THREE-DIMENSIONAL ANALYSIS OF GEOGRID REINFORCEMENT USED IN PILE-SUPPORTED EMBANKMENTS

Kyle A. Halvordson

Thesis submitted to the faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Master of Science
In
Civil Engineering

Raymond H. Plaut, Chairman

George M. Filz Elisa D. Sotelino

December 3, 2007
Blacksburg, Virginia

Keywords: Geosynthetic Reinforcement, Geogrid, Minimization of Energy, Cable Net

THREE-DIMENSIONAL ANALYSIS OF GEOGRID REINFORCEMENT USED IN PILE-SUPPORTED EMBANKMENTS

Kyle A. Halvordson

ABSTRACT

Pile-supported geogrid-reinforced embankments are an exciting new foundation system that is utilized when sites are limited by a soft soil or clay. In this system, an embankment is supported by a bridging layer, consisting of granular fill and one or multiple layers of geogrid reinforcement. The bridging layer transfers the load to piles that have been driven into the soft soil or clay. The load from the embankment induces large deformations in the geogrid reinforcement, causing tensile forces in the ribs of the geogrid. Many of the current methods used to design geogrid reinforcement for this system simplify the approach by assuming that the reinforcement has a parabolic deformed shape. The purpose of this thesis is to thoroughly examine the behavior of the geogrid in a pile-supported embankment system, in an effort to determine the accuracy of the parabolic deformed shape, and identify the most important parameters that affect reinforcement design.

The geogrid was analyzed using a three-dimensional model that included a cable net to represent the geogrid and linear springs to represent the soil underneath the geogrid. A larger pressure was applied to the geogrid regions that are directly above the pile caps so that arching effects could be considered, and the stiffness of the springs on top of the pile were stiffer to account for the thin layer of soil between the geogrid and the pile cap. A Mathematica algorithm was used to solve this model using the minimization of energy method.

The results were compared to another model of this system that used a membrane to represent the geosynthetic reinforcement. Additionally, the maximum strain was compared to the strain obtained from a geosynthetic reinforcement design formula. A parametric study was performed using the Mathematica algorithm by varying the pile width, embankment pressure applied to the soil, embankment pressure applied to the pile, stiffness of the soil, stiffness of the soil on top of the pile, stiffness of the geogrid, geogrid orientation, rotational stiffness of the geogrid, and the layers of geogrid reinforcement.

Acknowledgements

I would like to express my sincere gratitude to Professor Raymond Plaut for inviting me to partake in this research project and for guiding my efforts throughout this endeavor. He has made several contributions to this research, and his assistance in reviewing this document is very appreciated. I would also like to thank my committee members Professor George Filz and Professor Elisa Sotelino, for their guidance and help throughout this research effort. Additionally I would like to thank Professor Filz for sharing his expertise in this field, and Professor Sotelino for her assistance in reviewing this document.

I would also like to thank Mike Woodworth and Brent Jones for helping me while I was learning Mathematica. I worked alongside Brent Jones while completing this research, and I think both of our works has been influenced by our countless discussions on the subject.

I would like to thank my family for their support and encouragement throughout my education. My Mother, for finding creative ways to send me care packages to keep my spirits high and my Father, for his guidance. Thanks to all my friends I've made and kept throughout this endeavor, for having a sense of humor and being there when I needed a break.

Financial support was provided by the National Science Foundation under Grant No. CMS-0408281.

Table of Contents

List of Figures

List of Tables

Chapter 1
Introduction and Literature Review

The definition provided by the International Geosynthetic Society for a geosynthetic is "A planar, polymeric (synthetic or natural) material used in contact with soil/rock and/or other geotechnical material in civil engineering applications." The first reported use of geosynthetic reinforcement was by the ancient cultures of Mesopotamia, who used a rudimentary form of geosynthetic to support their great pyramid shaped temples called Ziggurats. However, modern geosynthetic reinforcement is only a recent engineering innovation that was first introduced in Europe during the late 1960's (Pinto 2003).

Soil is an excellent compressive material, but there is no bond between grains, so soil has no resistance to tensile forces. Engineers have utilized this characteristic of soil by choosing materials for geosynthetic reinforcement that have little resistance to compression forces but are excellent in resistance to tensile forces for a perfect balance of composite materials. The soil transfers tensile stresses to the geosynthetic reinforcement through two mechanical processes, friction and the anchoring effect caused by the soil interlock through the apertures in the geosynthetic. The soil and geosynthetic interaction is dependent on the compaction and homogeneity of the embankment fill (Pinto 2003).

Geosynthetic reinforcement is used in several civil engineering applications, the most frequent being: roads and railways, foundations, embankments, steep slopes, and retaining walls. These applications can be subdivided into two groups, load-supporting structures and earth structures. Load-supporting structures support the structure's self-weight and an external load, while earth structures just support the structure's self-weight. Roads and foundations are examples of load-supporting structures, and embankments and steep slopes are earth structures (Pinto 2003).

