EQUIVALENT ELASTIC PROPERTIES OF LATTICES USING NON-UNIFORMLY DISTRIBUTED POINT CLOUD

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

ENGİN EGE YORMAZ

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
CIVIL ENGINEERING

NOVEMBER 2022

Approval of the thesis:

EQUIVALENT ELASTIC PROPERTIES OF LATTICES USING NON-UNIFORMLY DISTRIBUTED POINT CLOUD

submitted by **ENGİN EGE YORMAZ** in partial fulfillment of the requirements for the degree of **Master of Science in Civil Engineering**, **Middle East Technical University** by,

Prof. Dr. Halil Kalıpçılar
Dean, Graduate School of **Natural and Applied Sciences** _____

Prof. Dr. Erdem Canbay
Head of the Department, **Civil Engineering, METU** _____

Prof. Dr. Kağan Tuncay
Supervisor, **Civil Engineering, METU** _____

Examining Committee Members:

Prof. Dr. Özgür Kurç
Civil Engineering, METU _____

Prof. Dr. Kağan Tuncay
Civil Engineering, METU _____

Prof. Dr. Yalın Arıcı
Civil Engineering, METU _____

Assoc. Prof. Dr. Alper Aldemir
Civil Engineering, Hacettepe University _____

Assoc. Prof. Dr. Ozan Cem Çelik
Civil Engineering, METU _____

Date: 22.11.2022

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name: Engin Ege, Yormaz

Signature:

ABSTRACT

EQUIVALENT ELASTIC PROPERTIES OF LATTICES USING NON-UNIFORMLY DISTRIBUTED POINT CLOUD

Yormaz, Engin Ege
Master of Science, Civil Engineering
Supervisor: Prof. Dr. Kağan Tuncay

December 2022, 115 pages

A method based on the equivalency of strain energy density is proposed for finding the elastic parameters of non-uniformly distributed lattice models in 2D with a Poisson's ratio of 1/3. Lattice members are assumed to carry only normal force. Hence, the elasticity modulus times the area is the only defining parameter for each lattice member. The method is implemented in MATLAB and it is tested with regularly structured lattices whose solutions can be calculated exactly. A "User-Friendly Lattice Model Program with Graphical User Interface" is developed in MATLAB within the context of this study. The program can deal with complex geometries in 2D. Many examples with different levels of complexity and boundary conditions are solved using the developed program, and solutions are compared with the equivalent finite element solutions. The developed framework is expected to form the foundation for nonlinear analysis of concrete structures as it does not suffer from compression locking reported for structured lattice approaches in earlier studies.

Keywords: Lattice Model, Elastic Properties

ÖZ

DÜZESİZ NOKTA BULUTU İLE OLUŞTURULMUŞ LATİSLERİN EŞDEĞER ELASTİK ÖZELLİKLERİ

Yormaz, Engin Ege
Yüksek Lisans, İnşaat Mühendisliği
Tez Yöneticisi: Prof. Dr. Kağan Tuncay

Aralık 2018, 115 sayfa

2 boyutta ve Poisson oranları 1/3 olan düzensiz şekilde yapılanmış latis modellerinin elastik parametrelerini bulmak için şekil değiştirme enerjisi denkliğine bağlı bir yöntem önerilmiştir. Latis elemanlarının sadece eksenel yük taşıdığı varsayılmıştır. Bu sebeple her eleman için rlastisite modülü ve kesit alanı çarpımı yegâne parametredir. Bu yöntem, MATLAB kullanılarak somutlaştırılmış ve kesin çözümleri bilinen düzenli yapılanmış latisler ile test edilmiştir Bu çalışmanın bağlamında bir "Kullanıcı Dostu Grafiksel Kullanıcı Arayüzlü Latis Programı" MATLAB'da geliştirilmiştir. Bu program 2 boyutta karmaşık geometrileri çözebilmektedir. Farklı karmaşıklıkta ve farklı sınır koşulları olan birçok örnek geliştirilen bu program ile çözülmüştür ve bu çözümler eşdeğer sonlu eleman çözümleri ile karşılaştırılmıştır. Geliştrilmiş olan bu latis modelininin, düzgün yapılanmış latis modellerinde gözlemlenenin aksine sıkışma altında kilitlenmeyip, betonun doğrusal olmayan analizinde bir temel oluşturması beklenmektedir.

Anahtar Kelimeler: Latis Modeli, Elastik Özellikler

To my family.

ACKNOWLEDGMENTS

The author wishes to express his deepest gratitude to his supervisor Prof. Dr. Kağan Tuncay for his guidance, advice, criticism, encouragement and insight throughout the research.

TABLE OF CONTENTS

ABSTRACT v

ÖZ vi

ACKNOWLEDGMENTS viii

TABLE OF CONTENTS ix

LIST OF TABLES xii

LIST OF FIGURES xiii

CHAPTERS

1 INTRODUCTION 1

1.1 What is a Lattice Model? 1

1.2 Why are Lattice Models Used? 1

1.3 The Gap This Study Intends to Fill 2

1.4 Brief Overview of the Proposed Lattice Model 3

2 LITERATURE REVIEW 7

2.1 Early Studies on Lattice Models 7

2.2 Studies with Regular Lattices 9

2.3 Random Lattice Models 18

2.4 Classification of Lattice Models 28

3 METHODOLOGY 29

3.1 Types of Meshes Used 29

3.1.1 Random Mesh: Type 1, Delaunay Mesh 30

3.1.2 Random Mesh: Type 2, Delaunay Mesh with Crossing Elements 31

3.1.3 Random Mesh: Type 3, Constructed Using Proximity with Straight Edges 33

3.1.4 Random Mesh: Type 4, Constructed Using Proximity with Non-Constrained 34

3.1.5 Random Mesh Refinement 35
3.1.6 Regular Mesh 36
3.1.7 Regular Mesh – Rectangles with Two Diagonals 37
3.1.8 Regular Mesh – Triangular Mesh 40
3.1.9 Proposed Method for the Solution of EAs 41
3.2 Solution for The Deformation Field 60
3.3 The User-Friendly Lattice Model Program with Graphical User Interface 60
3.4 Exporting the Lattice Model to SAP2000 68
4 DISCUSSION OF RESULTS 70
4.1 Case Studies 70
4.2 Reference Shell Models 74
4.3 Regular Mesh 76
4.4 Random Mesh: Type 1 78
4.5 Random Mesh: Type 2 81
4.6 Random Mesh: Type 3 84
4.7 Random Mesh: Type 4 91
4.8 Performance Summary of Random Meshes 96
5 CONCLUSION 104
5.1 Summary 104
5.2 Convergence Performance of the Meshes and Solution Methods 105
5.3 Remarks on the User-Friendly Lattice Model Program with Graphical User Interface 105
5.4 Future Studies 106
REFERENCES 107
APPENDICES
A. Pseudocodes For Solving EAs 111

B. Runtime Performance of the LM 115

LIST OF TABLES

TABLES

Table 3.1: Theoretical solutions for cross sectional areas of the lattice elements in a lattice model with structured mesh of rectangles with two diagonals. 38

Table 3.2: Theoretical solutions for cross-sectional areas of the lattice elements in a lattice model with structured mesh – triangular shape. 41

Table 4.1 Theoretical solutions for cross sectional areas of the lattice elements in a lattice model with structured mesh - rectangular with two diagonals. 70

Table 4.2 Theoretical solutions for cross sectional areas of the lattice elements in a lattice model with structured mesh - rectangular with two diagonals. 70

Table 4.3: Summary of random meshes and their convergence performance when solved with different methods. 103

Table B.1: Runtime perfomance of 3x12 axially loaded rod, Random Mesh: Type 3, solved using the hybrid approach. 115

LIST OF FIGURES

FIGURES

Figure 1.1 Overview of the irregular mesh lattice model used. 5
Figure 1.2 Overview of the regular mesh rectangle lattice model used with two diagonals as in Hrennikoff (1940). 6
Figure 2.1 Two of the structured lattices that were presented by Hrennikoff (1940): (a) 'Triangular Pattern' (b) 'Simple Square Pattern with Intersecting Diagonals'… 8
Figure 2.2 Triangular lattice model modelled with stochastic defects. Three different strain conditions are shown; the last figure shows the fully separated lattice (Beale & Srolovitz, 1988). 10
Figure 2.3 (a) Test setup of a concrete beam (SEN Iosipescu type configuration), (b) corresponding regular lattice model of the test setup. (Schlangen & van Mier, 1992) 11
Figure 2.4 (a) Test setup of an embedment bar, (b) regular lattice model of the embedment bar simulating the experiment. Crack pattern is visible in the lattice model (Schlangen & van Mier, 1992)..... 12
Figure 2.5 Crack tip on a teared plate solved with peridynamic method (above) and conventional models (below) (Silling, 2000)..... 13
Figure 2.6 Node connectivities used in the OLM: (a) 8 nodes connectivity for internal nodes, (b) 28 nodes connectivity for internal nodes. (Aydin et al., 2018). 15
Figure 2.7 Damage patterns for uniaxial tension tests that were conducted using OLM with different horizons and solved with sequential linear analysis (Aydin et al., 2018). 16
Figure 2.8 Material curves used in the model and saw-tooth behavior of SLA: (a) steel (b) concrete. (Aydin et al., 2018)..... 17
Figure 2.9. Crack patterns of a concrete gravity dam under varying seismic loads, finite element and OLM comparison (Soysal et al., 2017). 18
Figure 2.10 Homogenized random lattice obtained by Schlangen & Garboczi (1996). Cracked pattern is visualized by darkening elements. 19

Figure 2.11 (a) Aggregates modelled as circles and cement paste in between. (b) Lattice model of a concrete specimen that is modelled with interface element modeling method (Zubelewicz & Bažant, 1987). 20
Figure 2.12 (a) Neighboring aggregates modeled as circles and the cement matrix around. Corresponding lattice element is at the right side. (b) Specimen consisting of randomly distributed aggregates and corresponding lattice truss model. (c) stress-strain curve for the cement matrix bonds. 22
Figure 2.13 Ice crack patterns with structured lattice with no random variation of lattice strength. The angle between structured lattice and impact direction is (a) 0º, (b) 15º, (c) 30º, (d) 45º. 23
Figure 2.14 Ice crack patterns with structured lattice with probabilistic random variation of lattice strength. The angle between structured lattice and impact direction is (a) 0º, (b) 22.5º, (c) 45º. 23
Figure 2.15 Ice crack patterns with random lattice when compression failure of lattices: (a) not allowed, (b) allowed. 24
Figure 2.16 (a) Random mesh as proposed by Jirásek & Bazant (1995). (b) Accompanying stress-strain diagram for lattices. 24
Figure 2.17 Sample mesh showing the lattice mesh and normal spring and shear spring interactions between lattice nodes (G.-F. Zhao, 2014). 25
Figure 2.18 (a) Diagram showing a discretized body into Voronoi cells and node connectivities. (b) Diagram showing the rigid body spring model (RBSM) components: rigid cells and 3 springs at each Voroni boundary (Bolander & Sukumar, 2005). 26
Figure 2.19 Sample regular and random meshes used to model the concrete under compression (Aydin et al., 2021). 27
Figure 2.20. Force vs displacement curve for an over-reinforced concrete beam. Deflections observed using simulation (indicating “random mesh”) and uniform mesh are indicated with lines and experimental observation was given by the grey zone (Aydin et al., 2021). 28

Figure 2.21. Classification of lattice models according to the literature review by Pan et al. (2018). 28
Figure 3.1 Sample Random Mesh: Type 1..... 31
Figure 3.2 Creation of 'crossing elements' in Random Mesh: Type 2. 32
Figure 3.3 Sample beam with Random Mesh: Type 2. (a) Delaunay triangulation plus 'crossing elements' form the lattice elements. (b) Voroni areas calculated for nodes. 32
Figure 3.4 Close-up view of an irregular mesh. Black dots indicate nodes. Bold lines indicate lattice elements. Thin lines and light grey hatch indicate Voronoi areas tributary to each node. 33
Figure 3.5 Random Mesh: Type 3 with a sample node marked with a red dot and its range of connectivity is indicated with a red circle (3.01d). Connectivities in the mesh are drawn with black lines. 34
Figure 3.6 Comparison of (a) Random mesh: Type 3 and (b) Random mesh: Type 4..... 35
Figure 3.7 Graph showing the refinement rules for the refined irregular mesh..... 36
Figure 3.8 (a) Sample regular mesh – rectangles with two diagonals. Black dots indicate node locations. Note that there is no node at the 'intersection' of the two diagonals. (b) Close-up view of the regular mesh – rectangles with two diagonals showing the theoretical cross-sectional areas of the lattice members..... 39
Figure 3.9 (a) Sample regular mesh – triangular mesh. Black dots indicate node locations. Note that the pattern does not repeat at the edges if a rectangular shape is desired, so the theoretical solutions for cross sectional area values are incorrect at the boundaries. 41
Figure 3.10 (a) Sample plate prototype. (b) Corresponding lattice model to represent the behavior of the plate prototype. 42
Figure 3.11 Diagram of a sample piece of lattice showing the parameters used in EA calibration calculations. 44
Figure 3.12 Diagram showing a sample piece of the prototype under consideration tributary to each node..... 45

Figure 3.13 Regular lattice meshed with rectangle with two diagonals, element size is 1 EA values are written besides the elements. Exact EAs are assigned to the elements that are found from analytical methods (Hrennikoff, 1940). 56
Figure 3.14 Diagram showing a 20x20 square regular lattice meshed with rectangle with two diagonals, element size is 1. Lattice elements are solved using the local approach and colored according to their EA values. 56
Figure 3.15 (a) Corner of the lattice that is shown at Figure 3.14 zoomed in. Error values solved for each lattice element is given with numbers. (b) Center of the lattice that is shown at Figure 3.14 zoomed in. Relative error in EA values solved for each lattice element compared to the exact EA values are given with numbers. 57
Figure 3.16 (a) Diagram showing a 20x20 square regular lattice meshed with rectangle with two diagonals, element size is 1. Lattice elements are solved using Jirásek & Bazant (1995) approach and colored according to their EA values. (b) Closeup view of the mesh at the corner.....58
Figure 3.17 (a) Diagram showing a 20x20 square regular lattice meshed with rectangle with two diagonals, element size is 1. Lattice elements are solved using the hybrid approach and colored according to their EA values. (b) Closeup view of the mesh at the corner.....59
Figure 3.18 Close up view of a non-uniform lattice. Average element size is 1. Lattice elements are colored according to their EA values. 60
Figure 3.19 LM user interface showing a sample cantilevered wall-like structure under a lateral force. 61
Figure 3.20 Control panel of LM. User interactive fields and buttons are numbered within red circles. 62
Figure 3.21 Different types of meshes with “Page Size” =15. Left column shows meshes with “Approx. Element Size” = 0.5. Right column shows meshes with “Approx. Element Size” = 0.25.....66
Figure 3.22 Sample lattice model showing the effect of a roller support that is released along the x direction. 67

Figure 3.23 Elements of the lattice model are colored with a gradient of color according to their EAs. 67
Figure 3.24 Lattice model showing the tributary areas obtained by Voroni Tesselation hatched corresponding to each node. 68
Figure 3.25 Deflections of a lattice model of a square shaped structure is solved by the built-in solver of LM in (a) and solved in SAP2000 (b). The displacements are identical. 68
Figure 3.26 Deformed shape of a lattice model of a L-shaped structure is solved by the built-in solver of LM and solved in SAP2000 (b). Deflections of a lattice model of a simply supported beam is solved by the built-in solver of LM (c) and solved in SAP2000 (d). 69
Figure 4.1 Lattice model of the axially loaded rod. 71
Figure 4.2 Lattice model of the cantilevered beam. 71
Figure 4.3 Lattice model of the beam with hole. 72
Figure 4.4 Lattice model of the L shaped structure. 73
Figure 4.5 SAP2000 shell solutions of the sample cantilevered beam. (a) Shell size is 1, (b) Shell size is 1/2, (c) Shell size is 1/4, (d) Shell size is 1/8. 74
Figure 4.6 SAP2000 Shell solutions. (a) Axially loaded rod, shell size is 1/8, (b) Beam with hole, shell size is 1/8, (c) L shaped structure, shell size is 1/8. 75
Figure 4.7 Distribution of a tip load among the nodes for both in the reference shell models and in the lattice models. 75
Figure 4.8 Lattice model of the axially loaded rod, structured mesh. (a) d=1, (b) d=1/2, (c) d=1/4, (d) d=1/8. 76
Figure 4.9 Lattice model of the cantilevered beam, structured mesh. (a) d=1, (b) d=1/2, (c) d=1/4, (d) d=1/8. 76
Figure 4.10 Lattice model of the beam with hole, structured mesh. (a) d=1/2, (b) d=1/4, (c) d=1/8. 77
Figure 4.11 Lattice model of the L shaped structure, structured mesh. (a) d=1, (b) d=1/2, (c) d=1/4, (d) d=1/8. 77

Figure 4.12 Lattice mesh used for the axially loaded rod, and beam with hole Random Mesh: Type 1. (a) d=1, (b) d=1/2, (c) d=1/4, (d) d=1/8. 78
Figure 4.13 Lattice model of beam with hole, Random Mesh: Type 1. (a) d=1, (b) d=1/2, (c) d=1/4. 78
Figure 4.14 Lattice model of the L shaped structure, Random Mesh: Type 1. (a) d=1, (b) d=1/2, (a) d=1/4, (b) d=1/8. 79
Figure 4.15 Axially loaded rod relative errors of lattice model vs shell model, Random Mesh: Type1, solved by assigning EA_{Bazant} to the lattices.. (a) d=1, 100 samples run; (b) d=1/2, 200 samples run; (c) d=1/4, 400 samples run; (d) d=1/8, 800 samples run. 80
Figure 4.16 Lattice model of the axially loaded rod, Random Mesh: Type 2. (a) d=1, (b) d=1/2, (c) d=1/4, (d) d=1/8. 81
Figure 4.17 Lattice model of the cantilevered beam, Random Mesh: Type 2. (a) d=1, (b) d=1/2, (c) d=1/4, (d) d=1/8. 81
Figure 4.18 Lattice model of the beam with hole, Random Mesh: Type 2. (a) d=1/2, (b) d=1/4, (c) d=1/8. 82
Figure 4.19 Lattice model of the L shaped structure, Random Mesh: Type 2. (a) d=1, (b) d=1/2, (c) d=1/4, (d) d=1/8. 82
Figure 4.20 Sample histograms for Axially loaded rod relative errors of lattice model vs shell model, Random Mesh: Type 2, solved by assigning EA_{Bazant} to the lattices. (a) d=1, 100 samples run; (b) d=1/2, 200 samples run; (c) d=1/4, 400 samples run; (d) d=1/8, 800 samples run. 83
Figure 4.21 Lattice model of the axially loaded rod, and cantilevered beam Random Mesh: Type 3. (a) d=1, (b) d=1/2, (c) d=1/4, (d) d=1/8. 84
Figure 4.22 Lattice model of the beam with hole, Random Mesh: Type 3. (a) d=1/2, (b) d=1/4, (c) d=1/8. 84
Figure 4.23 Lattice model of the L shaped structure, Random Mesh: Type 3. (a) d=1, (b) d=1/2, (c) d=1/4, (d) d=1/8. 85
Figure 4.24 Axially loaded rod relative errors of lattice model vs shell model, Random Mesh: Type 3, solved by assigning EA_{Bazant} to the lattices. (a) d=1, 100

samples run; (b) d=1/2, 200 samples run; (c) d=1/4, 400 samples run; (d) d=1/8, 800 samples run. 86
Figure 4.25 Axially loaded rod tip deflections, Random Mesh: Type 3, solved by assigning EA_{Bazant} to the lattices. (a) d=1, (b) d=1/8..... 87
Figure 4.26 Axially loaded rod deflected shape, Random Mesh: Type 3, solved by assigning EA_{Bazant} to the lattices, deflections are exaggerated for visibility. (a) d=1, (b) d=1/8..... 88
Figure 4.27 Histogram of frequency of occurrence of element lengths, Random Mesh: Type 3, solved by assigning EA_{Bazant} to the lattices. (a) d=1, (b) d=1/8. 89
Figure 4.28 Axially loaded rod, x vs u_x plot, Random Mesh: Type 3, solved by assigning EA_{Bazant} to the lattices. (a) d=1, (b) d=1/8..... 90
Figure 4.29 Lattice model of the axially loaded rod and cantilevered beam, Random Mesh: Type 4. (a) d=1, (b) d=1/2, (c) d=1/4, (d) d=1/8. 91
Figure 4.30 Axially loaded rod relative errors of lattice model vs shell model, Random Mesh: Type 4, solved by assigning EA_{Bazant} to the lattices. (a) d=1, 100 samples run; (b) d=1/2, 200 samples run; (c) d=1/4, 400 samples run; (d) d=1/8, 800 samples run. 92
Figure 4.31 Sample axially loaded rod tip deflections, Random Mesh: Type 4, solved by assigning EA_{Bazant} to the lattices. (a) d=1, (b) d=1/8 93
Figure 4.32 Sample axially loaded rod deflected shape, Random Mesh: Type 4, solved by assigning EA_{Bazant} to the lattices, deflections are exaggerated for visibility. (a) d=1, (b) d=1/8..... 94
Figure 4.33 Histogram of frequency of occurrence of element lengths, Random Mesh: Type 4, solved by assigning EA_{Bazant} to the lattices. (a) d=1, (b) d=1/8. 95
Figure 4.34 Axially loaded rod, x vs u_x plot, Random Mesh: Type 4, solved by assigning EA_{Bazant} to the lattices. (a) d=1, (b) d=1/8..... 96
Figure 4.35 Relative deflection errors versus lattice size, axially loaded rod..... 99
Figure 4.36 Relative deflection errors versus lattice size, cantilevered beam. 100
Figure 4.37 Relative deflection errors versus lattice size, beam with hole..... 101
Figure 4.38 Relative deflection errors versus lattice size, L shaped structure..... 102

CHAPTER 1

INTRODUCTION

1.1 What is a Lattice Model?

The lattice model may be explained as a type of analog model for a solid continuum using a point cloud that fills the body. Force-deformation relationships between these points are the material properties that should be chosen appropriately to approximate the behavior of the solid continuum. The extensive literature review on lattice models by Pan et al. (2018) points out that the term 'Lattice' in mathematics generally means a group of points that are positioned with a certain pattern. Thus, they point out that although rigid-body spring networks—due to their definition—may not fall into this category, they are still named lattice models in the literature. An example of a rigid-body spring network is given in Bolander & Sukumar (2005). Therefore, it can be stated that models in which a continuum is modeled by nodes connected with different elements such as trusses, springs, beams, and shear springs can be called lattice models. In addition to these, models in which a continuum is tessellated into rigid cells that interact with the neighboring cells through translational/shear/rotational springs are also regarded as lattice models in the literature (as in Bolander & Sukumar (2005)). It should also be noted that there are different names used in the literature for lattice models. The lattice models are sometimes named 'articulated frameworks' (Hrennikoff, 1940) or sometimes simply 'networks' (Garboczi & Thorpe, 1986).

1.2 Why are Lattice Models Used?

An advantage of lattice models over continuum-based models is that they enable the modeling of non-homogeneous materials with greater disorder (Schlangen & Van

Mier, 1992). Another advantage of the lattice models is the lack of singularities in the nonlinear analysis that generally appear in continuum-based finite element methods (Chen et al., 2014). These properties make lattice models suitable for studying fracture in brittle materials such as concrete and ceramics, especially for large deformation problems where fragmentation occurs. One other property of the lattice models is that models resemble atoms and inter-atomic forces in the matter. This advantage led to the use of the lattice models to study cracks in the atomic scale and in the macro scales. (Paskin et al., 1980; Berton & Bolander, 2006).

These advantages, however, were not the only reason, why lattice models were used. Originally in Hrennikoff (1940), the lattice approach was proposed with the main goal of being able to solve the forces and deformations for in-plane stress (or in-plane-strain) problems by hand, simply because continuum-based approaches rarely allow exact solutions. These exact solutions were obtained for special cases where the geometry and loadings were very simple. Since Hrennikoff's time, the amount of research effort spent on lattice models has been continuously increasing due to the ease of access to high-performance computers. It is now a common practice to solve large deformation behavior of structures under dynamic loadings with both geometric and material non-linearities using lattice models (Hwang et al., 2020).

1.3 The Gap This Study Intends to Fill

From the above brief review of the development of lattice models, it is clear that the theoretical background of the lattice models was well studied, both from the physics (Feng & Sen, 1984) and from the structural mechanics and fracture mechanics perspective (Schlangen & Garboczi, 1997; Hwang et al., 2020). The researchers established how to calibrate different types of lattices, whether the structure is regular or random (Hrennikoff, 1940; Hwang et al., 2020). Lattice models were performing well in predicting the cracking behavior that was observed in experiments (Berton & Bolander, 2006). Lattice models were even applied for

studying more complicated problems than simple shape specimens such as high-pressure tunnels. Examples are given in Işık (2019) and Tuncay et al. (2016).

A review of the literature has shown that there is a gap in our understanding of elastic lattice models when a non-uniformly distributed point cloud is used in the analysis. The extensive literature review by Pan et al. (2018) shows that there is more interest in non-uniform lattices recently as structured lattices tend to have directionally biased crack patterns.

In this study, an energy-based approach is developed and implemented in MATLAB that can calibrate the parameters of random lattice spring models. The overall structural deflections are going to be studied for lattice spring models in relation to the energy-based element EA calibrations. A user-friendly interface is also developed that can handle different types of lattice meshes, complex geometries, boundary conditions, and loadings. This program is expected to serve as a platform for the nonlinear analysis of brittle materials.

1.4 Brief Overview of the Proposed Lattice Model

The lattice model used in this study consists of lattice elements that transfer only axial forces. No flexural moment is transferred at nodes. No shear springs are present either. Thus, this lattice model can be considered a truss network. This truss analogy can be visualized in Figure 1.1 and Figure **1.2**. This lattice can be classified as a 'lattice spring model' (LSM) according to the lattice model classifications given by Pan et al. (2018). LSM is chosen in this study because of its simplicity and popularity when compared to LBM (lattice beam models) in recent years (Pan et al., 2018).

Since the properties of regular lattices can be solved exactly, they are studied as benchmark problems in this study. A regular mesh consisting of squares with two diagonals and a random mesh is studied (see Section 3.1.1 for details). Each of the lattice networks is desired to represent the behavior of the corresponding elastic medium. The considered geometries are cantilevered beams and beam-column

joints. In general, any geometry with orthogonal edges can be studied using the algorithms presented in this study. Different boundary conditions (pin supports or rolling supports) or different loading conditions are handled with ease. To make it available to the research community, a user-friendly software is developed in MATLAB that allows the user to draw the geometry of the problem, and choose boundary conditions and loading scenarios.

In order to represent an elastic medium, properties of the truss elements need to be calibrated. The only mechanical property of a truss element is its stiffness EA/L (elastic modulus times the cross-sectional area divided by its length) value. The length of a truss is determined by the positions of the end nodes of the truss. Assuming that trusses are of uniform cross-section, then only EA is left to be calibrated. If truss members and the prototype have the same elasticity modulus, then only the cross-sectional area parameter is left to be calculated. This calibration is carried out by using the strain energy density equivalence between the prototype and the lattice model. The performance of this calibration is discussed in Chapter 4.

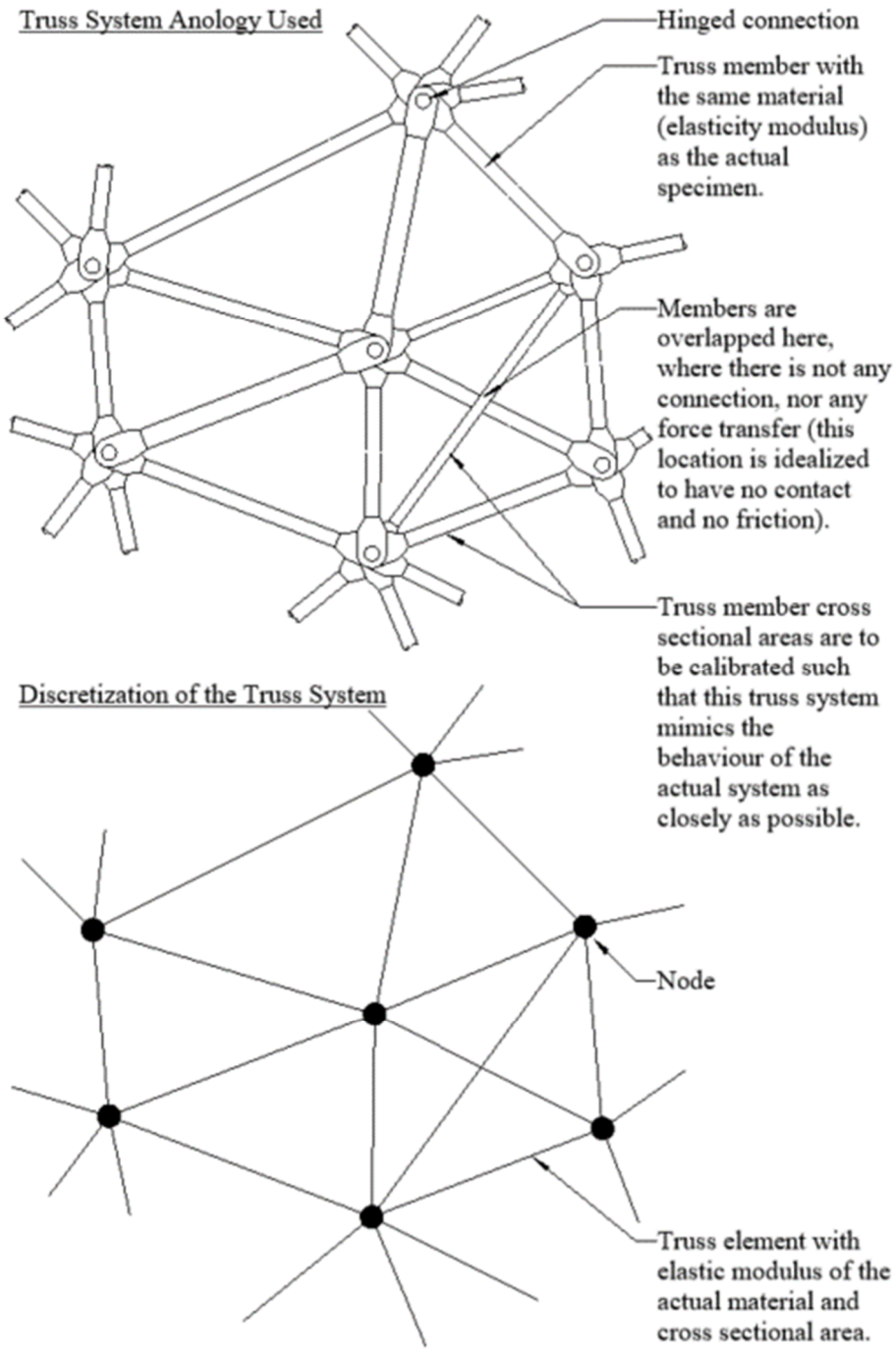


Figure 1.1 Overview of the irregular mesh lattice model used.

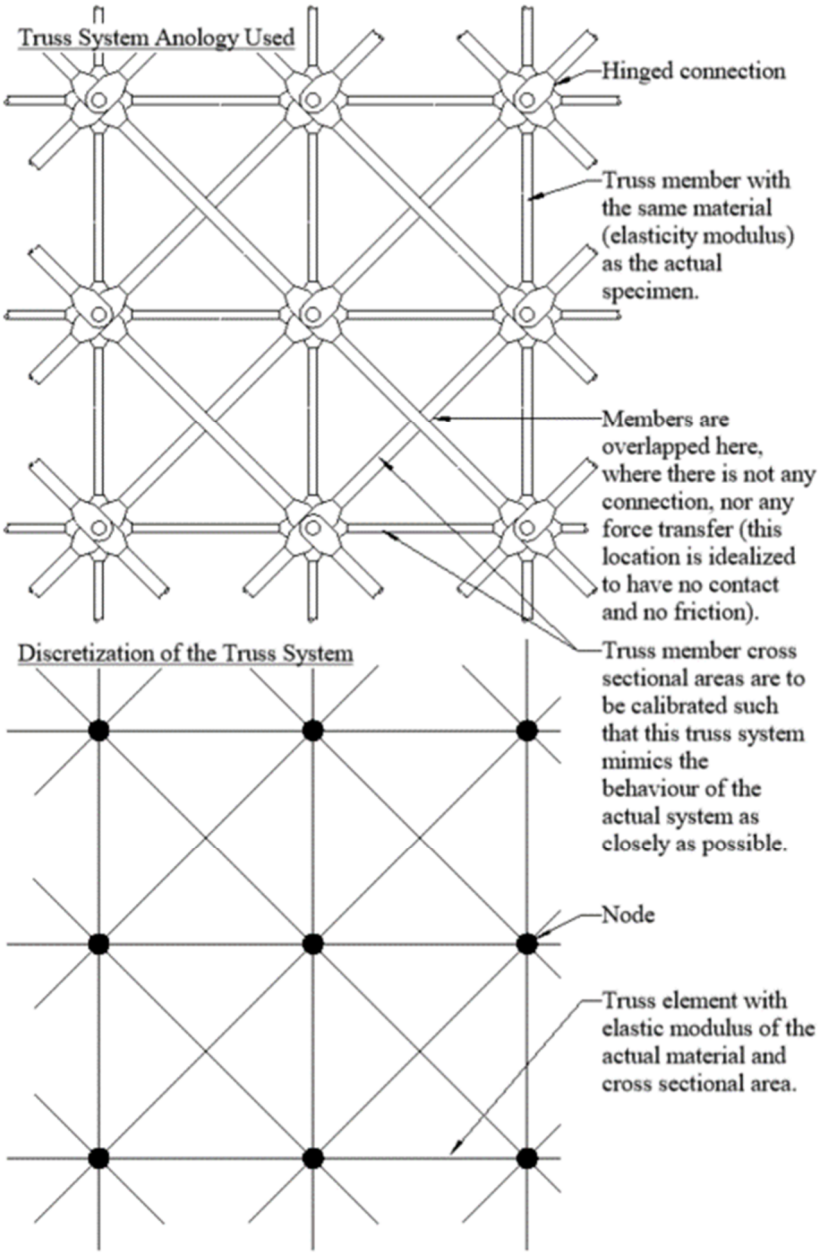


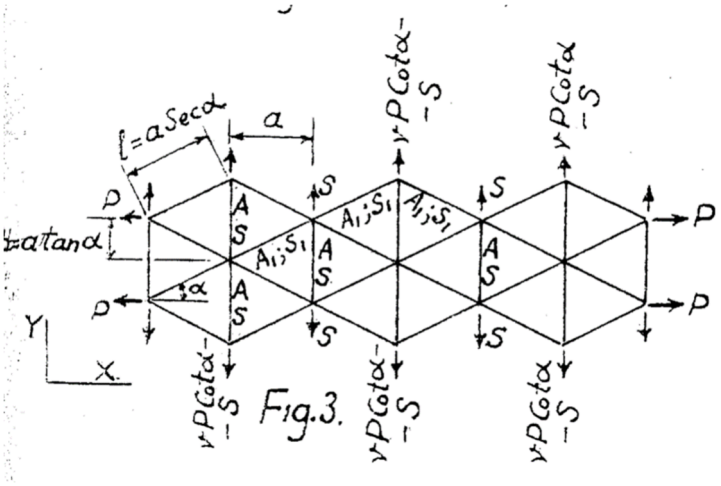
Figure 1.2 Overview of the regular mesh rectangle lattice model used with two diagonals as in Hrennikoff (1940).

