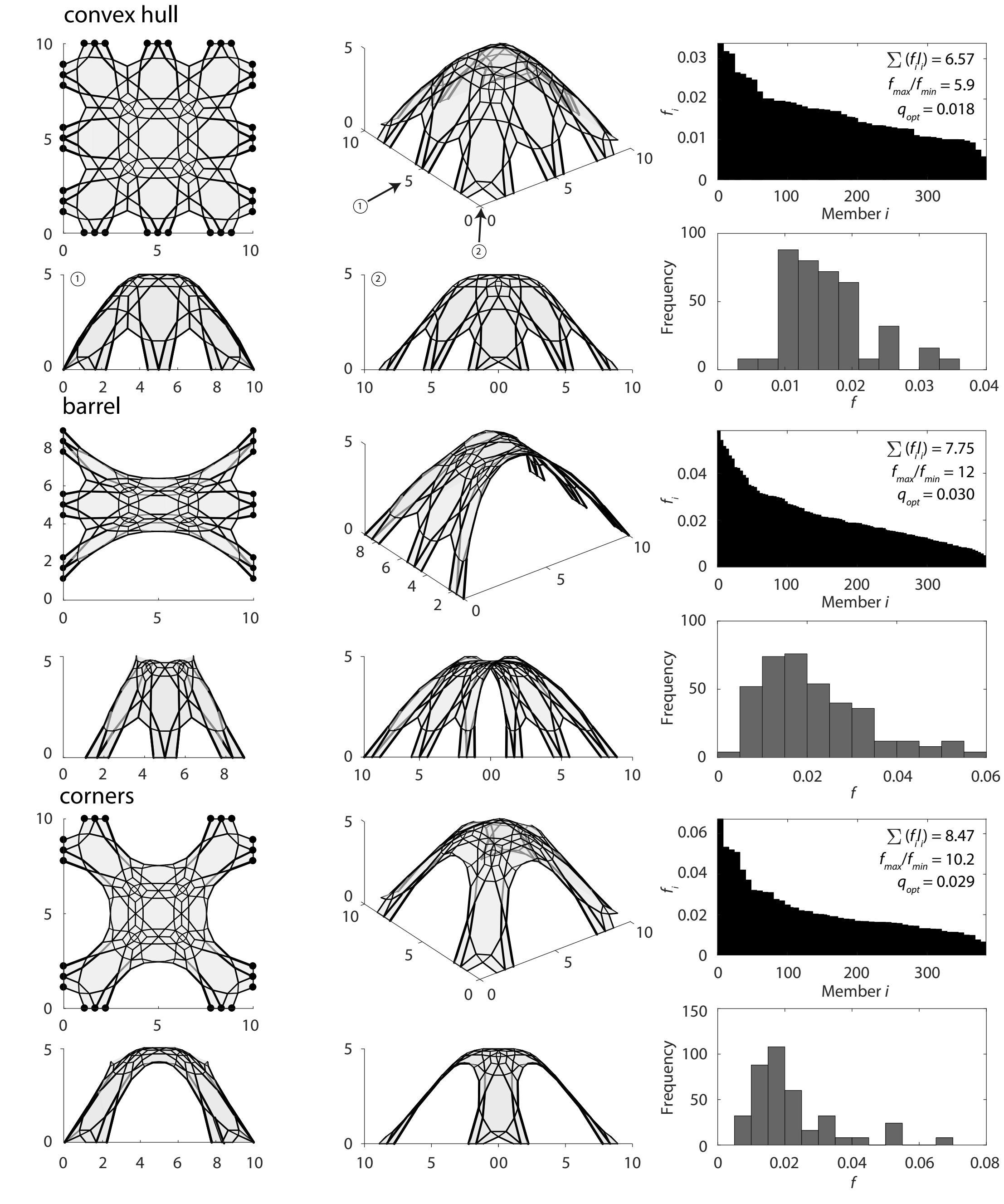

Figure 3-6: Sample output results for all three boundary conditions for a single pattern topology. Here, the results are shown for the parameterization where $d = 3, o_1 = 0.5$, and $o_2 = 0.5$ for the implementation in phase one. The form found structures are shown in plan view, perspective view, and in two section viewpoints indicated by the numbered arrows. The results of each boundary condition are accompanied by a bar chart that plots the internal force, f_i of each member i, in addition to a histogram of internal forces, f. The performance metrics are also shown alongside the optimal applied force density, q_{opt}.

Chapter 4

Results

The primary research contributions stem from an expansion of the grid shell design vocabulary via topological and geometric studies of patterns that are at the intersection of history, culture and mathematics. This chapter shows the outcomes of the grid shell case study and demonstrates the resulting design diversity. In addition, findings from the two design phases are outlined, and the broader implications that the work may have within architecture and structural engineering practices are discussed.

4.1 Phase one: Pattern topology

Given that the application of the force density method in phase one constrains the form finding results to the four topology categories in Figure 3-3 (b) without geometric effects, data from this parametric study is obtained rather quickly. For this reason, the results presented in this section are an appropriate starting point for early stage grid shell design explorations. The generated forms, accompanied by the corresponding performance metrics, therefore serve as a benchmark for the more detailed geometrically-driven investigations in phase two, here presented in sequence.

This first design investigation entails conducting a parametric study of the grid shell design space for the four aforementioned topologies. Grid shells forms are obtained for each of the three boundary conditions, for target heights within the range $0.1\sqrt{a} \leq h_{target} \leq \sqrt{a}$, and for pattern densities $2 \leq d \leq 10$. Since the sensitivity to pattern geometry is concealed once the force density method is applied with equal member force densities, the exact choice of values for the offset parameters when $0 < o_1 \leq 1$ or $0 < o_2 \leq 1$ is arbitrary; for simplicity, $o_1 = 0.5$ and $o_2 = 0.5$ are chosen. In total, $1,080$ distinct grid shells are studied in this phase.

A subset of form-found structures derived within this phase is shown in Figure 4-1. The results are presented for $h_{target} = 5$ in the form of a grid, in order to demonstrate the visual and aesthetic effects of varying the pattern density, as well as the boundary conditions and category of pattern topology. These grid shells are also evaluated relative to one another, with respect to the chosen performance metrics: $\Sigma(f_i l_i)$ and $\frac{f_{max}}{f_{min}}$. Figure 4-2 plots structural performance, as measured by $\Sigma(f_i l_i)$, versus the height, h_{target}, for the four topology categories for the convex hull boundary condition. The results corresponding to the barrel and corners boundary conditions are shown in Figures 4-3 and 4-4, respectively. Upon examining the data through the topology lens, comparable grid shell performance is achieved, however, the $o_1 = 0.5$, $o_2 = 0$ topology appears to correspond to a slightly lower structural weight and is hardly affected by pattern density values. This could be explained by the fact that the offset in the Islamic pattern creates additional load paths for the applied load to be transferred, resulting with lower internal forces in each member. By the same rationale, higher pattern densities tend to correspond to slightly improved structural performance as compared to their counterparts with fewer members. Another interesting finding is that optimal values for the load path are achieved when the height of the grid shell is approximately half of the width of the square plan ($0.5\sqrt{a} = 5$). It is important to note, however, that while trends are indeed observed in d and h_{target}, they are not strongly correlated with the structural performance. Unless a grid shell is particularly shallow, or has a very sparse pattern, the structural performance, as dictated by the load path, is generally comparable, meaning that the designer has the creative agency to control the aesthetic qualities of their design without having a negative impact on the structural performance.

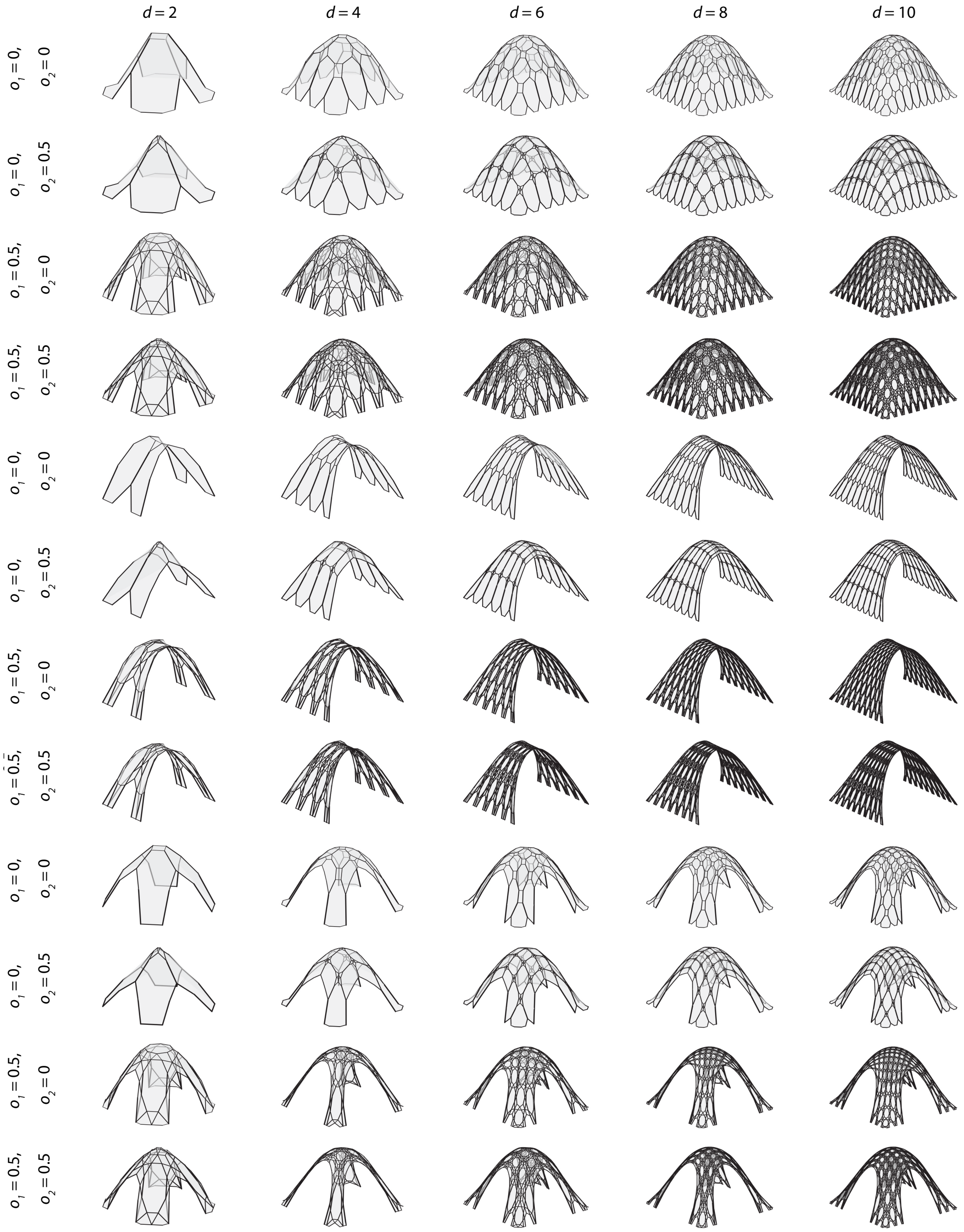

Figure 4-1: Phase one: A subset of form-found structures that corresponds to the results in Figures 4-2 to 4-7. The grid shells are presented for $d = 2$ to $d = 10$ in increments of 2, for each of the four possible topology categories shown in Figure 3-3 (b). The first four rows correspond to the convex hull boundary condition, and are followed by barrel and corners, consecutively.