This thesis is focused on the use of geosynthetic-reinforced pile-supported (GRPS) embankment systems. This system may be used as an earth structure (e.g., an embankment) or a load-supporting structure (e.g., an embankment designed to support a

road or railway). In the GRPS system, a "bridging layer" consisting of granular fill and geosynthetic reinforcement transfers the external load and the self-weight of the embankment to piles that have been driven into soft soil or clay. The load transferred to the piles is then transferred directly to the bedrock below the soft soil. Figure 1.1 illustrates all the components of this system. This thesis is subsequent work to a dissertation written by Stewart (2005) on the efficiency of the GRPS embankment system in transferring load to the piles. The results obtained from Stewart were used to determine the load acting directly on the geosynthetic reinforcement so the performance of the geosynthetic reinforcement in this system could be evaluated.

The geosynthetic reinforcement serves three functions in this system: transfer load to the piles through the membrane effect, oppose the axial thrust that occurs at the sides of the embankment, and separate the embankment fill from the clay or soft soil (Pinto 2003). The scope of this thesis does not cover the latter two functions. Unloaded geosynthetic reinforcement cannot resist normal forces until deformation has occurred because the

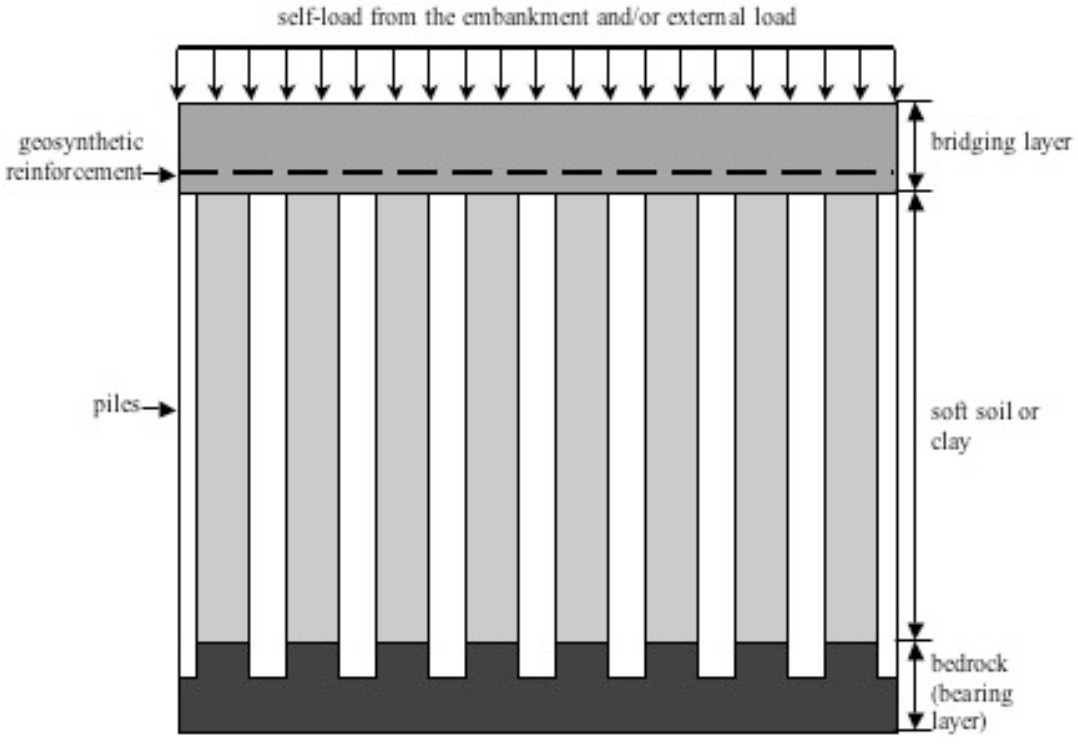

Figure 1.1 Geosynthetic-Reinforced Pile-Supported Foundation System

reinforcement rests flat on the soil, and most geosynthetic material offers virtually no shear resistance (Koerner 2005, Villard et al. 2002). The geosynthetic reinforcement's ability to increase its resistance to normal loads as deformations increase is called the membrane effect. When large deformations occur, the curvature of the reinforcement causes the vertical stress to distribute to soil regions that have less load or no load at all, reducing the vertical stress acting on the loaded region (Burd 1995).

Many load-supporting and earth construction projects have utilized geosynthetic reinforcement and reported significant time and cost savings. Some estimate that geosynthetic reinforcement is responsible for a cost savings of 30 percent (Pinto 2003). Geosynthetic reinforced structural systems save money because less granular fill is required for the bridging layer and piles may be spaced farther apart. Geosynthetic-reinforced systems also speed up construction and may be installed in most weather conditions, so reinforced systems permit more efficient designs and construction techniques.

1.1 Design of Geosynthetic-Reinforced Embankments

There are three steps to designing a GRPS embankment system. First, the amount of load carried by the pile is calculated by summing the load acting directly on the pile and the additional load caused by soil arching. The next step in the design process is to calculate the tension in the geosynthetic reinforcement caused by the vertical load acting on the reinforcement that is transferred to the piles. The final step is to calculate the tension that is superimposed on the reinforcement by the lateral spreading of the embankment (Russell and Pierpoint 1997). As stated above, the final step will not be discussed in this thesis. The tension design force of the geosynthetic reinforcement is found by assuming a deformed shape and then calculating the tension using mechanical principles. Some researchers dispute the methods used for the design of geosynthetic reinforcement, because the methods used to calculate the tension stress in the geosynthetic make inaccurate assumptions and the methods used to calculate the vertical load acting on the tensile reinforcement don't provide consistent results.