CHAPTER 2

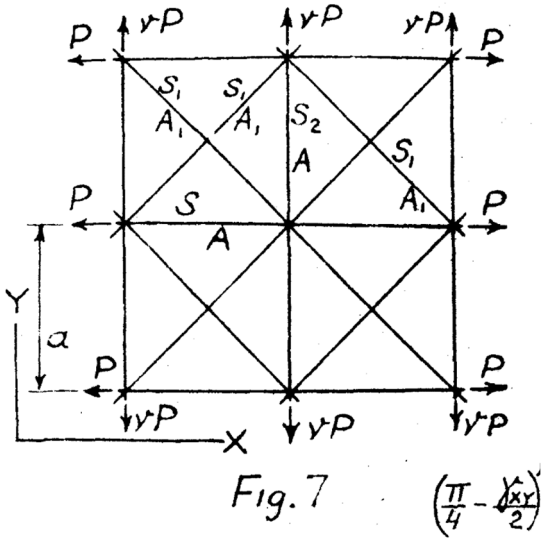
LITERATURE REVIEW

2.1 Early Studies on Lattice Models

The research that sets the groundwork for the lattice models is the Ph.D. thesis of Hrennikoff in 1940. Hrennikoff describes an 'articulated framework' (see Figure 2.1) and gives the theoretical solutions for the lattice element parameter for various regular lattice shapes. He proved that these lattice models only work with a Poisson's ratio of 1/3. Then he showed that as these regular lattice shapes get smaller and smaller, the solution of plain stress (or plain strain) should be identical to these of a plate problem solved using the differential equations that govern its deformation. It should be noted that Hrennikoff focused on accurately calculating the deflections and forces by using the articulated framework. He then provided the hand solution to an elastic steel gusset plate by using this method to show that the 'articulated framework' model gives somewhat accurate (excellent in solving deflections, acceptable in solving bending moments and shears) results for a real steel plate. In the absence of computers in 1940, Hrennikoff stated in his paper that random patterns may also be possible theoretically but not practical. When computers are invented, future work would prove this to be true (Schlangen & Garboczi, 1996). Although it is stated that Hrennikoff's research set the groundwork for lattice models, one should note that Hrennikoff stated that he took inspiration from two papers published earlier by Wieghardt (1906) and Riedel (1927).



(a)



(b)

Figure 2.1 Two of the structured lattices that were presented by Hrennikoff (1940): (a) 'Triangular Pattern' (b) 'Simple Square Pattern with Intersecting Diagonals'.

2.2 Studies with Regular Lattices

After the establishment of the groundwork by Hrennikoff, lattice models were used in the field of theoretical physics to study the mechanics of fracture in the 1970s (Ziman, 1979) and 1980s (Feng & Sen, 1984),. However, the vast majority of theoretical and computational studies focused on regular lattices.

Paskin et al. (1980) developed a computational framework to solve a triangular lattice network to investigate the initiation of cracks in brittle materials. The researchers found that results of the cracks modeled by using this framework method are in accord with the theoretical formulations. Beale & Srolovitz (1988) performed a study that modeled defects in lattice models. They used regular triangular lattices that transfer only axial forces (see Figure 2.2). According to a predefined random rule, the lattices would or would not be omitted, thus simulating defects in the medium. Lattices are assumed to break when their strain limits are reached. This breaking was assumed to be irreversible. Researchers concluded that based on their results, the proposed method was applicable to brittle materials such as ceramics, amalgamites, and sintered powders.

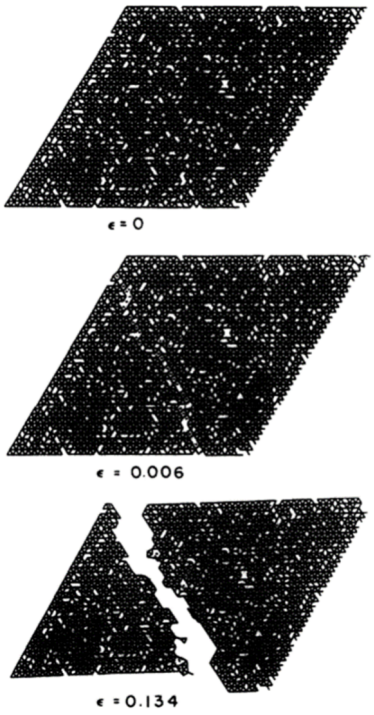
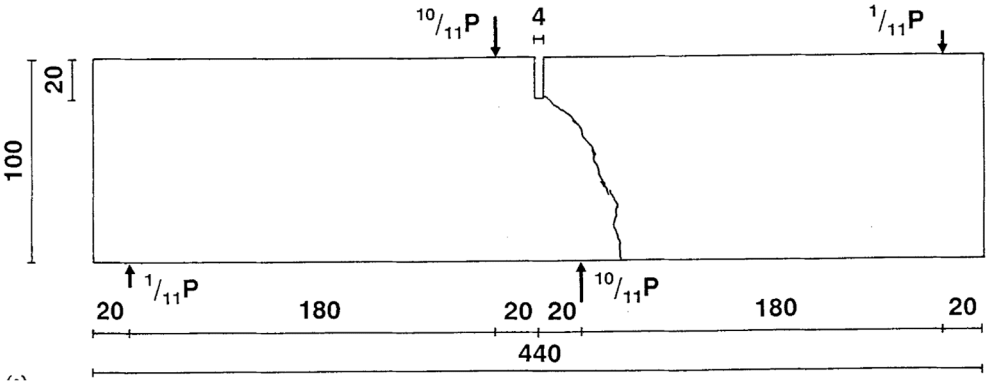
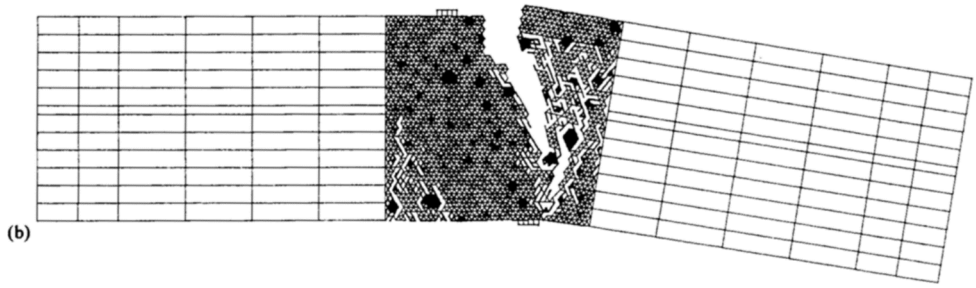


Figure 2.2 Triangular lattice model modelled with stochastic defects. Three different strain conditions are shown; the last figure shows the fully separated lattice (Beale & Srolovitz, 1988).

Schlangen & van Mier (1992) used a regular triangular lattice network that transmits flexural moments at nodes to study concrete crackingThey considered the grain structure of the concrete and assigned elastic properties and failure criteria to the lattice elements according to the grain where the lattice element is located. If the lattice element was within an aggregate, then they assumed the elastic properties of the aggregate, if it was located within the cement region, they used nonlinear cement properties accordingly. Three distinct problems including anchor pullout were modeled (see Figure 2.3 and Figure 2.4). They then compared laboratory experiments carried out under the same conditions with the lattice model results. Their observation was that the lattice models were able to provide fair matches with the experimental observations.



(a)



(b)

Figure 2.3 (a) Test setup of a concrete beam (SEN Iosipescu type configuration), (b) corresponding regular lattice model of the test setup. (Schlangen & van Mier, 1992)

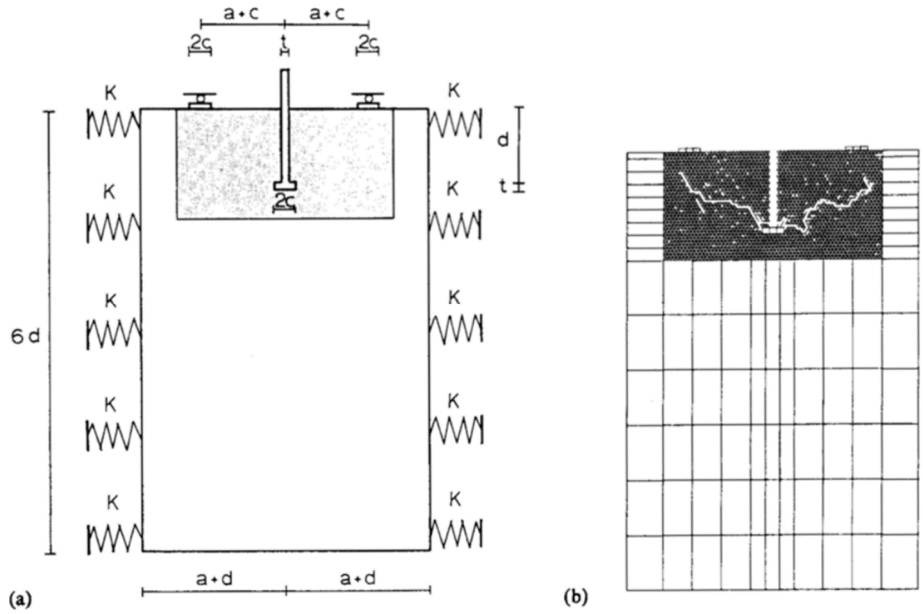


Figure 2.4 (a) Test setup of an embedment bar, (b) regular lattice model of the embedment bar simulating the experiment. Crack pattern is visible in the lattice model (Schlangen & van Mier, 1992).

The "peridynamic" formulation approach was proposed to solve the material behavior at crack tips and near crack locations (Silling, 2000). The method became quite popular quickly as it does not need additional consideration at the crack tips and the crack locations needed not be known a-priori. Traditionally, nodes in a lattice model were connected with the neighboring nodes. But it was noted that the stresses that occur at a point in a body are not only dependent on the strains at the point, but also on the gradient of the strains at the point. Thus, in the peridynamic model, the nodes in a lattice model are connected not only with the neighboring nodes but with more nodes that are second and third neighbors, etc. This approach was able to accurately estimate crack locations and could predict the crack tip propagation better than conventional continuum-based methods (see Figure 2.5). By using the mentioned theoretical background on the peridynamic model, Macek & Silling (2007) developed a hybrid finite element framework and solved the quasi-static behavior of a cracking plate and rigid ball penetration into a brittle plate. In this

study, however, they needed to make a hybrid model of finite elements used for parts that are away from the crack/impact locations, and framework (or lattice) meshes were used at the locations of crack/impact. This hybrid modeling was necessary as each interior node in the lattice regions had more than 100 connectivities and millions of lattice elements would be needed if the whole region was modeled with lattices, which was computationally too expensive with the available computational sources at that time.

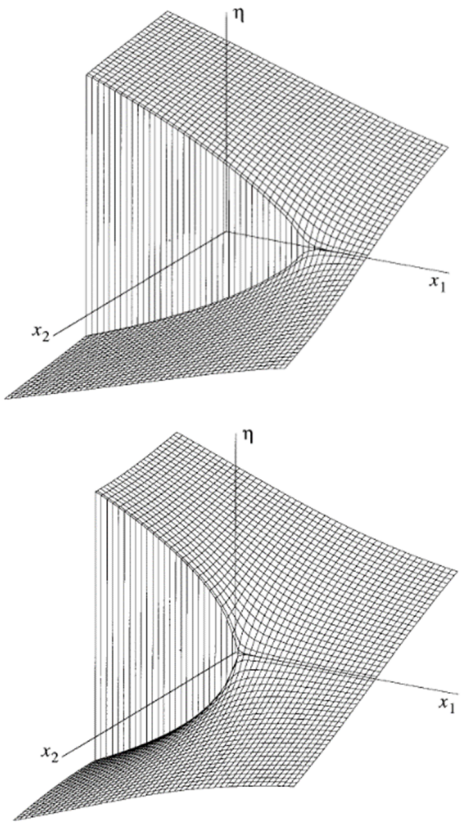


Figure 2.5 Crack tip on a teared plate solved with peridynamic method (above) and conventional models (below) (Silling, 2000).

By using the concept of the peridynamic approach mentioned above, an overlapping lattice model (OLM) is developed to study the behavior of concrete (Aydin et al., 2018). The overlapping lattice method consisted of nodes connected with the nearby nodes within a horizon distance. This horizon distance was taken as 1.5 times the

orthogonal node spacing and 3.01 times the orthogonal node spacing, which created 8 and 28 connectivities, respectively for internal nodes (See Figure 2.6). Each connectivity was assigned a lattice element that would carry only axial forces and no bending moments. The elasticity times the lattice cross-sectional area for each truss element was calibrated using the energy equivalency. A constant strain field was applied along one direction to the original model. Elastic energy stored was recorded. The same strain field was then applied to the lattice model, initially assuming elasticity times the lattice cross-sectional area for each element was taken equal to unity. The ratio of the elastic energy stored in the original model and the elastic energy stored in the lattice model was calculated. This ratio was used as lattice cross-sectional area for each element. Researchers used the OLM to model a concrete tension test as shown in Figure 2.7. The researchers used sequential elastic analysis (SLA), which was originally proposed by Rots (2001) to calculate the deformations in the lattice model. In this method, a number of linear analyses were performed rather than performing a direct non-linear calculation to model the stress-strain behavior of concrete and steel materials (see Figure 2.8). SLA had the advantage of being able to handle concrete softening without any convergence problems unlike methods based on the Newton-Rapson approach have for highly nonlinear problems. However, it is only appropriate for quasi-static problems under non-cyclic loadings.

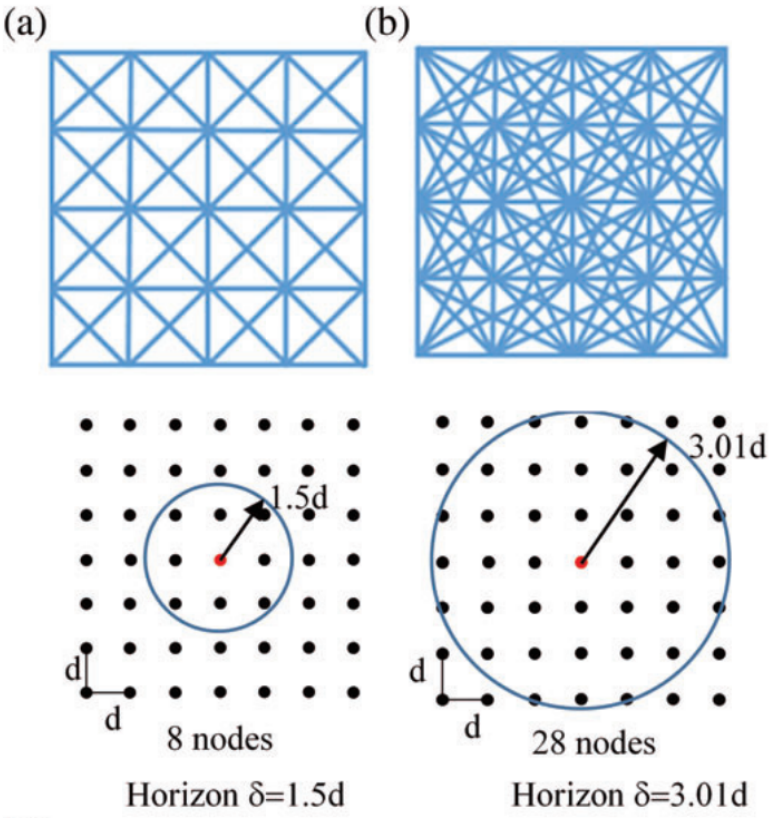


Figure 2.6 Node connectivities used in the OLM: (a) 8 nodes connectivity for internal nodes, (b) 28 nodes connectivity for internal nodes. (Aydin et al., 2018).

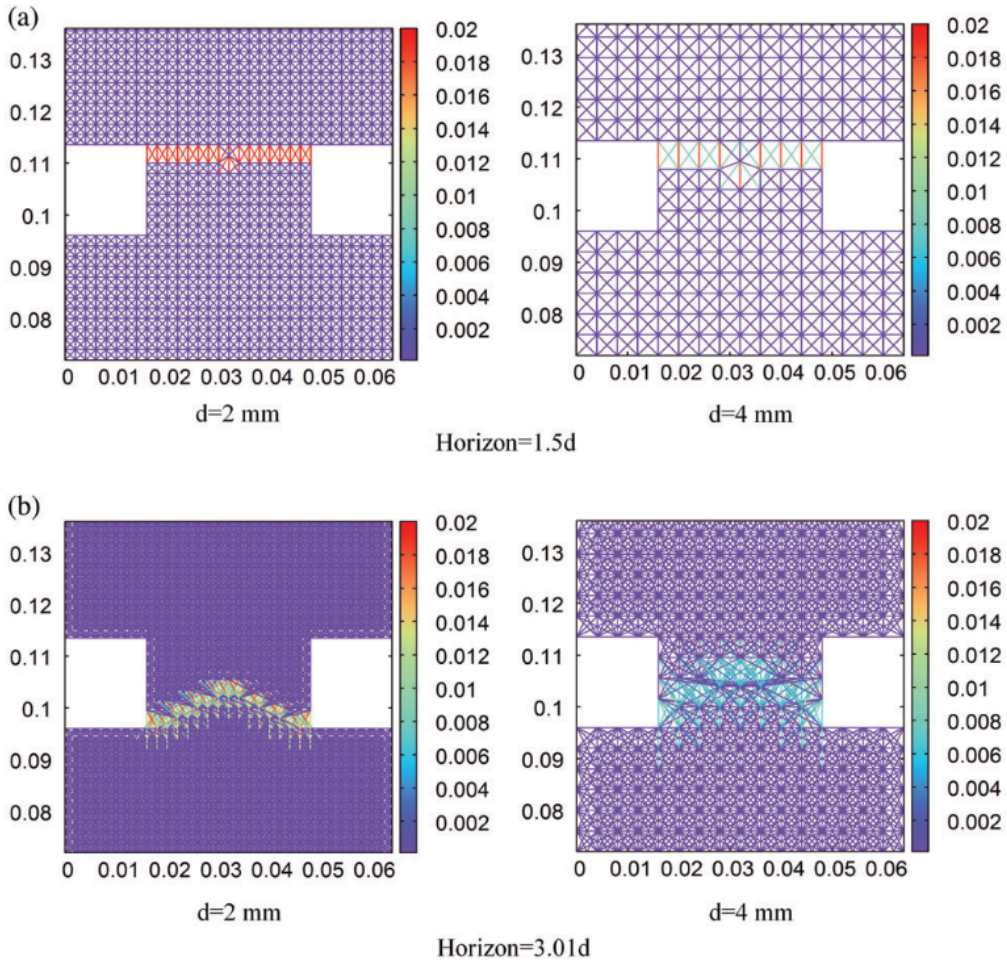
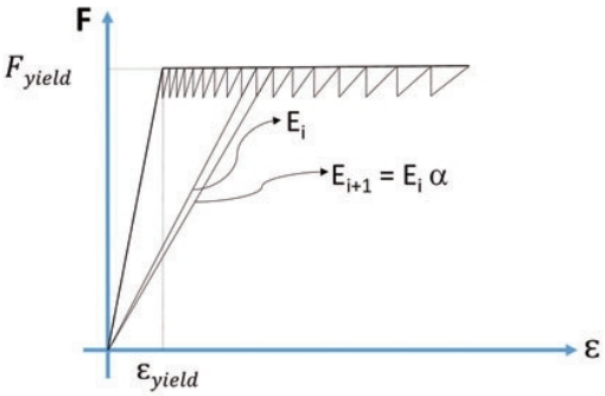
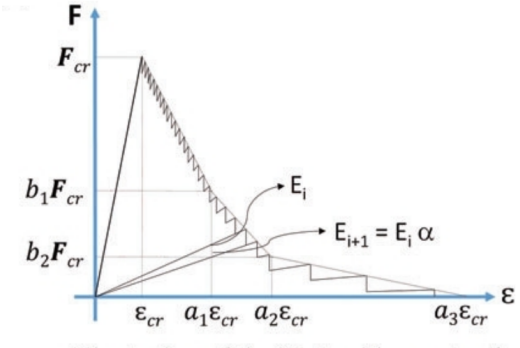


Figure 2.7 Damage patterns for uniaxial tension tests that were conducted using OLM with different horizons and solved with sequential linear analysis (Aydin et al., 2018).



(a)



(b)

Figure 2.8 Material curves used in the model and saw-tooth behavior of SLA: (a) steel (b) concrete. (Aydin et al., 2018).

Researchers also compared OLM to finite element methods for modeling concrete gravity dams under seismic loads (Soysal et al., 2017). Two shapes of monoliths were considered under varying lateral loads due to different peak ground accelerations (see Figure 2.9). It was observed that with either of the methods, the global behavior of the dam and demands were similar.

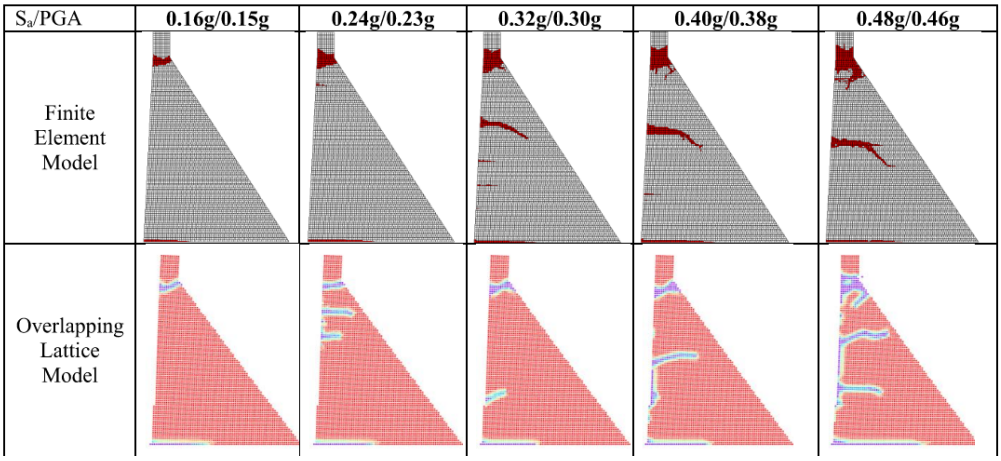


Figure 2.9. Crack patterns of a concrete gravity dam under varying seismic loads, finite element and OLM comparison (Soysal et al., 2017).

2.3 Random Lattice Models

Schlangen & Garboczi (1996) proposed a method utilizing random node distribution to reduce the bias of crack direction (Figure 2.10). This random lattice could transfer moments at the nodes and lattice properties were calibrated for a homogeneous elastic continuum. They used the method outlined in Moukarzel & Herrmann (1992) to achieve this elastic uniformity. The method required multiple program phases to estimate elasticity modulus times cross-sectional areas, EAs and elasticity modulus times moment of inertias, EIs of lattice elements. First, the lattice would be subjected to hydrostatic pressure and the EAs of the elements were determined. This alone, however, was not always sufficient to achieve elastic uniformity. Then, the second phase of a calibration algorithm is performed where the lattice is subjected to uniaxial strain, and EIs were solved for each lattice element. An interesting observation of the researchers was that if the crack patterns to be simulated were simple in shape, then lattice elements without bending provided accurate results, but when the crack shape that is to be simulated was complicated, then the lattice

elements that transfer flexure gave more accurate results. They elaborated further on this research in their next study (Schlangen & Garboczi, 1997).

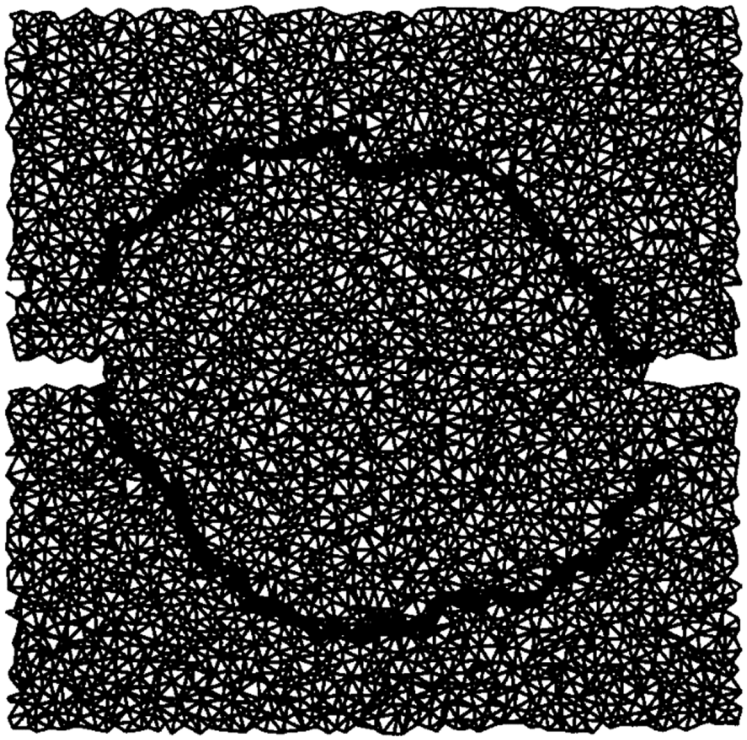
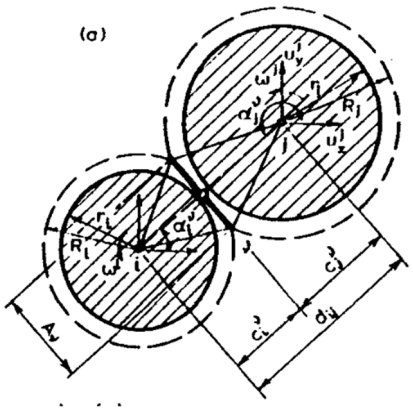
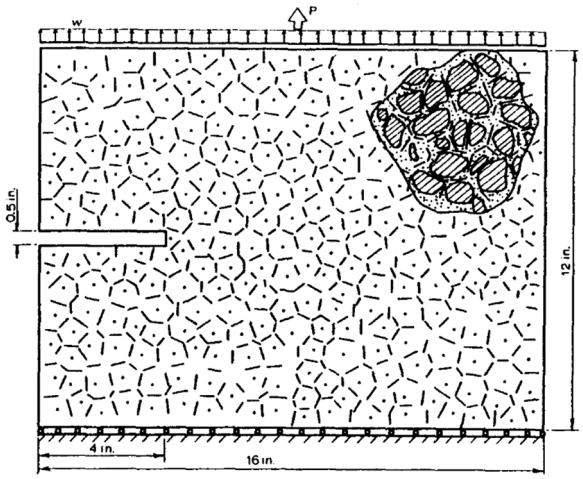


Figure 2.10 Homogenized random lattice obtained by Schlangen & Garboczi (1996). Cracked pattern is visualized by darkening elements.

Zubelewicz & Bažant (1987) proposed a method for modeling the nonlinear behavior of concrete. In the model, aggregates were taken to be rigid, and the cement matrix was modeled as deformable. The clear distance between aggregate elements that were neighboring each other determined how stiff the interaction between these two aggregates was. Aggregates were assumed to be in a circular shape for simplicity. They named this approach the interface element model.



(a)



(b)

Figure 2.11 (a) Aggregates modelled as circles and cement paste in between. (b) Lattice model of a concrete specimen that is modelled with interface element modeling method (Zubelewicz & Bažant, 1987).

Bažant et al. (1990) further developed this approach and proposed a model called 'the Random Particle Model'. In this model, a particle system was generated with different sizes, and these particles would be connected with axial interactions only, i.e., lattice elements. The configuration of aggregates and cement matrix was similar to that of Zubelewicz & Bažant (1987), but this time, the aggregates were taken to

be elastic (as opposed to rigid in the earlier study). The main objective of the study was to investigate the effects of specimen size, concrete softening, and microcracking zones in macro-scale behavior. The stiffness of cement matrix connection between neighboring aggregates was defined in the below equations, which are analogous to springs connected in series:

$$S = \left(S_i^{-1} + S_n^{-1} + S_J^{-1}\right)^{-1} \tag{2.1}$$

$$S_i = \frac{E_a A_i}{L_i} \tag{2.2}$$

$$S_m = \frac{E_m A_m}{L_m} \tag{2.3}$$

$$S_i = \frac{E_a A_j}{L_j} \tag{2.4}$$

where E_a is the elasticity modulus of the aggregate, E_m is the elasticity modulus of the cement paste, and A_j, A_m, and A_i are the cross-sectional areas of the lattice element obtained (Figure 2.12). They concluded that their model was successful at predicting the size effects of specimens and post-peak load behavior.

Jirásek & Bažant (1995a) made comparisons between regular and random lattices focusing on the brittle behavior of sea ice. They used lattices that transfer only axial forces. They studied how a steel ball would crack sea ice. First, they used regular lattices with no variation in lattice element parameters (as shown in Figure 2.13). Secondly, they used regular lattices again but introduced random displacements in the location of nodes (as shown in Figure 2.14). They also considered a fully random lattice (as shown in Figure 2.15). It was observed that when regular lattices were used, the crack pattern was biased (even when randomness was introduced). Fully random meshes were capable of providing crack patterns that were more realistic and similar to experimentally observed crack patterns. In a follow-up study, they improved their models and named them ‘the particle model for fracture’ (Jirásek & Bazant, 1995b).

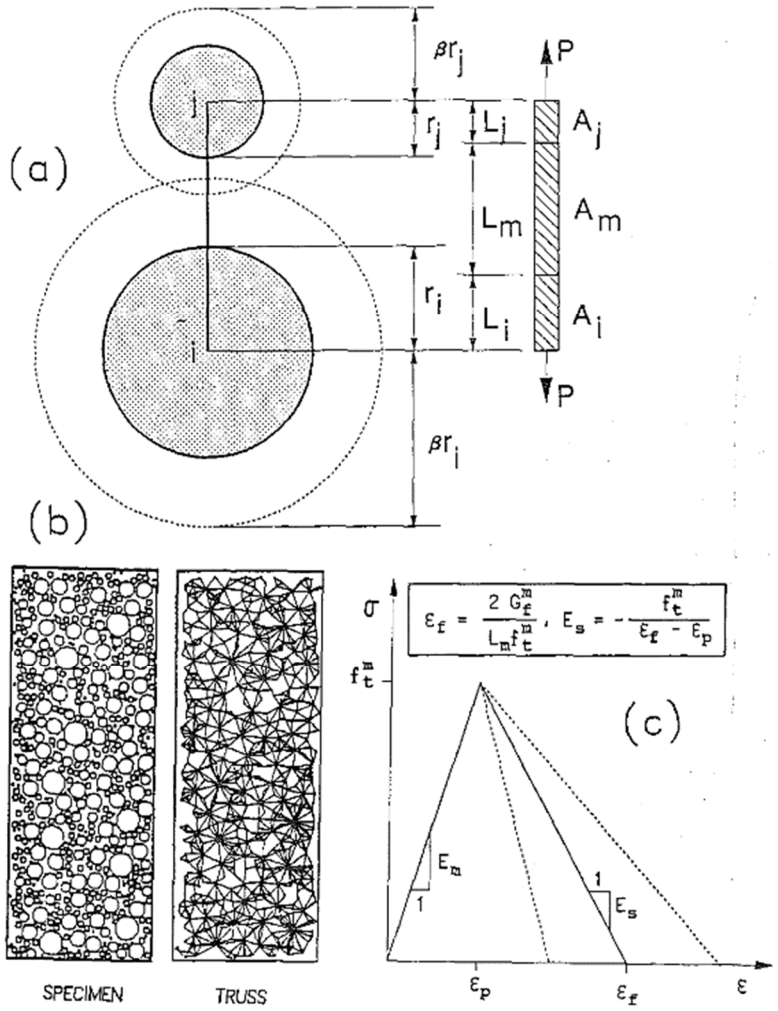


Figure 2.12 (a) Neighboring aggregates modeled as circles and the cement matrix around. Corresponding lattice element is at the right side. (b) Specimen consisting of randomly distributed aggregates and corresponding lattice truss model. (c) stress-strain curve for the cement matrix bonds.

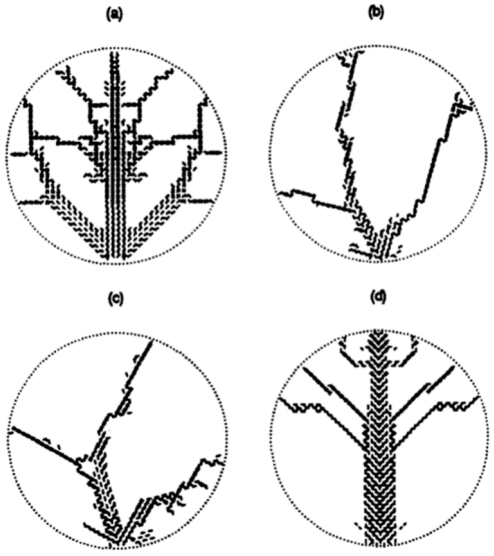


Figure 2.13 Ice crack patterns with structured lattice with no random variation of lattice strength. The angle between structured lattice and impact direction is (a) 0º, (b) 15º, (c) 30º, (d) 45º.

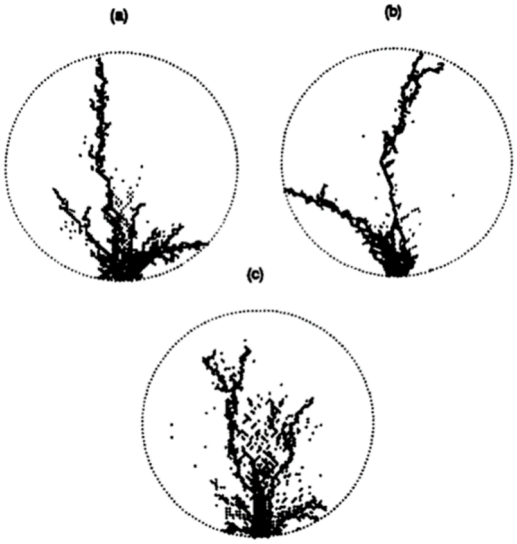


Figure 2.14 Ice crack patterns with structured lattice with probabilistic random variation of lattice strength. The angle between structured lattice and impact direction is (a) 0º, (b) 22.5º, (c) 45º.

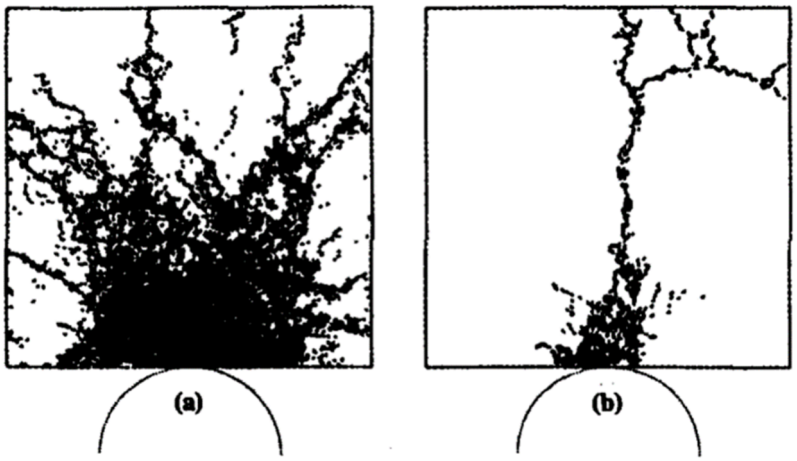


Figure 2.15 Ice crack patterns with random lattice when compression failure of lattices: (a) not allowed, (b) allowed.