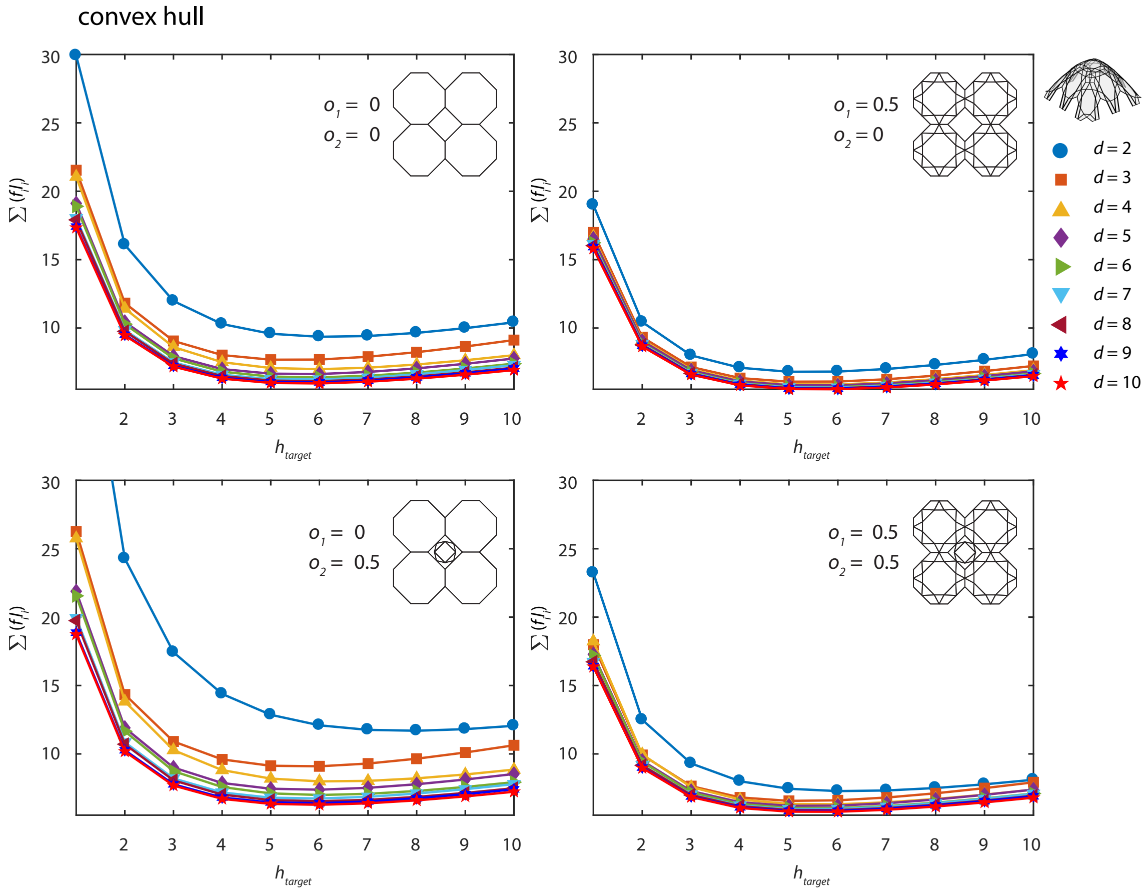

Figure 4-2: Phase one: Plots of structural performance, as measured by $\Sigma(f_i l_i)$, versus the height, h_{target}, of the grid shell for each $o_1 - o_2$ combination for the convex hull boundary condition. Each data set corresponds to a given pattern density, d. Considering all four topologies, the approximate optimal grid shell height is equivalent to half the width of the square plan ($h_{target} \approx 5$). Lower pattern densities correspond to slightly poorer performance, while the metrics are comparable across the four topologies. Structures with $o_1 = 0.5$, $o_2 = 0$ are marginally more efficient.

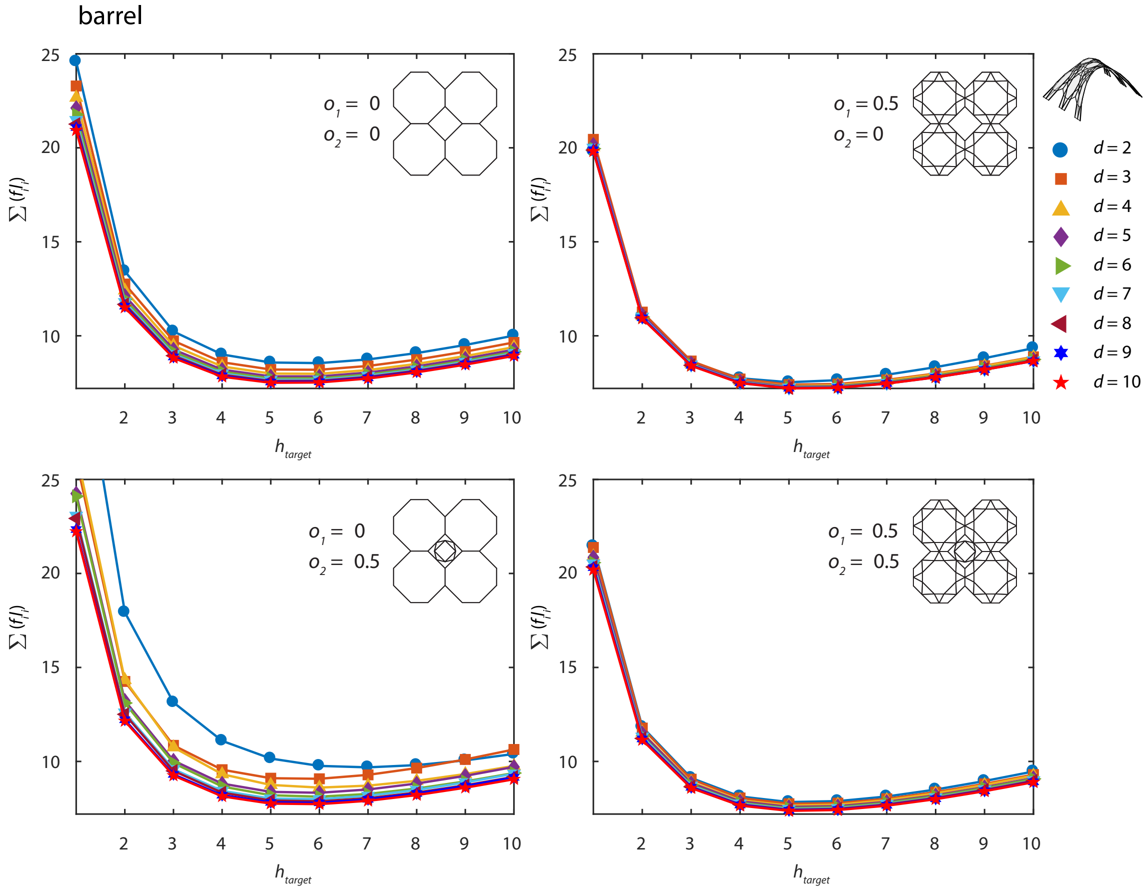

Figure 4-3: Phase one: Plots of structural performance, as measured by $\Sigma(f_i l_i)$, versus the height, h_{target}, of the grid shell for each $o_1 - o_2$ combination for the barrel boundary condition. Each data set corresponds to a given pattern density, d. The approximate optimal grid shell height is equivalent to half the width of the square plan ($h_{target} \approx 5$). Pattern density does not have a significant effect on the performance, even when the pattern is fairly sparse (e.g. $d = 2$).

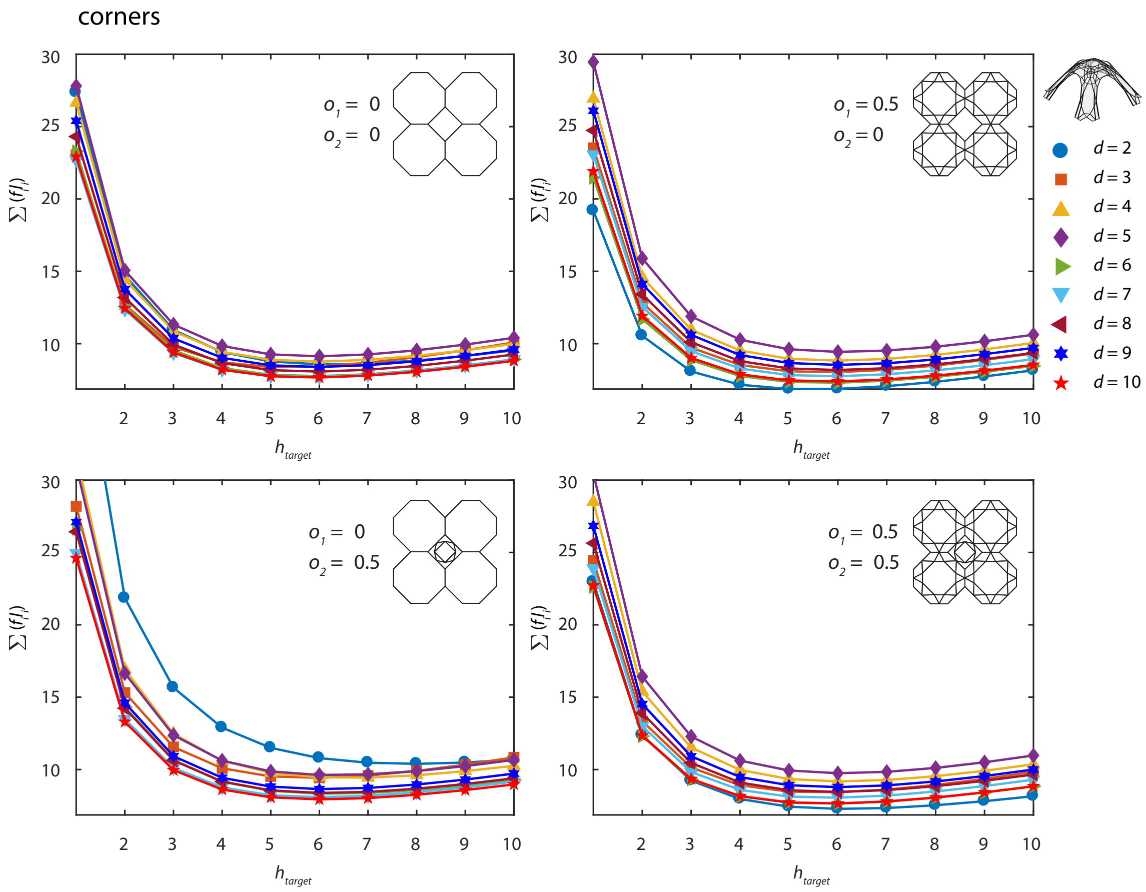

Figure 4-4: Phase one: Plots of structural performance, as measured by $\Sigma(f_i l_i)$, versus the height, h_{target}, of the grid shell for each $o_1 - o_2$ combination for the corners boundary condition. Each data set corresponds to a given pattern density, d. Again, the optimal height of the structure is approximately half the width of the square plan.

Coupling these observed trends with the data in Figures 4-5, 4-6 and 4-7 allows the designer to gain a more holistic overview of the grid shell performance. Now, the four topologies exhibit varying degrees of performances per the $\frac{f_{max}}{f_{min}}$ metric. In some cases, the addition of a non-zero o_1 or o_2 offset creates a larger distribution of forces, which would correspond to a wider variety of member sizes, and perhaps a less efficient structure with more complex construction procedures and connection details. Moreover, the performance is now worsened as the pattern is densified for some of the topologies and as the grid shell height increases, which opposes the findings in Figure 4-2. Combining the trends extracted from the two metrics reveals that perhaps the $o_1 = 0.5$, $o_2 = 0$ topology performs best for the convex hull boundary condition case, albeit marginally so. Overall, the results prove that the pattern parameters do not have a very strong influence on the grid shell performance, particularly with respect to the load path, and therefore provide the designer with the intended creative freedom that fuels this work.

Shifting perspectives to the design of a grid shell of prescribed height, $h_{target} = 5$, Figure 4-8 plots the structural performance for the four combinations of topology parameters for all three considered boundary conditions. Similar trends are observed to those previously discussed, and the data also reiterates the fact that design tradeoffs can be made between the pattern parameters while still retaining structural integrity and efficiency. This dashboard of results is therefore useful for conceptual design, as non-intuitive behavior, particularly for the barrel and corners boundary conditions, are explicitly revealed and can be taken into account in decision-making processes. Most importantly though, this figure plots the results derived from Islamic patterned grid shells as compared to their near-equivalent quadrilateral counterparts, proving that the proposed structures can perform as well as, if not better than conventional grid shells. In addition, Figure 4-9 compares grid shells generated from both quadrilateral and Islamic patterns, demonstrating that visual diversity and aesthetics can be brought to the fore via culturally significant patterns without having to sacrifice structural performance.