1.1.1 Vertical Loads on the Geosynthetic Reinforcement

The major source of disagreement among the different design methods is the effect soil arching has on the distribution of load between the soil and the piles. Soil arching occurs when the shear resistance between the supported and unsupported soil overcomes the weight of the unsupported soil, causing the unsupported soil to become stable. Arching can be visualized as a dome resting on four columns (Russell and Pierpoint 1997). The dome is comprised of the soil and the columns represent the piles.

None of the six accepted design methods, BS8006, adapted Terzaghi 1 and 2, Hewlett and Randolph, the adapted Guido, and Carlsson, account for the load supported by the clay or soft soil beneath the geosynthetic. It is conservatively assumed that the vertical load acting between the piles that is not transferred to the pile through soil arching, is transferred to the pile through the geosynthetic reinforcement. In this manner, the load transferred to the pile through the geosynthetic reinforcement is calculated by subtracting the load transferred to the pile through arching and the load directly resting on the pile, from the total load on the structure. Stewart and Filz (2005) disagree with this assumption and they have included the support provided by the soft soil, between the piles, in their design method.

A comparison of the SRR values calculated using the six design methods is shown in Table 1.1. In the table, a is the width of the pile, s is the pile spacing, and H is the height of the embankment.

Table 1.1 Comparison of the SRR values for Six Design Methods (Stewart 2005)

	SRR					
	a/s=0.30		**a/s=0.40**		**a/s=0.50**	
Method	**H/s=1.5**	**H/s=4**	**H/s=1.5**	**H/s=4**	**H/s=1.5**	**H/s=4**
BS8006	0.74	0.274	0.401	0.144	0.089	0.022
Adapted Terzaghi 1	0.596	0.313	0.49	0.225	0.39	0.162
Adapted Terzaghi 2	0.769	0.522	0.689	0.411	0.602	0.314
Hewlett & Randolph	0.529	0.506	0.425	0.33	0.337	0.205
Adapted Guido	0.11	0.041	0.094	0.035	0.079	0.029
Carlsson	0.435	0.163	0.373	0.14	0.311	0.117

Russell and Pierpoint (1997) conclude that no present design method is reliable and the only accurate method of determining the settlement and tension in the geosynthetic is by performing a three-dimensional numerical analysis. The BS8006 method overpredicts and underpredicts the stress reduction ratio, depending on the variables, and the adapted Guido method consistently underpredicts the stress reduction ratio. The BS8006 method does not satisfy vertical or horizontal equilibrium, and the adapted Guido method is based on empirical results of small-scale experiments with weak reinforcement layers (Love and Milligan 2003). The Terazaghi methods and Hewlett and Randolph method have the closest results when compared to a three-dimensional analysis.

1.1.2 Geogrid Reinforcement

There are four failure modes for a GRPS system. The granular fill above the geosynthetic can undergo bearing failure if the reinforcement is placed too low in the embankment. A second failure can occur from anchorage pullout, a result of an inadequate embedment length. A third failure mode, however very rare, is the tensile failure of the geosynthetic reinforcement. Finally, creep may cause failure in the geosynthetic reinforcement (Koerner 2005).

The materials used for geosynthetic reinforcement often suffer from creep, so the short-term and long-term strengths may be quite different (Pinto 2003). Creep effects may be minimized by loading the geosynthetic beyond the yield stress during the manufacturing process (Koerner 1998).

The design strain has a large effect on the design load determined using many of the accepted design methods. To maintain accuracy in design, it is important to maintain compatibility between the settlement and the strain. A design strain of 5% is usually assumed, and it is common practice in the United States to design a reinforcement system for which the reinforcement does not exceed this strain (Russell and Pierpoint 1997).

Many design methods assume that the geosynthetic reinforcement undergoes a constant strain throughout the length of the deformed shape. The methods also assume that the deformed shape is circular or parabolic (Espinoza 1993). The strain is only constant for the entire length of reinforcement if the deformed shape is perfectly circular or parabolic, and the deformed shape of the geosynthetic reinforcement is never perfectly circular for any reinforcement application. No in-depth numerical analysis that has been performed on a pile-supported embankment model has predicted a parabolically deformed shape for the reinforcement. For a design method to be accurate, it is necessary to account for a variable strain in the geosynthetic reinforcement, and since none of the accepted design methods account for a variable strain, they are not accurate.

1.2 Previous Research

The research that has been done in analyzing geosynthetic reinforced foundation systems has been very extensive, but many models have been analyzed in two dimensions and there is a limited amount of research that has included three dimensions. A few research studies have compared field measurements of actual geosynthetic-reinforced sites to finite element analysis results. A few studies that have been conducted on geosynthetic-reinforced pile-supported embankments will be discussed briefly, and several suggestions made by past researchers that have affected the present research will be noted.