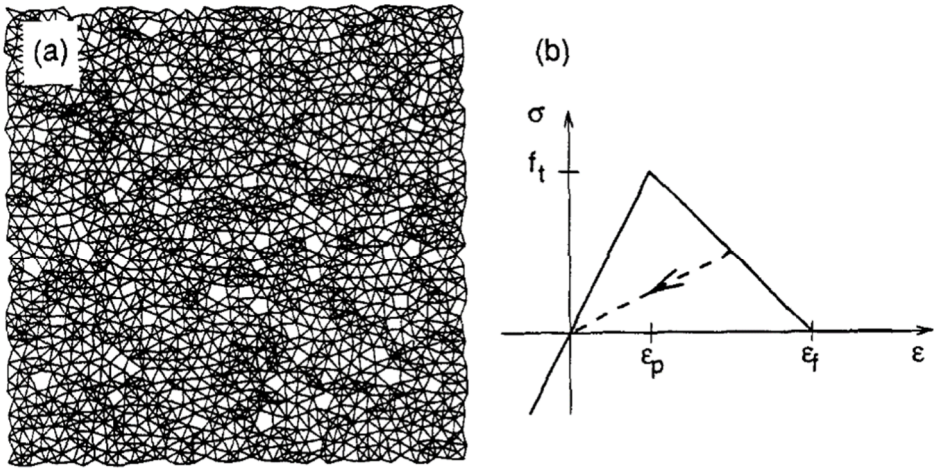


Figure 1: (a) Random lattice, (b) bilinear stress-strain diagram

Figure 2.16 (a) Random mesh as proposed by Jirásek & Bazant (1995). (b) Accompanying stress-strain diagram for lattices.

Zhao (2014) provided groundwork for studying lattice models under dynamic loadings. He used regular triangular meshes for modeling the body. Nodes were

connected with axial and shear springs. The researcher calls this model a distinct lattice spring model (DLSM). In this context, the lattice models that consist of elements that only transmit axial forces are called LSM. It should be noted that the main improvement in this study is the addition of shear springs which allows one to model elastic bodies having Poisson's ratios other than 1/3 (in 2D) and 1/4 (in 3D).

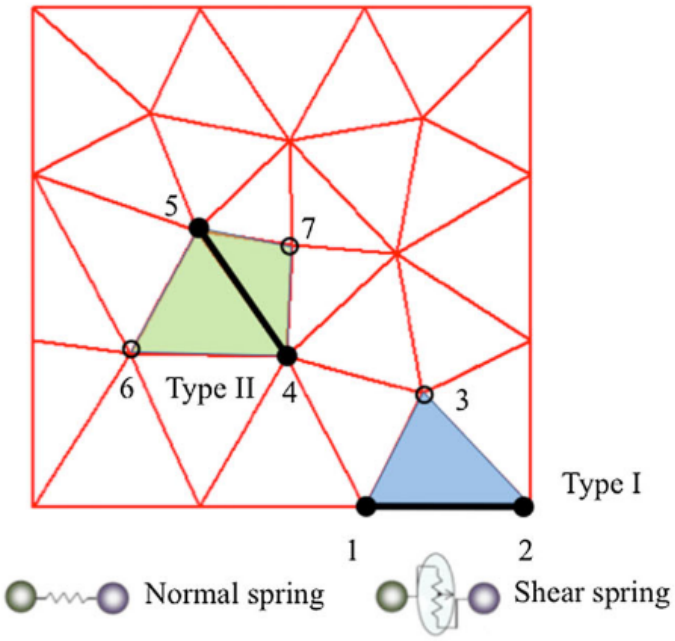
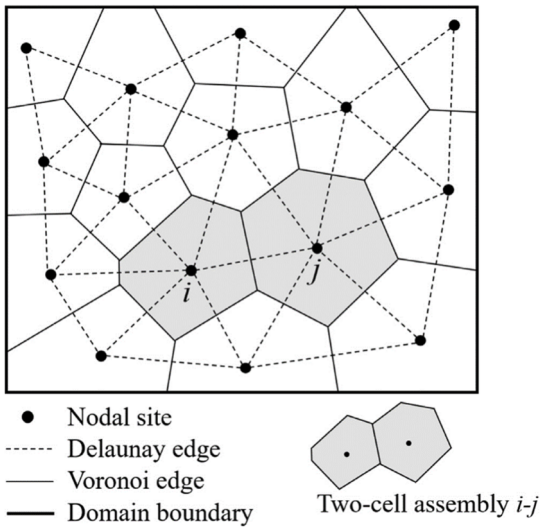
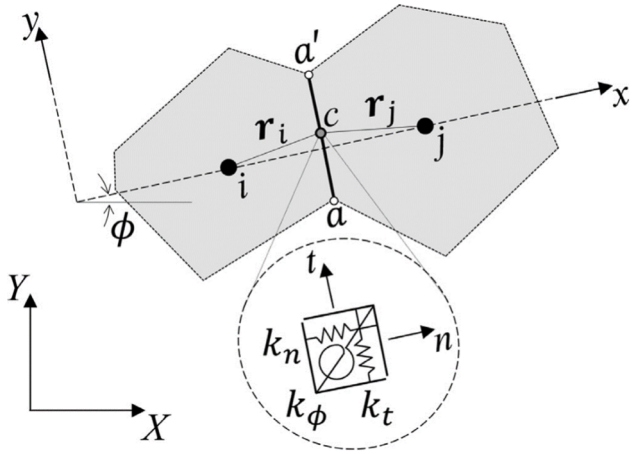


Figure 2.17 Sample mesh showing the lattice mesh and normal spring and shear spring interactions between lattice nodes (G.-F. Zhao, 2014).

Hwang et al. (2020) proposed a method to study the dynamic behavior accounting for non-linearities due to large displacements using random lattices. Their method was utilizing Voronoi areas as rigid cells and translational, rotational and shear springs located at each boundary shared between two rigid cells. This method was called the Rigid Body Spring Model (RBSM) taking its roots from Kawai (1978) and it was further developed by Bolander & Sukumar (2005) and later on by Yamamoto (2019) to study the localized failure of reinforced concrete.



(a)



(b)

Figure 2.18 (a) Diagram showing a discretized body into Voronoi cells and node connectivities. (b) Diagram showing the rigid body spring model (RBSM) components: rigid cells and 3 springs at each Voroni boundary (Bolander & Sukumar, 2005).

Regular lattices have the shortcoming of being locked under compression for non-linearly analysis and they, most of the time, cannot be used to model the indirect tension failure of concrete (Aydin et al., 2021). Concrete needs to be analyzed non-linearly to capture the behavior of indirect tension failure. To overcome this, a "random lattice model" developed by constructing a regular lattice model and applying a certain amount of perturbation to each node (similar to the approach used by Aydin et al. (2021) was proposed. A sample lattice is shown in Figure 2.19. This lattice allows for the second-order effects to be captured. It was observed that this "random mesh" was representing the actual tensile failure behavior of the concrete specimen better than "regular meshes" (Figure 2.20). The experimental data for a reinforced concrete beam bending test was taken from the study of Belgin & Şener (2008). The method for non-linear analysis that was conducted in this study is actually done by SLA with non-linear tension behavior and linear compression behavior (Aydin et al., 2018).

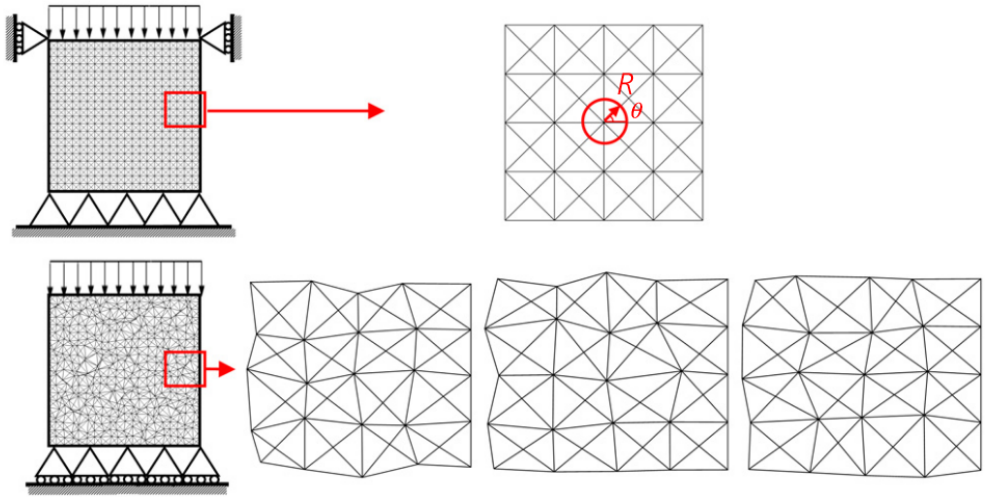


Figure 2.19 Sample regular and random meshes used to model the concrete under compression (Aydin et al., 2021).

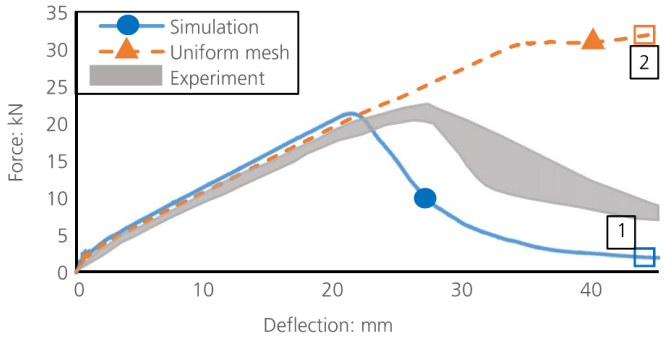


Figure 2.20. Force vs displacement curve for an over-reinforced concrete beam. Deflections observed using simulation (indicating "random mesh") and uniform mesh are indicated with lines and experimental observation was given by the grey zone (Aydin et al., 2021).

2.4 Classification of Lattice Models

Nikolić et al. (2018) and Pan et al. (2018) provided the most recent and comprehensive literature review of lattice models. Figure 2.21 illustrates the different types of lattice model classifications (Pan et al., 2018). In this context, the present study falls under the category of LSM with both regular and random lattices. Element behavior is assumed to be elastic. Although the applied loading is static, extension to account for dynamic loadings is quite trivial.

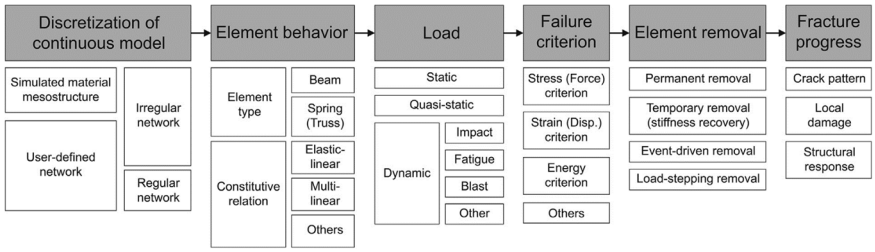


Figure 2.21. Classification of lattice models according to the literature review by Pan et al. (2018).

CHAPTER 3

METHODOLOGY

To study the elastic lattices, these steps are followed:

- A lattice mesh is generated.
- The EAs of the lattice elements are calibrated.
- Boundary conditions are defined and loads are applied.
- Elastic deformations are solved.
- Deflections are calculated and stored.
- The deflections are compared with the deflections that are obtained from finite element shell models of the same problem under consideration.

3.1 Types of Meshes Used

There are six types of meshes considered in this study. Four of them are random meshes and two of them are regular meshes. These meshes can be listed as follows:

- Random Mesh: Type 1, Delaunay mesh
- Random Mesh: Type 2, Delaunay mesh with crossing elements
- Random Mesh: Type 3, constructed using proximity with straight edges
- Random Mesh: Type 4, constructed using proximity with non-constrained edges
- Structured Mesh: Type 1, rectangles with two diagonals
- Structured Mesh: Type 2, triangular

The regular meshes (or structured meshes) are constructed by repeating patterns of nodes and element connectivities. In this study, the pattern of “square with two diagonals” is given the most attention. The second type of mesh is the random mesh

(or non-structured mesh). They are constructed with a randomly generated but homogeneous point cloud connected by truss elements. The connectivity rules depend on the type of random mesh. It is also possible to refine random meshes at selected locations of the model by increasing the mesh density. The following sections explain the mesh generation algorithms in detail.

3.1.1 Random Mesh: Type 1, Delaunay Mesh

First boundaries are constructed with regularly spaced nodes. Then, a random point cloud distribution for nodes is introduced. At each iteration, the algorithm checks a node candidate with random x and y coordinates within the shape boundary. The node candidate must not be closer to the previously placed nodes than a predefined distance. If this condition is met, then the node candidate is accepted and added to the node list. If not, then this node candidate is rejected and a new node candidate is randomly generated. The algorithm runs for a sufficiently large number of iterations. This fixed number is much higher than the number of nodes that can fit within the boundary because with each node placed, the probability of having the next node candidate satisfying the placement criteria decreases. This is because the coordinates of the node candidates are random. The fixed number of iterations is chosen such that increasing the number of iterations would not increase the number of nodes placed.

A Voronoi tesselation algorithm is run to determine the tributary area due to each node. Then, a Delaunay triangulation algorithm is run to obtain a homogeneously connected lattice model. The node connectivity list is stored in the computer memory. This node connectivity list defines the lattice elements whose EAs need to be estimated (presented in Section 3.1.9). A sample Delaunay triangulation mesh is shown in Figure 3.1.

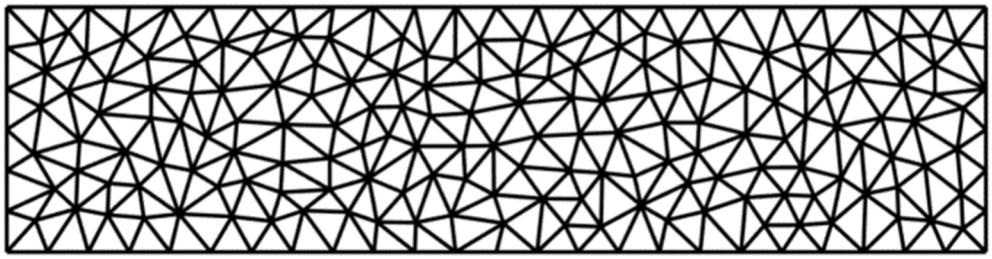


Figure 3.1 Sample Random Mesh: Type 1.

3.1.2 Random Mesh: Type 2, Delaunay Mesh with Crossing Elements

Initial phase of the algorithm is the same as the Random Mesh Type 1, but there is an additional step that is performed to construct this mesh. In this step, crossing elements are added to the connectivity list. Crossing element candidates are chosen between nodes separated by a distance smaller than a pre-defined distance. This prevents very long elements due to being created within the mesh. The crossing element candidate is checked with the list of elements to see if it results in a very narrow angle with the other elements. The minimum angle that is allowed that a crossing element can make with other elements is pre-defined such that a homogeneously distributed elements are created. Figure 3.2 illustrates the crossing element creation criteria. See Figure 3.3 for a sample of this mesh and the corresponding Voronoi tessellation. Figure 3.4 shows the close-up view of that mesh with Voroni tessellation.

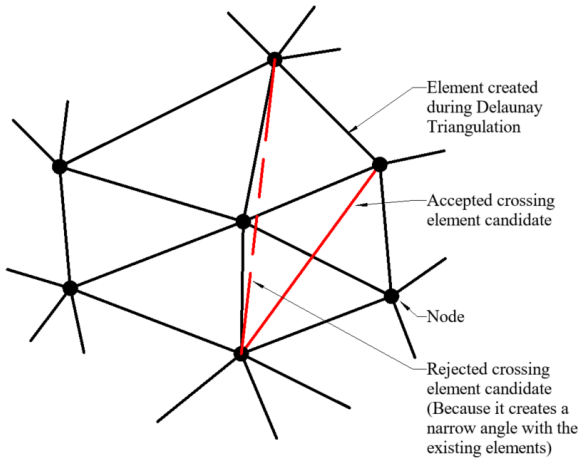
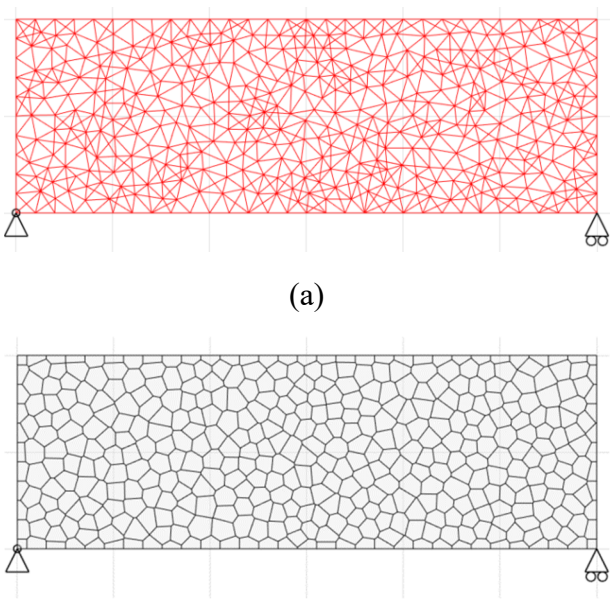


Figure 3.2 Creation of 'crossing elements' in Random Mesh: Type 2.



(a)

(b)

Figure 3.3 Sample beam with Random Mesh: Type 2. (a) Delaunay triangulation plus 'crossing elements' form the lattice elements. (b) Voroni areas calculated for nodes.

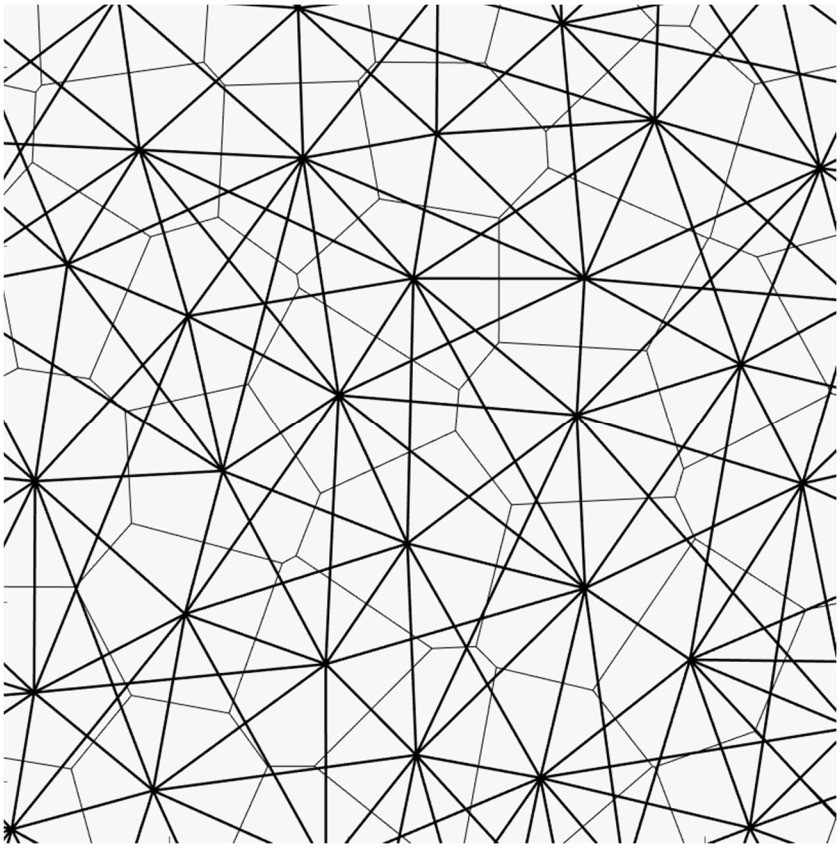


Figure 3.4 Close-up view of an irregular mesh. Black dots indicate nodes. Bold lines indicate lattice elements. Thin lines and light grey hatch indicate Voronoi areas tributary to each node.

3.1.3 Random Mesh: Type 3, Constructed Using Proximity with Straight Edges

The node creation phase of this type is the same as Random Mesh: Type 1, but this time a different method is followed to create the connectivities. All nodes are connected with the nearby nodes. The density is determined based on reference length, d. Nodes are placed such that none of the nodes are closer to each other by 0.75d. If a node is closer than 3.01d to another node, then they are considered nearby and connected. Narrow angles and overlapping elements are allowed in this type of mesh (Figure 3.4). It should be noted that the interconnectedness of this mesh is

resembling the interconnected regular meshes that are used to make peridynamic models (Aydin et al., 2018).

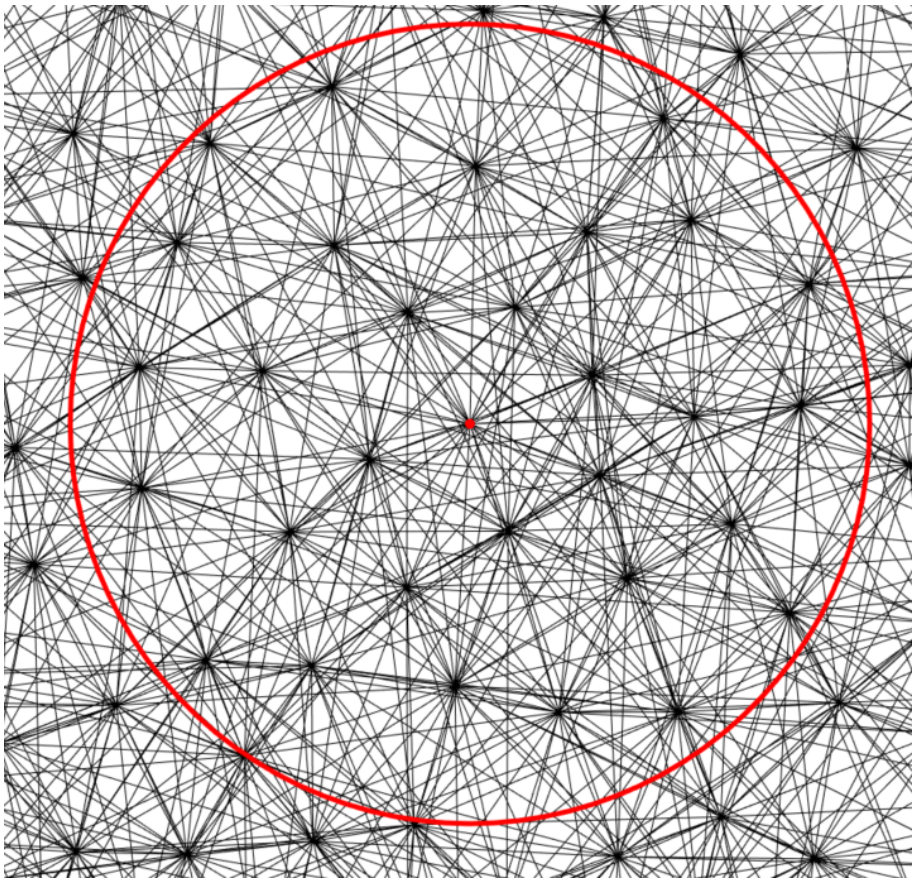
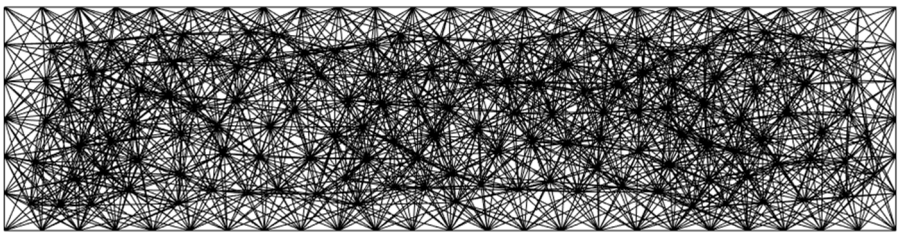


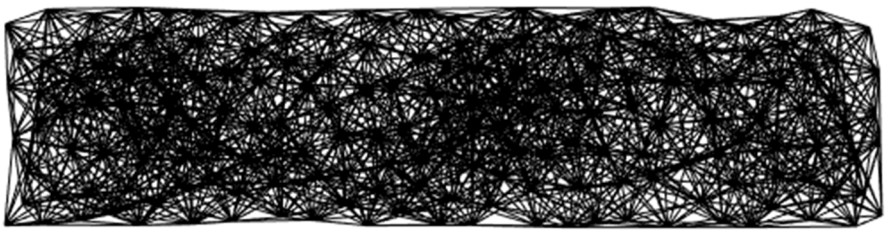
Figure 3.5 Random Mesh: Type 3 with a sample node marked with a red dot and its range of connectivity is indicated with a red circle (3.01d). Connectivities in the mesh are drawn with black lines.

3.1.4 Random Mesh: Type 4, Constructed Using Proximity with Non-Constrained

This mesh type is similar to Random Mesh: Type 3, but in this type, there are no straight edges imposed, and edges are allowed to be irregular. A comparison of Random Mesh: Type 3 and Type 4 is shown in Figure 3.6.



(a)



(b)

Figure 3.6 Comparison of (a) Random mesh: Type 3 and (b) Random mesh: Type 4.

3.1.5 Random Mesh Refinement

Additional nodes are added to the boundaries which are closer to refinement points by a predefined distance. The closer the refinement point to the boundary is, the more nodes are added at the boundary. Nodes are prevented from being placed very close to each other such that a non-homogeneous element distribution is obtained. For the generation of interior nodes, iterations are performed. At each iteration, the computer checks a node candidate with random x and y coordinates. This node candidate must not be closer to the nodes than a predefined distance. This predefined distance gets smaller as the node candidate gets closer to the refinement point and is capped at a certain minimum value to prevent very small element sizes. This distance gets larger as the node candidate gets away from the refinement and is capped at a certain maximum value to prevent very large element sizes. The rest of the procedure is identical to the ones given for random meshes.

In Figure 3.7, d is the typical element length at the boundaries, apr is the approximate expected element length at not refined locations. In this study, the parameters are taken as c_1=1, c_2=1, c_3=1/3, and c_4=4. It is possible to use different values as required by the nature of the problem that is considered by going into the source code, but the user was not given the option to change these variables at the user interface as the intention is to keep the graphical user interface as simple as possible.

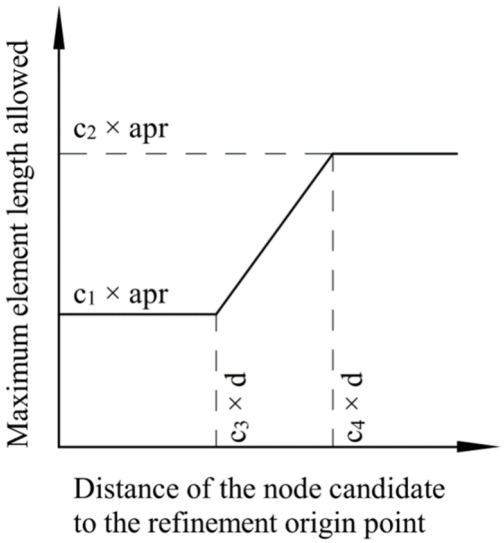


Figure 3.7 Graph showing the refinement rules for the refined irregular mesh.

3.1.6 Regular Mesh

There are many types of regular meshes that were used for regularly structured lattice models (Hrennikoff, 1940). Each has its own advantages and disadvantages. In this study, a structured pattern consisting of rectangles with two diagonals is used in the computer software with a graphical user interface. The reason for choosing this type of regular pattern is that it is scalable to rectangular shapes (provided that each side of the rectangle is divisible by the length of one orthogonal element of the structured mesh pattern). This type of structure does not have the hardships experienced at the rectangular boundaries that a triangular shape would experience. The triangular

lattice pattern was tested to make sure that the procedure gives results in agreement with the analytical calculations. This triangular lattice option, however, was not included in the computer software.

3.1.7 Regular Mesh – Rectangles with Two Diagonals

This type of mesh consists of uniformly distributed nodes at the boundaries. Elements connect these nodes in both orthogonal directions. At each rectangular zone created, two diagonal members are added. Although these two diagonals seem to 'intersect' each other in the middle, this is only the visual case, there is no connection in the middle structurally, and thus there is no node in the middle of each rectangle.

This type of lattice is quite popular as it can repeat itself in a pattern that rectangular areas can be filled (provided that both sides of the rectangle are divisible by one orthogonal element length). Since the diagonal members are not actually connected in the middle, this also keeps the number of nodes low, as a consequence, the number of total degrees of freedom in the system is kept small. All the joints are 'pin' connections, so no moment is transferred at the nodes between the elements.

Theoretical EA values for orthogonal elements and diagonal elements as well as the Poisson's ratio of this system can be calculated by using the energy density equations or by simply equilibrium equations such that this truss system would mimic the behavior of the real in-plane system. As it was already calculated by Hrennikoff (1940), the parameters of this pattern can be summarized as follows.

$$A = \frac{at}{1+\nu} \tag{3.1}$$

$$A_1 = \frac{\nu\sqrt{2}}{1-\nu^2}at \tag{3.2}$$

$$\nu = \frac{1}{3} \tag{3.3}$$

A is the cross-sectional area of the orthogonal lattice element, A_1 is the cross-sectional area of the diagonal lattice elements, a is the length of the orthogonal lattice elements, ν is the Poisson's Ratio and t is the thickness of the plate under consideration.

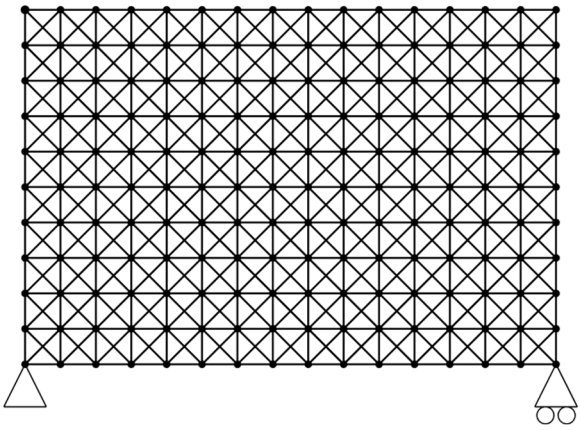
One peculiar observation to be made here is that Poisson's ratio needs to be 1/3 always for this system to work. Hrennikoff (1940) showed that Poisson's ratios other than 1/3 were not possible with this type of regular mesh. It should be noted that similar comments were made by Bažant et al. (1990).

Table 3.1 shows the A and A_1 for different values of a. These values are used to check the accuracy of the solutions obtained with the proposed numerical method.

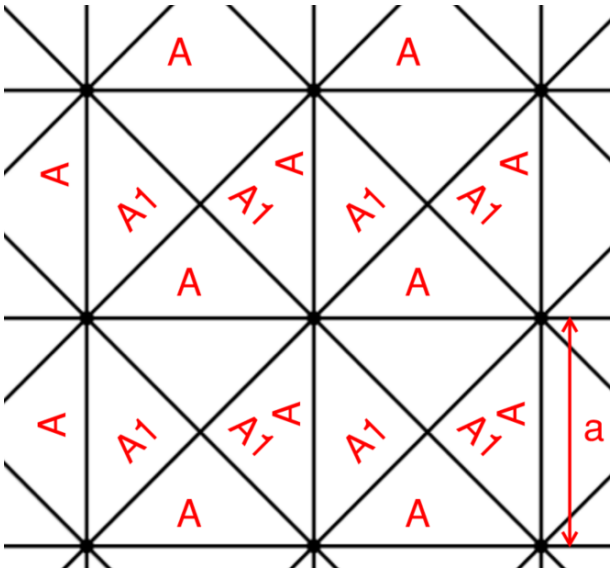
Table 3.1: Theoretical solutions for cross sectional areas of the lattice elements in a lattice model with structured mesh of rectangles with two diagonals.

a	A	A_1
1/8	0.094	0.066
1/4	0.188	0.133
1/3	0.250	0.177
1/2	0.375	0.265
1	0.750	0.530
2	1.500	1.061

Note: t=1, ν=1/3



(a)



(b)

Figure 3.8 (a) Sample regular mesh – rectangles with two diagonals. Black dots indicate node locations. Note that there is no node at the 'intersection' of the two diagonals. (b) Close-up view of the regular mesh – rectangles with two diagonals showing the theoretical cross-sectional areas of the lattice members.

3.1.8 Regular Mesh – Triangular Mesh

This pattern may be the simplest of all possible regular meshes since a square pattern is not possible (it would be structurally unstable without its diagonals; see the previous section for rectangular mesh with two diagonals). The triangular pattern considered here consists of lattice elements that create equilateral triangles and nodes at the intersection points of these lattice elements. There is no flexural moment transfer at nodes, so all elements can be considered truss elements.

One undesirable trait of this pattern may be that it is not possible to make a rectangular-shaped structure without breaking the pattern. Hence, if the pattern is desired to repeat in a strict manner, there are a few options. One may use a trapezoidal shape angled 60º, or a rectangular bound may be used, but edges do not always lay on the boundaries. Alternatively, one can break the pattern and close the edges at the bounds, but then the properties of elements at the edges cannot be calculated theoretically in a practical manner.

The triangular pattern will be used to verify the accuracy of the computational method that is proposed. As it was already calculated by Hrennikoff (1940), the parameters of this pattern can be summarized as follows, where A is the cross-sectional area of the triangular lattice elements regardless of the orientation and a_1 is the side of one triangle and t is the thickness of the prototype plate.

$$A = \frac{\sqrt{3}}{2} a_1 t \tag{3.4}$$

$$\nu = \frac{1}{3} \tag{3.5}$$

Table 3.2 shows the A for different values of a_1.

Table 3.2: Theoretical solutions for cross-sectional areas of the lattice elements in a lattice model with structured mesh – triangular shape.

a_1	A
1/4	0.217
1/3	0.289
1/2	0.433
1	0.866
2	1.732

Note: t=1, ν=1/3

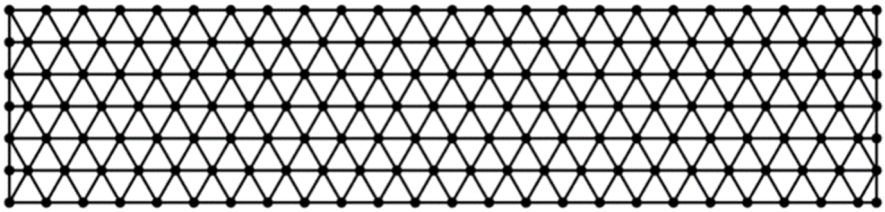


Figure 3.9 (a) Sample regular mesh – triangular mesh. Black dots indicate node locations. Note that the pattern does not repeat at the edges if a rectangular shape is desired, so the theoretical solutions for cross sectional area values are incorrect at the boundaries.