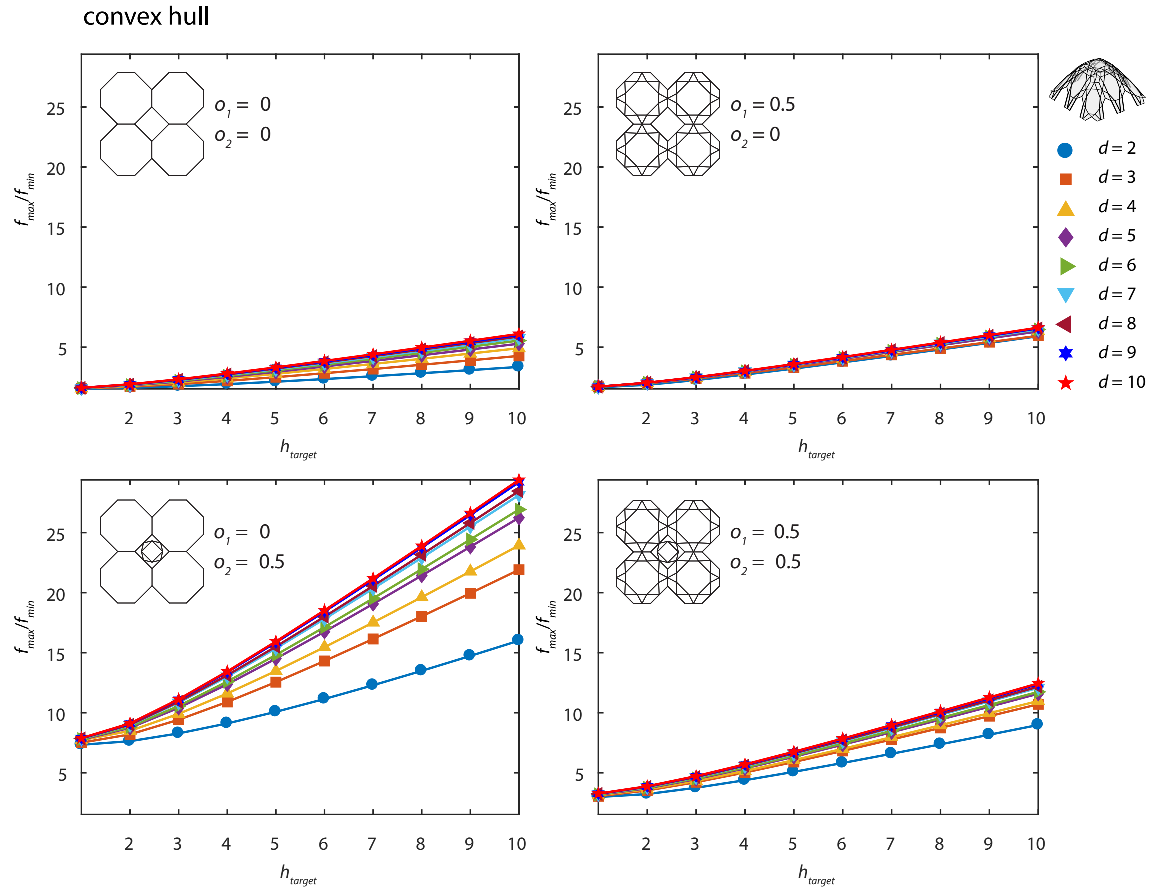

Figure 4-5: Phase one: Plots of structural performance, as measured by $\frac{f_{max}}{f_{min}}$, versus the height, h_{target}, of the grid shell for each $o_1 - o_2$ combination for the convex hull boundary condition. Each data set corresponds to a given pattern density, d. Conversely, lower pattern densities now correspond to slightly improved performance with respect to this metric, while the four topologies exhibit varying levels of performance.

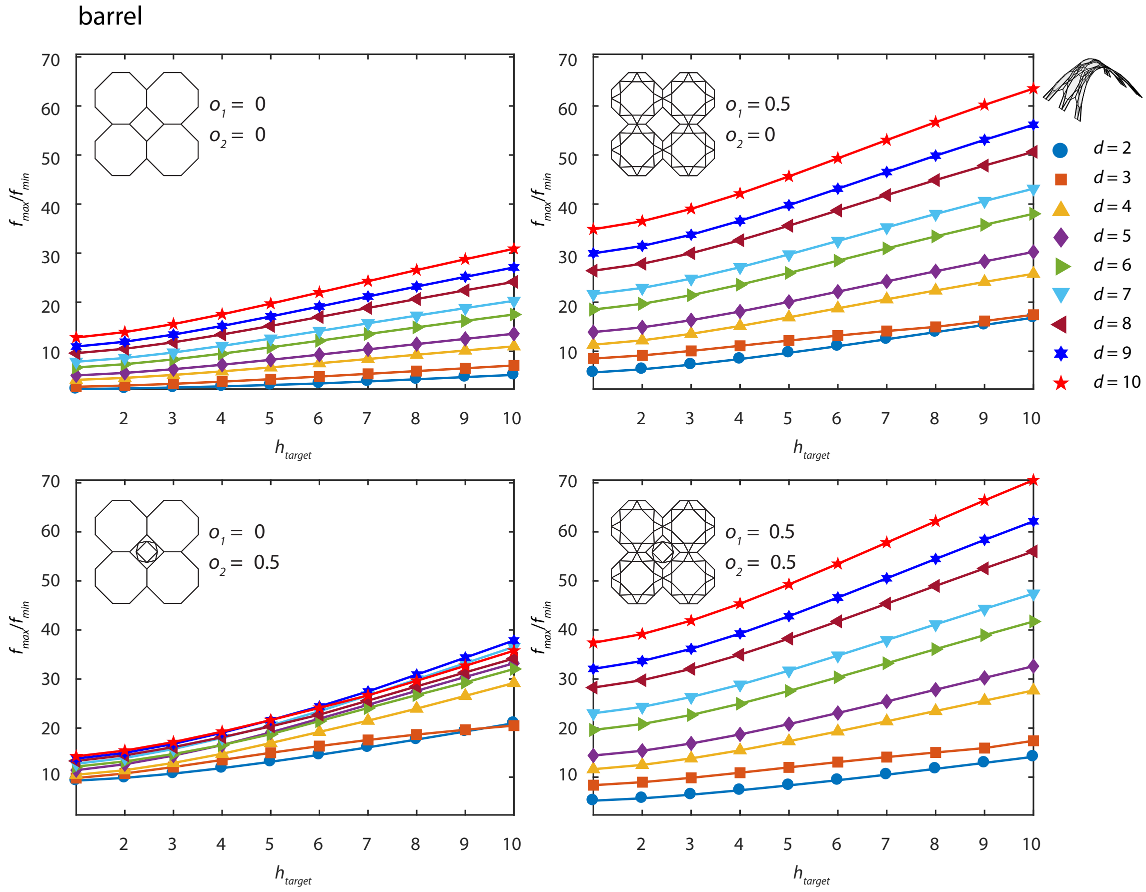

Figure 4-6: Phase one: Plots of structural performance, as measured by $\frac{f_{max}}{f_{min}}$, versus the height, h_{target}, of the grid shell for each $o_1 - o_2$ combination for the barrel boundary condition. Each data set corresponds to a given pattern density, d. The force distribution ratio generally worsens with increasing height and pattern density.

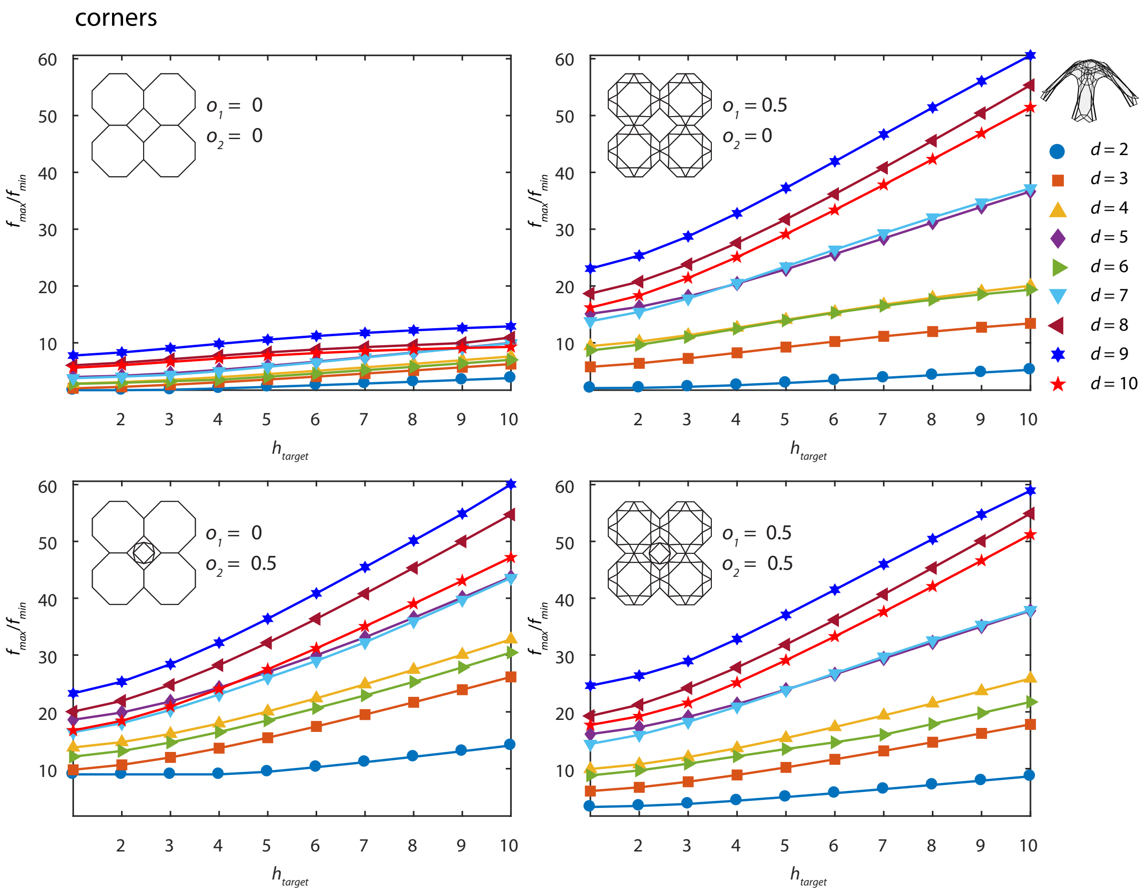

Figure 4-7: Phase one: Plots of structural performance, as measured by $\frac{f_{max}}{f_{min}}$, versus the height, h_{target}, of the grid shell for each $o_1 - o_2$ combination for the corners boundary condition. Each data set corresponds to a given pattern density, d. The force distribution generally worsens with increasing grid shell height and pattern density, but the trend in the pattern density results is less intuitive. Therefore, comparing the results quantitatively, in addition to aesthetically, is key during the conceptual design process.