There are two methods of analysis, equilibrium and continuum or compatibility analysis. Equilibrium analysis is usually the preferred method among researchers, but the geosynthetic-reinforced embankment system is indeterminate so assumptions have to be made for the reactions or the deformed shape of the system. These assumptions may not be accurate. Equilibrium analysis doesn't typically predict displacements, and an equilibrium analysis doesn't maintain compatibility among the displacements. So the displacements calculated using a finite element or finite difference analysis are more accurate, because finite element and finite difference are both continuum analyses (Agaiby and Jones 1995). This thesis investigates how the geosynthetic reinforcement

performs under specific loading conditions, and the compatibility between the soil and geosynthetic is not an issue because the soil is modeled as a layer of linear springs.

Madhav and Poorooshasb (1989) derived a two-dimensional model that used a Pasternak shear layer in conjunction with a Winkler foundation model. Shukla, Kumar, and Chandra (1994) also used a Pasternak shear layer in a mechanical model of a strip loading applied to a layer of geosynthetic reinforcement. The model was solved using the finite difference method. A few years later, Shukla and Chandra (1996) compared the Pasternak foundation model to the Kerr and Winkler models. Deb, Chandra, and Basudhar (2005) expanded on this model by adding multiple layers of reinforcement separated by Pasternak shear layers. The study found that the bearing capacity doubled when three layers were used instead of no reinforcement at all, and the increase of effectiveness that is caused by the addition of reinforcement is reduced as more layers of reinforcement are added. The study also found that the ideal position for reinforcement is nearest to the interface of the granular and soft soil.

Russell and Pierpoint (1997) performed a numerical analysis using a finite difference program called FLAC 3D. The reinforcement was represented as one-dimensional linearly elastic cable elements, and the soft soil below the reinforcement was not modeled so that an accurate comparison could be made between the numerical analysis and the design methods. To the author's knowledge, this is the only analysis that has been conducted where the reinforcement was represented as cables in a three-dimensional pile-reinforced cell.

Villard and Giraud (1998) conducted a finite element analysis of a geosynthetic woven membrane and compared the results to numerical analysis results obtained from other authors. Several years later, Villard et al. (2002) analyzed a pile-supported embankment model in three dimensions with three different geosynthetic reinforcements: a geosynthetic with fibers parallel to the lines of piles, a geosynthetic with fibers oriented 45 degrees to the lines of piles, and an elastic membrane. The model only included a quarter of the entire cell to save computer memory and time. A quarter could be modeled

because of the cell's symmetry. The geosynthetic with fibers parallel to the lines of piles had the largest displacement in the center of the cell, and the geosynthetic that had fibers oriented 45 degrees to the lines of piles had very similar displacements at the center and center of the cell edge.

Rogbeck, Gustavsson, Sodergren, and Lindquist (1998) took strain gage and settlement readings at the site of a road embankment supported by a geogrid and pile foundation system. They reported that the two design methods that were used, BS8006 and Carlsson, were too conservative, but a two-dimensional finite difference model solved using FLAC was very accurate. The authors suggested that if the model could have been formulated in three dimensions, the differences between the field measurements and the analysis results might have been corrected.

Fakher and Jones (2001) studied the effect of bending stiffness on the analysis of geosynthetic reinforcement systems and concluded that bending stiffness is insignificant unless the geosynthetic is reinforcing super soft clays. Most geosynthetic reinforces compacted granular fill, so bending stiffness should not be a concern in a pile-supported embankment system. The actual bending stiffness of geosynthetic reinforcement was not mentioned in the report, but the bending stiffness often varies between different types of geosynthetic and is usually minimal.

Han and Gabr (2002) performed a numerical analysis that included arching effects, and found that geosynthetics improve the distribution of load to the piles. The numerical analysis also found that the maximum tension stresses occur near the edge of the pile.

Huang, Han, and Collin (2005) reported on a geogrid-reinforced pile system that supports a railway embankment. Field measurements of the settlements and strains were collected for six years, and the results were compared to a numerical analysis performed with FLAC 3D. The foundation soil was modeled as a linearly elastic-perfectly plastic material. The numerical analysis produced similar settlements to those reported from the field measurements, but the tension forces were not as close as expected. The FLAC

analysis of the geogrid system consisting of three layers of reinforcement found that the three layers together exhibited similar behavior to beams. The top layer had the largest tensile forces occur at the edge of the pile, and the bottom layer had the largest tensile forces occur at the mid-point of the pile spacing.

A parametric study performed by Yan, Yang, and Han (2006) concluded that a geosynthetic stiffness of 3000 kilo-Newtons per meter and a pile spacing of 2 to 2.5 meters provides the best economy for a GRPS system. If the stiffness of the geosynthetic is increased or the pile spacing is decreased beyond these values, the settlements do not decrease significantly. The study also found that the tensile stress concentration and maximum tension of the geosynthetic increase when the height of the embankment, stiffness of the geosynthetic, or modulus of elasticity of the pile increase. The settlement at the top of the embankment decreases when the number of geosynthetic layers, stiffness of the geosynthetic, pile stiffness, or soil stiffness increases.

Cao, Chen, and Chen (2006) also analyzed pile-reinforced embankments using a mechanical model and found that the results compared well with field measurements obtained from a highway site. The analysis was performed using the principle of minimum potential energy. A parametric study revealed that the pile spacing, embedment height, cap size, pile stiffness, and soil stiffness had an impact on the distribution of load that the pile supported, but the shear modulus of the fill and the geomembrane stiffness had little effect.