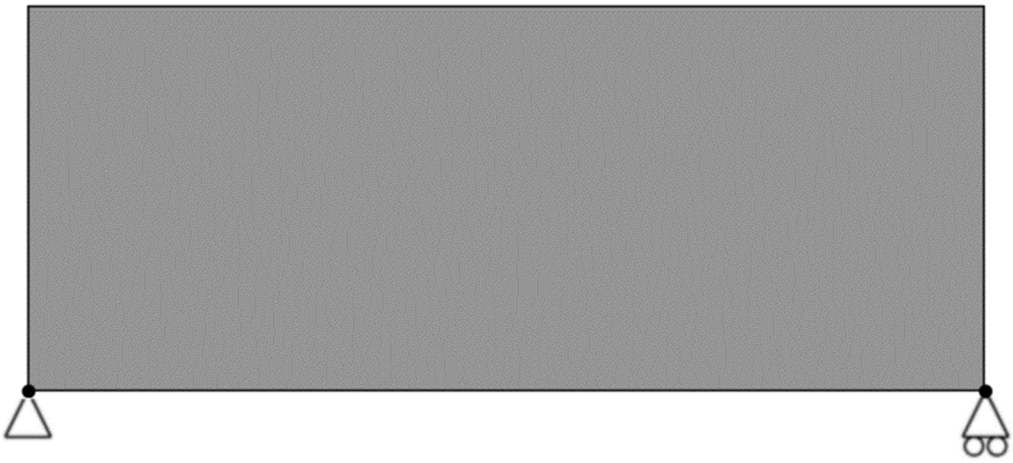
3.1.9 Proposed Method for the Solution of EAs

The core part of the developed program solves the lattice element EAs. The background of the algorithm is based on the energy density equations of continuum mechanics. Based on these energy density equations, a set of equations can be expressed relating the elastic medium and the lattice model (see Figure 3.10). The solution of these EAs may be divided into two phases. These are:

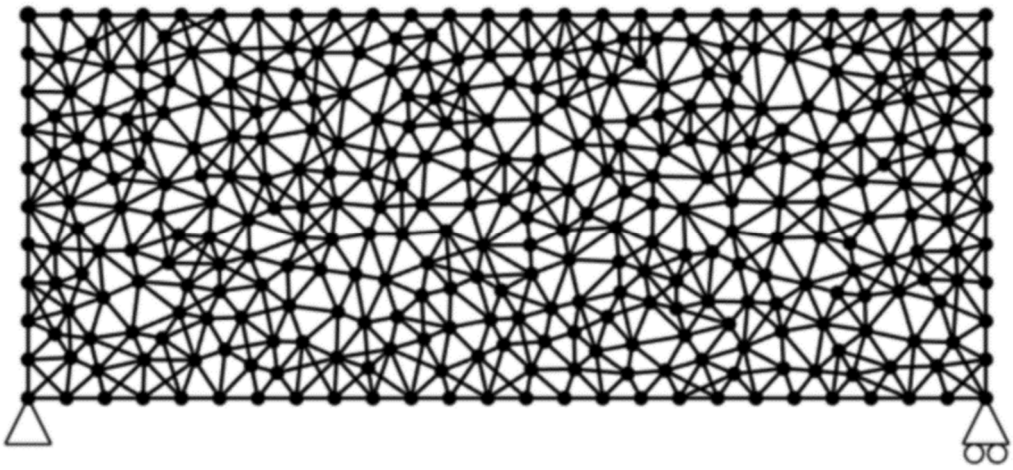
1. Equation creation phase
2. Solution phase

There are 3 different approaches studied for calibration lattice element EAs. These can be listed as follows:

1. The global approach of Jirásek & Bazant (1995)
2. The local approach
3. The hybrid global-local approach



(a)



(b)

Figure 3.10 (a) Sample plate prototype. (b) Corresponding lattice model to represent the behavior of the plate prototype.

3.1.9.1 The Global Approach of Jirásek & Bazant (1995)

Jirásek & Bazant (1995) gives a solution to EAs for random lattices assuming a constant value for all elements regardless of connection or length considerations. This method can be summarized as follows. Consider equibiaxial stress state where there is constant normal stresses and constant normal strains on the body: $\sigma=\sigma_x=\sigma_y$ and $\varepsilon=\varepsilon_x=\varepsilon_y$. The total strain energy of the prototype is equal to:

$$U_{prototype} = \frac{EA_f\varepsilon^2}{1-\nu} \tag{3.6}$$

where E is the elasticity modulus of the prototype, A_f is the total area of the prototype, and ν is the Poisson's ratio and is equal to 1/3.

The total strain energy of the lattice model is equal to:

$$U_{lattice} = \sum_{i=1}^{n} \frac{EAL_i\varepsilon^2}{2} \tag{3.7}$$

where L_i is the length of the ith element of the lattice model, and n is the total number of lattices in the system. E is the elasticity modulus that is assigned to all lattice models and is also equal to the elasticity modulus of the lattice elements.

Since the energies of the two systems must be equal:

$$U_{lattice} = U_{prototype} \tag{3.8}$$

The below equation can be derived:

$$EA_{Bazant} = \frac{3EA_f}{\sum_{i=1}^{n} \mathrm{L_i}}$$

where EA_{Bazant} is the cross-sectional area times the elasticity modulus that is going to be assigned to all of the lattice elements regardless of their positions and lengths (Jirásek & Bazant, 1995).

3.1.9.2 The Local Solution

Equations are expressed by using the strain energy density equivalency between the prototype and the lattice model. In essence, the method is to write these equivalencies and take their partial derivatives such that the equivalencies become independent of the strain components. These energy equivalencies are considered for each node. Tributary body area and the tributary lattice elements for a sample node are shown in Figure 3.11.

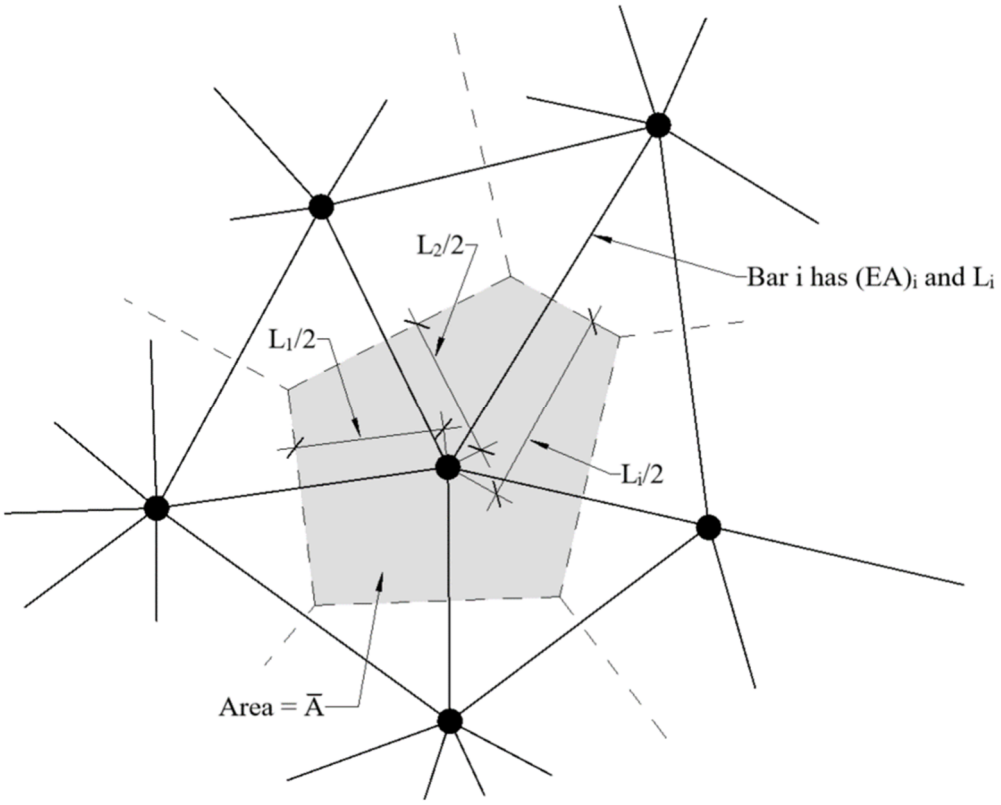


Figure 3.11 Diagram of a sample piece of lattice showing the parameters used in EA calibration calculations.

The strain energy U of a prismatic bar is defined below

$$U = \frac{P^2 L}{2AE} \tag{3.9}$$

In this expression, P is the axial force, A is the cross-sectional area of that bar and E is the elastic modulus of the bar material. Instead of writing the energy equation in terms of forces, the equation can be written in terms of deformation as follows

$$U = \frac{EA\,\delta^2}{2L} \tag{3.10}$$

where δ indicates change in length. The strain energy of a prismatic expression can be expressed in terms of strain as

$$U = \frac{EA\,L\varepsilon^{\,2}}{2} \tag{3.11}$$

Now consider a single node within a lattice model and the tributary piece of plate to that node (Figure 3.12). Consider that this plate of thickness t is subjected to a strain field normal strain along the x direction, ε_x, normal strain along the y direction, ε_y, and in-plane shear strain, γ_{xy}. This is the most general case for the plane stress problem. ε_x, ε_y, and γ_{xy} can take arbitrary values. There are a number of lattice elements that connect to that node. The strain along the element of index i due to this general field of strain can be defined as ε_i as given below:

$$\varepsilon_i = \varepsilon_x \cos^2\theta_i + \varepsilon_y \sin^2\theta_i + \gamma_{xy} \sin\theta_i \cos\theta_i \tag{3.12}$$

Where θ_i is the angle of the lattice element measured from the x axis.

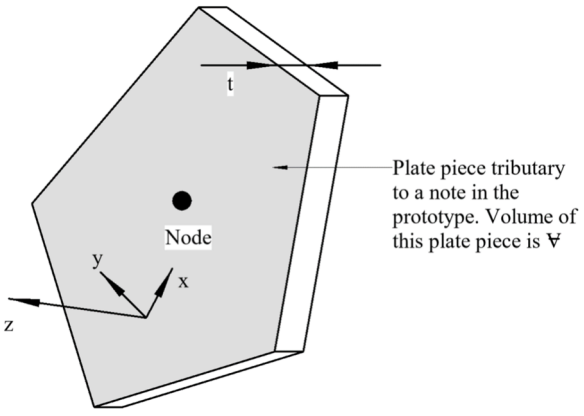


Figure 3.12 Diagram showing a sample piece of the prototype under consideration tributary to each node.

Summing the strain energies due to this general strain field at each element gives:

$$U = \sum_{i=1}^{n} \frac{(EA)_i L_i}{2} \left[\varepsilon_x c^2 + \varepsilon_y s^2 + \gamma_{xy} cs\right]^2 \tag{3.13}$$

Where the total number of elements connected to the node under consideration is n and *c* and *s* are defined as follows:

$$c = \cos\theta_i \tag{3.14}$$

$$s = \sin\theta_i \tag{3.15}$$

If this sum is then divided by the tributary volume to the node (i.e. the tributary area to the node, $\bar{A}$, times the thickness of the plate, t, as shown in Figure 3.12):

$$w = \sum_{i=1}^{n} \frac{(EA)_i L_i}{2\bar{A}t} \left[\varepsilon_x c^2 + \varepsilon_y s^2 + \gamma_{xy} cs\right]^2 \tag{3.16}$$

Where w is the strain energy density for the considered node in the lattice model. If the partial derivatives of the strain energy density w are taken with respect to strains:

$$\frac{\partial^2 w}{\partial \varepsilon_x^2} = \sum_{i=1}^{n} \frac{(EA)_i L_i c^4}{\bar{A}t} \tag{3.17}$$

$$\frac{\partial^2 w}{\partial \varepsilon_y^2} = \sum_{i=1}^{n} \frac{(EA)_i L_i s^4}{\bar{A}t} \tag{3.18}$$

$$\frac{\partial^2 w}{\partial \gamma_{xy}^2} = \sum_{i=1}^{n} \frac{(EA)_i L_i c^2 s^2}{\bar{A}t} \tag{3.19}$$

$$\frac{\partial^2 w}{\partial \varepsilon_x \partial \gamma_{xy}} = \sum_{i=1}^{n} \frac{(EA)_i L_i c^3 s}{\bar{A}t} \tag{3.20}$$

$$\frac{\partial^2 w}{\partial \varepsilon_x \partial \gamma_{xy}} = \sum_{i=1}^{n} \frac{(EA)_i L_i c s^3}{\bar{A} t} \tag{3.21}$$

$$\frac{\partial^2 w}{\partial \varepsilon_x \partial \varepsilon_y} = \sum_{i=1}^{n} \frac{(EA)_i L_i c^2 s^2}{\bar{A} t} \tag{3.22}$$

Now consider the same problem of the elastic plate that is tributary to a node that is under consideration (as illustrated in Figure 3.12). This time, the prototype system is an elastic plate. This system is taken to be a plane stress problem where the x and y directions are the in-plane directions and z is the out-of-plane direction. Hence, the following stresses on the z face are zero:

$$\sigma_z = 0 \tag{3.23}$$

$$\tau_{xz} = \tau_{yz} = 0 \tag{3.24}$$

$$\gamma_{xz} = \gamma_{yz} = 0 \tag{3.25}$$

The stress-strain relationships of this plane stress problem then become:

$$\varepsilon_x = \frac{\sigma_x}{E} - \frac{\nu}{E}\sigma_y \tag{3.26}$$

$$\varepsilon_y = \frac{\sigma_y}{E} - \frac{\nu}{E}\sigma_x \tag{3.27}$$

Strain energy stored in this piece of plate, U can be written as below.

$$U = \frac{1}{2} \forall \varepsilon \sigma \tag{3.28}$$

Where $\forall$ is the volume of this plate, and ε and σ indicate the strains and stresses on this plate. Extracting the stresses and strains in this explicit form for each direction gives this expression below.

$$U = \frac{\forall}{2} \left[\varepsilon_x \sigma_x + \varepsilon_y \sigma_y + \tau_{xy} \gamma_{xy} \right] \tag{3.29}$$

Inserting Equations 3.21 and 3.22 into Equation 3.24 gives this expression:

$$U = \frac{\forall}{2}\frac{E}{1-\nu^2}\varepsilon_x^2 + \frac{\forall}{2}\frac{E}{1-\nu^2}\varepsilon_y^2 + \forall\frac{Ev}{1-v^2}\varepsilon_x\varepsilon_y + \frac{\forall}{2}G\gamma_{xy}^2 \tag{3.30}$$

Where G is the shear modulus and defined as:

$$G = \frac{E}{2(1+\nu)} \tag{3.31}$$

Strain energy density, w is then obtained by dividing the above expression by the volume of the plate piece:

$$w = \frac{1}{2}\frac{E}{1-\nu^2}\varepsilon_x^2 + \frac{1}{2}\frac{E}{1-\nu^2}\varepsilon_y^2 + \frac{Ev}{1-v^2}\varepsilon_x\varepsilon_y + \frac{1}{2}G\gamma_{xy}^2 \tag{3.32}$$

By inserting Equation 3.26 into Equation 3.27, the below expression is obtained for the strain energy density. Note that this expression is only in terms of the elasticity modulus, the Poisson's ratio, and strains.

$$w = \frac{1}{2}\frac{E}{1-\nu^2}\varepsilon_x^2 + \frac{1}{2}\frac{E}{1-\nu^2}\varepsilon_y^2 + \frac{Ev}{1-v^2}\varepsilon_x\varepsilon_y + \frac{1}{2}\frac{E}{2(1+\nu)}\gamma_{xy}^2 \tag{3.33}$$

Taking partial derivatives, all the strain terms can be eliminated from the expression, these expressions are very useful as they are independent of the strain field.

$$\frac{\partial^2 w}{\partial\varepsilon_x^2} = \frac{E}{1-\nu^2} \tag{3.34}$$

$$\frac{\partial^2 w}{\partial\varepsilon_y^2} = \frac{E}{1-\nu^2} \tag{3.35}$$

$$\frac{\partial^2 w}{\partial\gamma_{xy}^2} = \frac{E}{2(1+\nu)} \tag{3.36}$$

$$\frac{\partial^2 w}{\partial\varepsilon_x\partial\gamma_{xy}} = 0 \tag{3.37}$$

$$\frac{\partial^2 w}{\partial\varepsilon_x\partial\gamma_{xy}} = 0 \tag{3.38}$$

$$\frac{\partial^2 w}{\partial \varepsilon_x \partial \varepsilon_y} = \frac{E\nu}{1-\nu^2} \tag{3.39}$$

By equating Equation (3.17) through Equation (3.22) to Equation (3.34) through Equation (3.39), the below expressions are obtained. These expressions relate the lattice system and the prototype in terms of just the elasticity modulus, Poisson's ratio, tributary area, and lattice element lengths and orientations.

$$\frac{\partial^2 w}{\partial \varepsilon_x^2} = \frac{E}{1-\nu^2} = \sum_{i=1}^{n} \frac{(EA)_i L_i c^4}{\bar{A}t} \tag{3.40}$$

$$\frac{\partial^2 w}{\partial \varepsilon_y^2} = \frac{E}{1-\nu^2} = \sum_{i=1}^{n} \frac{(EA)_i L_i s^4}{\bar{A}t} \tag{3.41}$$

$$\frac{\partial^2 w}{\partial \gamma_{xy}^2} = \frac{E}{2(1+\nu)} = \sum_{i=1}^{n} \frac{(EA)_i L_i c^2 s^2}{\bar{A}t} \tag{3.42}$$

$$\frac{\partial^2 w}{\partial \varepsilon_x \partial \gamma_{xy}} = 0 = \sum_{i=1}^{n} \frac{(EA)_i L_i c^3 s}{\bar{A}t} \tag{3.43}$$

$$\frac{\partial^2 w}{\partial \varepsilon_x \partial \gamma_{xy}} = 0 = \sum_{i=1}^{n} \frac{(EA)_i L_i c s^3}{\bar{A}t} \tag{3.44}$$

$$\frac{\partial^2 w}{\partial \varepsilon_x \partial \varepsilon_y} = \frac{E\nu}{1-\nu^2} = \sum_{i=1}^{n} \frac{(EA)_i L_i c^2 s^2}{\bar{A}t} \tag{3.45}$$

Note that Equation (3.42) and Equation (3.45) are identical for ν=1/3. Hence, only for this value of ν, a perfect analogy may exist.

EA_i (EA for node i) are the unknowns to be solved in this set of equations. The number of unknowns in this system is equal to the number of elements in the lattice model. The number of equations, on the other hand, is 5×number of nodes. Although the number of independent equations in this system is lower than the total number of

equations, it is not possible to assess which equations or which chain of equations would yield a redundancy. So all of these equations are considered during the solution phase.

By using this method of equating the lattice model to the prototype on the basis of strain energy densities, the EAs of lattice elements can be solved. For doing so, all the nodes in the lattice model are considered and this procedure of finding the EAs due to each node needs to be repeated. At each node considered, for each element, there appears a coefficient for the $(EA)_i$, and the unknown $(EA)_i$ itself, and one 'solution' for the strain energy density. That coefficient is stored in the matrix $\underline{\underline{A}}$, the 'solution' is stored in the column vector $\underline{rhs}$. Vector $\underline{EA}$ includes unknown parameters.

Parameter *nel* indicates the number of elements in the lattice model and the parameter *nn* indicates the number of nodes in the lattice system. $\underline{\underline{A}}$ is a coefficient matrix in the form of

$$\underline{\underline{A}} = \begin{bmatrix} A_{1,1} & \cdots & A_{1,nn\times 5} \\ \vdots & & \vdots \\ \vdots & & \vdots \\ \vdots & & \vdots \\ \vdots & & \vdots \\ A_{nel,1} & \cdots & A_{nel,nn\times 5} \end{bmatrix} \tag{3.46}$$

Note that $\underline{\underline{A}}$ is not a square matrix. It has *nn*×5 number of rows, but *nel* number of columns.

$$\underline{EA} = \begin{bmatrix} EA_1 \\ \vdots \\ EA_{nel} \end{bmatrix} \tag{3.47}$$

Column vector $\underline{EA}$ has *nel* number of columns, thus the system has *nel* number of unknowns to be solved.

$$\underline{\text{rhs}} = \begin{bmatrix} rhs_1 \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ rhs_{nn\times 5} \end{bmatrix} \tag{3.48}$$

Column vector $\underline{\text{rhs}}$ has $nn\times 5$ number of rows, thus the system has $nn\times 5$ number of equations, but note that the unique number of equations is actually lower.

It is possible to write the system equation as a matrix expression as

$$\underline{\underline{A}}\ \underline{EA} = \underline{rhs} \tag{3.49}$$

Alternatively, one can write these equations in a more explicit manner:

$$\begin{bmatrix} A_{1,1} & \cdots & A_{1,nel} \\ \vdots & \ddots & \vdots \\ \vdots & \ddots & \vdots \\ \vdots & \ddots & \vdots \\ A_{nn\times 5,1} & \cdots & A_{nn\times 5,nel} \end{bmatrix} \begin{bmatrix} EA_1 \\ \vdots \\ EA_{nel} \end{bmatrix} = \begin{bmatrix} rhs_1 \\ \vdots \\ \vdots \\ \vdots \\ A_{nn\times 5} \end{bmatrix} \tag{3.50}$$

If the $\underline{\underline{A}}$ and $\underline{EA}$ are known and $\underline{\text{rhs}}$ would be sought, it would be possible to solve this system in a straightforward manner, but this is not the case. Here, matrix $\underline{\underline{A}}$ and vector $\underline{\text{rhs}}$ are known, and vector $\underline{EA}$ is to be solved.

There is no unique solution to the set of equations. Alternatively, one can consider this as a minimization problem. An $\underline{EA}$ vector is sought so as to minimize the norm of the error $\underline{e}$ vector.

$$\underline{e} = \underline{\underline{A}}\ \underline{EA} - \underline{rhs} \tag{3.51}$$

A least-squares approach can be derived by multiplying both sides of the equation by $\underline{\underline{A}}^T$ from the left:

$$\underline{\underline{A}}^T \, \underline{\underline{A}} \, \underline{EA} = \underline{\underline{A}}^T \, \underline{rhs} \tag{3.52}$$

A regularization is introduced to constrain the variation of EA values by introducing a constraint for the second norm of the EA vector

$$e = \frac{1}{2}\left(\underline{\underline{A}} \, \underline{EA} - \underline{rhs}\right)^{\boldsymbol{T}} \left(\underline{\underline{A}} \, \underline{EA} - \underline{rhs}\right) + \lambda\left(\frac{\underline{\underline{EA}}^T \, \underline{\underline{EA}}}{2} - R\right) \tag{3.53}$$

Here e is the scalar error to be minimized and R is a constraint defined as:

$$R = \frac{\underline{\underline{EA}}^T \, \underline{\underline{EA}}}{2} \tag{3.54}$$

λ is a Lagrange parameter that is unknown. Hence, there are two unknowns to be found, EA and λ. A number of trial-and-error solutions are required to determine which λ gives the best solution to the EAs such that the lattice model behavior is reasonably close to the reference finite element shell model. It was found that for the random meshes, $\lambda=10^{-8}$ gives the best results for this particular study. Although a fixed λ was used throughout this study, it should be noted that λ may or may not be dependent on the size of the problem under consideration and a further study is suggested for the selection of this parameter.

The regularized solution (minimizing Equation 3.52) takes the form

$$(\underline{\underline{A}}^T \, \underline{\underline{A}} + \lambda \underline{\underline{I}}) \, \underline{EA} = \underline{\underline{A}}^T \, \underline{rhs} \tag{3.55}$$

where I is the identity matrix. Note that Equation (3.55) closely resembles Equation (3.52), except for the addition of λ to the diagonal of the coefficient matrix.

Solving EA gives the elastic modulus times the cross-sectional area for each of the lattice elements. EAs for the elements that are within the boundaries are obtained using this approach. Elements laying at the boundaries however are assigned the average EA of internal elements.

3.1.9.3 Hybrid Global-Local Solution

EA_{Bazant} obtained in Section 3.1.9.1 can be utilized to regularize the Local Approach presented earlier which will be called the Hybrid Global-Local Solution. The scalar error term in Equation (3.53) can be modified as:

$$e = \frac{1}{2}\left(\underline{\underline{A}}\,\underline{EA} - \underline{rhs}\right)^T \left(\underline{\underline{A}}\,\underline{EA} - \underline{rhs}\right) + \frac{\lambda}{2}\left(\underline{EA} - \underline{EA}_{Bazant}\right)^T \left(\underline{EA} - \underline{EA}_{Bazant}\right) \tag{3.56}$$

Opening the equations into explicit forms:

$$e = \frac{1}{2}\left(\underline{EA}^T \underline{\underline{A}}^T \underline{\underline{A}}\,\underline{EA} - \underline{EA}^T \underline{\underline{A}}^T \underline{rhs} - \underline{rhs}^T \underline{\underline{A}}\,\underline{EA} + \underline{rhs}^T \underline{rhs}\right) + \frac{\lambda}{2}\left(\underline{EA}^T\,\underline{EA} - \underline{EA}^T\,\underline{EA}_{Bazant} - \underline{EA}_{Bazant}^T\,\underline{EA} + \underline{EA}_{Bazant}^T\,\underline{EA}\right) \tag{3.57}$$

$$e = \frac{1}{2}\underline{EA}^T \underline{\underline{A}}^T \underline{\underline{A}}\;\underline{EA} - \underline{EA}^T \underline{\underline{A}}^T \underline{rhs} - \underline{rhs}^T \underline{\underline{A}}\;\underline{EA} + \frac{1}{2}\underline{rhs}^T\,\underline{rhs} + \frac{\lambda}{2}\,\underline{EA}^T\,\underline{EA} - \lambda \underline{EA}^T\,\underline{EA}_{Bazant} - \frac{\lambda}{2}\underline{EA}_{Bazant}^T\,\underline{EA} \tag{3.58}$$

when the error is minimized by taking partial derivatives, the below equation is obtained

$$\left(\underline{\underline{A}}^T \underline{\underline{A}} + \lambda \underline{\underline{I}}\right)\underline{EA} = \underline{\underline{A}}^T \underline{rhs} + \lambda\,\underline{EA}_{Bazant} \tag{3.59}$$

This equation is an improved form of Equation (3.55), such that it takes both global energy constraints and local energy constraints at each of the Voronoi cells into consideration.

The factor λ determines how close the solution will be to the local solution. If λ is large, then the solution is close to the solution in Jirásek & Bazant (1995). If λ is

small, then the solution is close to the solution shown in Equation (3.55). For the solutions that are presented in this study, λ is chosen as the greater of absolute maximum of the element of matrix A and absolute maximum of element of matrix rhs.

This selection provides that the value of λ, which has the same balancing effect on the solution regardless of the problem size, element lengths, elasticity of the material, etc. This value was observed to give reasonable results for the elastic deformation problems considered.

3.1.9.4 Comparison of EA Solutions with the Exact Solution for the Regular Mesh

Calculated values of the exact values by Hrennikoff (1940) are given in Section 3.1.7. When the element size is 1, the exact solution for such lattices is 0.53 for orthogonal elements and 0.75 for diagonal elements. The distribution of exact EA values is given in Figure 3.13.

The distribution of EAs calculated by the procedure as described in Section 3.1.9 for a regular lattice (rectangles with two diagonals) is compared with the exact solutions. Figure 3.14 shows the comparison of EAs.

It was observed that the method is able to calibrate EAs near the exact values, but there appears a pattern due to boundary effects. The high EA elements and low EA elements alternate at each strip starting from the boundaries through the center of the shape (Figure 3.14). When the elements are located near the boundaries, then the deviation of EA values from the exact values is higher, but when the elements are located near the boundaries, the EAs found are slightly different than the exact values. These errors are shown in Figure 3.15. Errors in the figure are defined as follows:

$$Error\ in\ lattice\ EA = \frac{(EA_{lattice} - EA_{exact})}{EA_{exact}} \tag{3.60}$$

The difference can be attributed to the nodes at the boundaries having fewer elements connecting to them. It must be noted that though these differences are small, the EA values near the boundaries are very close to the exact EA values.

It was observed that when a random lattice is used, the EA values are obtained around the exact EA values for a regular lattice, but there appears no pattern in the distribution of EAs to the naked eye (Figure 3.18). This is expected because the node distribution and connectivity are random (generated with the rules explained in Section 3.1.1).

Figure 3.16 shows the distribution of errors in EA values (as defined in (3.60)). As the EA values are all the same in the solution of Jirásek & Bazant (1995), there occurs two types of errors, one in the orthogonal elements and the other in the diagonal elements. In the hybrid solution, errors are similar to the solution of Jirásek & Bazant (1995), but the edges seem to have slightly different EA error values. From these figures, it cannot be stated that the EA distribution necessarily improved or worsened.

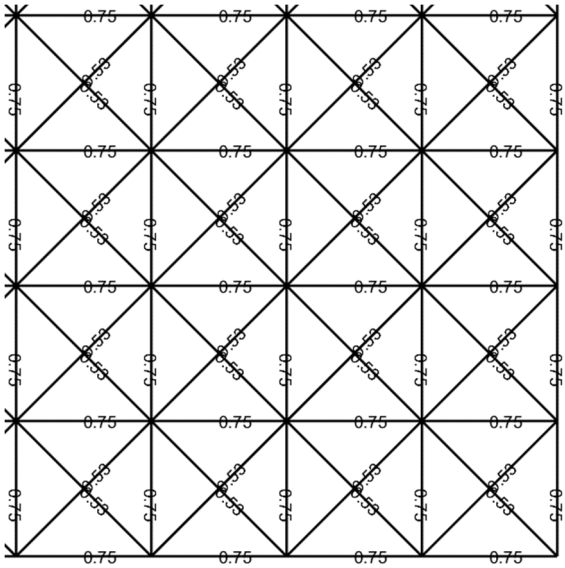
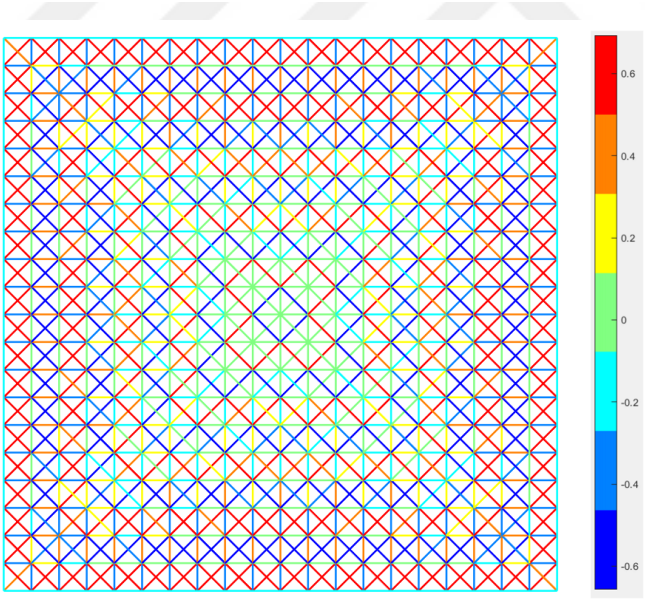
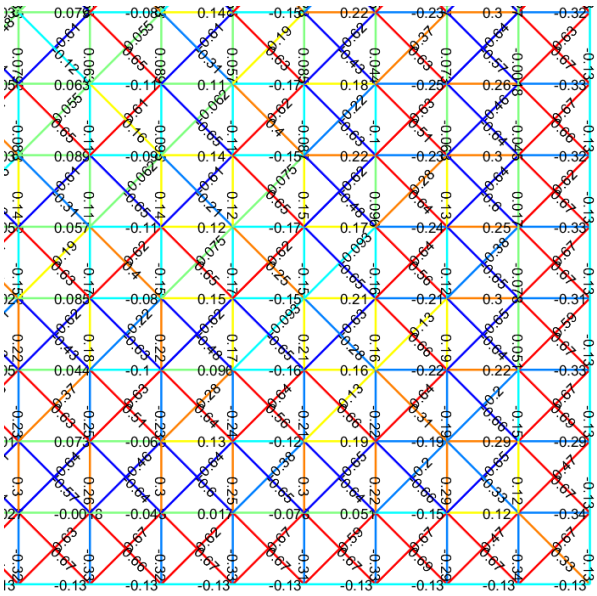


Figure 3.13 Regular lattice meshed with rectangle with two diagonals, element size is 1 EA values are written besides the elements. Exact EAs are assigned to the elements that are found from analytical methods (Hrennikoff, 1940).

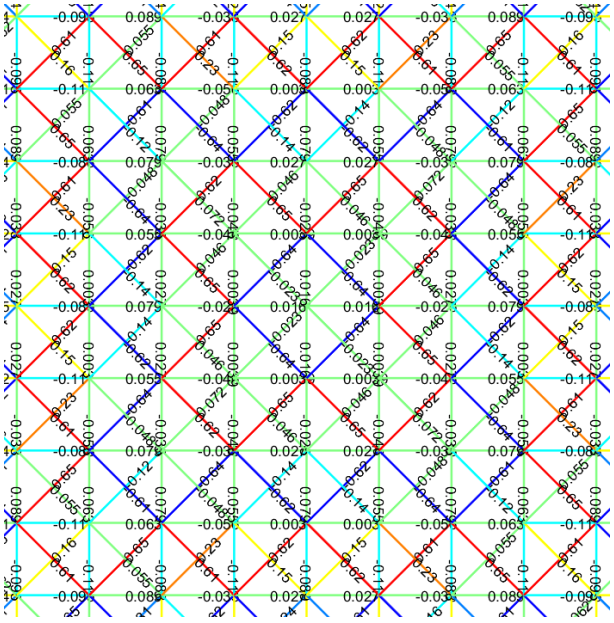


(a)

Figure 3.14 Diagram showing a 20x20 square regular lattice meshed with rectangle with two diagonals, element size is 1. Lattice elements are solved using the local approach and colored according to their EA values.

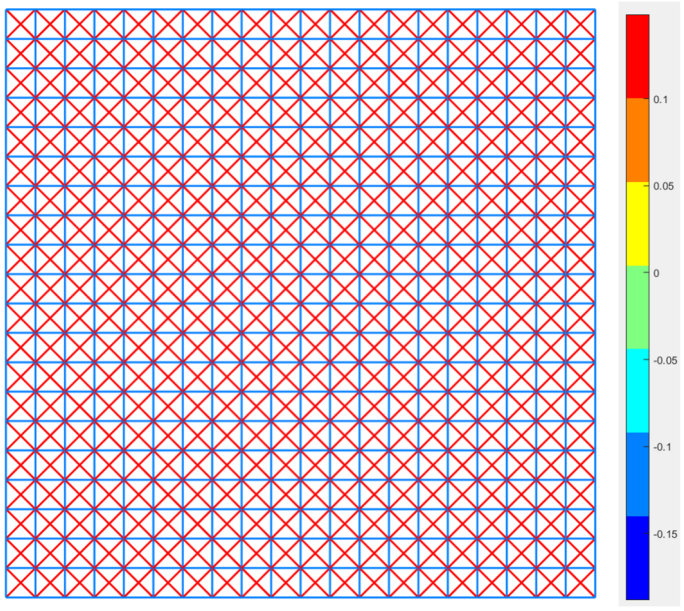


(a)

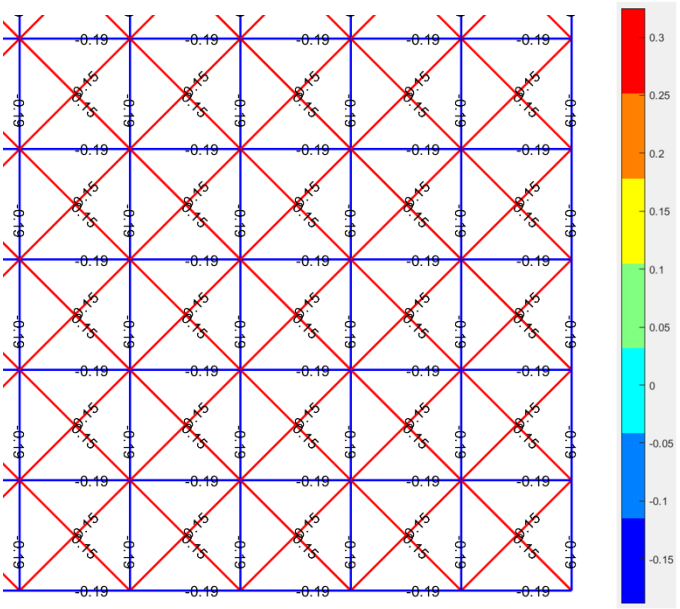


(b)

Figure 3.15 (a) Corner of the lattice that is shown at Figure 3.14 zoomed in. Error values solved for each lattice element is given with numbers. (b) Center of the lattice that is shown at Figure 3.14 zoomed in. Relative error in EA values solved for each lattice element compared to the exact EA values are given with numbers.