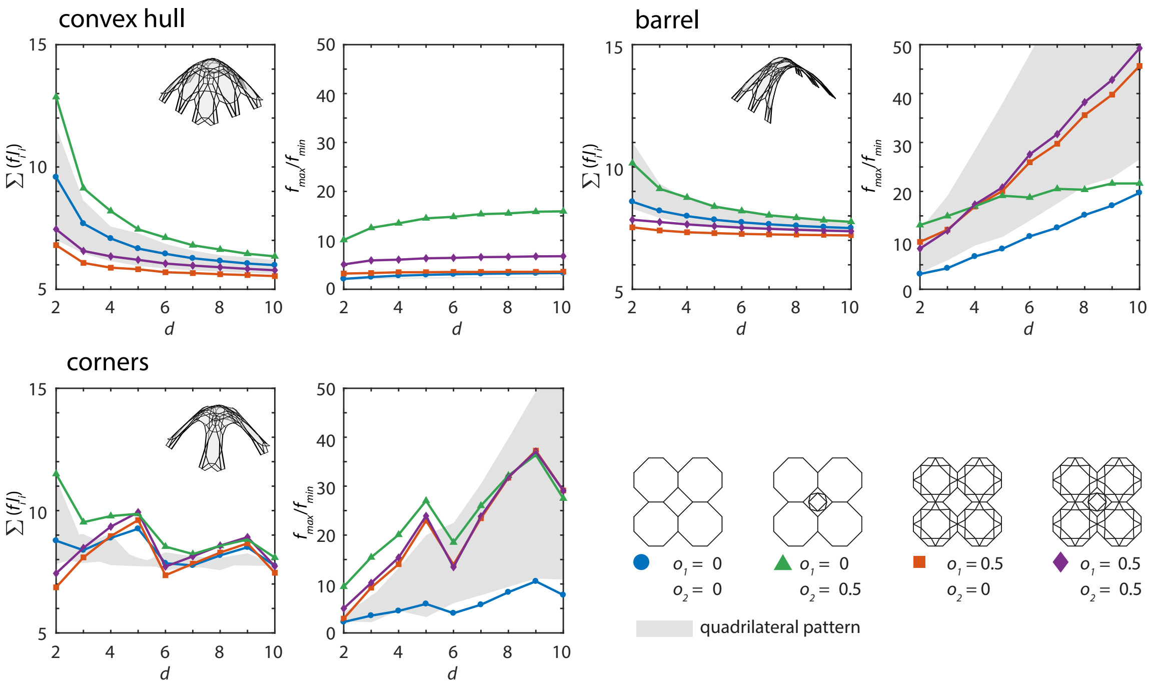

Figure 4-8: Phase one: Plots of structural performance, $\Sigma(f_i l_i)$ and $\frac{f_{max}}{f_{min}}$, versus pattern density, d, for grid shells of height $h_{target} = 5$ for all three boundary conditions. Each data set corresponds to one of the four considered topology categories. The shaded area represents the performance bounds exhibited by the near-equivalent quadrilateral grid shells. The results prove that similar or better performance can be achieved using Islamic patterns. Note that the shaded area for the convex hull $\frac{f_{max}}{f_{min}}$ plot is hardly discernible, as it is approximately a fine horizontal line at $\frac{f_{max}}{f_{min}} = 2$.

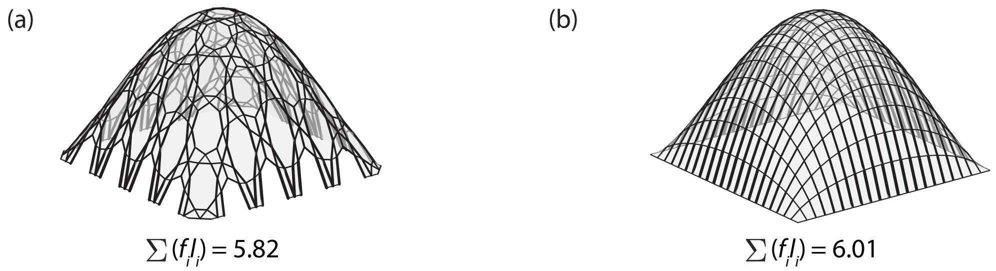

Figure 4-9: (a) Islamic patterned grid shell generated in phase one with the following parameters: $d = 5, o_1 = 0.5$, and $o_2 = 0$ (b) Near-equivalent quadrilateral grid shell with $d_{quad} = 21$. This side by side comparison shows that visual diversity and aesthetic qualities can be prioritized in design without sacrificing structural performance.

4.2 Phase two: Pattern geometry

Phase one examines four Islamic pattern topologies for three boundary conditions and a range of grid shell heights, identifying general performance trends. Building upon this information, the core objective of phase two is to not only offer the luxury of increased structural diversity, but also an opportunity to improve, or at least match, the performance benchmarks defined in phase one through geometric considerations. Once the designer selects a grid shell height and pattern density to work with, o_1 and o_2 can be sampled at a higher resolution as a result of geometrically scaling the member force densities. For the purpose of this discussion, a grid shell of height $h_{target} = 5$ and pattern density $d = 3$ are chosen, and o_1 and o_2 are sampled within the range $0 \leq o_1 \leq 1, 0 \leq o_2 \leq 1$ in increments of 0.2. These chosen constraints yield a total of 108 grid shells, which are studied in this phase. Figure 4-10 demonstrates the

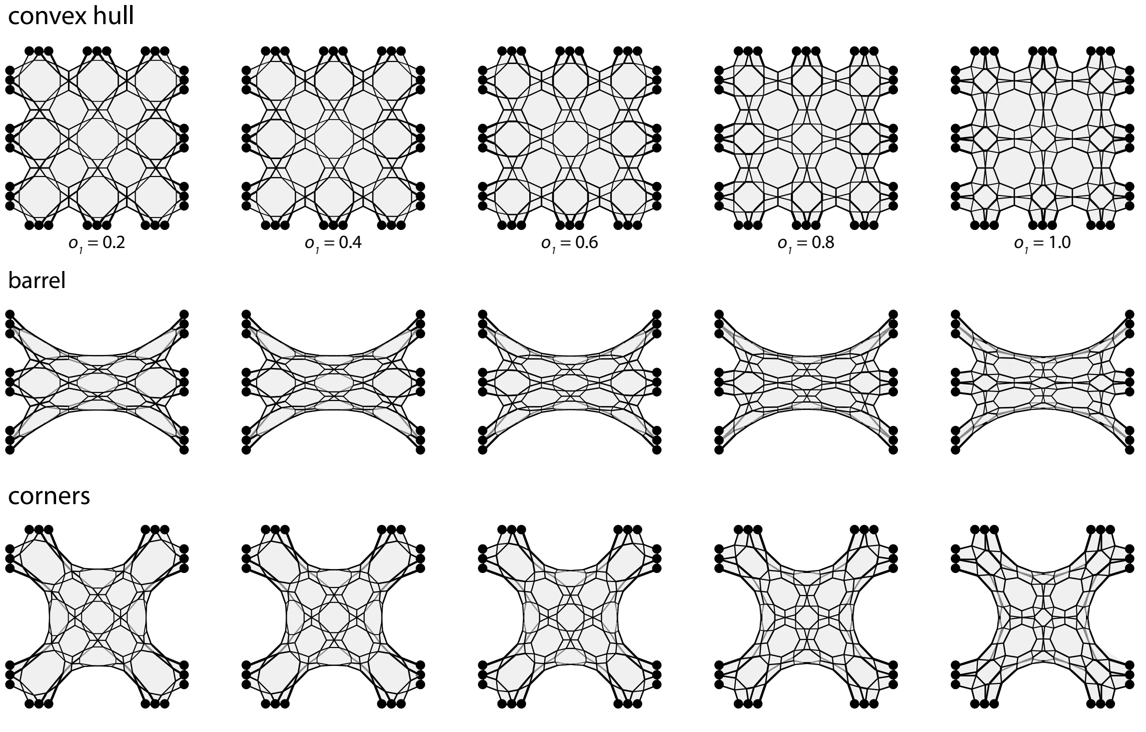

Figure 4-10: Phase two: Plan views of form-found structures are shown for the following parameterization, for each boundary condition: $d = 3$, $o_2 = 0$, and $h_{target} = 5$. The o_1 offset variable is varied from 0.2 to 1 in increments of 0.2. The achieved offset gradient shows that the scaled force densities recover the structural diversity that is lost via the implementation of the force density method in phase one.

visual effects of scaling force densities according to the initial member lengths through a series of structures shown in plan. Here, o_2 is held constant at 0, while o_1 is modified, resulting with smaller open spaces in the grid shell as the offset is increased. Thus, with this implementation, geometric diversity is recovered and is exhibited through a true gradient of offset values. The designer is no longer restricted to four patterns as a result of the limitation intrinsic to the force density method where member force densities are equal throughout, and unlimited grid shell designs can now be uncovered from alterations to the pattern geometry in the $o_1 - o_2$ space. In fact, assuming an increment of 0.2 in o_1 and o_2, if both h_{target} and d were sampled at the same resolution as in phase one, a collection of $9,720$ unique designs would be possible, as compared to their $1,380$ quadrilateral counterparts. Thus, 9 times more grid shell options are made possible with this implementation than in phase one. Three of these additional grid shell designs, where both o_1 and o_2 are varied, are shown in Figure 4-11.

Considering the structural performance of the sampled pattern geometries with respect to the $o_1 - o_2$ space, Figure 4-12 shows that for all three boundary conditions, the load path is hardly affected by the offset values. This means that grid shells with significantly different aesthetic qualities have a comparable structural performance, reinforcing the notion of design freedom. However, this metric does slightly prefer low o_2 values, while $o_1 = 0$ is unfavorable. It is also interesting to note that the stems, representing the results from the original implementation in phase one, are almost on equal par with the scaled raw data points. Therefore, scaling the force densities can indeed meet the performance benchmarks set by the preceding design phase, but may in certain cases worsen the performance. The $\frac{f_{max}}{f_{min}}$ metric, on the other hand, is less predictable since the design space is not flat. Minimum values, and thus the most equally distributed member forces, are found in structures with $o_1 = 0$ and $o_2 = 0$, while $o_1 = 1$ results in poor performance for the barrel and corner boundary conditions. Nonetheless, the combination of options that are encompassed in this design space offer the designer many valuable avenues of exploration in preliminary design investigations by simultaneously prioritizing architectural and structural objectives.

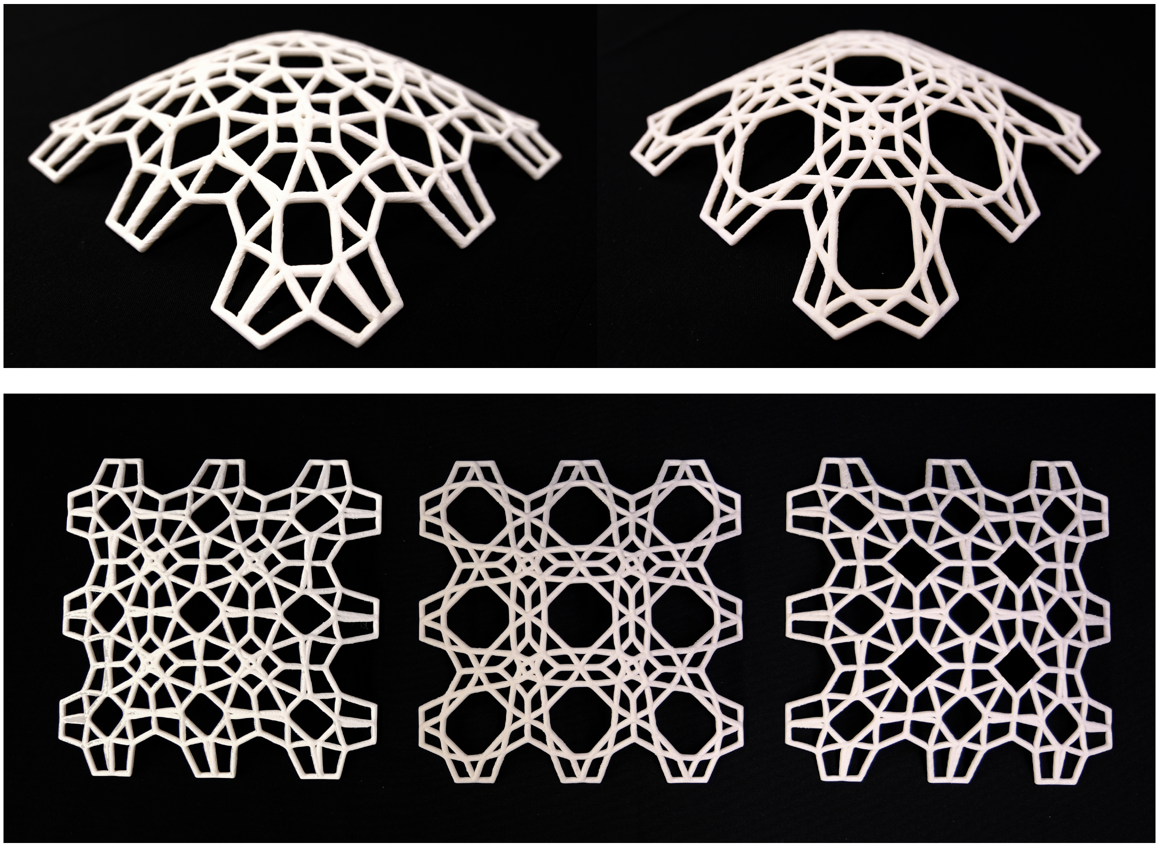

Figure 4-11: Phase two: 3D-printed form-found structures designed using the scaled force density implementation in phase two. The designs were generated for the convex hull boundary condition with $h_{target} = 0.2\sqrt{a}$ and $d = 3$. The following combinations of offset parameters were used: (bottom left) $o_1 = 1, o_2 = 1$, (bottom middle) $o_1 = 0.6, o_2 = 0.8$, and (bottom right) $o_1 = 1, o_2 = 0$. These structures provide additional visualizations to those in Figure 4-10, and show the aesthetic effects when the o_2 offset parameter is also varied.