Chen and Yang (2006) outlined a simple method for determining the settlement, and used it to perform a parametric study with varying cell length, soil stiffness, pile diameter, height of embankment, and ratios of pile to soil loading.

Several suggestions were taken from past research and used in the present study:

- Pile caps should cover 10 to 25 percent of the embankment area (Milligan and Love 2003).

- Cables more accurately predict the membrane effect when deformations in the reinforcement are large, because the cable elements do not have bending stiffness or shear resistance (Villard et al. 2002).
- Settlement of the surface is almost negligible, while the deformations at the base of the embankment are significant (Russell and Pierpoint 1997).
- Shukla and Chandra (1996) believe that until accurate elastic parameter values for soil are known, a finite element analysis is not more accurate than the solution of a mechanical model, and a mechanical model saves computer time.
- Moraci and Recalcati (2006) suggest that the soil and geogrid interaction does not have a significant effect on the tensile strength of a geogrid in soil because the tensile strength of a geogrid obtained from an air pullout test is almost identical to the strength obtained from a soil pullout test.

Shukla and Chandra's conclusion influenced the research performed for this thesis because a mechanical model was used instead of a finite element analysis, and the geogrid was analyzed as a linear-elastic cable net to more accurately predict the membrane effect. The results focus on the settlement of the geogrid and the tensile forces that were developed in it because the settlement at the surface of the embankment is usually negligible, and the soil-geogrid interaction was not taken into account because of the conclusion made by Moraci and Recalcati. The parametric study that was performed using the mechanical model included several pile caps that covered from 10 to 25 percent of the embankment area.

1.3 Overview of Thesis

In Chapter Two, the model and the parameters used in the analysis are introduced. The assumptions made to analyze the model are described.

Chapter Three describes the procedure and results obtained from tensile rupture tests that were performed on samples of geogrid reinforcement.

Chapter Four outlines several methods that are used to analyze cable net structures, and then derives the minimization of energy method that was applied to analyze the model described in Chapter Two. The minimization of energy method was used to solve four example problems in order to verify the accuracy of the method.

Chapter Five describes in detail the Mathematica implementation of the minimization of energy method. It describes the purpose of all the routines used in the analysis and how they work. All the modifications that were made to the program to solve different case studies are also discussed.

The results of a parametric study are presented in Chapter Six. The programming routines used to find the results in Mathematica are described, and the variables and modifications used in the model are discussed for each case study. The effects of vertical pressure, pile stiffness, soil stiffness, pile width, geogrid stiffness, geogrid orientation, joint bending stiffness, and multiple layers of reinforcement on the mechanical model are investigated, and a brief discussion of the analysis is included. The results are compared to a similar model that used a membrane to represent the reinforcement. Additionally, the maximum strain obtained from the analysis is compared to the strain calculated using a geosynthetic design formula.

Chapter Seven summarizes the results of the parametric study and discusses conclusions that were made from the results of this investigation. The ability of the mechanical model to predict the actual behavior of a geogrid-reinforced pile-supported system is discussed, and recommendations are made for future research in this field.

Chapter 2
Model and Geometry

Geogrids have become an increasingly popular type of geosynthetic reinforcement, and are the primary type of geosynthetic used in pile-supported embankment systems. Geogrids are usually made from polymer that has been woven into ribs and laid in two directions orthogonally so that the ribs meet at square junctions. This orientation creates rectangular holes called apertures that allow soil in between the closely spaced ribs. The woven ribs are ideal for tensile forces but allow little resistance to bending or compression forces. Specific geogrid properties such as flexibility, stiffness, and aperture size vary greatly between different products. After examining a geogrid sample and investigating the properties of the woven ribs, it becomes very apparent that a cable net can represent a geogrid very well. This chapter discusses the geometry and variables of the cable net, loading, and springs that were used in a mechanical model to analyze the performance of a geogrid in a geogrid-reinforced pile-supported embankment system. To analyze this reinforcing system, several assumptions were made to make the model both solvable and simpler.

2.1 Geometry

In order to simplify the three-dimensional model of this system, only one unit cell was examined. The unit cell, shown in Figure 2.1, was a square consisting of sides of length L. The cell was made up of four square piles, and the centerlines of each pile made up the edges of the cell.

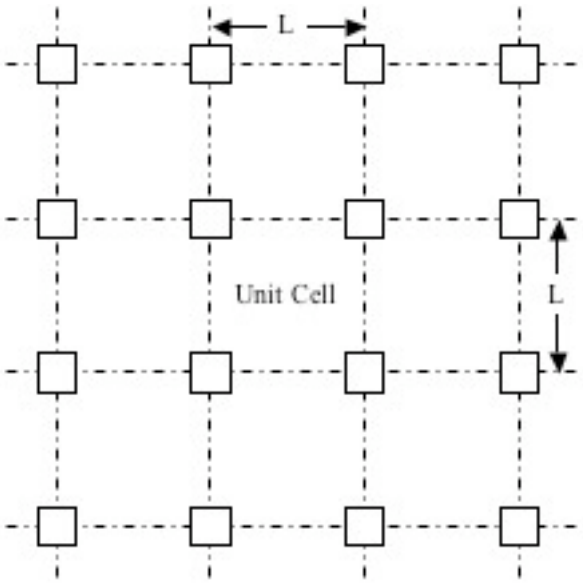

Figure 2.1 Pile Layout and Unit Cell

It was unnecessary to examine other cells because of the redundancy of this expansive multiple-pile system. The vertical pressure on the adjacent cells was assumed to be of equal magnitude causing the geogrid reinforcement to remain stationary along the pile gridlines. In other words, the location of this cell permitted the perimeter of the cell to be restrained in the direction perpendicular to each edge. A model consisting of one central cell was chosen so that the redundancy of analyzing additional similar cells was eliminated, and because a geogrid on the edge could experience unrestrained non-linear behavior if the geogrid was not anchored properly. The model primarily focused on the performance of the geogrid reinforcement, and not on the effect improper anchorage would cause.