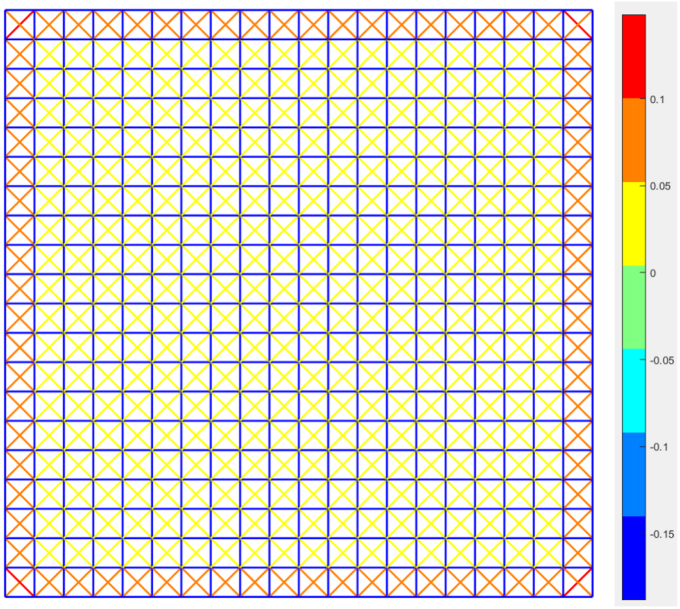


(a)

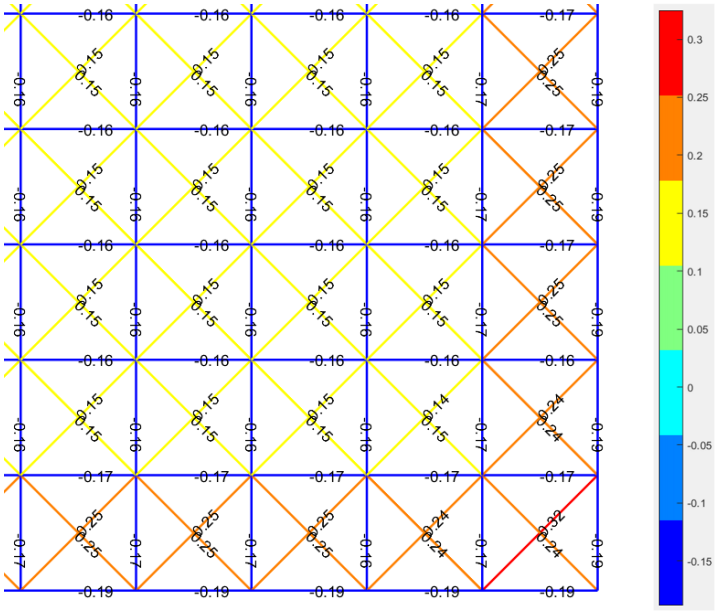


(b)

Figure 3.16 (a) Diagram showing a 20x20 square regular lattice meshed with rectangle with two diagonals, element size is 1. Lattice elements are solved using Jirásek & Bazant (1995) approach and colored according to their EA values. (b) Closeup view of the mesh at the corner.



(a)



(b)

Figure 3.17 (a) Diagram showing a 20x20 square regular lattice meshed with rectangle with two diagonals, element size is 1. Lattice elements are solved using the hybrid approach and colored according to their EA values. (b) Closeup view of the mesh at the corner.

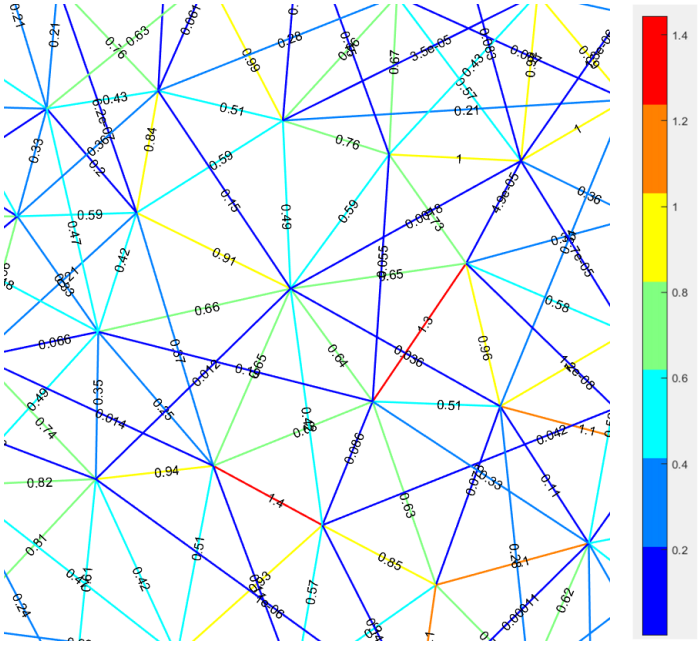


Figure 3.18 Close up view of a non-uniform lattice. Average element size is 1. Lattice elements are colored according to their EA values.

3.2 Solution for The Deformation Field

The nodal displacements are calculated assuming small deformations and linear elasticity. Each element has an assigned EA value. The solution is carried out using the direct stiffness method.

3.3 The User-Friendly Lattice Model Program with Graphical User Interface

Although there is considerable academic material on the theoretical background of lattice models (see Chapter 2), there was little effort to make the lattice models useable in daily calculations. It is believed that creating versatile and daily-use software is one of the important steps to making lattice models available for both the research community and practicing engineers. The developed software allows users

to draw 2D orthogonal shapes with arbitrary boundary conditions and loads. This program is named “User-Friendly Lattice Model Program with Graphical User Interface” throughout this work and abbreviated as LM. Figure 3.19 shows a screenshot of the program. MATLAB platform is used to create this program as it is a relatively high-level language making the programming phase easier. On the other hand, as a downside, this increases the computing times. But since this is a proof-of-concept study, one can recreate this software or a part of this software as needed with any other lower-level programming language as desired. One other advantage of MATLAB is that it has built-in functions to create a graphical user interface with relative ease.

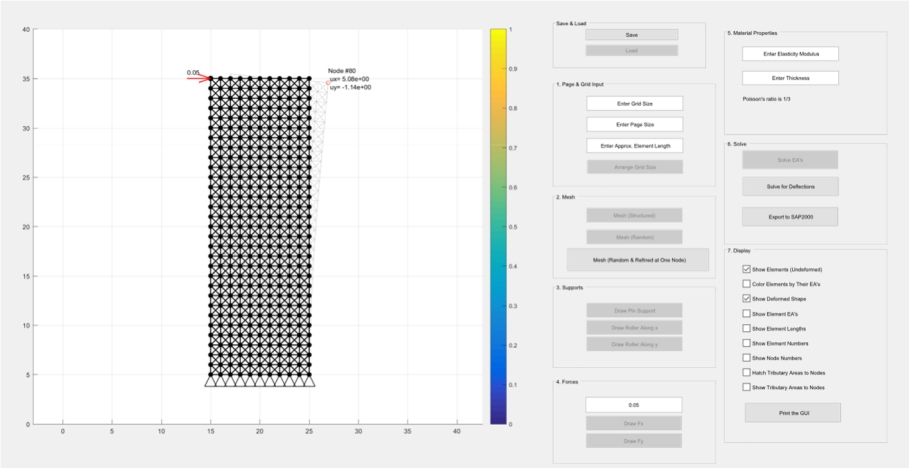


Figure 3.19 LM user interface showing a sample cantilevered wall-like structure under a lateral force.

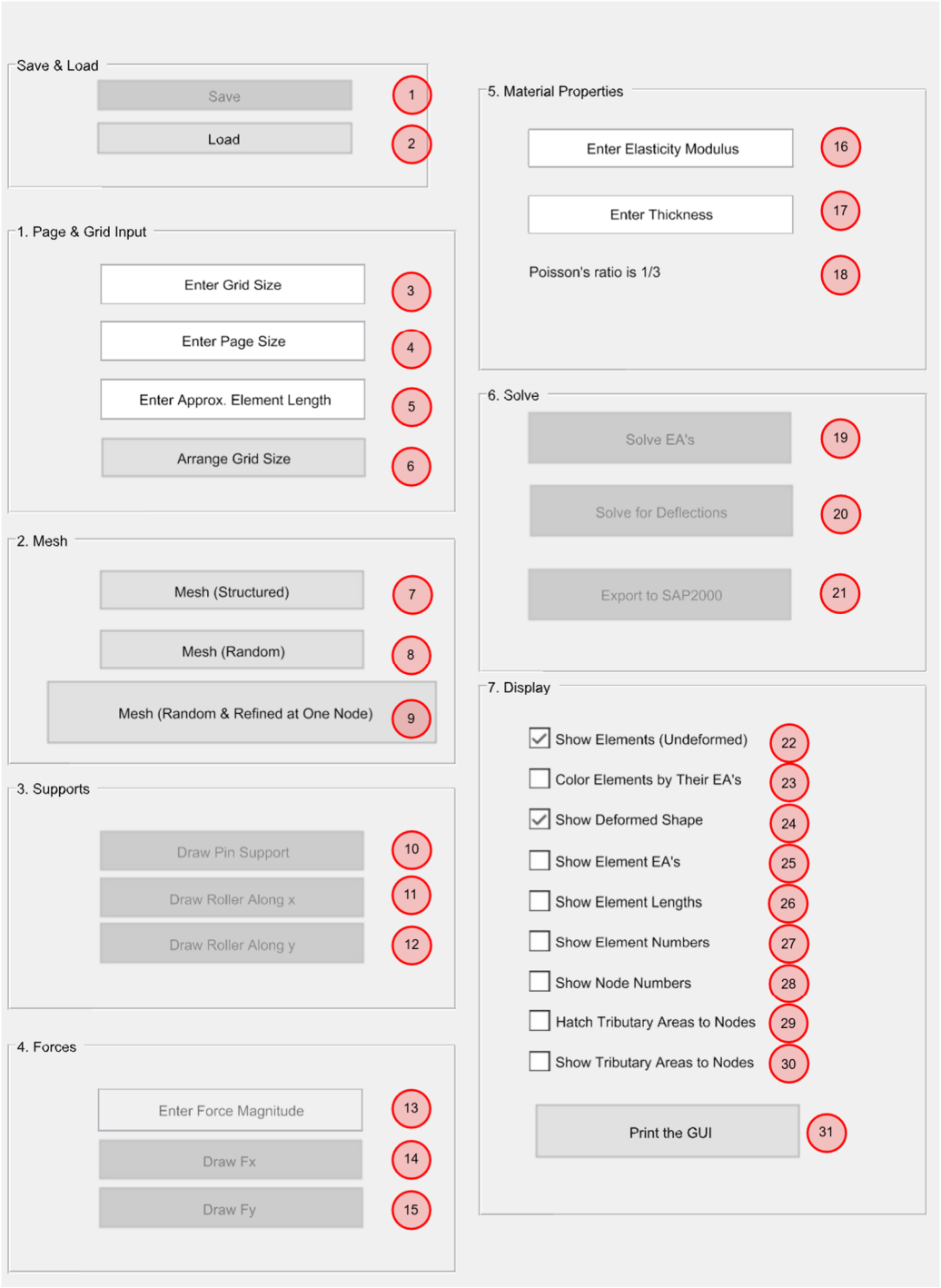


Figure 3.20 Control panel of LM. User interactive fields and buttons are numbered within red circles.

Explanations of buttons in the control panel are given below (numbers are indicated in Figure 3.20):

1. Save: This button saves the finalized state of the calculation so that it can be recalled later. All the node information, member information, EAs, deflections, and figures are saved. This button is only enabled after all the calculations are completed (deflected shape has been calculated). The location of the saved file is the location of the MATLAB script. The work is saved with the date and time of the instance of the button being clicked so as to prevent any overwriting of previously saved works.
2. Load: This button loads finalized state of a calculation. All the node information, member information, EAs, deflections, and figures are recalled and displayed. This button is only enabled if the user has not clicked to draw any shape. After the user clicks on the shape window, then the program assumes that the user does not intend to load any previous work, but to create a new calculation, so the button is disabled automatically to prevent the creation of any overlapping models.
3. Enter Grid Size: This text box is used to take the input of the grid size from the user. Since the program only allows the user to draw by snapping to the grids, this field is necessary to be edited according to the fineness of the geometry to be studied.
4. Enter Page Size: This text box is used to take the input of the page size from the user. Since the program only allows the user to draw within the window limits, this field is necessary to be edited according to the overall size of the geometry to be studied. See Figure 3.21 for a sample page size of 15 units.
5. Enter Approx. Element Length: This text box is used to take input for the approximate element lattice element length. If a regular mesh is to be used, then the “approximate element size” is exactly equal to the actual element size to be created. If a random mesh is to be used, then “Approx. Element Size” gives the average element size to be created in the lattice mesh.

6. Arrange Grid Size - This button sets and displays the user's entered grid size and page sizes on the screen.
7. Mesh (Structured) - Creates a structured mesh within the bounds that the user has drawn. This structured mesh is a “Rectangle with two diagonals” as explained in Section 3.1.7.
8. Mesh (Random) - Creates a type 2 random mesh (without refinement) within the bounds that the user has drawn. This type of mesh is explained in Section n3.1.2.
9. Mesh (Random & Refined at One Node) - Creates a random mesh (with refinement) within the bounds that the user has drawn.
10. Draw Pin Support - Places a pin support at the closest node to the user cursor click location.
11. Draw Roller Along x - Places a roller support along x direction at the closest node to the user cursor click location. These supports restrain their nodes along the y direction, but they are released along the x direction. See Figure 3.22 for a demonstration of the roller supports along the x direction.
12. Draw Roller Along y - Places roller support along y direction at the closest node to the user cursor click location. These supports restrain their nodes along the x direction, but they are released along the y direction.
13. Enter Force Magnitude - This field receives the user input for the force magnitude to be applied.
14. Draw Fx - Draws a force along the x direction at the closest node to the user cursor click location. The magnitude of this force is input at the field “Enter Force Magnitude”.
15. Draw Fy - Draws a force along the y direction at the closest node to the user cursor click location. The magnitude of this force is input at the field “Enter Force Magnitude”.
16. Enter Elasticity Modulus - This field takes the input from the user for the Elasticity Modulus for the in-plane problem under consideration.

17. Enter Thickness - This field takes the input from the user for the thickness of the in-plane problem under consideration.
18. Solve EAs. The program solves the EAs of the lattice elements. The method for solving the EAs is explained in section 3.1.9.2.
19. Solve for Deflections - Nodal deflections are solved elastically.
20. Export to SAP2000 - This button is used to export the lattice model into SAP2000. A text file in .e$k format that is recognized by SAP2000 is created. Then the user can import that .e$k file into SAP2000. All the node information, element information, releases, stiffness modifiers, and loads are exported. One may also see the contents of .e$k by opening it with a text editor. Also, when opened with a text editor, this file is generally long, but the syntax is rather readable.
21. Show Elements (Undeformed) - This shows the undeformed lattice model.
22. Color Elements by Their EAs - Elements are displayed with a gradient of color according to their EA values (See Figure 3.23).
23. Show Deformed Shape - Shows the deformed lattice model under the loads applied. This deformed shape is scaled such that it does not go off-screen and it is suitably sized for user inspection.
24. Show Element Lengths – Element lengths are displayed in the figure.
25. Show Element EAs - Element EAs are displayed in the figure.
26. Show Element Numbers - Element numbers are displayed in the figure.
27. Show Node Numbers - Node numbers are displayed in the figure.
28. Hatch Tributary Areas to Nodes - Voronoi tesselation of the lattice model is displayed. The area tributary to each node is hatched with a light grey color (See Figure 3.24).
29. Show Tributary Areas to Nodes - Area values tributary to each node is displayed.
30. Print the GUI – Prints the entire interface of the program with the current figure on the display. Prints are saved as Scalable Vector Graphics files with the extensions of .svg so that they do not lose their quality when resized.

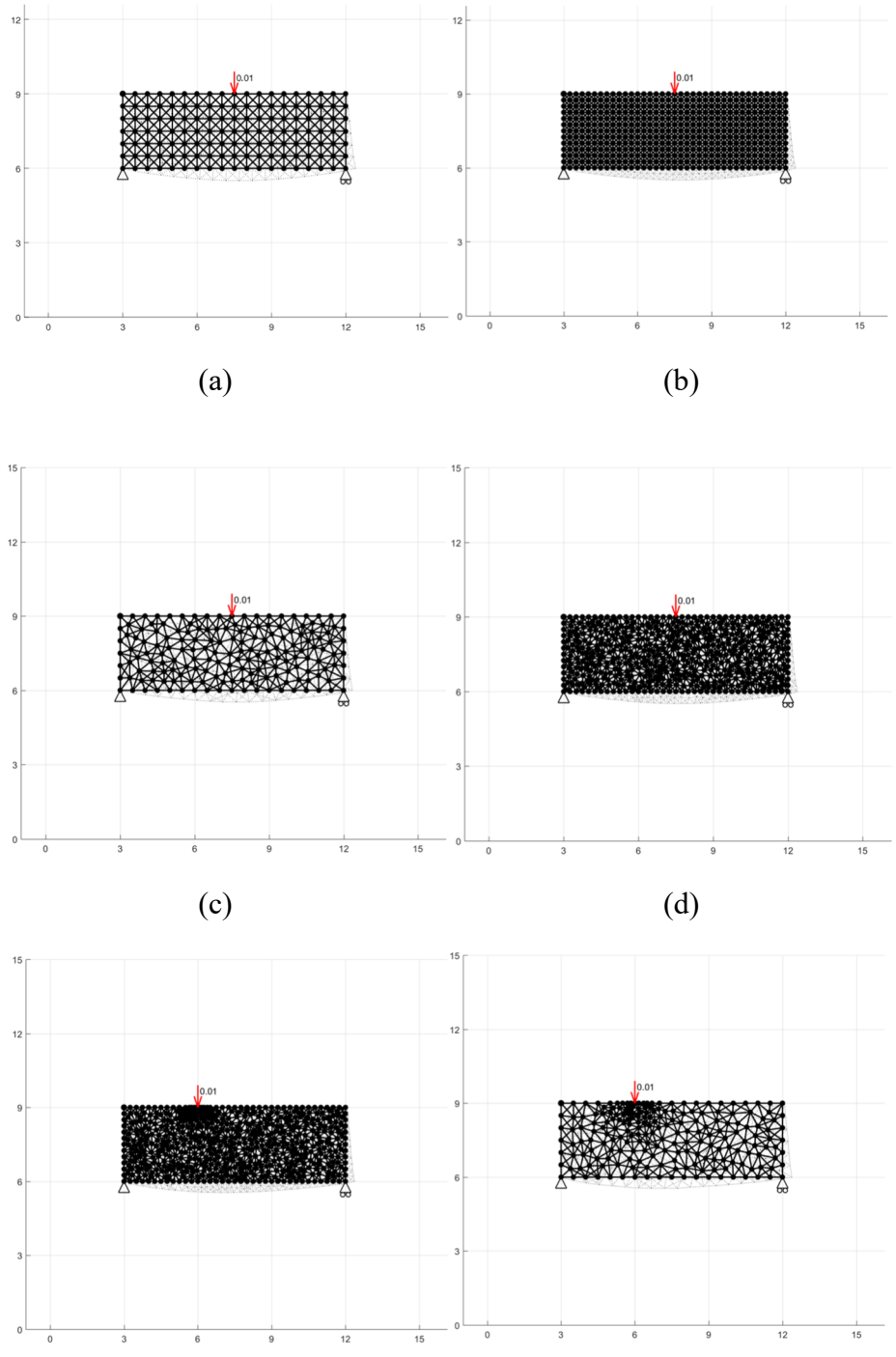


Figure 3.21 Different types of meshes with “Page Size” =15. Left column shows meshes with “Approx. Element Size” = 0.5. Right column shows meshes with “Approx. Element Size” = 0.25.

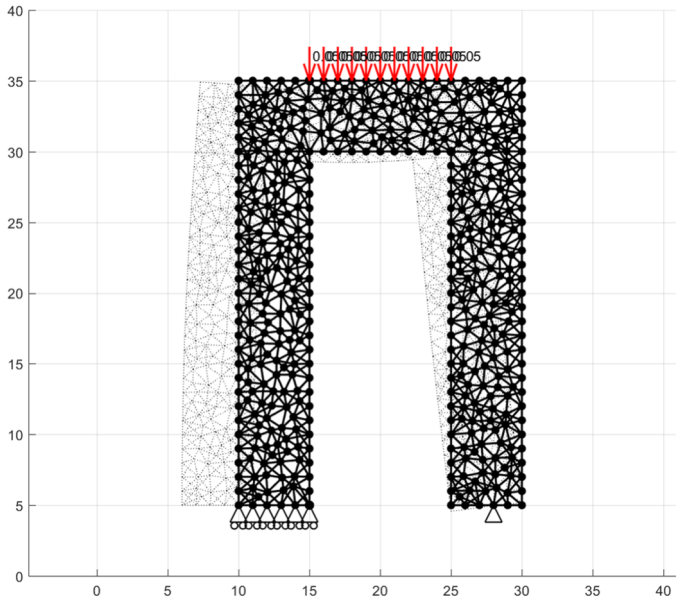


Figure 3.22 Sample lattice model showing the effect of a roller support that is released along the x direction.

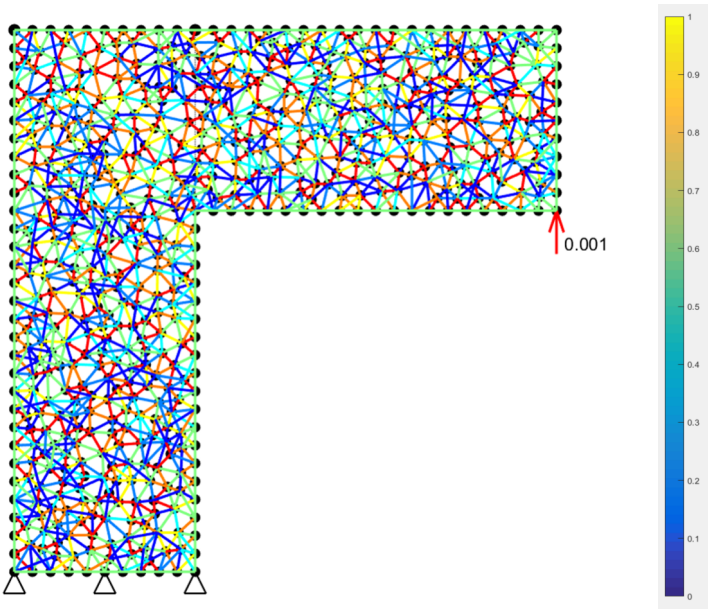


Figure 3.23 Elements of the lattice model are colored with a gradient of color according to their EAs.

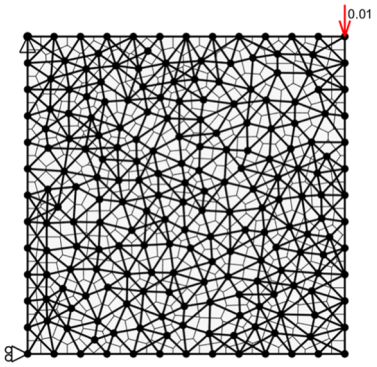
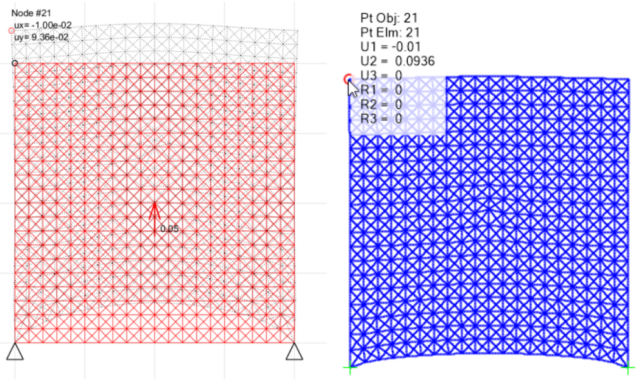


Figure 3.24 Lattice model showing the tributary areas obtained by Voroni Tesselation hatched corresponding to each node.

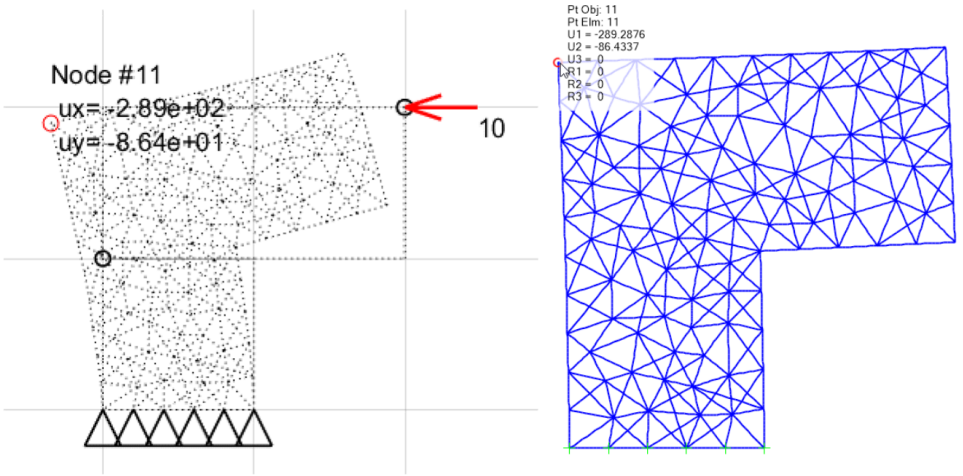
3.4 Exporting the Lattice Model to SAP2000

The LM can export the lattice model into a .e$k file. This is a type of file that SAP2000 can read. Then the user can import that .e$k file into SAP2000 after opening the software. It was observed that the results that were obtained in SAP2000 were identical to those obtained from the built-in solver of LM. A number of comparisons are presented below to show that calculated deformations are identical.

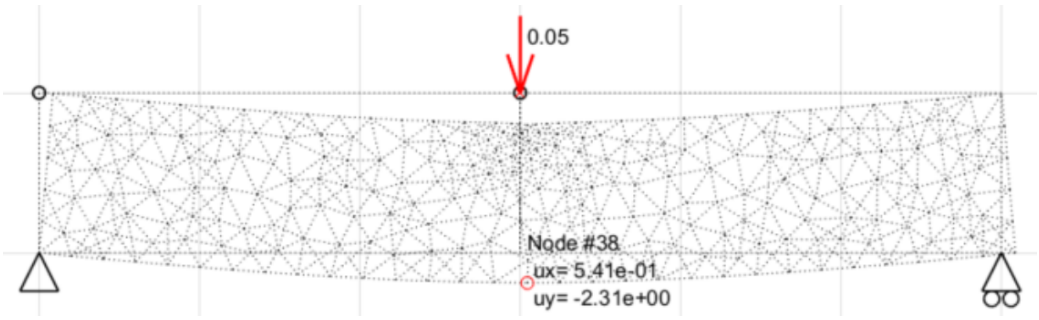


(a) (b)

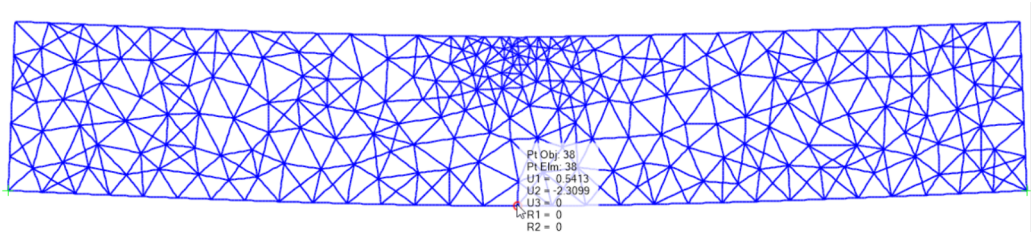
Figure 3.25 Deflections of a lattice model of a square shaped structure is solved by the built-in solver of LM in (a) and solved in SAP2000 (b). The displacements are identical.



(a) (b)



(c)



(d)

Figure 3.26 Deformed shape of a lattice model of a L-shaped structure is solved by the built-in solver of LM and solved in SAP2000 (b). Deflections of a lattice model of a simply supported beam is solved by the built-in solver of LM (c) and solved in SAP2000 (d).

CHAPTER 4

DISCUSSION OF RESULTS

4.1 Case Studies

A number of different examples were solved to test the proposed method of calibrating EAs. Lattice models are compared with the equivalent shell models. A sufficiently fine shell model is obtained by gradually making denser mesh. The deflection of the reference node is given in Table 4.1. A quadratic element size of 1/8 is found to be sufficiently fine.

Table 4.1 Theoretical solutions for cross sectional areas of the lattice elements in a lattice model with structured mesh - rectangular with two diagonals.

Quadratic shell element size	Deflection of the reference node
1	0.2643
1/2	0.2666
1/4	0.2676
1/8	0.2675

Four types of sample structures are solved with lattice models and then compared with the equivalent shell models. For all structures, the Poisson's ratio is 1/3, the elasticity modulus is 1, and the plate thickness is 1. These values are chosen to simplify the presentation of the results.

The first problem is the axially loaded rod. General dimensions, loading, and boundary conditions of the structure are visually summarized in Figure 4.1. The nodes at the left tip are assigned pin supports. The nodes at the right tip are assigned loads that put the rod under tension. The magnitude of these loads sums up to 0.001. These loads at the tip are distributed such that the corners receive one part load and the nodes that are not at the corners are receiving two parts of the load in order to be

able to represent a distributed load as much as possible. Note that only random mesh with d=1/4 is shown in the figure, but regular meshes and different d values are also investigated.

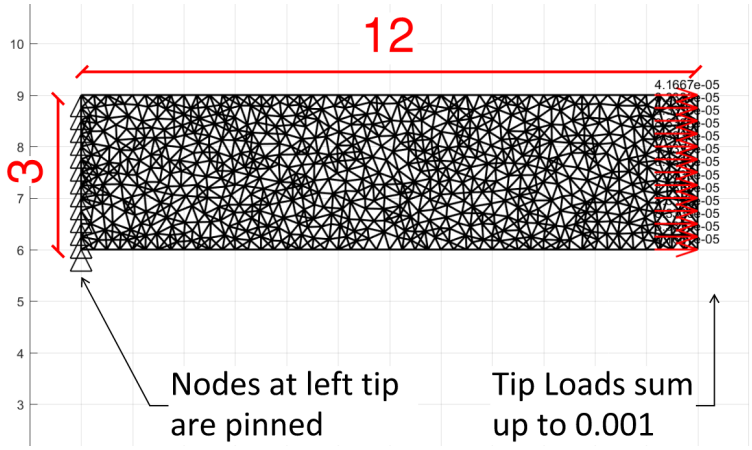


Figure 4.1 Lattice model of the axially loaded rod.

The second problem is the cantilever beam. Dimensions and boundary conditions of the structure are the same as the axially loaded rod, but the only difference is that the loads that are applied at the tip of the structure are along the y direction. Figure 4.2 visualizes the structure.

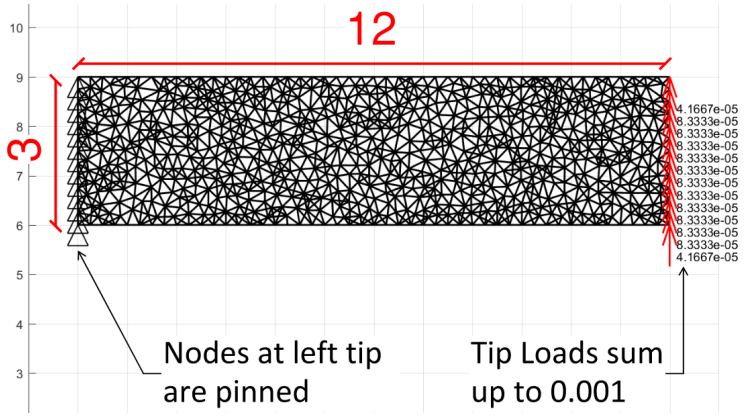


Figure 4.2 Lattice model of the cantilevered beam.

The third problem is the cantilever beam with a hole in the middle. This is the same problem as the cantilevered beam, but the only difference is that this structure has a hole in the center as shown in Figure 4.3.

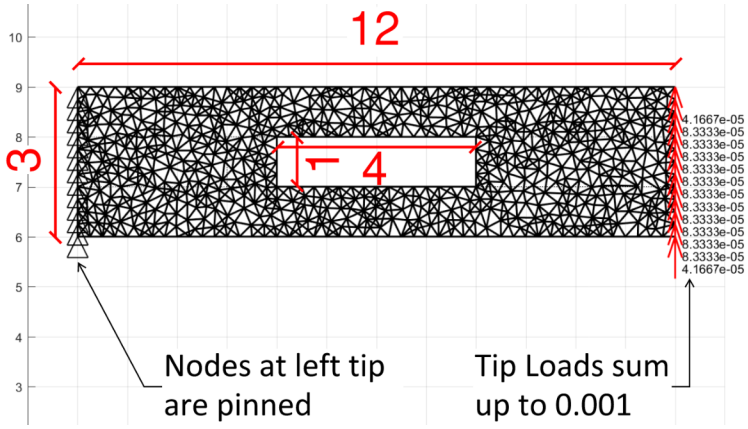


Figure 4.3 Lattice model of the beam with hole.

The fourth problem is the L-shaped structure. Dimensions, loading, and boundary conditions of the structure are visually summarized in Figure 4.4. The nodes at the bottom of the column are assigned pin supports. The nodes at the right tip are assigned loads that put the structure under flexure similar to the cantilevered beam problem.

Each of these structures are built by using Random Mesh: Type 1, Random Mesh: Type 2 and Random Mesh: Type 3 (as explained in Sections 3.1.1, 3.1.2, 3.1.3) and each of these mesh types are solved using the global approach (Section 3.1.9.3), the local approach (Section 3.1.9.2) and the hybrid approach (Section 3.1.9.3). Random Mesh: Type 4 (as explained in Section 3.1.4) is only solved with the global approach.

Each of the above cases are run 100 times when the element size, d is equal to 1. The amount of runs increased to 200 when d=1/2, to 400 when d=1/4 and to 800 when d=1/8.

Structured meshes for these structures are run only once by assigning the exact EA values (as in 3.1.7) as the exact solution give the same result every time.

The nodal deflections at the 'tips' are stored. The 'tip' of the structure is the edge that received the distributed load. For example, if the structure is the cantilevered beam with length 12 and depth 3 and if the element size, d is 1/4, then there are 3×4+1=13 nodes at the 'tip'. And the nodal deflections of the corresponding shell model are also stored at the tip. Then, the relative error for deflection is calculated for each node after 100 runs. The relative error of deflection under the applied load is defined as:

$$\text{Deflection relative error} = \frac{\Delta_{\text{lattice}} - \Delta_{\text{shell}}}{\Delta_{\text{shell}}} \tag{4.1}$$

where Δ_{lattice} denotes the deflection of the node in the lattice model and Δ_{shell} denotes the deflection of the node in the shell model that is solved using the finite element method.

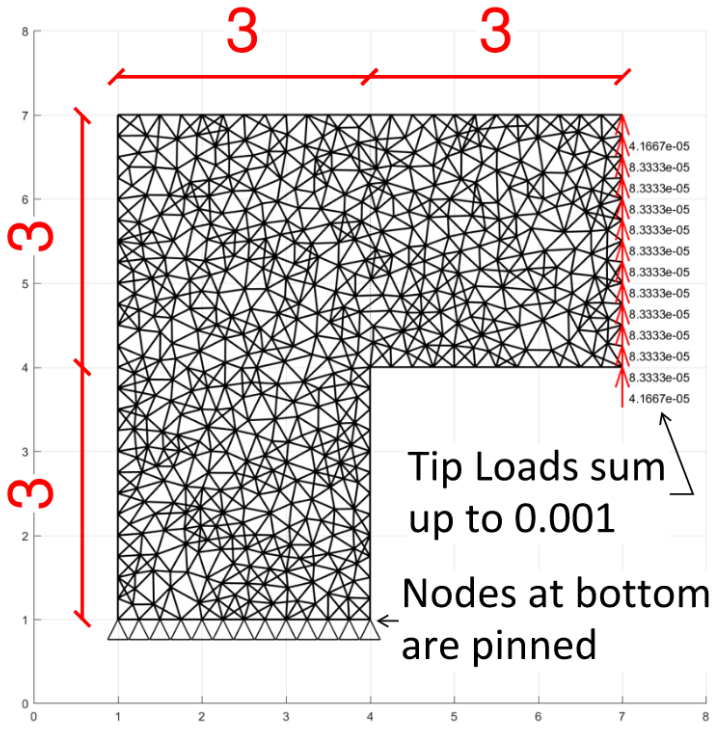


Figure 4.4 Lattice model of the L shaped structure.