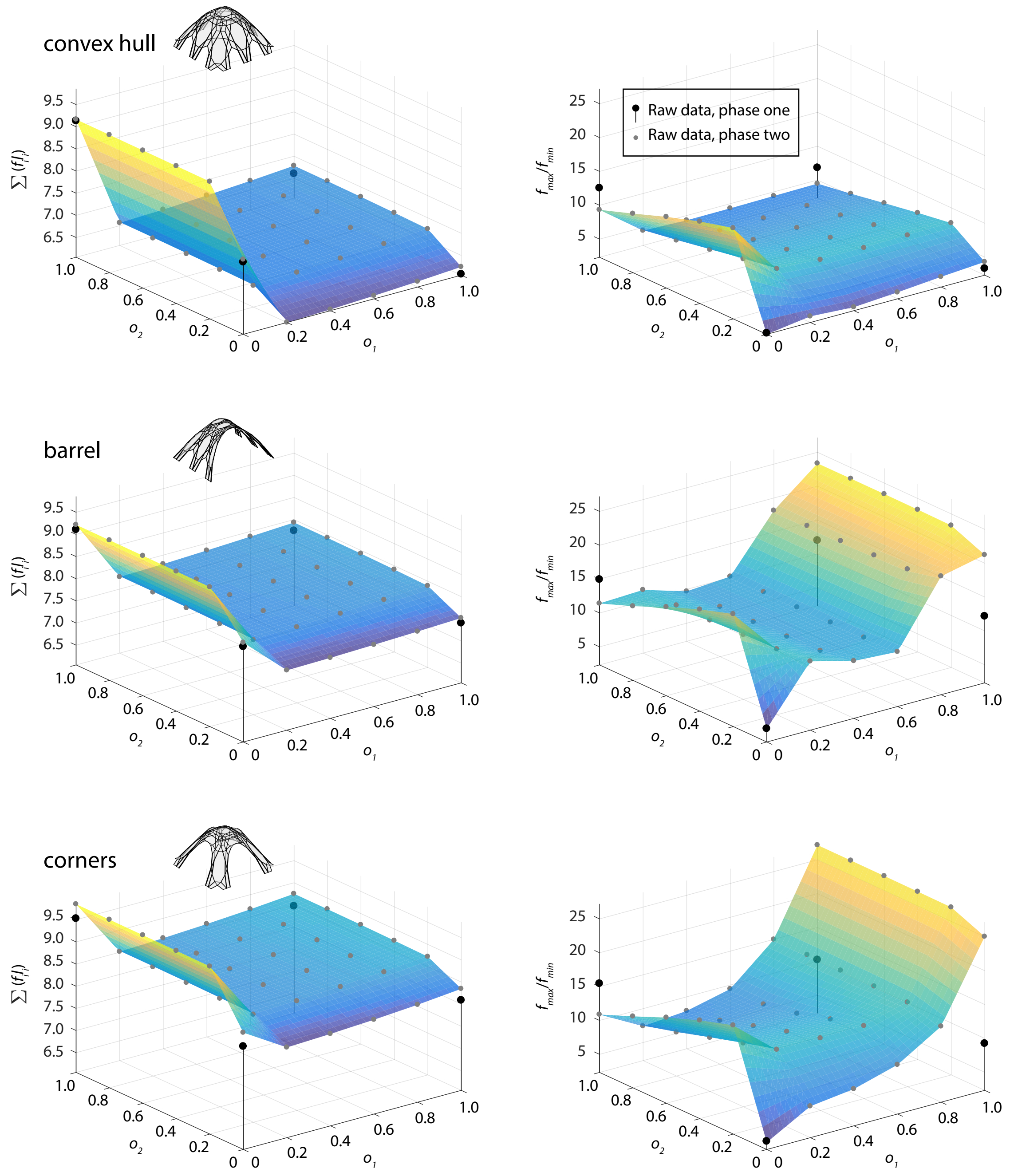

Figure 4-12: Phase two: The results shown here correspond to grid shells where $d = 3$, $h_{target} = 5$, and where both o_1 and o_2 are varied from 0 to 1 in increments of 0.2. Structural performance metrics, $\Sigma(f_i l_i)$ and $\frac{f_{max}}{f_{min}}$, are plot versus the $o_1 - o_2$ space for each boundary condition. Sampled data, for which the force densities are scaled, are shown as points, while the corresponding data from the four topologies (Figure 3-3 (b)) examined in phase one are shown as stems. The three-dimensional surfaces represent the interpolated structural performance for phase two. The $\Sigma(f_i l_i)$ metric prefers low o_2 values across the three boundary conditions, offering more flexibility with modifications in o_1. The $\frac{f_{max}}{f_{min}}$ are less predictable, but minimum values, and thus the most equally distributed member forces, are found in structures with $o_1 = 0$ and $o_2 = 0$. High-performing, yet aesthetically distinct structures are produced in phase two.

Chapter 5

Conclusion

This thesis deeply engages with the mathematical fundamentals of Islamic patterns in order to ascribe a new, structural purpose to designs that historically served a decorative or shading purpose. The patterns are reinterpreted through an engineering lens, and are considered to be an abundant source of topologies and geometries to be used in the design of grid shell structural systems. The high-performing and diverse set of grid shells studied in this work uncovers a new design potential for these patterns, as the patterns are transformed from the delicate ornamental scale to the architectural scale.

This chapter summarizes the key contributions presented in this thesis, while also proposing potential avenues for future work.

5.1 Summary of contributions and potential impact

This research presents a new approach to the conceptual design of funicular grid shells wherein the design space of traditional, culturally meaningful patterns is used to the designer's advantage. Grid shell design is shown to benefit from the introduction of interesting, topologically and geometrically diverse patterns in order to achieve both aesthetic and structural design objectives. In fact, a total of $1,080$ unique grid shells are studied in phase one, where topological effects are isolated, while 108 out of $9,720$ possible designs from phase two are sampled. Thus, the novel geometric force density scaling strategy that is implemented in phase two yields 9 times as many grid shells as in phase one, and also produces a design space that is 7 times more diverse than that of near-equivalent quadrilateral structures.

In addition, the proposed methodology is proven to produce a wide variety of design options that can perform just as well as, or better than conventional quadrilateral grid shells. One such design example is shown as a courtyard roof structure in the concept rendering in Figure 5-1. The resulting expansion of the grid shell vocabulary introduces a new type of creativity into the design process and leaves many design possibilities that are yet to be discovered and constructed in the built environment, and would have not otherwise been considered without the embedded geometric precedents of Islamic patterns. This work therefore encompasses a return to the synergistic approach to architectural and structural design, where aesthetic architectural features and structural systems become more tightly intertwined through geometry.

5.2 Future work and concluding remarks

The very nature of Islamic patterns allows for endless topological and geometrical alternatives to be explored in the context of grid shells, beyond what is demonstrated in this thesis. One conceivable extension of this work could therefore be to implement the proposed methodology for different Islamic patterns, or perhaps even for other cultural or mathematical patterns of the designer's choosing. Moreover, this design approach, emphasizing topology and geometry as applied to grid shells, serves as a facet of a broader outlook on the use of these patterns in modern structural design. Other discrete structural typologies such as braced frames, space frames and trusses, could also benefit from the introduced creative freedom.

Furthermore, the notion of a performance-driven conceptual design methodology could also be further expanded to encompass additional structural criteria, such as local and global buckling constraints, as well as strain energy calculations so as to provide the designer with a more holistic understanding of grid shell performance.

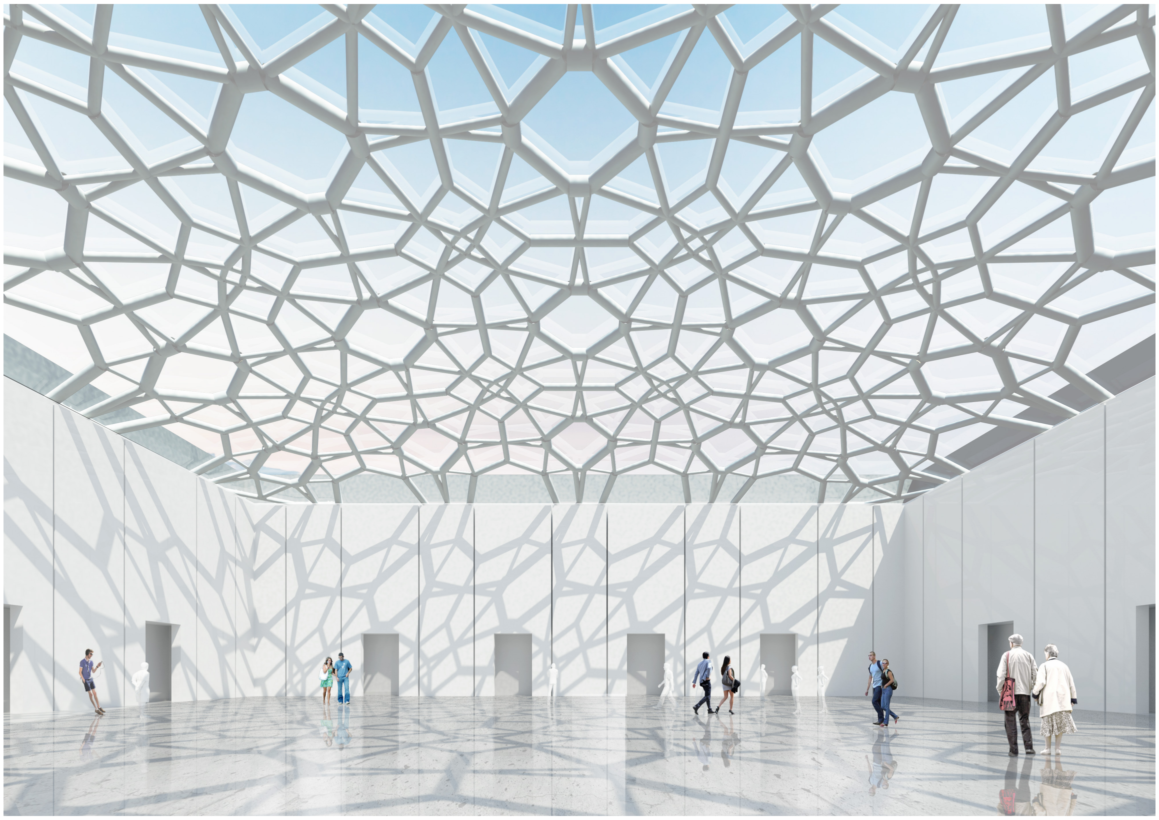

Figure 5-1: Concept rendering of a grid shell roof structure designed in phase two, where the force densities are scaled. The variables used to generate the form are as follows: $d = 5$, $o_1 = 1.0, o_2 = 1.0$, and $h_{target} = 1$. *Rendering produced by Daniel Marshall, MIT.*

Another interesting consideration to build into the evaluation methodology would be a measure for how planar the faces of the grid shell are to account for cladding and constructability issues [2]. In addition, the framework in its current form is only suitable for conceptual design explorations, where speed and quantity of design options are a priority. Therefore, another possible avenue for future work could be to build upon the existing functionality to allow the designer to refine a single grid shell design through several form finding iterations where the loading conditions are updated according to the structure's self-weight and nodal tributary areas.