2.2 Model

The geogrid was modeled as a linearly elastic cable net with properties based on actual values provided in the geogrid product literature. In practice, the “bridging layer” of an embankment may be reinforced with one layer or several layers of geogrid reinforcement. This model only had one cable net layer, but an additional analysis was performed with two cable net layers. The soil was modeled as a distribution of vertical linear springs that

were of comparable compressive stiffness to soil. The springs were theoretically fixed to square pile caps at the corners of the cell or a deep bedrock layer.

Experiments have found that it was reasonable to assume that load acts at the cable net junctions for small cable lengths (Buchholdt 1985). The distance from junction to junction of a geogrid is typically less than 5 cm and rarely greater than 10 cm, so this assumption can be used for geogrids as well. The uniform pressure was applied as equivalent point loads and the soil was modeled as springs that acted at each cable net junction. In addition, the point loads at the pile locations were strengthened while the other point loads were reduced to account for the arching effect described in Chapter 1. The springs at the pile locations were stiffened because there was only a thin layer of soil between the pile cap and the geogrid. In this manner, two variables were necessary to represent the two values of point loads that acted on the cable net, one variable for a point load that acted over the pile cap and another variable for the point load that acted over the soil. Two variables were also necessary to represent the two spring stiffness values that acted on the cable net, the spring stiffness that acted at the pile cap and the spring stiffness that acted at the soil locations.

The properties of geogrid reinforcement vary substantially between different manufacturers and product lines. Some geogrid product lines have anisotropic properties and others are isotropic. The model assumed that the geogrid had isotropic properties; however, a separate analysis was performed to investigate the effect of an anisotropic geogrid on the results. Two variables were used to analyze the cable net in this model, the product of the modulus of elasticity and the cross-sectional area of a geogrid rib, EA, and the distance between adjacent geogrid junctions, H. The geogrid was isotropic, so the cable net was a square grid and its stiffness was equal in both directions.

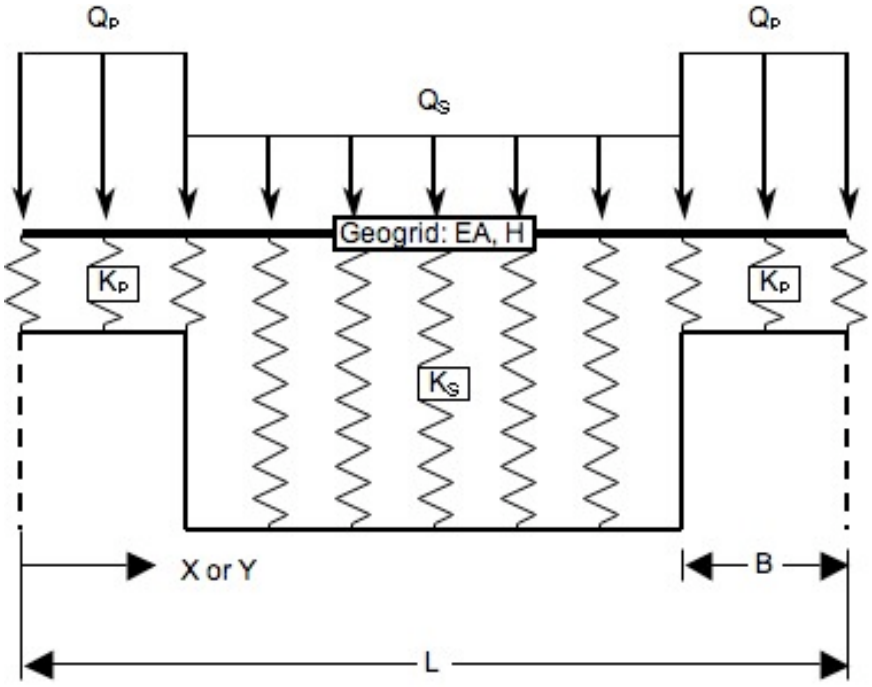

Figure 2.2 Elevation View of Model

A two-dimensional cross-section of the unit cell and the dimensional parameters used for this model are shown in Figure 2.2. Q is the uniform pressure applied to the geogrid and K is the stiffness per area of the soil. The subscripts P and S represent the load or stiffness over the pile and soil, respectively. B is half of the pile width, and L, as described above, is the distance from centerline to centerline of the piles. The unit cell was a square, so the horizontal axis could be either x or y, and the vertical axis is z.