4.2 Reference Shell Models

Reference shell models are modeled by using SAP2000 software. Shell solutions are obtained using the finite element method. Deflections of the reference nodes were recorded for comparison. Reference nodes are the nodes to that the external loads are applied. Figures 4.5 and 4.6 show these shell models. Figure 4.7 illustrates how the tip loads are applied to the shells. Corner nodes receive 1 part of point load and intermediate nodes receive 2 parts of point load. Sum of the nodal loads applied to the tip of the structure is 0.01.

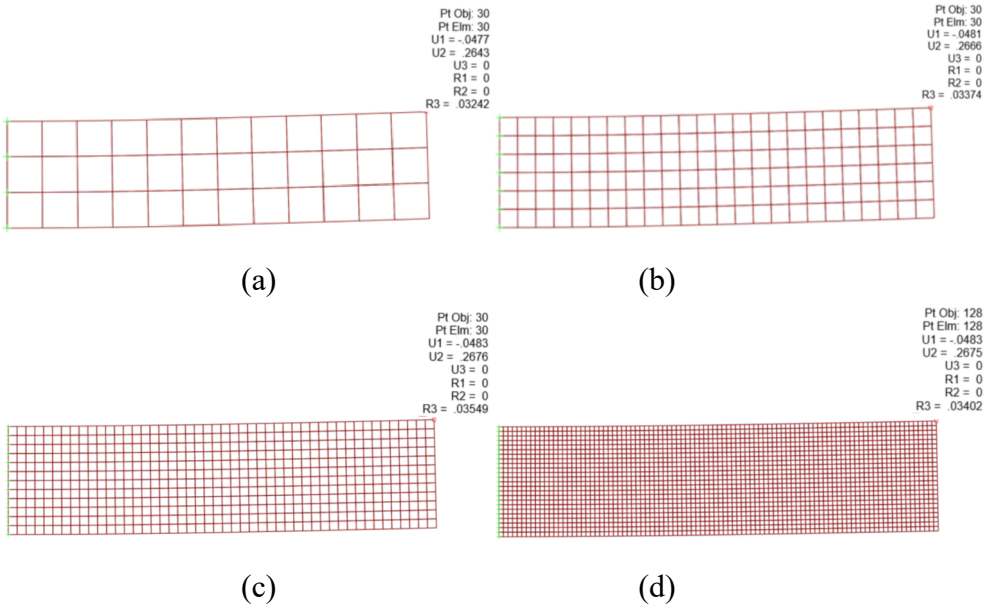


Figure 4.5 SAP2000 shell solutions of the sample cantilevered beam. (a) Shell size is 1, (b) Shell size is 1/2, (c) Shell size is 1/4, (d) Shell size is 1/8.

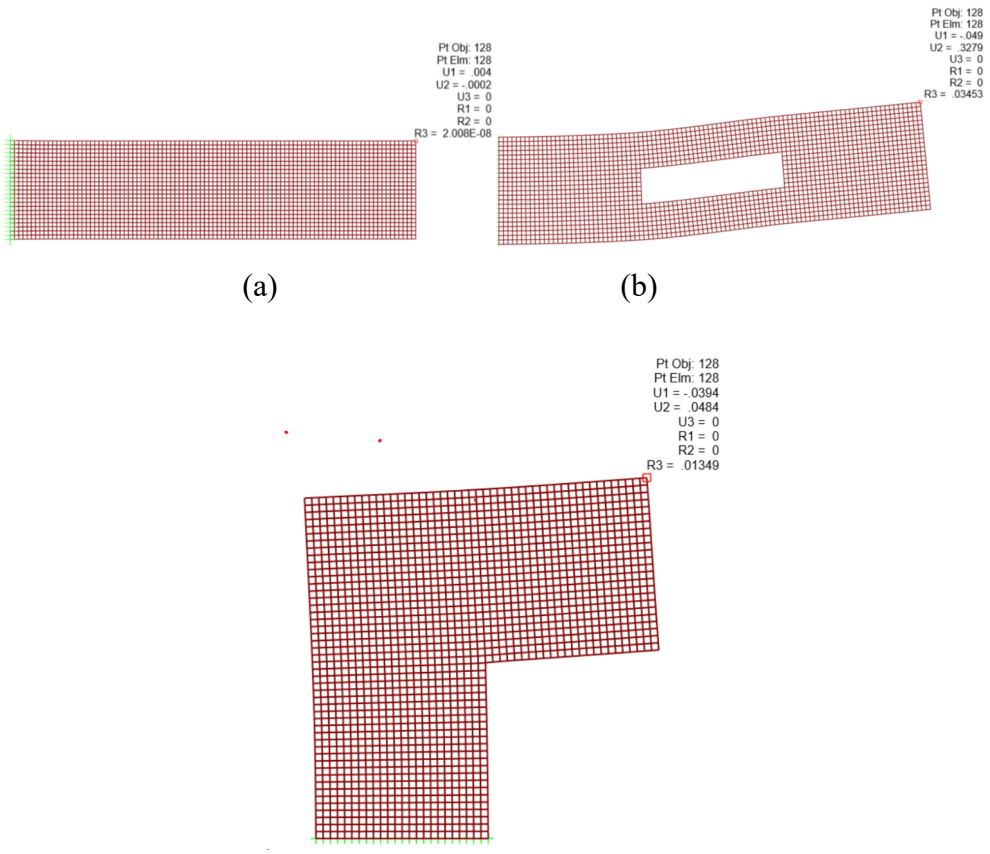


Figure 4.6 SAP2000 Shell solutions. (a) Axially loaded rod, shell size is 1/8, (b) Beam with hole, shell size is 1/8, (c) L shaped structure, shell size is 1/8.

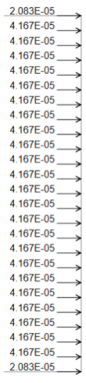


Figure 4.7 Distribution of a tip load among the nodes for both in the reference shell models and in the lattice models.

4.3 Regular Mesh

Structured lattice meshes studied are presented here from Figures 4.8 through Figure 4.11.

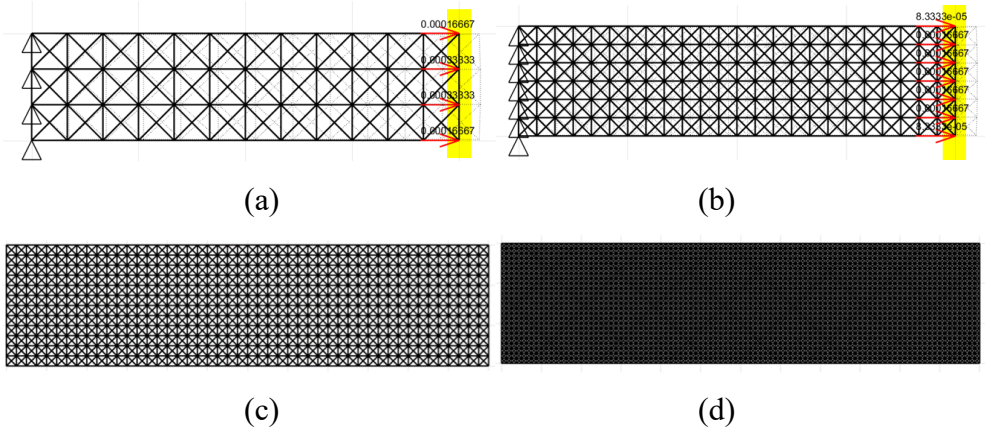


Figure 4.8 Lattice model of the axially loaded rod, structured mesh. (a) d=1, (b) d=1/2, (c) d=1/4, (d) d=1/8.

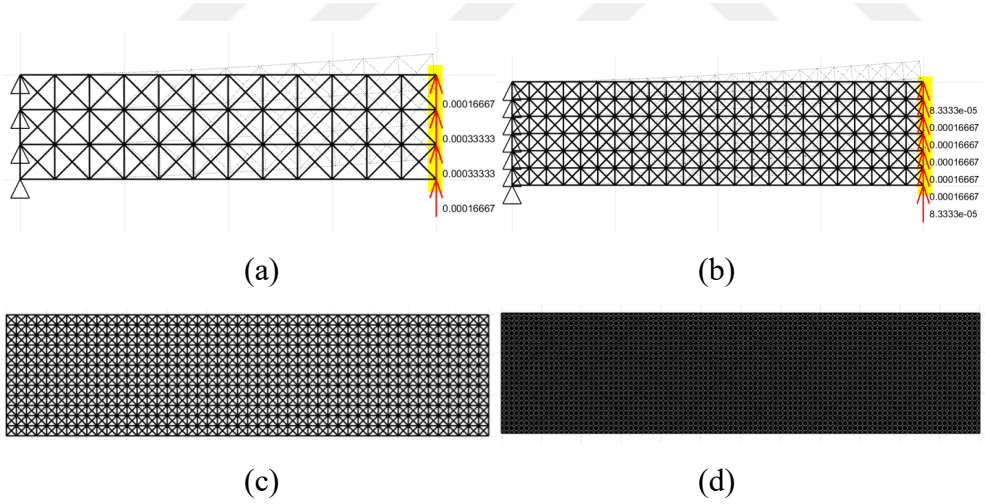
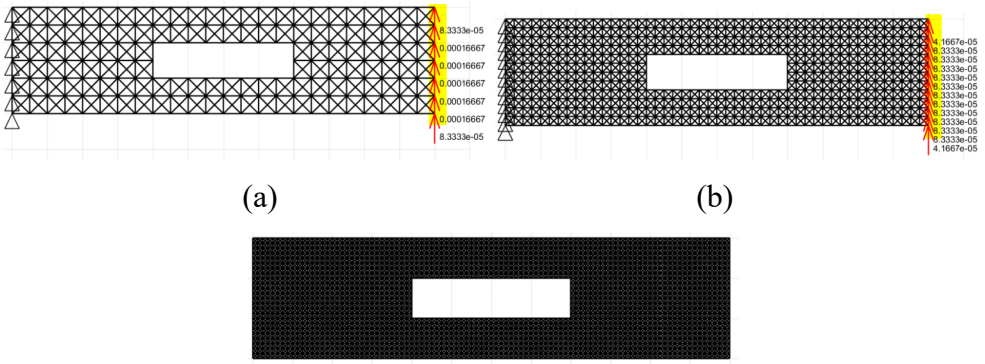


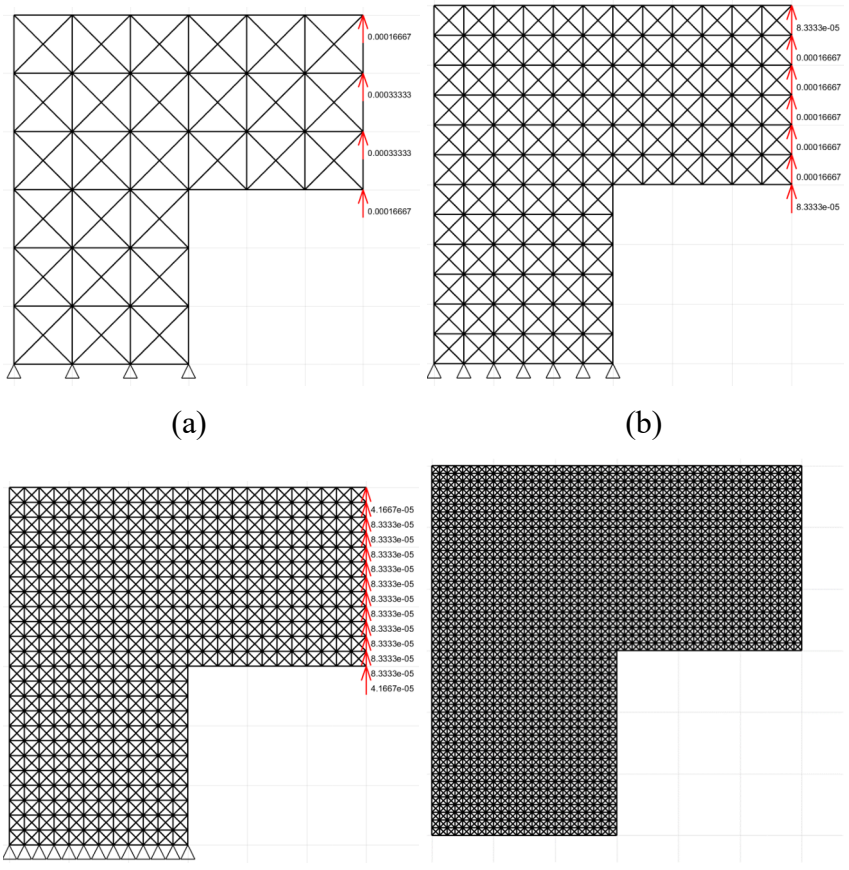
Figure 4.9 Lattice model of the cantilevered beam, structured mesh. (a) d=1, (b) d=1/2, (c) d=1/4, (d) d=1/8.



(a) (b)

(c)

Figure 4.10 Lattice model of the beam with hole, structured mesh. (a) d=1/2, (b) d=1/4, (c) d=1/8.



(a) (b)

(c) (d)

Figure 4.11 Lattice model of the L shaped structure, structured mesh. (a) d=1, (b) d=1/2, (c) d=1/4, (d) d=1/8.

4.4 Random Mesh: Type 1

Type 1 random lattice meshes (that consist of the Delaunay triangulation studied are presented here from Figure 4.16 through Figure 4.19.

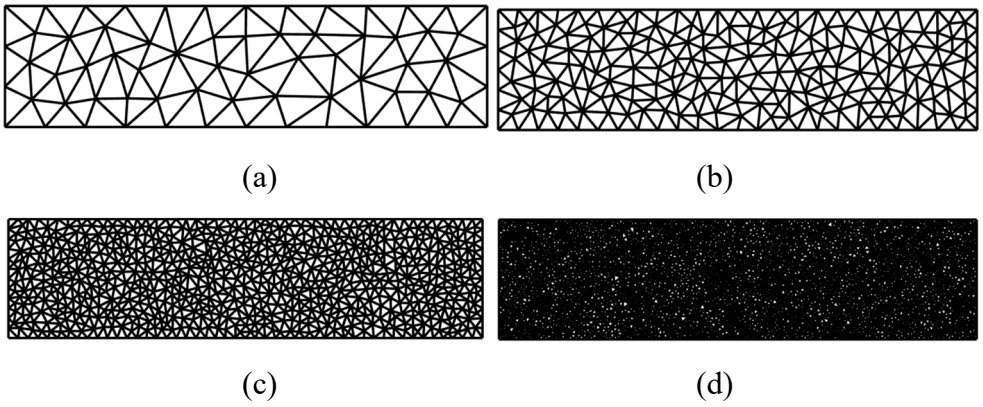


Figure 4.12 Lattice mesh used for the axially loaded rod, and beam with hole Random Mesh: Type 1. (a) d=1, (b) d=1/2, (c) d=1/4, (d) d=1/8.

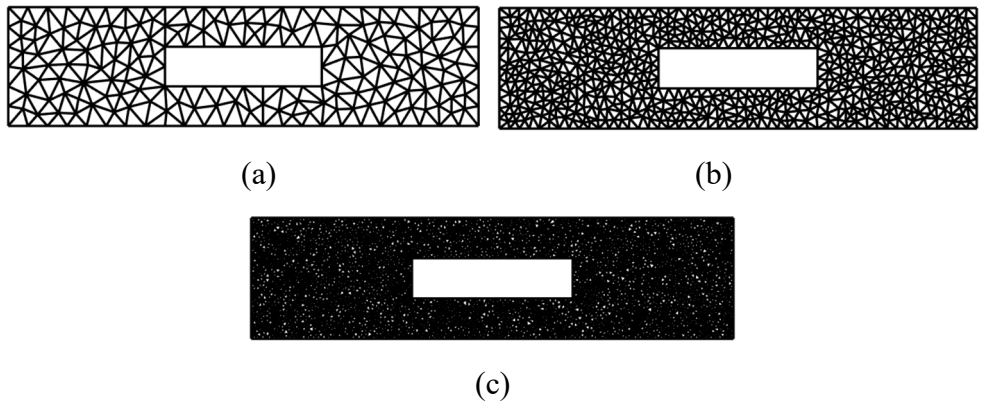
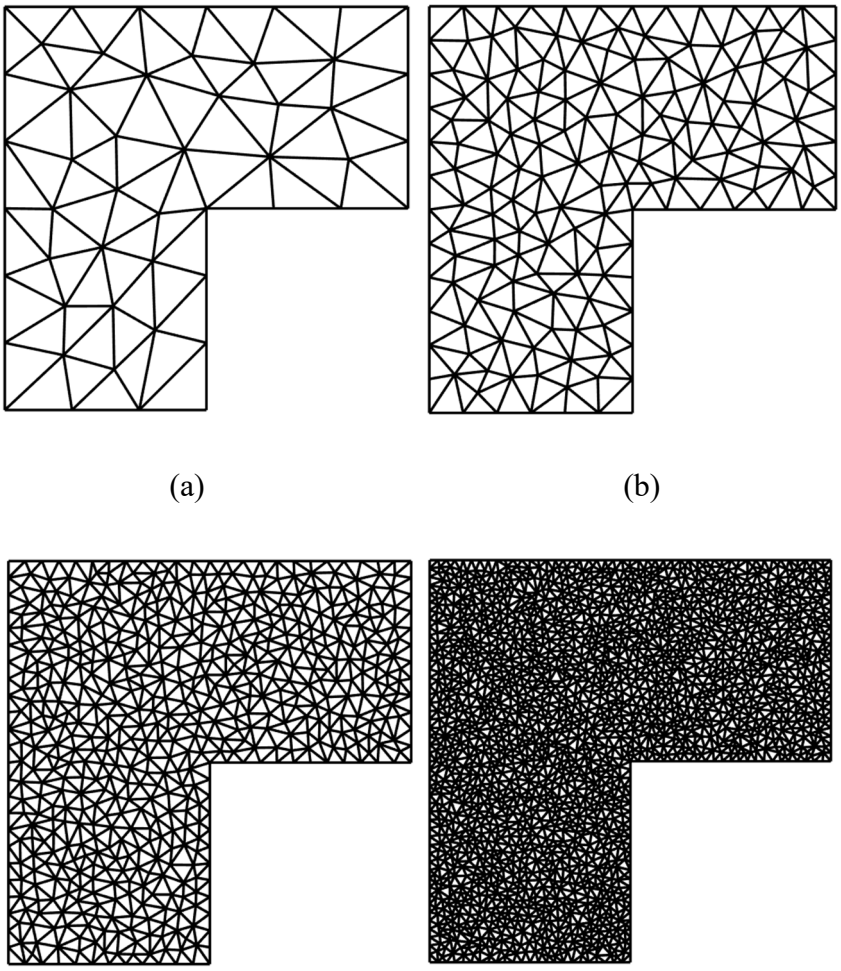


Figure 4.13 Lattice model of beam with hole, Random Mesh: Type 1. (a) d=1, (b) d=1/2, (c) d=1/4.



(a) (b)

(c) (d)

Figure 4.14 Lattice model of the L shaped structure, Random Mesh: Type 1. (a) d=1, (b) d=1/2, (a) d=1/4, (b) d=1/8.

Sample observations for the behavior of Type 1 random meshes are given here. Figure 4.15 show histograms of relative errors in random lattice models compared to shell models for and axially loaded rod. In this sample figure, the EA values are solved according to the energy equivalency method that is described in Section 3.1.9.1, but it should be noted that similar histograms are also obtained for the solutions that were carried out by using the methods described in 3.1.9.2 and 3.1.9.3. Regardless of the structure type studied, the shape of the histograms generally follows a bell curve shape. The structures with element size, d=1 are run 100 times; the ones with d=1/2 are run 200 times; the ones with d=1/4 are run 400 times, and; the ones with d=1/8 are run 800 times so as to obtain smooth histograms.

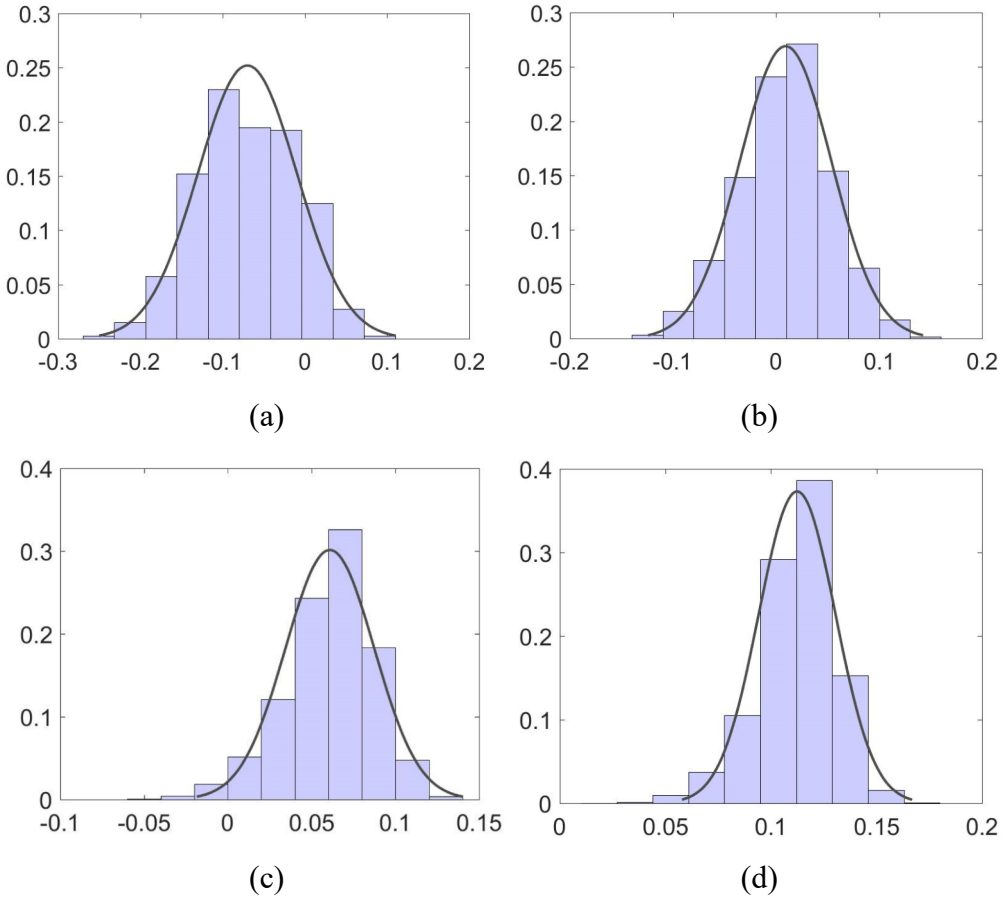


Figure 4.15 Axially loaded rod relative errors of lattice model vs shell model, Random Mesh: Type1, solved by assigning EA_{Bazant} to the lattices.. (a) d=1, 100 samples run; (b) d=1/2, 200 samples run; (c) d=1/4, 400 samples run; (d) d=1/8, 800 samples run.

Results obtained for cantilevered beam, beam with hole and L shaped structure show similar histograms following bell shapes as in Figure 4.15.

4.5 Random Mesh: Type 2

Type 2 random lattice meshes (that consist of Delaunay triangulation and crossing elements) studied are presented here Figure 4.20 through Figure 4.19. External loads and supports are only shown for d=1 and d=1/2 for clarity.

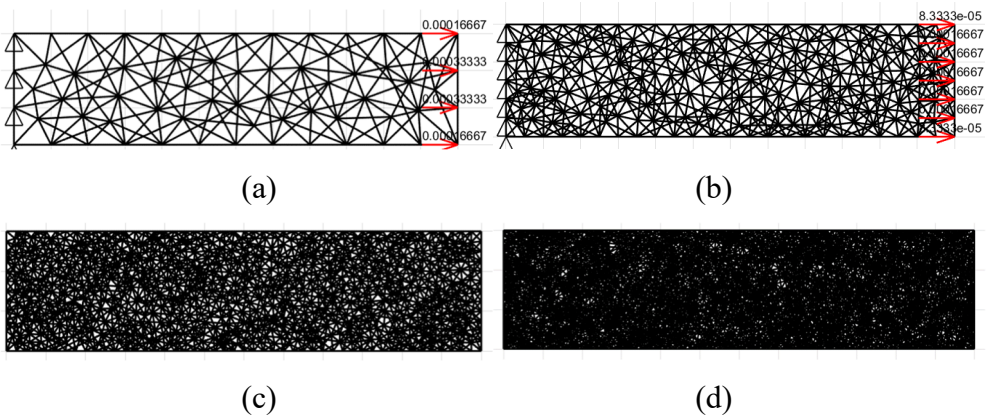


Figure 4.16 Lattice model of the axially loaded rod, Random Mesh: Type 2. (a) d=1, (b) d=1/2, (c) d=1/4, (d) d=1/8.

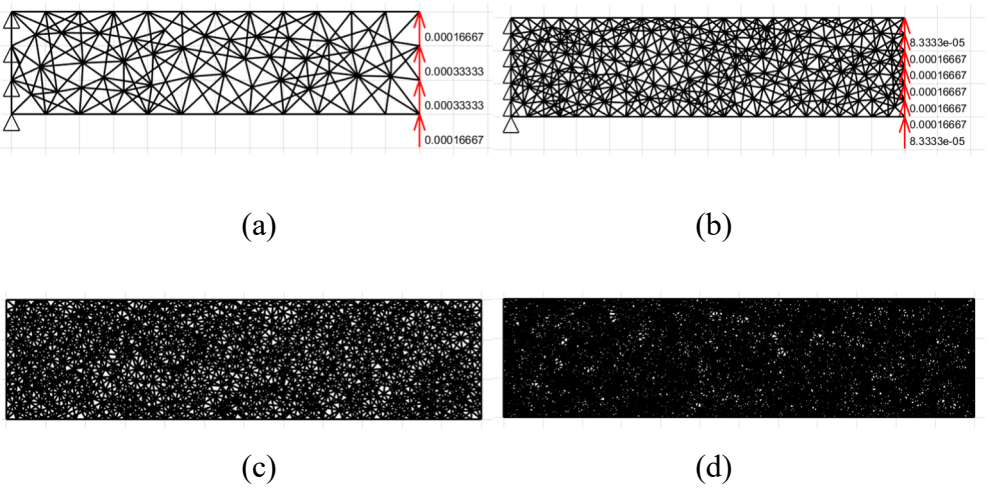
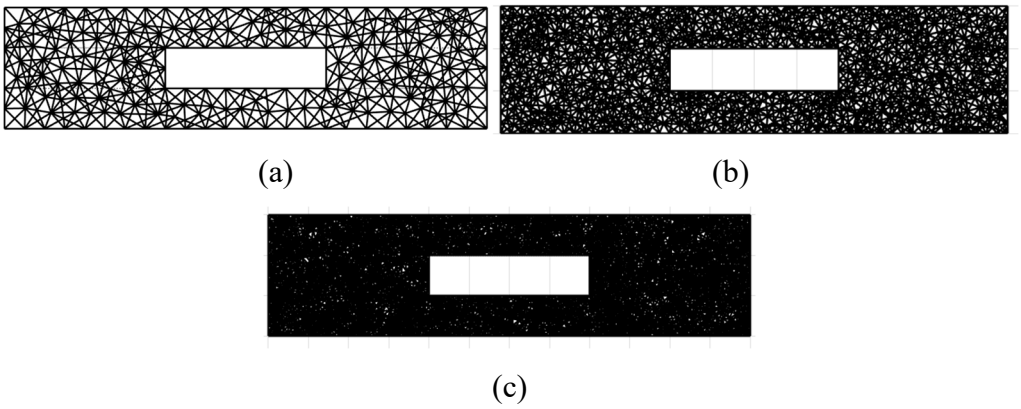


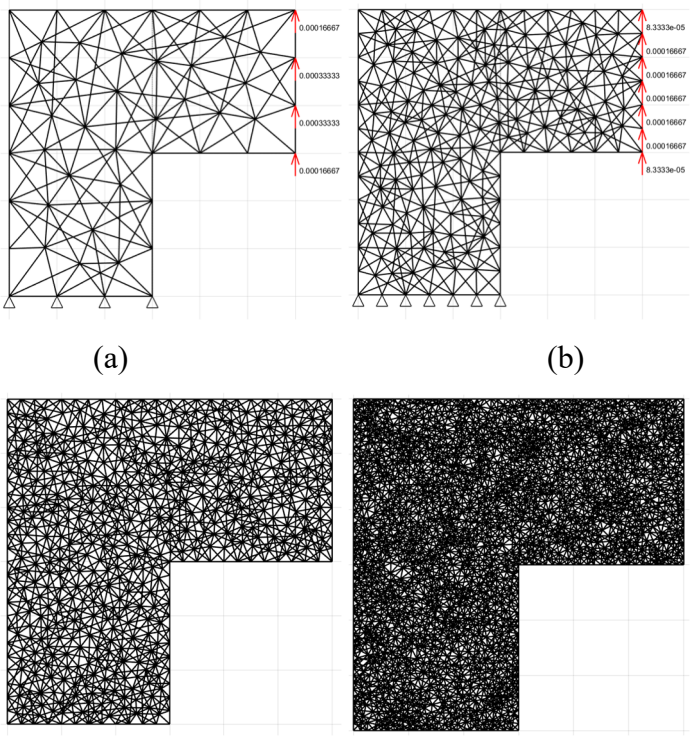
Figure 4.17 Lattice model of the cantilevered beam, Random Mesh: Type 2. (a) d=1, (b) d=1/2, (c) d=1/4, (d) d=1/8.



(a) (b)

(c)

Figure 4.18 Lattice model of the beam with hole, Random Mesh: Type 2. (a) d=1/2, (b) d=1/4, (c) d=1/8.



(a) (b)

(c) (d)

Figure 4.19 Lattice model of the L shaped structure, Random Mesh: Type 2. (a) d=1, (b) d=1/2, (c) d=1/4, (d) d=1/8.

Sample observations for the behaviour of Type 2 random meshes are given here. Figure 4.20 show histograms of relative errors in random lattice models compared to shell models for and axially loaded rod. In this sample figure, the EA values are solved according to the energy equivalency method that is described in section 3.1.9.1, but it should be noted that similar histograms are obtained for the solutions that were carried out by using the methods described in 3.1.9.2 and 3.1.9.3 too. Also regardless of the structure type studied, the shape of the histograms generally follows a bell curve shape. The structures with element size, d=1 are run 100 times; the ones with d=1/2 are run 200 times; the ones with d=1/4 are run 400 times, and; the ones with d=1/8 are run 800 times so as to obtain smooth histograms.

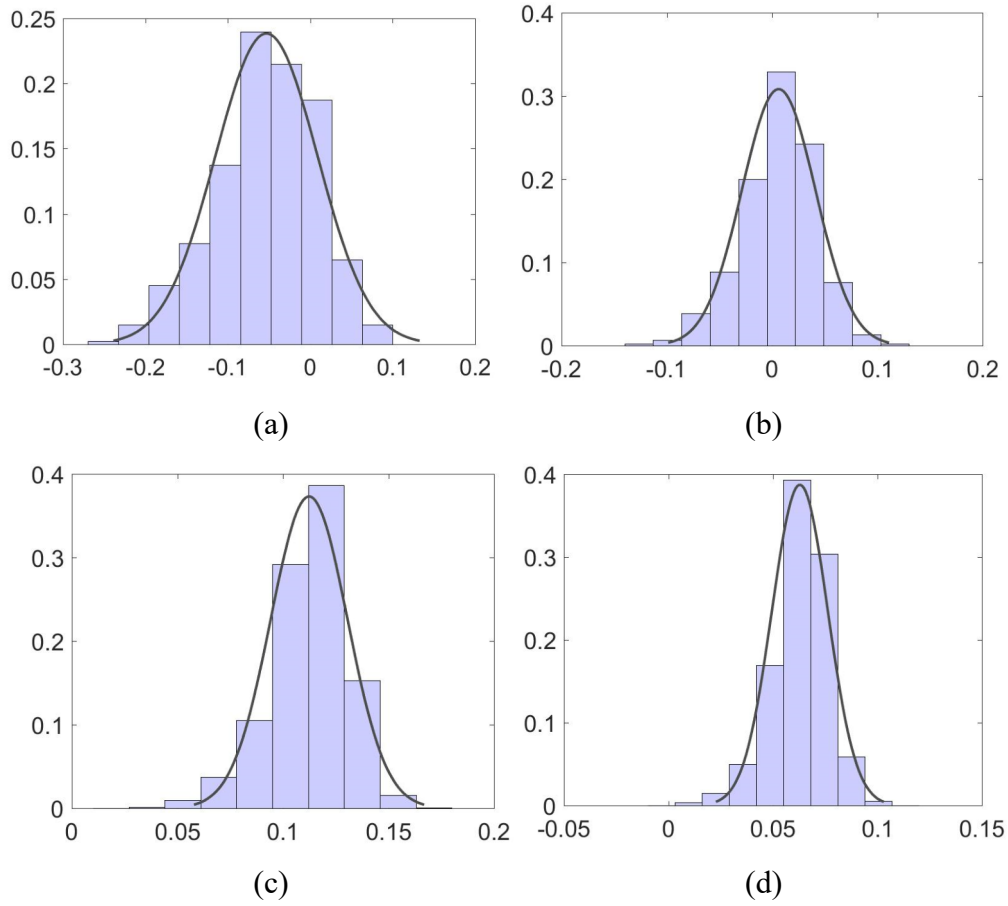


Figure 4.20 Sample histograms for Axially loaded rod relative errors of lattice model vs shell model, Random Mesh: Type 2, solved by assigning EA_{Bazant} to the lattices. (a) d=1, 100 samples run; (b) d=1/2, 200 samples run; (c) d=1/4, 400 samples run; (d) d=1/8, 800 samples run.

Results obtained for cantilevered beam, beam with hole and L shaped structure show similar histograms following bell shapes as in Figure 4.20.

4.6 Random Mesh: Type 3

Type 3 Random Meshes (constructed using proximity with straight edges) studied are given in Figure 4.21 through Figure 4.23. Loads and supports are not shown for clarity.

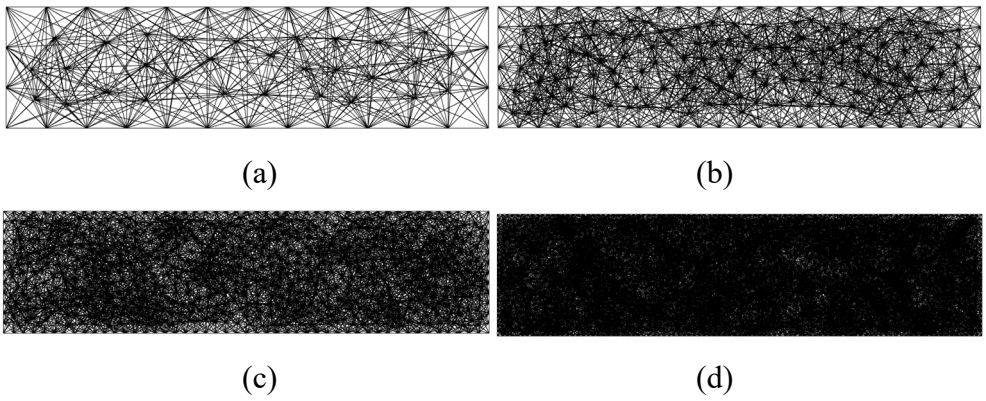


Figure 4.21 Lattice model of the axially loaded rod, and cantilevered beam Random Mesh: Type 3. (a) d=1, (b) d=1/2, (c) d=1/4, (d) d=1/8.