Perhaps most interestingly though, Islamic patterns open yet another set of topological possibilities through parquet deformations [27], or spatially varying geometric patterns, which could make for an even richer grid shell design space. This directly aligns with the core objectives and contributions of the research presented in this thesis, where topology and geometry present creative design opportunities beyond those derived from topology optimization methodologies [11]. Historic and cultural patterns are shown to produce produce diverse structures that perform well while also responding to aesthetic priorities, thus offering an additional means for designers to reference vernacular traditions in the modern built environment.

Appendix A

MATLAB Code

This appendix contains a collection of ten MATLAB scripts and functions that enable a designer to conduct parametric grid shell design studies on a particular set of patterns. The enclosed code pertains to the latter two stages of the computational framework, the details of which are outlined in Sections 3.3 and 3.4. In order to produce a collection of grid shells, the designer only needs to change the input parameters and run the main script in Appendix A.1, which calls upon the remaining complementary functions. The code is commented, where appropriate, to define the variables and function inputs and outputs.

A.1 Main script

mainIGP.m - Main script to perform form finding using Islamic patterns. This script enables parametric studies of the design space to be conducted, and is the only code that the designer needs to interact with.

```
clear variables; clc;

%% USER INPUTS
saveBoolean = true; % true = saves the images that are produced
dataBoolean = true; % true = writes data to text file
fdmScaleBoolean = true; % true = force densities are scaled (phase ...
    two), false = phase one
quadBoolean = true; % true = comparison to quadrilateral pattern is ...
    conducted

% FDM
totalForce = 1; % total force applied to structure to be divided into...
     point loads
forceDens = 50; % guess value for force density optimization loop - ...
    to be normalised by no. members
minH = 1; % this value will be divided by 10 to compute the minimum ...
    hfactor (% of length of pattern)
maxH = 10; % used to compute the maximum hfactor
BCtype = 1; % 1 = convex hull, 2 = barrel, 3 = corners

% Islamic pattern
% Directory where Grasshopper pattern data is stored
datadir = 'Data_IGP/Data_grasshopper/';

% Writing images and data
% Input the directories of your choosing for data storage; the ...
    folders need to be created and specified prior to running the ...
    code:
imgdir = 'Data_IGP/Data_images/'; % where images of structures are ...
    saved
```

```
resdir = 'Data_IGP/Data_results/'; % where text files of results are ...
    saved
forcedir = 'Data_IGP/Data_forces/'; % where member forces are saved
nodedir = 'Data_IGP/Data_nodes/'; % where pattern topology and node ...
    coordinates are saved
datafilename = [resdir,'parametric_study.txt']; % name of the text ...
    file where data will be written

% -----------------------------------------------------------------
%% COMPUTATION
tic

listing = dir([datadir,'*.csv']); % extracts all .csv files from ...
    directory

poolobj = parpool('local', 8); % sets up parallel pool object; number...
     of workers will depend on your computer

for h = minH:maxH % parametric study for a range of heights

    parfor loopvar = 1:length(listing) % parallel processing

        % Pattern generation
        [gridpts, edges,n,m,constarea,l_initial] = patterngenIGP(...
            datadir, listing(loopvar).name);

        % FDM + PERFORMANCE ANALYSIS
        nodesFind = setBC(gridpts,BCtype);
        nF = length(nodesFind);

        % Point force applied to all free nodes
        p = zeros(n-nF,1); %
        p(:,3) = totalForce/(n-nF); % set z component of force (...
            normalized by number of free nodes)

        % Parsing data
        [pathstr,name,ext] = fileparts(listing(loopvar).name);
        % Parsing the file name output from grasshopper to determine ...
           the system parameters
        % Sample name: (datadir)/dx_2_dy_2_o1_0.5_o2_0.5.csv
        ind = strfind(name, '_');
        % number of polygonal units in x direction (= pattern density...
            in square plan)
        dx = str2double(name(ind(1)+1:ind(2)-1));
        % number of polygonal units in y direction (dx=dy for square)
        dy = str2double(name(ind(3)+1:ind(4)-1));
        o1 = str2double(name(ind(5)+1:ind(6)-1));
        o2 = str2double(name(ind(7)+1:end));

        % Force Density Method + results visualization
        qStart = forceDens/m;

        % Height selection optimization
        hfactor = h/10;
        L = sqrt(constarea); % dimension of square grid.
        targetH = hfactor*L; % target height for the structure
        picname = [name,'_nF_',num2str(nF),'_bc_',num2str(BCtype),'...
           _h_',num2str(targetH)];
        [optq, diff,exitflag,output] = OptFDMHeight(gridpts, edges, ...
           nodesFind, n, m, p, qStart, targetH,l_initial,...
           fdmScaleBoolean);

        % FDM implementation
        if fdmScaleBoolean == false % Phase one: NO SCALING
            [nodes,f,l] = fdm(gridpts, edges, nodesFind, n, m, p, ...
               optq*ones(m,1));
        else % Phase two: SCALED
            [nodes,f,l] = fdm(gridpts, edges, nodesFind, n, m, p,...
               optq./l_initial);
        end

        % Performance analysis
        [sumFL,fMAXmin] = performance(f, l);

        % Plot the form-found structure
        formvis(edges, nodes, f, gridpts, nodesFind, picname,imgdir, ...
           saveBoolean, sumFL, fMAXmin, optq)

        % Comparison to quadrilateral grid
        if quadBoolean == true
            [dquad, sumFLquad,fMAXminquad,optqquad,nodesquad,...
               edgesquad,fquad,nFquad] = quadpattern(totalForce, ...
               forceDens, targetH, BCtype,constarea, m);
        else
            % Initialize quad parameters (for case where quadBoolean ...
               == false
            dquad = []; sumFLquad = []; fMAXminquad = []; optqquad = ...
               []; nodesquad = []; edgesquad = []; fquad = []; ...
               nFquad = [];
        end

        % WRITE DATA TO TEXT FILE
        writeData(dataBoolean,datafilename,forcedir,nodedir,picname,...
           dx,dy,o1,o2,edges,nodes,fMAXmin,sumFL,f,optq,BCtype,...
           quadBoolean,dquad,fMAXminquad,sumFLquad,optqquad,...
           nodesquad,edgesquad,fquad,nFquad);

    end % parfor END

end

delete(poolobj);

toc
```

A.2 Generating Islamic patterns

patterngenIGP.m - This function reads the.csv file that contains the pattern data that is generated in Grasshopper. This serves as the interface between Grasshopper and MATLAB, where the pattern is recreated.

```
%% VARIABLE DEFINITIONS

% INPUTS
% cfile       Name of current pattern .csv file

% OUTPUTS
% gridpts     Matrix of coordinates of all nodes comprising pattern
%             network (x,y,z=0); (nx3)
% edges       (mx2) matrix of pattern topology
%             Each row represents the connectivity of a single member
%             (column 1 = node i index, column 2 = node j index)
% n           Number of nodes in the pattern
% m           Number of members (edges) in the pattern
% constarea   Area of the square plan enclosing the pattern. Length
%             and width of base = sqrt(constarea)
% l_initial   Initial lengths of all of the members in the 2D pattern

% -------------------------------------------------------------------

function [gridpts, edges,n,m,constarea,l_initial] = patterngenIGP(...
    datadir, cfile)

% Sample Grasshopper pattern file name: dx_2_dy_2_o1_0.5_o2_0.5.csv
filename = [datadir, cfile];

% Reading data from Grasshopper .csv output
% .csv file row 1 = x coordinates of pattern nodes
% .csv file row 2 = y coordinates of pattern nodes
% .csv file row 3 = z coordinates of pattern nodes
% .csv file row 4 = x coordinates of member start nodes
% .csv file row 5 = y coordinates of member start nodes
% .csv file row 6 = z coordinates of member start nodes
% .csv file row 7 = x coordinates of member end nodes
% .csv file row 8 = y coordinates of member end nodes
% .csv file row 9 = z coordinates of member end nodes
fid = fopen(filename, 'r');
data = csvread(filename);
fclose(fid);

% Obtaining data from .csv file
gridpts = [transpose(data(1,:)), transpose(data(2,:)), transpose(data...
    (3,:))];
gridpts = gridpts(any(gridpts,2),:); % removes zero rows generated by...
     grasshopper csv write
gridpts = round(gridpts,12); % rounds the coordinate values to 12dp

n = size(gridpts,1);
lineStartpts = [transpose(data(4,:)), transpose(data(5,:)), transpose...
    (data(6,:))];
lineStartpts = round(lineStartpts,12);
m = size(lineStartpts,1);
lineEndpts = [transpose(data(7,:)), transpose(data(8,:)), transpose(...
    data(9,:))];
lineEndpts = round(lineEndpts,12);

constarea = data(10,1);

% Initial lengths of the line segments in the base pattern
l_initial = diag(pdist2(lineStartpts,lineEndpts));

% Initializing edges matrix representing grid topology
edges = zeros(length(lineStartpts),2);

% Constructing edges matrix from start and end coordinates of lines
for i=1:n
    current_gridpts = gridpts(i,:);

    for j=1:m
        current_lineStartpts = lineStartpts(j,:);
        current_lineEndpts = lineEndpts(j,:);

        if isequal(current_gridpts,current_lineStartpts)
            edges(j,1)=i;
        end

        if isequal(current_gridpts,current_lineEndpts)
            edges(j,2)=i;
        end
    end
end
end
```

A.3 Setting boundary conditions

setBC.m - This function automates the selection of three typical boundary conditions in order to standardize the comparisons across grid shells: 1- convex hull, 2- barrel and 3- corners.

```
%% VARIABLE DEFINITIONS

% OUTPUTS
% nodesFind    Column vector of fixed node indices

% ------------------------------------------------------------------

function [nodesFind] = setBC(gridpts,bctype)

xvalues = gridpts(:,1); % isolating x coordinates of pattern nodes
yvalues = gridpts(:,2);

min_x = min(xvalues);
max_x = max(xvalues);
min_y = min(yvalues);
max_y = max(yvalues);

% Convex hull
if bctype==1
    nodesFind = unique(boundary(xvalues,yvalues,0));

% Barrel; fixed along y direction
elseif bctype==2
    nodesminx = find(xvalues==min_x);
    nodesmaxx = find(xvalues==max_x);
    nodesFind = vertcat(nodesminx,nodesmaxx);

% Corners
elseif bctype==3
```

```
    % Convex hull nodes within the radius equivalent to cornerDist ...
        from the corner nodes will be fixed
    cornerDist = (max_x - min_x) * 0.25; % NOTE: this is defined ...
        heuristically
    nodesFind = [];
    convexNodesFind = unique(boundary(xvalues,yvalues,0));
    convexNodes = gridpts(convexNodesFind,:);
    corners = [min_x, min_y, 0; min_x, max_y, 0; max_x, min_y, 0; ...
        max_x, max_y, 0];
    for i = 1:size(corners,1)
        corner = corners(i,:);
        for j = 1:size(convexNodes,1)
            node = convexNodes(j,:);
            if norm(corner-node,2) ≤ cornerDist
                nodesFind = vertcat(nodesFind, convexNodesFind(j));
            end
        end
    end

end
```