The dimensional parameters were converted to non-dimensional parameters so that each parametric study case was applicable to multiple dimensional cases. The length units were eliminated by multiplying Q by L squared and K by L cubed, and the force units were eliminated in both parameters by dividing by EA:

$$q = \frac{Q \bullet L^2}{EA} \tag{2.1}$$

and

$$k = \frac{K \bullet L^3}{EA} \tag{2.2}$$

The dimensional values B and H were divided by L so that

$$b = \frac{B}{L} \tag{2.3}$$

and

$$h = \frac{H}{L} \tag{2.4}$$

The non-dimensional point load, p, and spring stiffness, t, that were applied at each node were found by multiplying q and k by the tributary area of the node, or h squared:

$$p = q \bullet h^2 \tag{2.5}$$

and

$$t = k \bullet h^2 \tag{2.6}$$

2.3 Assumptions

The geogrid reinforcement was modeled as a flat, linearly elastic, cable net. The assumption that geogrid reinforcement behaves linear elastically was tested. The results and a description of the testing procedure can be found in Chapter 3.

Both the friction caused by the interaction between the geogrid surface and the soil, and the lateral soil stiffness within the geogrid apertures, were assumed to be negligible. As mentioned in Chapter 1, an investigation of pullout results had found that this assumption was reasonable (Moraci and Recalcati 2006).

Two assumptions were made to account for the soil's uncertain behavior. The soil above the geogrid acted on the geogrid as two vertical pressures, a large pressure above the pile and a smaller pressure between the piles. However, it was very unlikely that the vertical load between the piles was uniform because soil arching does not evenly distribute the load over the unsupported soil. The soil supporting the geogrid was assumed to act like a distribution of linearly elastic springs, yet the validity of this assumption is unknown and it is very unlikely that soil behaves linear elastically.

Chapter 3
Geogrid Testing

Two different Huesker geogrid products were tested, Fortrac and Fornit. Among the tested products were four variations of Fortrac geogrids, 110, 80, 55, and 35 (PET), and two variations of Fornit geogrid, 20 and 30. Each variation was rated for a different strength. The tests were performed strictly to obtain the relationship of the tensile strength per width and strain of several geogrid products, and these tests were in no way an impartial verification of the reported strengths in the product literature. ASTM D6637 was used as a guideline and the procedure was modified for the testing equipment that was available.

3.1 Procedure

The procedure was adapted from ASTM D6637, but a two-inch specimen width was used because of limitations in the test apparatus. At first two ribs were tested for each specimen, but it became apparent that it would be more difficult to obtain consistent results using two ribs, so only one rib was used for the remainder of the tests. The two adjacent ribs to the tested rib were cut so that the ribs on either side provided no load path. This setup corresponds to method A of the ASTM D6637 testing procedure. The upper edge and lower edge of the geogrid were placed between the clamps of the testing apparatus, so that the rib was loaded on center. The clamps were padded with thin pieces of cardboard to prevent damage to the geogrid by the clamps. Enough of the specimen was between the clamps so that the ends were secure and would not slip during testing. The testing system was zeroed and a load rate of 5 mm per minute was set. At first the initial length of the specimen (i.e., the length of the specimen between the clamp edges) was recorded so that an accurate strain could be calculated. After comparing the hand measured initial lengths to the initial length recorded by the testing apparatus, it was decided that the hand measured initial lengths could be discontinued because the testing apparatus provided a more accurate initial length. Only the first eight test specimens were measured by hand. After the test specimen was secure and the testing apparatus was set,

the testing apparatus was initiated and the geogrid was pulled until rupture. The failure strength and elongation were recorded at failure. Thirty tests were performed over the course of two lab days. On September 29, 2006, ten geogrid specimens were tested and on October 13, twenty specimens were tested. Pictures of the testing apparatus and two geogrid specimens are shown in Figures 3.1-3.3.

3.2 Observations and Results

The majority of the specimens failed at locations close to the clamps and rib junctions. It was concluded that the specimens failed in close proximity to the clamps because the rib junction locations were near the clamps.