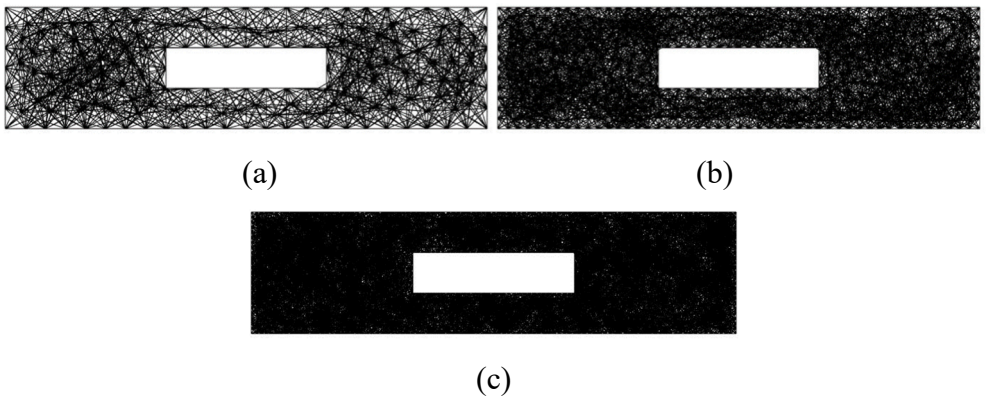


Figure 4.22 Lattice model of the beam with hole, Random Mesh: Type 3. (a) d=1/2, (b) d=1/4, (c) d=1/8.

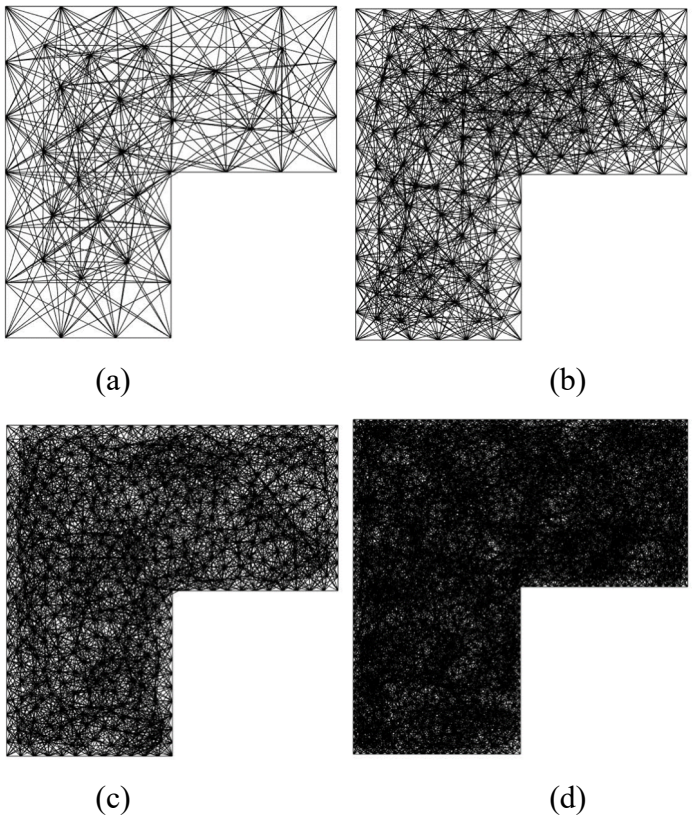


Figure 4.23 Lattice model of the L shaped structure, Random Mesh: Type 3. (a) d=1, (b) d=1/2, (c) d=1/4, (d) d=1/8.

Sample observations for the behavior of Type 3 random meshes are given here. Figure 4.24 shows histograms of relative errors in random lattice models compared to shell models for and axially loaded rod. In this sample figure, the EA values are solved according to the energy equivalency method that is described in section 3.1.9.1, but it should be noted that similar histograms are obtained for the solutions that were carried out by using the methods described in 3.1.9.2 and 3.1.9.3 too. Also regardless of the structure type studied, the shape of the histograms generally follows a bell curve shape. The structures with element size, d=1 are run 100 times; the ones with d=1/2 are run 200 times; the ones with d=1/4 are run 400 times, and; the ones with d=1/8 are run 800 times so as to obtain smooth histograms.

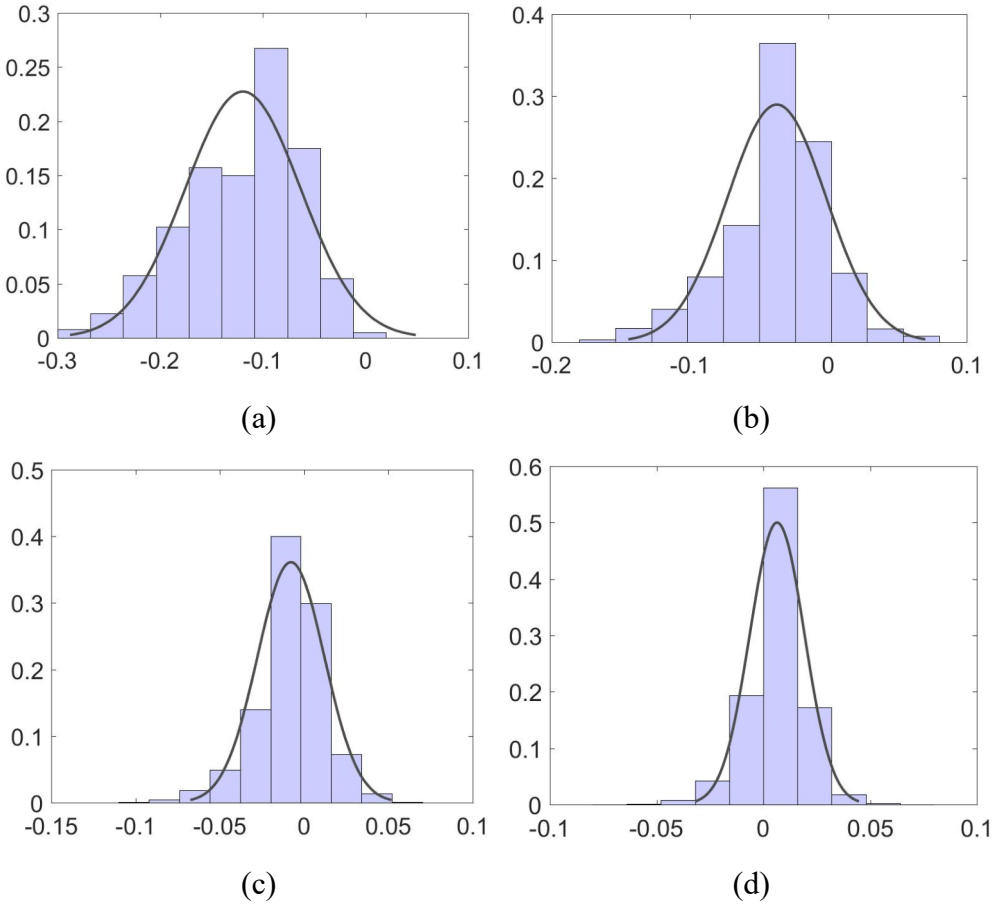
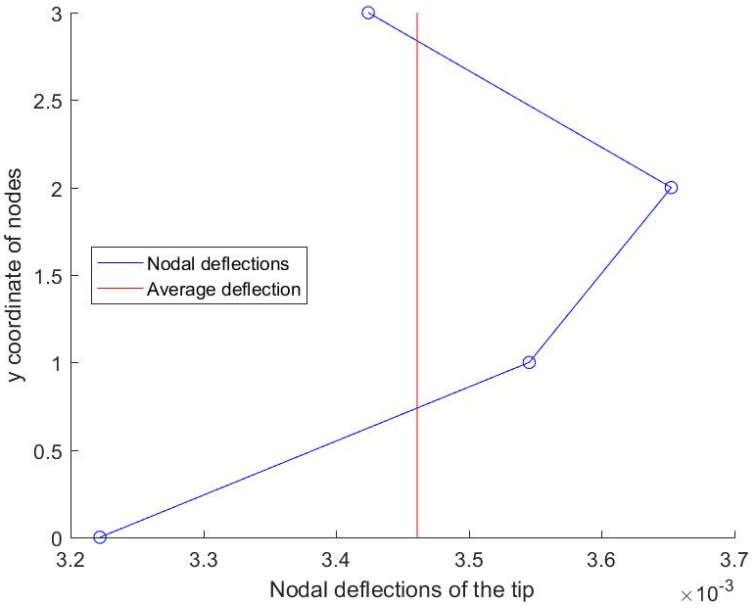


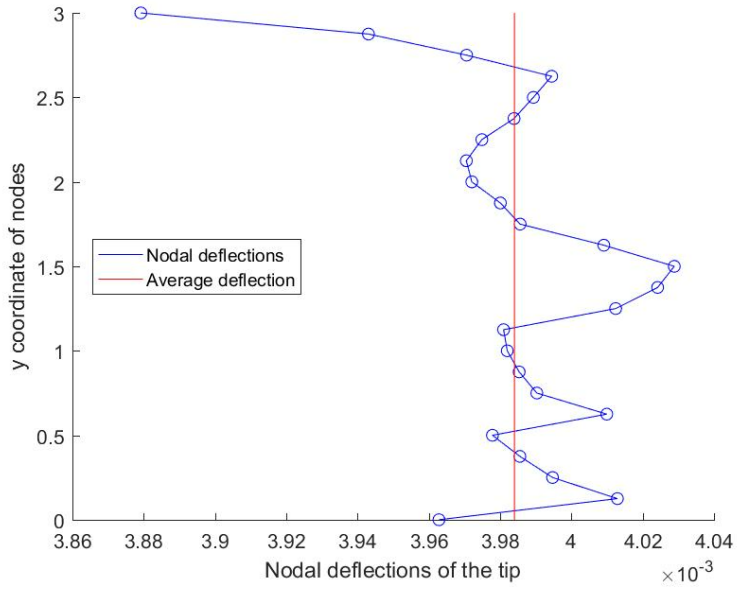
Figure 4.24 Axially loaded rod relative errors of lattice model vs shell model, Random Mesh: Type 3, solved by assigning EA_{Bazant} to the lattices. (a) d=1, 100 samples run; (b) d=1/2, 200 samples run; (c) d=1/4, 400 samples run; (d) d=1/8, 800 samples run.

Figure 4.25 illustrates the tip deflection of a sample axially loaded rod modeled with Type 3 Random Mesh. It is observed that when random meshes are used, the tips are not perfectly straight and there are perturbations observed when compared to the reference shell models. The tip deflection profile is observed to take a more straight shape as the element size, d gets smaller.

Results obtained for cantilevered beam, beam with hole and L shaped structure show similar histograms following bell shapes as in Figure 4.24.



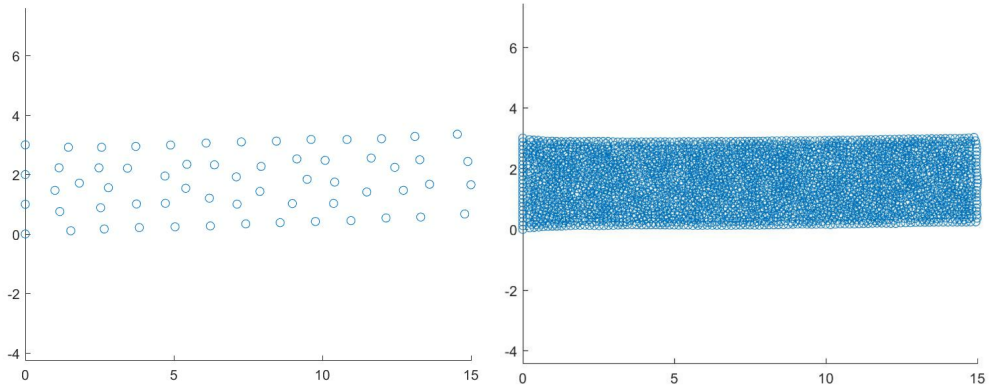
(a)



(b)

Figure 4.25 Axially loaded rod tip deflections, Random Mesh: Type 3, solved by assigning EA_{Bazant} to the lattices. (a) d=1, (b) d=1/8

Figure 4.26 shows the deflected shape of an axially loaded rod loaded modeled with Type 3 Random Mesh. It is observed that the bar, although loaded axially deflects in the perpendicular direction as well due to the randomness of the mesh, but as the element size, d gets smaller, it is seen that the perpendicular deflection gets small as well. Deflections in both directions in the figures are exaggerated so as to make the deflections easier to see.

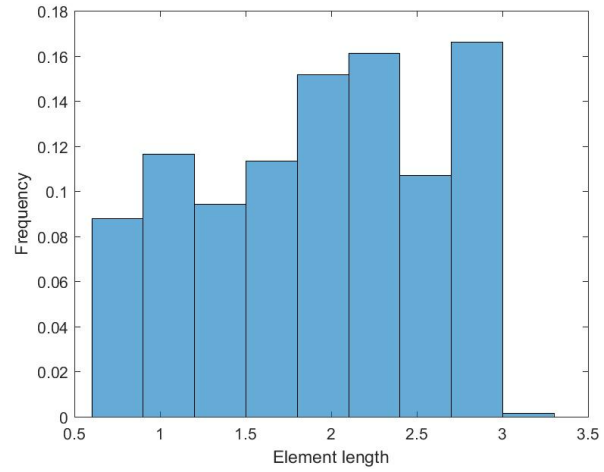


(a) (b)

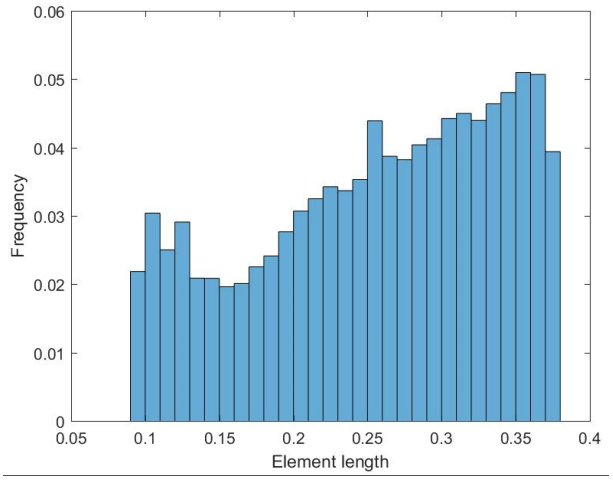
Figure 4.26 Axially loaded rod deflected shape, Random Mesh: Type 3, solved by assigning EA_{Bazant} to the lattices, deflections are exaggerated for visibility. (a) d=1, (b) d=1/8

Figure 4.27 shows the frequency of occurrence of element lengths in a Type 3 Random Mesh. It is seen that the element size frequencies are similar for all element lengths.

Figure 4.28 shows the displacement versus node position plot.

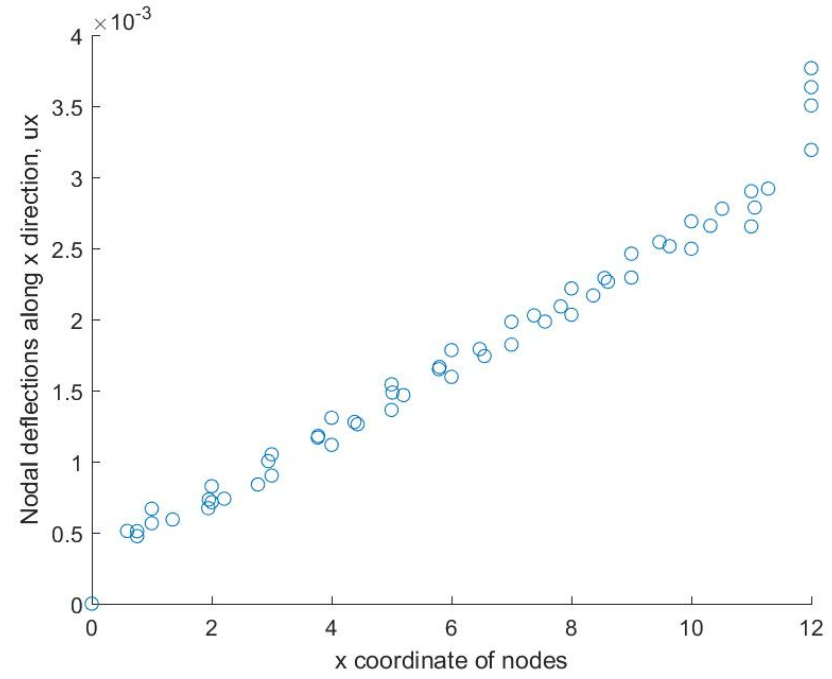


(a)

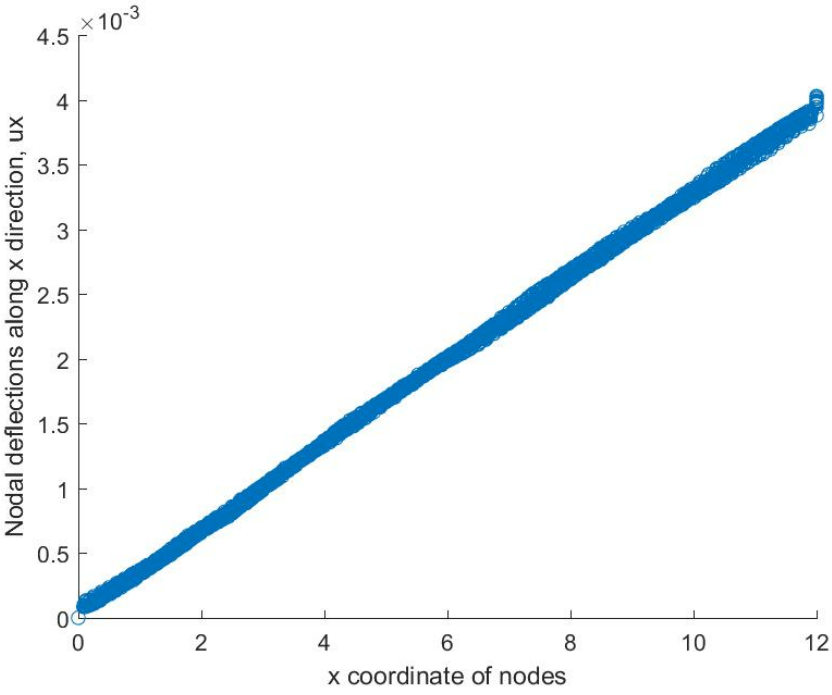


(b)

Figure 4.27 Histogram of frequency of occurrence of element lengths, Random Mesh: Type 3, solved by assigning EA_{Bazant} to the lattices. (a) d=1, (b) d=1/8.



(a)



(b)

Figure 4.28 Axially loaded rod, x vs u_x plot, Random Mesh: Type 3, solved by assigning EA_{Bazant} to the lattices. (a) d=1, (b) d=1/8

4.7 Random Mesh: Type 4

Type 4 Random Meshes (constructed using proximity with non-constrained edges) studied are given in Figure 4.29. Loads and supports are not shown for clarity.

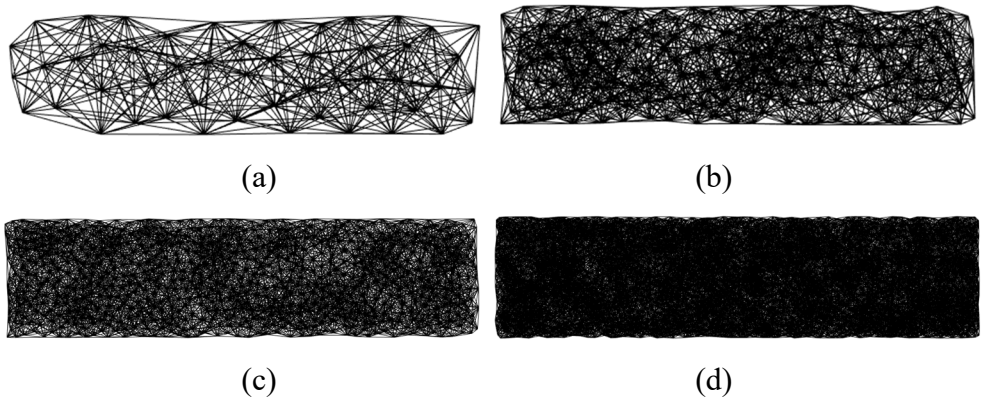


Figure 4.29 Lattice model of the axially loaded rod and cantilevered beam, Random Mesh: Type 4. (a) d=1, (b) d=1/2, (c) d=1/4, (d) d=1/8.

Sample observations for the behavior of Type 4 random meshes are given here. Figure 4.30 shows histograms of relative errors in random lattice models compared to shell models for and axially loaded rod. In this sample figure, the EA values are solved according to the energy equivalency method that is described in section 3.1.9.1, but it should be noted that similar histograms are obtained for the solutions that were carried out by using the methods described in 3.1.9.2 and 3.1.9.3 too. Only the axially loaded rod and the cantilevered beam structures are studied for Random Mesh: Type 4 . The structures with element size, d=1 are run 100 times; the ones with d=1/2 are run 200 times; the ones with d=1/4 are run 400 times, and; the ones with d=1/8 are run 800 times so as to obtain smooth histograms.

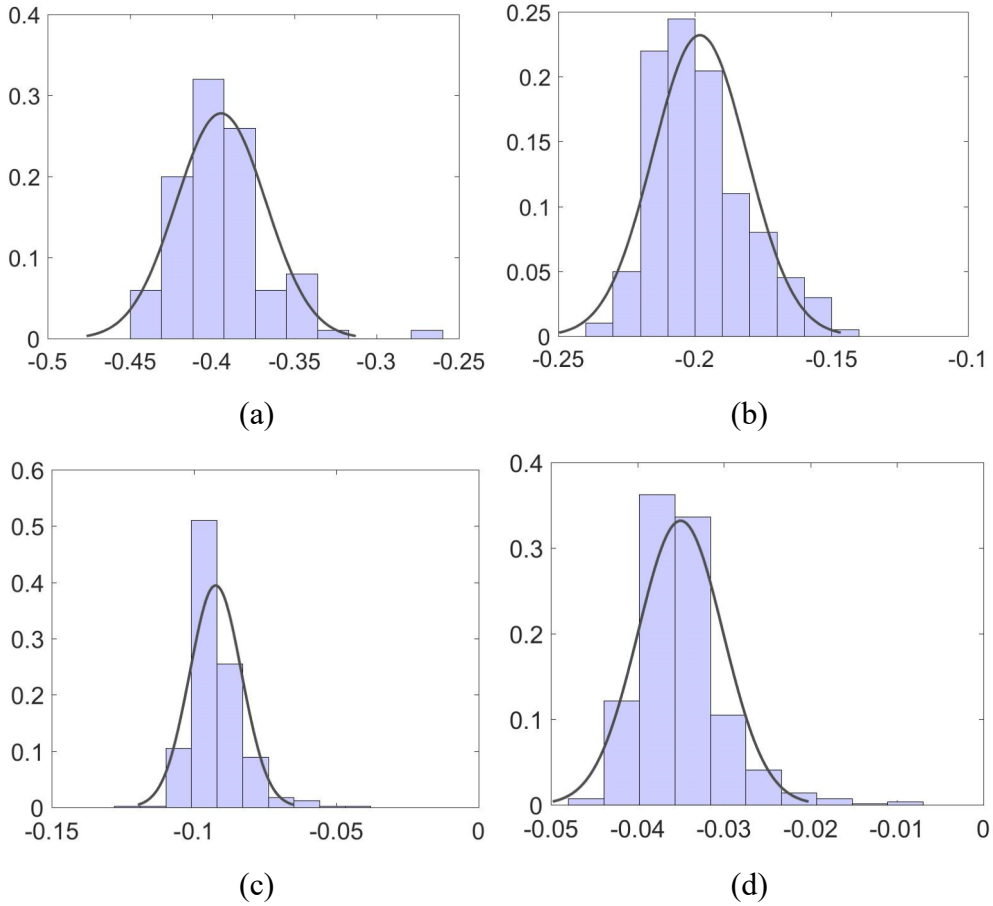
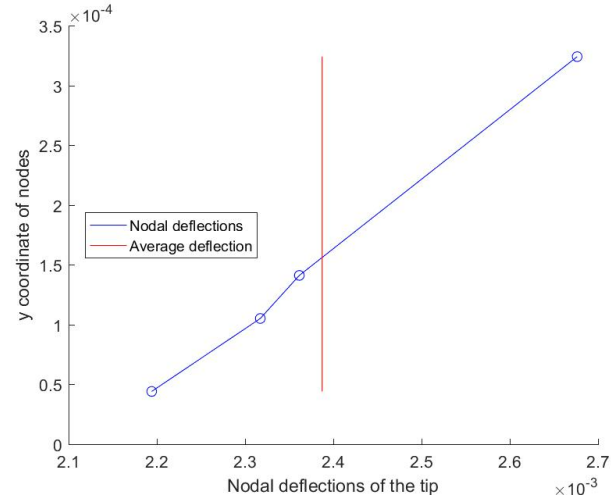
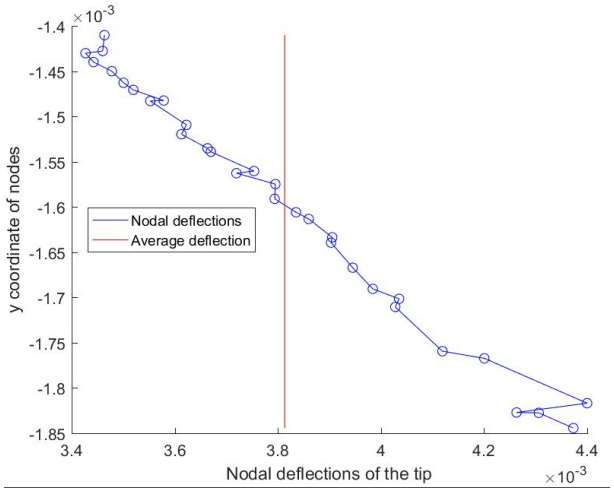


Figure 4.30 Axially loaded rod relative errors of lattice model vs shell model, Random Mesh: Type 4, solved by assigning EA_{Bazant} to the lattices. (a) d=1, 100 samples run; (b) d=1/2, 200 samples run; (c) d=1/4, 400 samples run; (d) d=1/8, 800 samples run.

Figure 4.31**Figure 4.30** illustrates the tip deflection of a sample axially loaded rod modeled with Type 4 Random Mesh. It is observed that the edge perturbations are much more unbalanced than these of Random Mesh: Type 3.



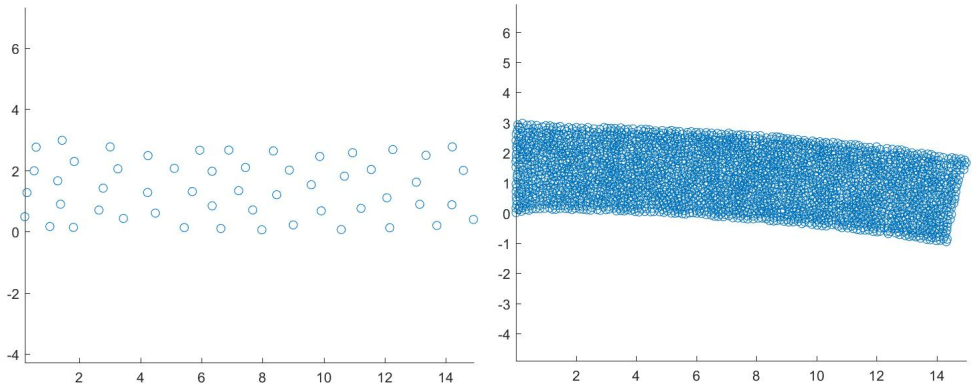
(a)



(b)

Figure 4.31 Sample axially loaded rod tip deflections, Random Mesh: Type 4, solved by assigning EA_{Bazant} to the lattices. (a) d=1, (b) d=1/8

Figure 4.32 shows the deflected shape of an axially loaded rod loaded modeled with Type 4 Random Mesh. Unlike Random Mesh: Type 3, the perpendicular deflections do not necessarily disappear as better in this type of mesh, so it is observed that this is a shortcoming of not having straight edges. Due to noticing this shortcoming, only axially loaded rod is studied for Random Mesh: Type 4. Cantilevered beam, beam with hole and L shaped structure are not studied for Random Mesh: Type 4.

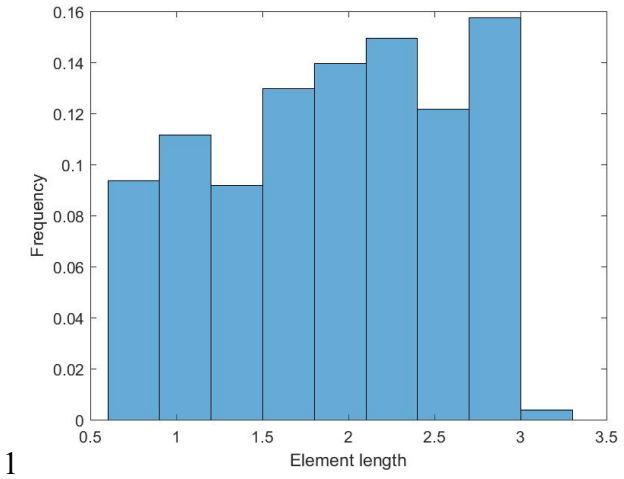


(a) (b)

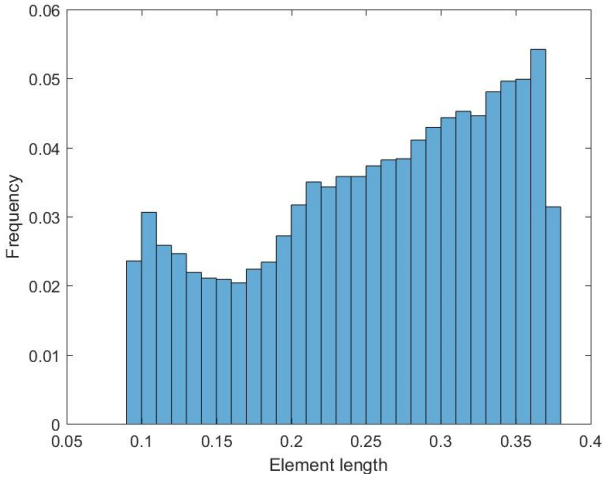
Figure 4.32 Sample axially loaded rod deflected shape, Random Mesh: Type 4, solved by assigning EA_{Bazant} to the lattices, deflections are exaggerated for visibility. (a) d=1, (b) d=1/8

Figure 4.33 shows the frequency of element length in a Type 3 Random Mesh. It is seen that the element size frequencies are similar for all element lengths.

Figure 4.34 shows the displacement versus node position plot. This plot indicates that the mesh is a homogeneous mesh and the elongation strains are almost equal at all locations measured from the support to the tip, but the straightness is not as apparent as it was in Random Mesh: Type 3.

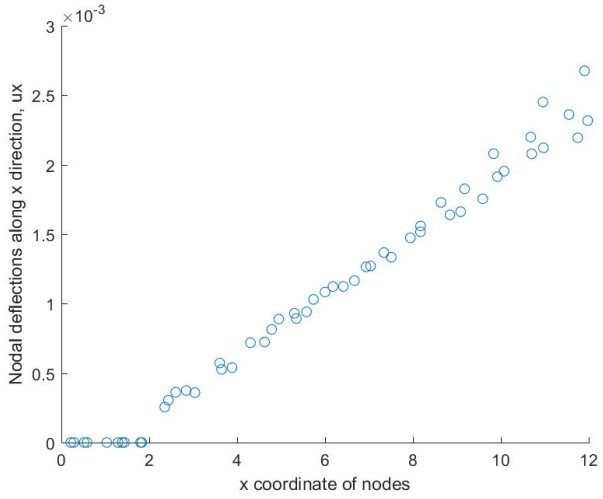


(a)

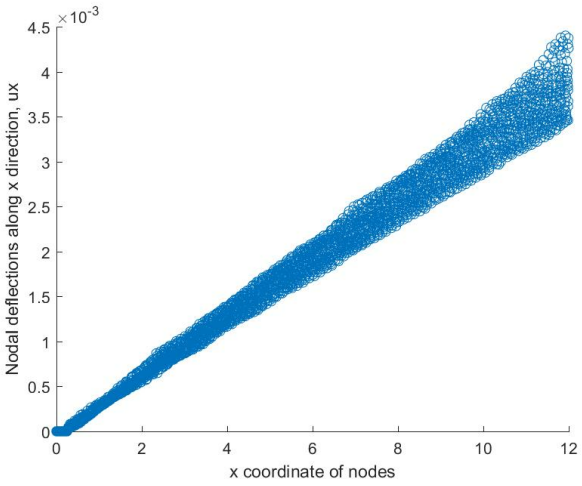


(b)

Figure 4.33 Histogram of frequency of occurrence of element lengths, Random Mesh: Type 4, solved by assigning EA_{Bazant} to the lattices. (a) d=1, (b) d=1/8.



(a)



(b)

Figure 4.34 Axially loaded rod, x vs u_x plot, Random Mesh: Type 4, solved by assigning EA_{Bazant} to the lattices. (a) d=1, (b) d=1/8

4.8 Performance Summary of Random Meshes

Mean nodal deflection errors of each mesh type for each structure are plotted in Figures 4.35 through 4.38. The method for obtaining these nodal deflection errors for Type 3 and Type 4 random meshes are explained in Sections 3.1.3. and 3.1.4. The methods used for Type 1 random mesh and Type 2 random mesh are the same.

In these graphs, each data point represents the mean error value corresponding to the lattice element size, d for the indicated mesh type with the indicated solution approach. This mean error is calculated based on a number of nodal error values such that the structures with element size, d=1 are run 100 times; the ones with d=1/2 are run 200 times; the ones with d=1/4 are run 400 times, and; the ones with d=1/8 are run 800 times. For each of these runs, the error values nodes at the at the 'tip' of the structure are recorded. 'Tip' of the structure is the edge of the structure that the nodal load are applied. So if the 'tip' of the structure has 13 nodes and the structure is run 100 times, there are 1300 errors obtained.

Exact results for structured meshes constructed with rectangles and two diagonals are given in these graphs for reference. These structured meshes are solved by assiging the exact EA values to the lattice elements (These exact EAs are explained and listed in Section 3.1.7).

It was initially expected that all the errors plotted in Figure 4.35 through Figure 4.38 would drop as the element size, got smaller, but it was seen that this was not necessarily the case and it was seen that the stability of errors were quite dependent on the mesh type and on the EA solution type as well.

Regardless of the solution method, the random meshes based on Delaunay triangulations (Random Mesh: Type 1 and Random Mesh: Type 2) were shown to not perform well in terms of convergence. When the element size, d gets smaller, the deflection values would not necessarily converge to the deflection values of the equivalent finite element shell models. When such types of meshes were solved by using the global approach or by the hybrid approach, errors do not seem to be stabilized as the element size got smaller, but the errors were still around the errors recorded for the structured meshes solved by assigning the exact EA values to the lattices. Furthermore, it needs to be noted that when Type 1 Random Meshes are solved by using the local approach (Section 3.1.9.2), the error values are not around the error values of the structured meshes either. Thus, Type 1 Random Meshes do

not show a satisfactory performance when used with the local solution approach and such results were not given in Figure 4.35 through Figure 4.38 due to this reason.

On the other hand, it was observed that the random meshes whose connectivities are constructed by using node proximity (Random Mesh: Type 3) have acceptable convergence behavior when solved with the method proposed in Jirásek & Bazant (1995) or when solved by using the hybrid local-global method (as described in 3.1.9.1). There are other indications that this mesh is well-behaving. In these, it is observed that the element size versus strain is following a straight line indicating the homogeneity of the model (Figure 4.28), and element size distribution is balanced having a similar number of elements for different element lengths (Figure 4.28). It is noted noted that when Type 3 Random Meshes are solved by using the local approach (3.1.9.2), there is no convergence observed and the error values obtained are not around the error values of the structured meshes either. Thus, Type 3 Random Meshes can be said to not compatible with the local approach.