A.4 Implementing the force density method

fdm.m - This function applies a standard implementation of the force density method [15]; it determines the three-dimensional form in pure compression corresponding to the input pattern, boundary conditions and loading conditions.

```
%% VARIABLE DEFINITIONS

% INPUTS
% p            Column vector of point forces applied to each free
%              node in the network (n-nf x 1)
% q            Column vector of the force densities of each of the
%              members in the pattern (mx1) (determined from
%              OptFDMHeight)

% -------------------------------------------------------------------

% OUTPUTS
% nodes        Matrix of equilibrium coordinates of all nodes
%              comprising form-found structure (x,y,z); (nx3)
% f            Vector of internal forces in each of the members
%              comprising the structure
% l            Vector of equilibrium lengths of all of the members

% -------------------------------------------------------------------

function [nodes,f,l] = fdm(gridpts, edges, nodesFind, n, m, p, q)

nodesFindex = sort(nodesFind); % sorts fixed nodes indices in ...
    ascending order

% Separating node coordinates into fixed and free
nodesF = gridpts(nodesFindex,:); % fixed node coordinates
xf = nodesF(:,1);
yf = nodesF(:,2);
zf = nodesF(:,3);
```

```
nodesNindex = setdiff(1:1:n, nodesFindex); % indices of free nodes

% Initialize branch node matrix
C = zeros(m, n);

% Fill in branch node matrix with node connectivities
for i=1:m
    startNode = edges(i,1);
    endNode = edges(i,2);

    C(i,startNode)=-1;
    C(i,endNode)=1;

end

% Separate branch node matrix
Cf = C(:,nodesFindex);
Cn = C(:,nodesNindex);

% Force densities
Q = diag(q);

Dn = Cn'*Q*Cn;
Df = Cn'*Q*Cf;

% Solve for coordinates of free nodes
xn = linsolve(Dn, p(:,1) - Df*xf);
yn = linsolve(Dn, p(:,2) - Df*yf);
zn = linsolve(Dn, p(:,3) - Df*zf);

% Solving for element forces
u = Cn*xn+Cf*xf;
v = Cn*yn+Cf*yf;
w = Cn*zn+Cf*zf;

U = diag(u);
```

```matlab
V = diag(v);
W = diag(w);

L = (U.^2 + V.^2 + W.^2).^0.5;
l = diag(L); % column vector of element lengths

f = L*q; % element forces

% Assemble coordinates of free nodes into a single matrix
nodesN = [xn, yn, zn];

% Assemble all node coordinates into a single matrix for plotting ...
    purposes
nodes = zeros(n,3);
nodes(nodesFindex,:) = nodesF;
nodes(nodesNindex,:) = nodesN;
end
```

A.5 Height-selection optimization loop

`OptFDMHeight.m` - This is a wrapper function that calls upon `Objective.m` to perform a constrained minimization wherein the optimal force density value is computed such that the final grid shell height is equal to the designer-prescribed target height.

```
%% VARIABLE DEFINITIONS

% OUTPUTS
% optq       Optimum force density determined according to Objective
%            function. Phase one applies the same force density to
%            each member, while phase two scales the force densities
%            according to the initial member length.
% -------------------------------------------------------------------

function [optq, diff,exitflag,output] = OptFDMHeight(gridpts, edges, ...
   nodesFind, n, m, p, qStart, targetH,l_initial,fdmScaleBoolean)
% c is a temporary cell that contains all of the inputs necessary to ...
   run fdm.m (except for q) when calling Objective.m

c{1} = gridpts;
c{2} = edges;
c{3} = nodesFind;
c{4} = n;
c{5} = m;
c{6} = p;
c{7} = targetH;
c{8} = l_initial;

% Constrained minimization: q≥0
options = optimoptions(@fmincon,'Display','iter','StepTolerance',1e...
   -08);
[optq,diff,exitflag,output] = fmincon(@(q) Objective(q,c,...
   fdmScaleBoolean),qStart,[],[],[],[],0,Inf,[],options);

end
```

A.6 Constrained minimization objective function

`Objective.m` - This function implements the force density method with either the same force density applied to every member in the structure (phase one), or with scaled force densities (phase two). The objective function which defines the height-selection procedure is defined in this code and corresponds to Equations 3.2 and 3.3.

```matlab
function [J] = Objective(q, c,fdmScaleBoolean)
gridpts = c{1};
edges = c{2};
nodesFind = c{3};
n = c{4};
m = c{5};
p = c{6};
targetH = c{7};
l_initial = c{8};

if fdmScaleBoolean == false % Phase one: NO SCALING
    % produces a single optimum q to result with the desired height
    [nodes,f,l] = fdm(gridpts, edges, nodesFind, n, m, p, q*ones(m,1)...
        );
else % Phase two: SCALED
    % scales force densities according to member lengths in intitial ...
        2D pattern
    [nodes,f,l] = fdm(gridpts, edges, nodesFind, n, m, p, q./...
        l_initial);
end

maxHeight = max(nodes(:,3));
J = abs(targetH - maxHeight);
end
```

A.7 Calculating structural performance metrics

`performance.m` - This function evaluates the performance of the form found structure by computing the values for the chosen structural performance metrics.

```
%% VARIABLE DEFINITIONS

% OUTPUTS
% sumFL       Load path of the structure; this is a proxy for the
%             material volume (or weight) of the structure
% fMAXmin     Ratio between the maximum and minimum internal member
%             forces within the structure

% -------------------------------------------------------------------

function [sumFL,fMAXmin] = performance(f, l)

% Load path
sumFL = sum(f.*l);

% Ratio between min and max forces
fMAXmin = max(f)/min(f);

end
```

A.8 Dashboard visualization of grid shell

`formvis.m` - This function plots the three-dimensional form-found structure presented in four viewpoints: 1- plan, 2- perspective, 3- elevation, and 4- section. A force bar chart and force histogram are also shown, and the corresponding performance metrics are included. This code makes use of the `save2pdf.m` function found on the MATLAB File Exchange [40] in order to save higher quality vector images.

```
function formvis(edges, nodes, f, gridpts, nodesFind, picname,imgdir,...
     saveBoolean, sumFL,fMAXmin,optq)
if saveBoolean == true
    % Member forces mapped to line widths for visualization purposes
    minLineW = 1;
    maxLineW = 3;
    [f_lineW,PS] = mapminmax(f',minLineW,maxLineW);

    % Plot axis limits
    maxX = max(nodes(:,1));
    maxY = max(nodes(:,2));
    maxZ = max(nodes(:,3));
    limits = [0 maxX 0 maxY 0 maxZ];

    % Plot title
    sf = 3; % no. significant figures
    % Sample picname: dx_2_dy_2_o1_0.5_o2_0.5_nF_28_bc_1_h_3
    cell = strsplit(picname,'_');
    dx = cell{2}; dy = cell{4}; o1 = str2double(cell{6}); o2 = ...
        str2double(cell{8}); nF = cell{10};
    name = [cell{1},' = ',dx,', ',cell{3},' = ',dy,', ',cell{5},...
        ' = ', num2str(o1,sf),', ',cell{7},' = ',num2str(o2,sf),', ',...
            cell{9},' = ', nF];
    line1 = [name,', q = ',num2str(optq,sf)];
    line2 = ['\Sigma (fl) = ', num2str(sumFL,sf),', f_{max}/f_{min} =...
        ',num2str(fMAXmin,sf)];

```

```
% PARENT FIGURE - to copy to subplot arrangement (different ...
   viewpoints of form-found structure)
fig1 = figure(1);
ax1 = axes('Parent',1);
for i = 1:length(edges)
    connectedNodes = edges(i,:);
    coordinates = nodes(connectedNodes,:);
    plot3(ax1,coordinates(:,1), coordinates(:,2), coordinates...
        (:,3),'k','LineWidth',f_lineW(i));
    hold all;
end
% Shading
DT = delaunayTriangulation(nodes(:,1:2));
faces = trisurf(DT.ConnectivityList, nodes(:,1), nodes(:,2), ...
   nodes(:,3));
set(faces, 'FaceColor', [.9 .9 .9], 'EdgeColor', 'none', '...
   FaceAlpha', .6);

hold off
set(gca,'Color','none');
axis(limits);
axis equal;

% SUBPLOTS
subRow = 2; % number of rows in subplot grid
subCol = 3; % number of columns in subplot grid
fontSize = 20;
titleSize = 30;

fig2 = figure('position',[0,0,2250,750]);

% Plan
ax2 = subplot(subRow,subCol,1);
copyobj(allchild(ax1),ax2);
hold on
scatter(gridpts(nodesFind,1),gridpts(nodesFind,2),'k','filled');
```

```
    view(2);
    set(gca,'Color','none','FontSize', fontSize);
    axis(limits);
    axis equal;
    hold off;

    % Perspective
    ax3 = subplot(subRow,subCol,2);
    copyobj(allchild(ax1),ax3);
    view(3);
    axis(limits);
    set(gca,'Color','none','FontSize', fontSize);
    axis equal;
    hold off;

    % Elevation
    ax4 = subplot(subRow,subCol,4);
    copyobj(allchild(ax1),ax4);
    view([90,0]);
    axis(limits);
    set(gca,'Color','none','FontSize', fontSize);
    axis equal;
    hold off;

    % Section
    ax5 = subplot(subRow,subCol,5);
    copyobj(allchild(ax1),ax5);
    view([-45,0]);
    axis(limits);
    set(gca,'Color','none','FontSize', fontSize);
    axis equal;
    hold off;

    % Plot of all member forces
    ax6 = subplot(subRow,subCol,3);
    bar(ax6,sort(f,'descend'),'FaceColor','k','EdgeColor','k')
    xlabel('Member number','FontSize',fontSize);
    ylabel('Member forces','FontSize',fontSize);
    axis([0 length(f) 0 max(f)]);
    set(gca,'Color','none','FontSize', fontSize);

    % Histogram of force distribution
    ax7 = subplot(subRow,subCol,6);
    histogram(ax7,f,'facecolor','k');
    xlabel('Member forces','FontSize',fontSize);
    ylabel('Frequency','FontSize',fontSize);
    set(gca,'Color','none','FontSize', fontSize);

    % Subplot title
    subTitle = suptitle({line1,line2});
    set(subTitle,'FontSize',titleSize);

    % Directory to save images
    filenamepdf = [imgdir,picname,'.pdf'];
    set(gcf,'renderer','painters');
    save2pdf(filenamepdf,gcf,800);
end
close all;
end
```

A.9 Comparing Islamic patterned grid shells to near-equivalent quadrilateral grid shells

`quadcompare.m` - This function computes the quadrilateral pattern density to satisfy the requirement that the number of members in the pattern is almost equivalent to the number of members in the Islamic pattern. An algorithm, developed to produce general quadrilateral patterns, is then employed to generate the quadrilateral grid upon which the force density method is applied. The corresponding performance metric values are output to facilitate the relative comparisons with the Islamic patterned grid shells.

```
function [dquad, sumFLquad,fMAXminquad,optqquad,edgesquad,nodesquad,...
    fquad,nFquad] = quadpattern(totalForce, forceDens, targetH, ...
    BCtype,constarea, m)
%% GENERATING QUADRILATERAL PATTERN FOR COMPARISON
L = sqrt(constarea);

% Solving for quadrilateral pattern density for near-equivalence to ...
    Islamic pattern
p = [2 2 -m];
r = roots(p);
dx = round(min(abs(r)));
dy = dx;
dquad = dx;

% Scaling square/ rectangle dimensions
ox = L/dx;
oy = ox;

% Initializing vectors
gridpts = zeros(1,3);
edgesquad = zeros(1,2);

base_pts = [0,0,0; 0,oy,0; ox,oy,0; ox,0,0];
```

```

% Stamping square / rectgangle to form quadrilateral patterm
for i = 0:dx-1
    for j = 0:dy-1
        new_shape = [base_pts(:,1)+i*ox, base_pts(:,2)+j*oy, base_pts...
            (:,3)];
        nodeID = [size(gridpts,1);size(gridpts,1)+1;...
            size(gridpts,1)+2;size(gridpts,1)+3];

        % Checking for duplicate nodes
        [Lia,Locb] = ismembertol(new_shape,gridpts,1e-3,'ByRows',true...
            );

        % Removing duplicate points
        for k = 1:length(Locb)
            if Locb(k) == 0 % not a duplicate point
                gridpts = vertcat(gridpts,new_shape(k,:));
                nodeID(k)= size(gridpts,1);
            else % duplicate point - replace node ID with ID of ...
                existing node
                nodeID(k)=Locb(k);
            end
        end

        edges_new = [nodeID(1),nodeID(2);nodeID(1),nodeID(4);...
            nodeID(2),nodeID(3);nodeID(3),nodeID(4)];

        % Sort: Swap columns of edges matrix such that node i < node ...
            j
        for p = 1:size(edges_new,1)
            if edges_new(p,1)> edges_new(p,2)
                edges_new(p,[1,2])=edges_new(p,[2,1]);
            end
        end

        % Checking for duplicate edges
        [Lia_edge,Locb_edge] = ismember(edges_new,edgesquad,'rows');
        for h = 1:length(Locb_edge)
            if Locb_edge(h) == 0 % not a duplicate edge
                edgesquad = vertcat(edgesquad,edges_new(h,:));
            end
        end
    end
end
edgesquad(1,:) = [];

%% FDM FOR QUADRILATERAL PATTERN

nquad = size(gridpts,1);
mquad = size(edgesquad,1);

nodesFind = setBC(gridpts,BCtype);
nFquad = length(nodesFind);

% Point force applied to all free nodes
p = zeros(nquad-nFquad,1);
p(:,3) = totalForce/(nquad-nFquad);

% Force Density Method with height selection
qStart = forceDens/mquad;
[optqquad, diff,exitflag,output] = OptFDMHeight(gridpts, edgesquad, ...
   nodesFind, nquad, mquad, p, qStart, targetH,ox,false);
[nodesquad,fquad,lquad] = fdm(gridpts, edgesquad, nodesFind, nquad, ...
   mquad, p, optqquad*ones(mquad,1));

% Performance analysis
[sumFLquad,fMAXminquad] = performance(fquad, lquad);

end
```