Type 4 Random meshes are solved only using the method of Jirásek & Bazant (1995), and the convergence of errors is observed. But these meshes have the shortcoming of another behavior. When the axially loaded rod model was tested with this type of mesh, it was observed that the mesh was deflecting along the y direction considerably (Figure 4.32). Since this behavior is even observed with the benchmark global method of Jirásek & Bazant (1995), these types of meshes were not tested with the hybrid approach and local approach. So it is concluded that straight boundary edges at boundaries (as in Random Mesh: Type 3) are performing better than non-constrained boundary edges.

Summary of these results are presented in Table 4.3.

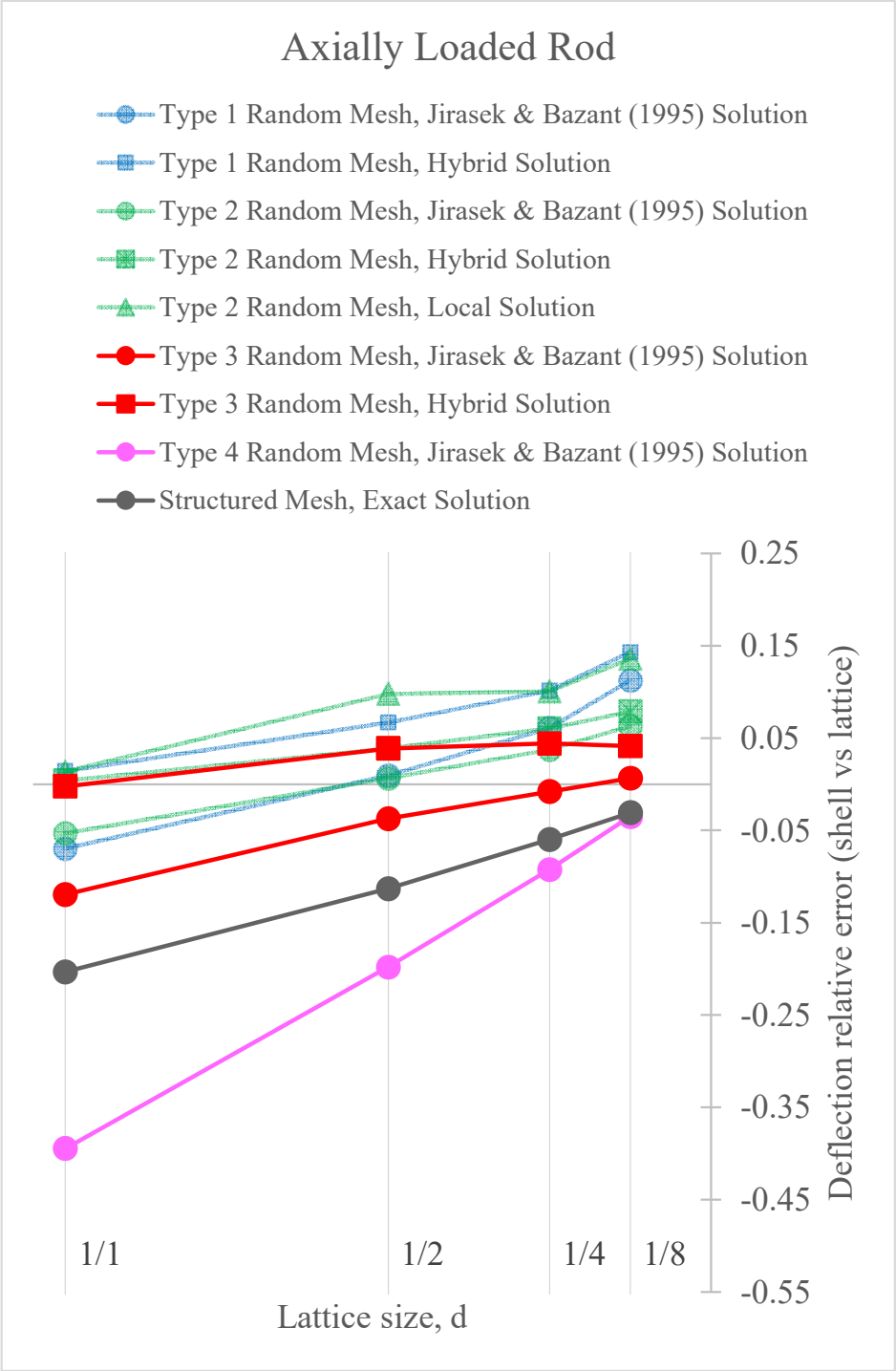


Figure 4.35 Relative deflection errors versus lattice size, axially loaded rod.

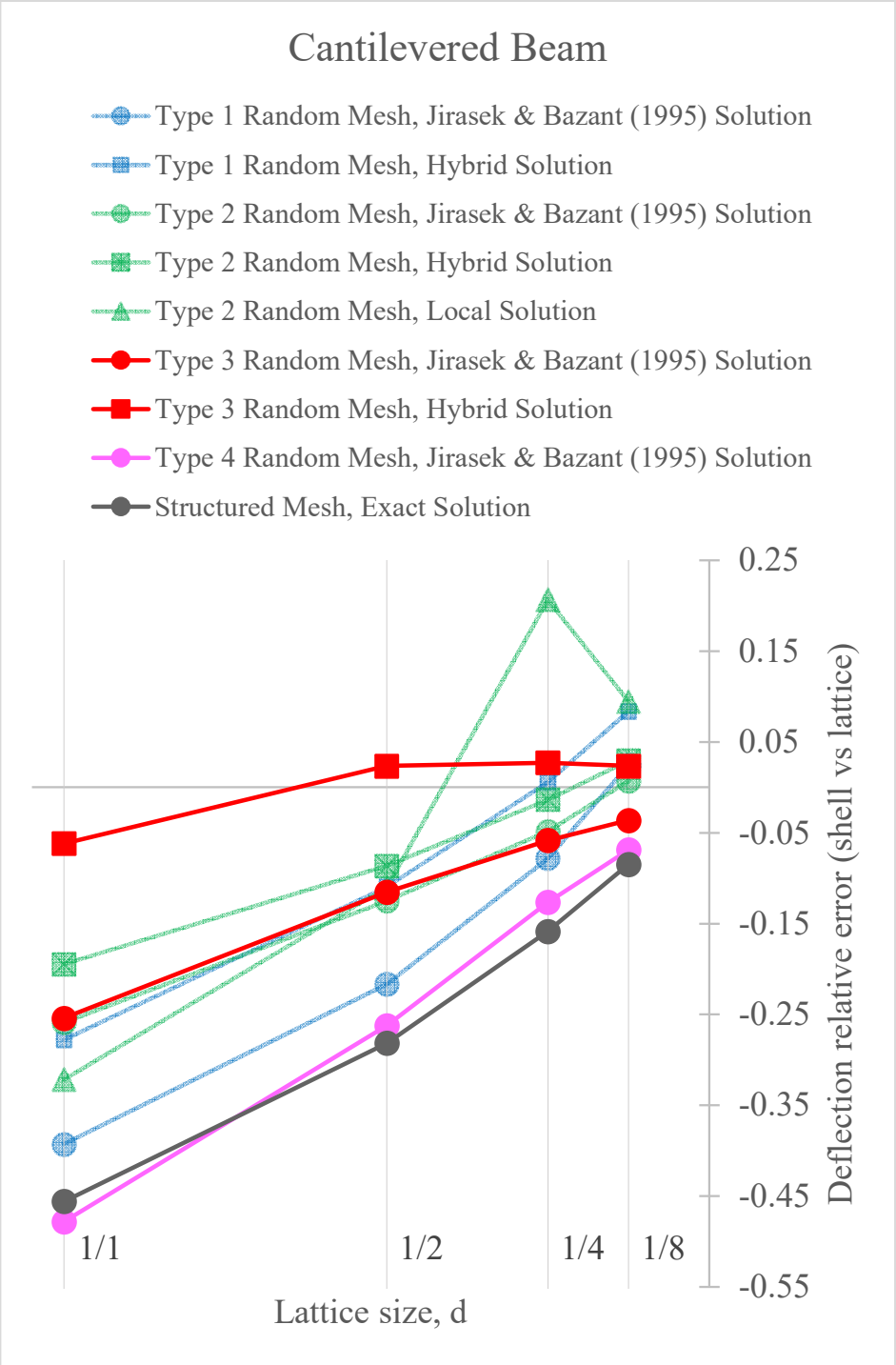


Figure 4.36 Relative deflection errors versus lattice size, cantilevered beam.

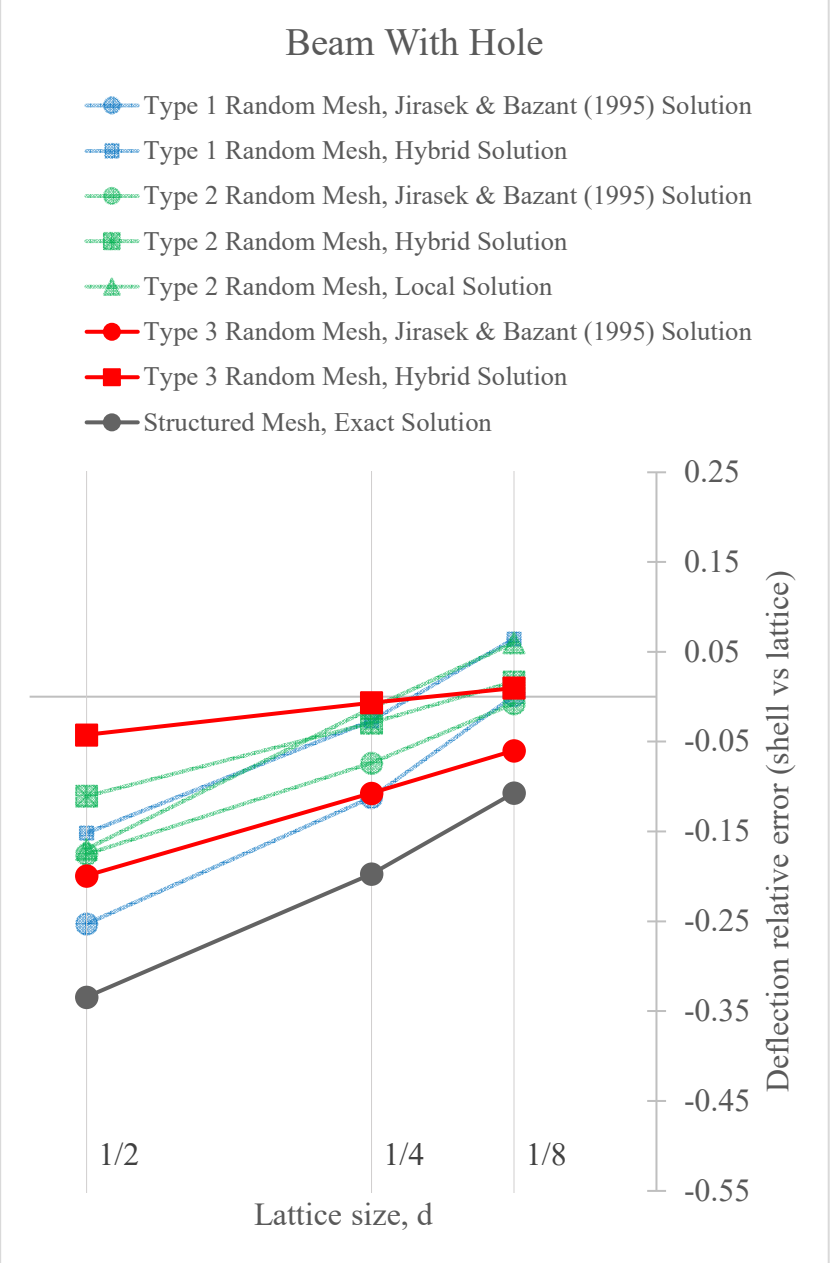


Figure 4.37 Relative deflection errors versus lattice size, beam with hole.

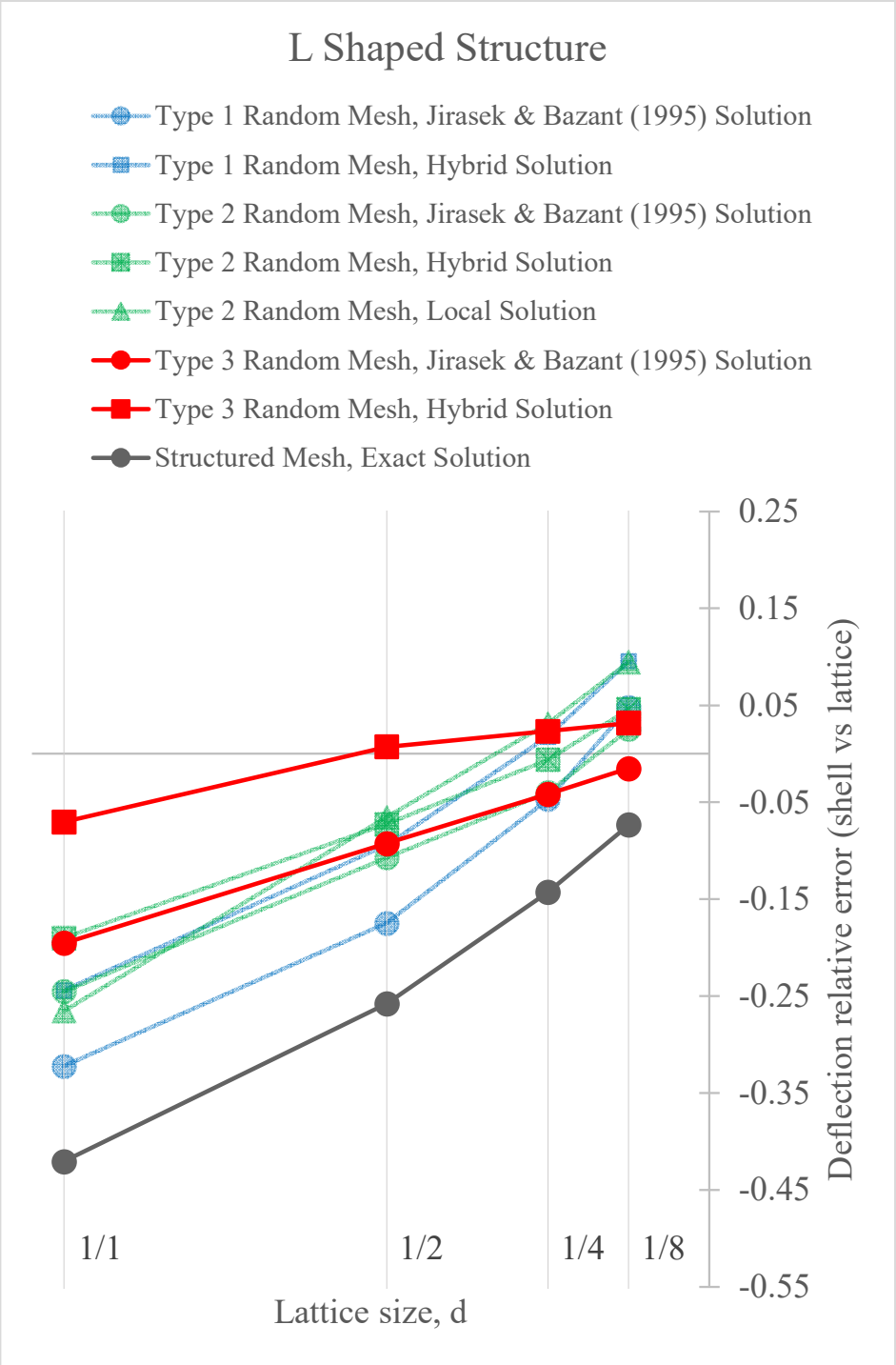
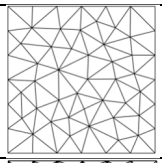
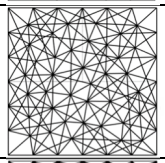
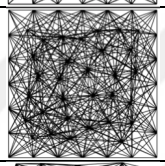
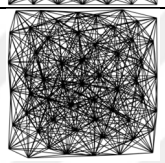


Figure 4.38 Relative deflection errors versus lattice size, L shaped structure..

Table 4.3: Summary of random meshes and their convergence performance when solved with different methods.

Random Mesh Type (Explained Section)	Sample Visual of the Mesh	Solution Type (Explained Section)		
		Bazant (3.1.9.1)	Local (3.1.9.2)	Hybrid (3.1.9.3)
Type 1 (3.1.1)		Not Good	Divergent	Not Good
Type 2 (3.1.2)		Not Good	Not Good	Not Good
Type 3 (3.1.3)		Acceptable	Divergent	Acceptable
Type 4 (3.1.4)		Acceptable But Edge Issues	Not Studied	Not Studied

It is can be seen from Table 4.3 that the convergence of such random LSM is quite dependent on the mesh algorithm and the EA calibration method as well. It is seen that the interconnectedness of the nodes is very important and methods based on Delaunay triangulation have the shortcoming of providing such interconnectedness. Meshes based on node proximity needed to be used to obtain convergent behavior.

Furthermore, local solutions solely based on node tributary areas are observed to give deflection results with the correct order of magnitude with the finite element shell deflections, but a good convergence could not be obtained. Solutions based on global approaches such as in Jirásek & Bazant (1995) and hybrid global-local solutions are observed to perform acceptably in terms of convergence. The performance of various random meshes and solution methods are summarized in Table 4.3.

CHAPTER 5

CONCLUSION

5.1 Summary

A method is proposed for calibrating the EA values of the lattice elements. The proposed method is based on the energy equivalency of the prototype and the lattice model. This method is tested against the already existing method of Jirásek & Bazant (1995) and a hybrid proposed method. Lattices that only transfer axial forces are used. So in that regard, this is a lattice spring model (LSM) study. Four different random meshes are constructed and their convergence performances are studied. Two of them are based on Delaunay triangulation and two of them are based on connectivity based on node proximity.

A number of different types of structures are chosen as examples to test the EA calibrating methods. These examples are drawn and solved using the MATLAB program. Two of these structures are simpler structures, a rod under axial load and a cantilevered beam, other two of the structures had non-trivial shapes, a beam with hole, and an L-shaped structure.

A computer program using MATLAB to carry out the tasks of this procedure. The program has a user interface to allow the user to draw geometries and assign boundary conditions and forces easily. The program creates the type of mesh (regular or random) with the input material parameters. This program is named “User-Friendly Lattice Model Program with Graphical User Interface”, shortly “LM”.

The elastic deflections for solved examples are compared with the equivalent finite element shell model solutions. The relative errors of deflections occurring at the tips of these structures are presented in histograms for many runs. Mean errors for each

of these structures for regular and random meshes are plotted against the mesh size used.

5.2 Convergence Performance of the Meshes and Solution Methods

Type 1 and Type 2 meshes have convergence issues and they are not advised to be used for lattice models unless the deflections are known a priori. Type 4 random meshes were observed to deflect perpendicular to the loading direction even with very small element sizes, so they are not advised to be used for lattice models.

It was seen that Type 3 random meshes that were solved with the method of Jirásek & Bazant (1995) or with the Hybrid global-local solution show the best convergence characteristics.

It is concluded that the interconnectedness of a mesh is important to obtain a good performance. Solutions that consider the energy density equations globally also seemed to perform better.

5.3 Remarks on the User-Friendly Lattice Model Program with Graphical User Interface

After the literature review conducted within the scope of this study, it was seen that the lattice models were mostly used for material behavior study purposes and there was not much focus on making the lattice models practicable in common engineering uses. Although the lattice models were beginning to be used to model real-world problems in some studies (Işık, 2019), it was seen that there was still room to improve the practicability of the lattice models. This was the motivation for developing a lattice program with practical use.

The LM was observed to be practical to model many in-plane lattice scenarios. This model is a proof of concept that the lattice models may be practicably used for engineering studies as well as in engineering practice in the future.

5.4 Future Studies

There are a number of topics that can be studied as a follow-up to this study. These are:

- Inelastic material behavior may be studied for the proposed random mesh. The method of sequential elastic analysis (SLA) can be a candidate for implementing such inelastic behavior as it has good convergence characteristics (Aydin et al., 2018).
- A study can be conducted to show that the proposed random mesh can be used to model the behavior of indirect tension failure in concrete since the random meshes do not have the shortcoming of being locked under compression as the regular meshes have.
- Other random types of lattice types may be implemented such as lattice beam model (LBM) or rigid body spring model (RBSM) with the "User-Friendly Lattice Model Program with Graphical User Interface" (LM).
- A structured overlapping lattice model (OLM) can be implemented with the MATLAB program developed.
- The MATLAB program that was developed can be improved to model the dynamic loads on elastic or inelastic lattices.
- A study may be conducted to seek a possible calibration method to cover deflection deviations of lattice models from shell models.

REFERENCES

Aydin, B. B., Binici, B., & Tuncay, K. (2021). Lattice simulation of concrete compressive behaviour as indirect tension failure. *Magazine of Concrete Research*, *73*(8), 394–409. https://doi.org/10.1680/jmacr.20.00206

Aydin, B. B., Tuncay, K., & Binici, B. (2018). Overlapping lattice modeling for concrete fracture simulations using sequentially linear analysis. *Structural Concrete*, *19*(2), 568–581. https://doi.org/10.1002/suco.201600196

Bažant, Z. P., Tabbara, M. R., Kazemi, M. T., & Pijaudier-Cabot, G. (1990). Random particle model for fracture of aggregate or fiber composites. *Journal of Engineering Mechanics*, *116*(8), 1686–1705. https://doi.org/10.1061/(asce)0733-9399(1990)116:8(1686)

Beale, P. D., & Srolovitz, D. J. (1988). Elastic fracture in random materials. *Physical Review B*, *37*(10), 5500–5507. https://doi.org/10.1103/PhysRevB.37.5500

Belgin, Ç. M., & Şener, S. (2008). Size effect on failure of overreinforced concrete beams. *Engineering Fracture Mechanics*, *75*(8), 2308–2319. https://doi.org/10.1016/j.engfracmech.2007.09.006

Berton, S., & Bolander, J. E. (2006). Crack band model of fracture in irregular lattices. *Computer Methods in Applied Mechanics and Engineering*, *195*(52), 7172–7181. https://doi.org/10.1016/j.cma.2005.04.020

Bolander, J. E., & Sukumar, N. (2005). Irregular lattice model for quasistatic crack propagation. *Physical Review B - Condensed Matter and Materials Physics*, *71*(9), 094106-1-094106-12. https://doi.org/10.1103/PhysRevB.71.094106

Chen, H., Lin, E., Jiao, Y., & Liu, Y. (2014). A generalized 2D non-local lattice spring model for fracture simulation. *Computational Mechanics*, *54*(6), 1541–1558. https://doi.org/10.1007/s00466-014-1075-4

Feng, S., & Sen, P. N. (1984). Percolation on elastic networks: New exponent and threshold. *Physical Review Letters*, *52*(3), 216–219. https://doi.org/10.1103/PhysRevLett.52.216

Garboczi, E. J., & Thorpe, M. F. (1986). Effective-medium theory of percolation on central-force elastic networks. III. the superelastic problem. *Physical Review B*, *33*(5), 3289–3294. https://doi.org/10.1103/PhysRevB.33.3289

G.-F. Zhao. (2014). Development of the distinct lattice spring model for large deformation analyses. *International Journal for Numerical and Analytical Methods in Geomechanics*, *38*, 1078–1100.

Hrennikoff, A. P. (1940). *Plane stress and bending of plates by method of articulated framework* (Doctoral dissertation, Massachusetts Institute of Technology).

Hwang, Y. K., Bolander, J. E., Hong, J. W., & Lim, Y. M. (2020). Irregular lattice model for geometrically nonlinear dynamics of structures. *Mechanics Research Communications*, *107*, 103554. https://doi.org/10.1016/j.mechrescom.2020.103554

Işık, G. (2019). *Numerical modelling of reinforced concrete pressure tunnels using the overlapping lattice method* [Thesis (M.S.) - Graduate School of Natural and Applied Sciences . Earthquake Studies.]. Middle East Technical University.

Jirásek, M., & Bazant, Z. (1995). Particle model for fracture and statistical micro-macro correlation of material constants. *Fracture Mechanics of Concrete Structures (Proc" 2nd Int. Conf, on Fracture Mech, of Concrete and Concrete Structures (FraMCoS-2), held at ETH, Zurich), ed, by F,H, Wittmann, Aedificatio Publishers, Freiburg, Germany, 955-964.*

Jirásek, M., & Bažant, Z. P. (1995). Particle model for quasibrittle fracture and application to sea ice. *Journal of Engineering Mechanics*, *121*(9), 1016–1025. https://doi.org/10.1061/(asce)0733-9399(1995)121:9(1016)

Kawai, T. (1978). New discrete models and their application to seismic response analysis of structures. *Nuclear Engineering and Design*, *48*(1), 207–229. https://doi.org/10.1016/0029-5493(78)90217-0

Macek, R. W., & Silling, S. A. (2007). Peridynamics via finite element analysis. *Finite Elements in Analysis and Design*, *43*(15), 1169–1178. https://doi.org/10.1016/j.finel.2007.08.012

Moukarzel, C., & Herrmann, H. J. (1992). A vectorizable random lattice. *Journal of Statistical Physics*, *68*(5–6), 911–923. https://doi.org/10.1007/BF01048880

Nikolić, M., Karavelić, E., Ibrahimbegovic, A., & Miščević, P. (2018). Lattice element models and their peculiarities. *Lattice Element Models and Their Peculiarities. Archives of Computational Methods in Engineering,* 2018, 25 (3), pp.753-784. ff10.1007/s11831-017-9210-y. ffhal-01997368f

Pan, Z., Ma, R., Wang, D., & Chen, A. (2018). A review of lattice type model in fracture mechanics: theory, applications, and perspectives. *Engineering Fracture Mechanics*, *190*, 382–409. https://doi.org/10.1016/j.engfracmech.2017.12.037

Paskin, A., Gohar, A., & Dienes, G. J. (1980). Computer simulation of crack propagation. *Physical Review Letters*, *44*(14), 940–943. https://doi.org/10.1103/PhysRevLett.44.940

Rots, J. G. (2001). Sequentially linear continuum model for concrete fracture. *Fracture Mechanics of Concrete Structures, Proceedings of the 4th International Conference on Fracture Mechanics of Concrete and Concrete Structures*, 831–839.

Schlangen, E., & Garboczi, E. J. (1997), *Fracture Simulations of Concrete Using Lattice Models: Computational Aspects, Engineering Fracture Mechanics*, Vol. 57, No. 2-3, 1997, pp. 319-332. doi:10.1016/S0013-7944(97)00010-6

Schlangen, E., & van Mier, J. G. M. (1992). Simple lattice model for numerical simulation of fracture of concrete materials and structures. *Materials and Structures*, *25*(9), 534–542. https://doi.org/10.1007/BF02472449

Schlangen, E., & Garboczi, E. J. (1996). New method for simulating fracture using an elastically uniform random geometry lattice. *International Journal of Engineering Science*, 34(10), 1131–1144. https://doi.org/10.1016/0020-7225(96)00019-5

Silling, S. A. (2000). Reformulation of elasticity theory for discontinuities and long-range forces. *Journal of the Mechanics and Physics of Solids*, *48*(1), 175–209. https://doi.org/10.1016/S0022-5096(99)00029-0

Soysal Albostan, B. F., Arıcı, Y., Binici, B., & Tuncay, K. (2017 0). *Comparison of the Overlapping Lattice and the Finite Element Approaches for the Prediction of the Collapse State of Concrete Gravity Dams.* 4th International Conference on Earthquake Engineering and Seismology, (11-13 October 2017), Eskişehir, Turkey. https://hdl.handle.net/11511/76256

Tuncay, K., Binici, B., & Yaman, İ. Ö. (2016 0). *Computational Modelling of Pressurized Reinforced Concrete Tunnel Linings*. https://hdl.handle.net/11511/81483

Yamamoto, Y. O. S. H. I. H. I. T. O., Isaji, Y., Nakamura, H. I. K. A. R. U., & Miura, T. A. I. T. O. (2019, June). Collapse simulation of reinforced concrete including localized failure and large rotation using extended RBSM. In *Proceedings of the 10th International Conference on Fracture Mechanics of Concrete and Concrete Structure (FraMCoS X), Bayonne, France* (pp. 24-26).

Zubelewicz, A., & Bažant, Z. P. (1987). interface element modeling of fracture in aggregate composites. *Journal of Engineering Mechanics*, *113*(11), 1619–1630. https://doi.org/10.1061/(asce)0733-9399(1987)113:11(1619)

APPENDICES

A. Pseudocodes For Solving EAs

The pseudocode written in MATLAB for the EA calibration method for the global approach (Described in3.1.9.1) is given below.

```
LengthIn=zeros(nel,1);
for i=1:1:nel
   x1=x(elnode(i,1));
   y1=y(elnode(i,1));
   x2=x(elnode(i,2));
   y2=y(elnode(i,2));

   LengthIn(i,1)=sqrt((x1-x2)^2+(y1-y2)^2);
end

%Check equality given in Jirasek & Bazant (1995)
Af=sum(ar); %floe area
sumL=sum(LengthIn);
Abazant=3*Af/sumL;

EA=Abazant*ones(nel,1);
```

The pseudocode written in MATLAB for the EA calibration method for the local approach (Described in 3.1.9.2) is given below.

```
AA=zeros(6*size(x,1),size(elnode,1));
rhss=zeros(1,6*size(x,1));
eqno=-5;
for i=1:nn
    eqno=eqno+6;
    x0=x(i);
    y0=y(i);
    n=nnel(i); %no of nodes connected to each node
    Abar=ar(i);

    xx=zeros(n,1);
    yy=zeros(n,1);
    L=zeros(n,1);
    c=zeros(n,1);
    s=zeros(n,1);

    for j=1:n
```

```
        if elnode(elements(i,j),1)==i %considered node is i
and other node of the element is kk
            kk=elnode(elements(i,j),2);
        else
            kk=elnode(elements(i,j),1);
        end
        xx(j)=x(kk);
        yy(j)=y(kk);

        xx(j)=(xx(j)+x0)/2; %needed to satifsy energy
dengsity eqns at nodes
        yy(j)=(yy(j)+y0)/2; %because only half the element
contributes

        L(j)=((xx(j)-x0)^2+(yy(j)-y0)^2)^0.5;
        c(j)=(xx(j)-x0)/L(j);
        s(j)=(yy(j)-y0)/L(j);
    end

    for j=1:n
        AA(eqno,elements(i,j))=L(j)*c(j)^4/(Abar*t);
        AA(eqno+1,elements(i,j))=L(j)*s(j)^4/(Abar*t);

AA(eqno+2,elements(i,j))=L(j)*s(j)^2*c(j)^2/(Abar*t);

AA(eqno+3,elements(i,j))=L(j)*s(j)^2*c(j)^2/(Abar*t);
        AA(eqno+4,elements(i,j))=L(j)*c(j)*s(j)^3/(Abar*t);
        AA(eqno+5,elements(i,j))=L(j)*c(j)^3*s(j)/(Abar*t);
    end

    rhss(eqno)=E/(1-nu^2);
    rhss(eqno+1)=E/(1-nu^2);
    rhss(eqno+2)=E/(2*(1+nu));
    rhss(eqno+3)=E*nu/(1-nu^2);
    rhss(eqno+4)=0;
    rhss(eqno+5)=0;
end
% size(x)
% size(elnode)
% size(AA)

Length=zeros(nel,1);
for i=1:1:nel
   x1=x(elnode(i,1));
   y1=y(elnode(i,1));
   x2=x(elnode(i,2));
   y2=y(elnode(i,2));

   Length(i,1)=sqrt((x1-x2)^2+(y1-y2)^2);
end
```

```
%Check equality given in Jirasek & Bazant (1995)
Af=sum(ar); %floe area
sumL=sum(Length);
Abazant=3*Af/sumL;

AA=sparse(AA);
AATrhssT=AA'*rhss';

KK=zeros(size(AA,1),size(AA,1));
KK=sparse(AA.'*AA);

Kmax=max(max(KK));
rhsmax=max(AATrhssT);
lambda=1*max(rhsmax,Kmax);

for j=1:nel
    %regularization
    KK(j,j)=KK(j,j)+lambda;
end
rhss2=AATrhssT+lambda*Abazant*ones(size(AATrhssT,1),1);

EA=sparse(KK)\rhss2;
```

The pseudocode written in MATLAB for the EA calibration method for the hybrid approach (Described in 3.1.9.3) is given below.

```
AA=zeros(6*size(x,1),size(elnode,1));
rhss=zeros(1,6*size(x,1));
eqno=-5;
for i=1:nn
    eqno=eqno+6;
    x0=x(i);
    y0=y(i);
    n=nnel(i); %no of nodes connected to each node
    Abar=ar(i);

    xx=zeros(n,1);
    yy=zeros(n,1);
    L=zeros(n,1);
    c=zeros(n,1);
    s=zeros(n,1);

    for j=1:n
        if elnode(elements(i,j),1)==i %considered node is i
and other node of the element is kk
            kk=elnode(elements(i,j),2);
        else
```

```
            kk=elnode(elements(i,j),1);
        end
        xx(j)=x(kk);
        yy(j)=y(kk);

        xx(j)=(xx(j)+x0)/2; %needed to satifsy energy
dengsity eqns at nodes
        yy(j)=(yy(j)+y0)/2; %because only half the element
contributes

        L(j)=((xx(j)-x0)^2+(yy(j)-y0)^2)^0.5;
        c(j)=(xx(j)-x0)/L(j);
        s(j)=(yy(j)-y0)/L(j);
    end

    for j=1:n
        AA(eqno,elements(i,j))=L(j)*c(j)^4/(Abar*t);
        AA(eqno+1,elements(i,j))=L(j)*s(j)^4/(Abar*t);

AA(eqno+2,elements(i,j))=L(j)*s(j)^2*c(j)^2/(Abar*t);

AA(eqno+3,elements(i,j))=L(j)*s(j)^2*c(j)^2/(Abar*t);
        AA(eqno+4,elements(i,j))=L(j)*c(j)*s(j)^3/(Abar*t);
        AA(eqno+5,elements(i,j))=L(j)*c(j)^3*s(j)/(Abar*t);
    end

    rhss(eqno)=E/(1-nu^2);
    rhss(eqno+1)=E/(1-nu^2);
    rhss(eqno+2)=E/(2*(1+nu));
    rhss(eqno+3)=E*nu/(1-nu^2);
    rhss(eqno+4)=0;
    rhss(eqno+5)=0;
end

AA=sparse(AA);
AATrhssT=AA'*rhss';

KK=zeros(size(AA,1),size(AA,1));
KK=sparse(AA.'*AA);

Kmax=max(max(KK));

for j=1:nel
    KK(j,j)=KK(j,j)+1e-8*Kmax; %for str;
end

rhss2=AATrhssT;

EA=sparse(KK)\rhss2;
```

B. Runtime Performance of the LM

Node numbers, element numbers and MATLAB run-times are presented to calibrate the EA values of a sample 3x12 axially loaded rod problem. Mesh generation, Voronoi algorithm, linear stiffness analysis, plotting, and other miscellaneous parts of the program are not included in these run times. The computer that is used to generate these run times have a processor of 2.30GHz, and 64GB RAM.

Table B.1: Runtime perfomance of 3x12 axially loaded rod, Random Mesh: Type 3, solved using the hybrid approach.

Reference Element length, d	Number of Nodes	Number of Elements	Runtime (s)
1	59	586	0.022659
1/2	203	2608	0.050178
1/4	706	9594	0.249868
1/8	2452	32087	1.794588