A.10 Writing data from parametric study to text file for post-processing

writeData.m - This function writes the key pattern data, along with grid shell performance metrics, and stores it in a text file for post-processing. The node coordinates of the form-found structure, along with the internal member forces and the pattern topology are also saved as individual.csv files.

```
function writeData(dataBoolean,datafilename,forcedir,nodedir,picname,...
    dx,dy,o1,o2,edges,nodes,fMAXmin,sumFL,f,optq,BCtype,quadBoolean,...
    dquad,fMAXminquad,sumFLquad,optqquad,nodesquad,edgesquad,fquad,...
    nFquad)
if dataBoolean == true

    % Saving the performance metrics for each form-found structure in...
         the same text file
    fid = fopen(datafilename,'at'); % append to end of existing file
    fprintf(fid,'%d\t %d\t %f\t %f\t %f\t %f\t %f\t %f\t %d\t %d\t %f...
        \t %f\t %f\n',dx,dy,o1,o2,max(nodes(:,3)),fMAXmin,sumFL,optq,...
        BCtype,dquad,fMAXminquad,sumFLquad,optqquad);
    fclose(fid);

    % Saving all forces in each form-found structure (Each structure ...
        has its own corresponding .csv file)
    forcefilename = [forcedir,picname,'.csv'];
    csvwrite(forcefilename,f);

    % Saving all deformed node coordinates and topology (Each ...
        structure has its own corresponding .csv file)
    nodefilename = [nodedir,picname,'_nodes.csv'];
    edgefilename = [nodedir,picname,'_edges.csv'];
    csvwrite(nodefilename,nodes);
    csvwrite(edgefilename,edges);

    % Saving the data for the quadrilateral pattern
    if quadBoolean == true
        quadforcefilename = [forcedir,'Quad/',picname,'-quad_d_',...
           num2str(dquad),'_nF_',num2str(nFquad),'.csv'];
        csvwrite(quadforcefilename,fquad);
        quadnodefilename = [nodedir,'Quad/',picname,'-quad_d_',...
           num2str(dquad),'_nF_',num2str(nFquad),'_nodes.csv'];
        quadedgefilename = [nodedir,'Quad/',picname,'-quad_d_',...
           num2str(dquad),'_nF_',num2str(nFquad),'_edges.csv'];
        csvwrite(quadnodefilename,nodesquad);
        csvwrite(quadedgefilename,edgesquad);
    end

end
end
```

Bibliography

[1] C. De Wolf, “Material quantities in building structures and their environmental impact,” Master’s thesis, Massachusetts Institute of Technology, 2014.

[2] S. Adriaenssens, L. Ney, E. Bodarwe, and C. Williams, “Finding the form of an irregular meshed steel and glass shell based on construction constraints.,” *Journal of Architectural Engineering*, vol. 18, no. 3, pp. 206 –213, 2012.

[3] H. Isler, *New shapes for shells.* International Association for Shell Structures, 1960.

[4] S. Huerta, “Structural design in the work of Gaudí,” *Architectural Science Review*, vol. 49, no. 4, pp. 324–339, 2006.

[5] F. Otto and B. Rasch, *Frei Otto, Bodo Rasch : finding form : towards an architecture of the minimal : the Werkbund shows Frei Otto, Frei Otto shows Bodo Rasch : exhibition in the Villa Stuck, Munich, on the occasion of the award of the 1992 Deutscher Werkbund Bayern prize to Frei Otto and Bodo Rasch.* Axel Menges, 1995.

[6] D. Veenendaal and P. Block, “An overview and comparison of structural form finding methods for general networks,” *International Journal of Solids and Structures*, vol. 49, no. 26, pp. 3741 –3753, 2012.

[7] A. Stawarz, “The Great Court in the British Museum,” 2010. Creative Commons Attribution-NoDerivs 2.0 Generic (CC BY-ND 2.0), [Online; accessed April 28, 2017].

[8] Flickr user: kaysgeog, “Amsterdam: Het Scheepvaartmuseum,” 2015. Creative Commons Attribution-NonCommercial-NoDerivs 2.0 Generic (CC BY-NC-ND 2.0), [Online; accessed April 28, 2017].

[9] S. R. Malek, *The effect of geometry and topology on the mechanics of grid shells.* PhD thesis, Massachusetts Institute of Technology, 2012.

[10] K. Verbeeck, L. Muller, and L. D. Laet, “Structural design by a dashboard approach,” in *Future Visions: Proceedings of the IASS Annual Symposium*, IASS, 2015.

[11] J. N. Richardson, S. Adriaenssens, R. F. Coelho, and P. Bouillard, “Coupled form-finding and grid optimization approach for single layer grid shells,” *Engineering Structures*, vol. 52, pp. 230 –239, 2013.

[12] S. Adriaenssens, P. Block, D. Veenendaal, and C. Williams, eds., *Shell Structures for Architecture: Form Finding and Optimization.* London: Taylor & Francis - Routledge, April 2014.

[13] E. Ramm, *Shape Finding Methods of Shells*, pp. 269–283. Vienna: Springer Vienna, 1992.

[14] A. Kilian, “Steering of form,” in *Shell structures for architecture : Form Finding and Optimization.* (S. Adriaenssens, P. Block, D. Veenendaal, and C. Williams, eds.), ch. 11, pp. 131–139, London and New York: Routledge - Taylor & Francis Group, 2014.

[15] H.-J. Schek, “The force density method for form finding and computation of general networks,” *Computer Methods in Applied Mechanics and Engineering*, vol. 3, no. 1, pp. 115 –134, 1974.

[16] E. Broug, *Islamic geometric patterns.* Thames & Hudson London, 2008.

[17] R. Kana’an, “Architectural decoration in Islam: History and techniques,” in *Encyclopaedia of the History of Science, Technology, and Medicine in Non-Western Cultures* (H. Selin, ed.), pp. 187–199, Springer Netherlands, 2008.

[18] J. Bourgoin, *Les Éléments de l’Art Arabe. Le trait des entrelacs.* Paris: Firmin-Didot et cie, 1879.

[19] L. M. Dabbour, “Geometric proportions: The underlying structure of design process for Islamic geometric patterns,” *Frontiers of Architectural Research*, vol. 1, no. 4, pp. 380 –391, 2012.

[20] E. Broug, *Islamic geometric design.* New York : Thames & Hudson Ltd., 2013., 2013.

[21] J. Goury, O. Jones, and P. d. Gayangos, *Plans, elevations, sections, and details of the Alhambra : from drawings taken on the spot in 1834 by Jules Goury, and in 1834 and 1837 by Owen Jones.* London : Published by Owen Jones : Vizetelly Brothers and Co., printers, MDCCCXLII-MDCCCXLV [1842-1845], 1842.

[22] E. P. d’Avennes, *L’Art arabe d’après les monuments du Kaire.* A. Morel et Cie, Libraires-Éditeurs, 1877.

[23] J. Bourgoin, *Arabic geometrical pattern and design.* Dover pictorial archive series, New York, Dover Publications [1973], 1973.

[24] E. H. Hankin, *The drawing of geometric patterns in Saracenic art.* Memoirs of the Archaeological Survey of India: no. 15, New Delhi : Archaeological Survey of India, 1998., 1998.

[25] B. Grünbaum and G. C. Shephard, "Interlace patterns in Islamic and Moorish art," *Leonardo*, vol. 25, no. 3/4, pp. 331–339, 1992.

[26] S. J. Abas and A. Salman, "Geometric and group-theoretic methods for computer graphic studies of Islamic symmetric patterns," *Computer Graphics Forum*, vol. 11, no. 1, pp. 43–53, 1992.

[27] C. S. Kaplan, *Computer graphics and geometric ornamental design*. PhD thesis, University of Washington, 2002.

[28] T. Yazar and S. Uysal, *Grasshopper İle Parametrik Modelleme*. Pusula Yayıncılık, 2016.

[29] T. Yazar, "Islamic patterns of semi-regular tessellations," 2012. [Online; accessed September 30, 2015].

[30] Y. Dold-Samplonius, "Practical Arabic mathematics: Measuring the Muqarnas by al-Kāshī," *Centaurus*, vol. 35, no. 3, pp. 193–242, 1992.

[31] N. Emami, A. Khodadadi, and P. V. Buelow, "Design of shading screens inspired by Persian geometric patterns: An integrated structural and daylighting performance evaluation," in *Shells, Membranes and Spatial Structures: Footprints: Proceedings of the IASS-SLTE Symposium* (R. M. L. R. F. Brasil and R. M. O. Pauletti, eds.), 2014.

[32] M. Lewinski, "Jali screens," 2009. Creative Commons Attribution-NonCommercial-ShareAlike 2.0 Generic (CC BY-NC-SA 2.0), [Online; accessed April 28, 2017].

[33] Flickr user: Funky Chickens, "Muqarnas style ceiling," 2006. Creative Commons Attribution-NonCommercial-NoDerivs 2.0 Generic (CC BY-NC-ND 2.0), [Online; accessed April 28, 2017].

[34] Flickr user: Inhabitat, "Al Bahr Towers by AEDAS," 2014. Creative Commons Attribution-NonCommercial-NoDerivs 2.0 Generic (CC BY-NC-ND 2.0), [Online; accessed April 28, 2017].

[35] J.-P. Dalbéra, "La bibliothèque de l'Institut du Monde Arabe la nuit (Paris)," 2017. Creative Commons Attribution 2.0 Generic (CC BY 2.0), [Online; accessed April 28, 2017].

[36] K. Linkwitz, "Force density method - design of a timber shell," in *Shell structures for architecture : Form Finding and Optimization.* (S. Adriaenssens, P. Block, D. Veenendaal, and C. Williams, eds.), ch. 6, pp. 59–69, London and New York: Routledge - Taylor & Francis Group, 2014.

[37] N. Brown, J. F. de Oliveira, J. Ochsendorf, and C. Mueller, "Early-stage integration of architectural and structural performance in a parametric multi-objective design tool," in *Proceedings of the International Conference on Structures and Architecture*, 2015.

[38] W. F. Baker, L. L. Beghini, A. Mazurek, J. Carrion, and A. Beghini, "Structural innovation: Combining classic theories with new technologies," *Engineering Journal - American Institute of Steel Construction*, vol. 52, no. 3, pp. 203–217, 2015.

[39] L. Loos, K. Verbeeck, and L. D. Laet, "Structurally informed decision-making by means of data visualisations during the conceptual design phase," in *Spatial Structures in the 21st Century: Proceedings of the IASS Annual Symposium* (K. Kawaguchi, M. Ohsaki, and T. Takeuchi, eds.), IASS, 2016.

[40] G. Hoffmann, "save2pdf," 2008. [Online; accessed October 23, 2016].