Figure 3.1 Testing Apparatus

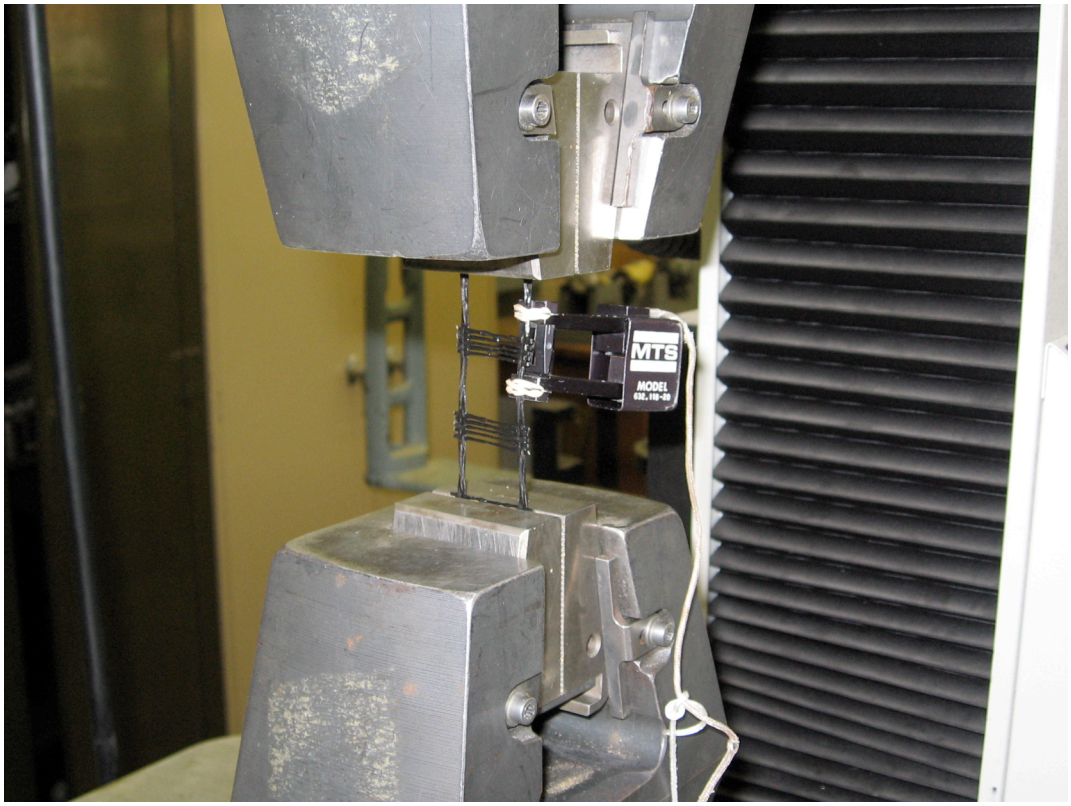

Figure 3.2 Two-Rib Specimen

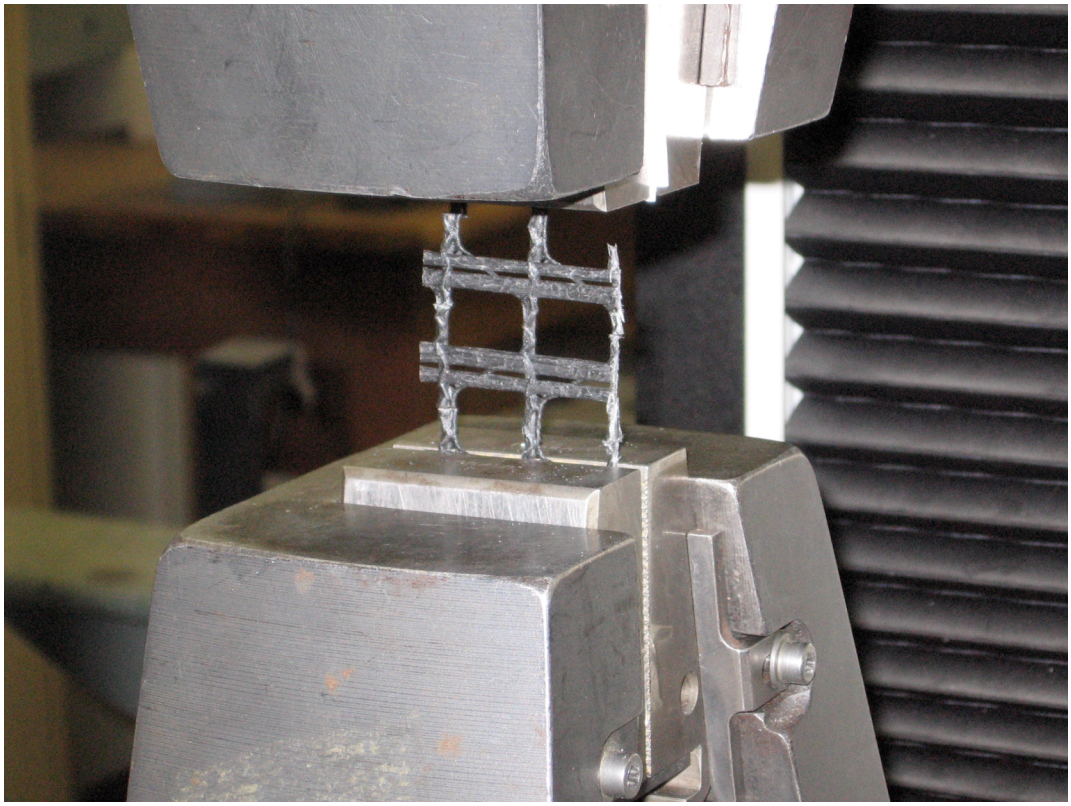

Figure 3.3 One-Rib Specimen with Adjacent Ribs Cut

During the first test date, the testing apparatus was calibrated so that the initial reading was equal to the initial distance between the clamp edges, but on the second test date, after testing had concluded, it was observed that the testing apparatus was not correctly calibrated. The initial readings of the last twenty tests were not equal to the initial length, and no hand measurements were taken to find the initial distance between clamps. In other words, an accurate strain could not be verified for the last twenty tests because the initial length of the geogrid was unknown. The results of the first ten tests provided a fairly accurate load per width to strain relationship, while the last twenty tests roughly verified the shape of the load per width versus strain curve, and all thirty tests provided an accurate reading of the rupture strength of a geogrid rib.

The testing apparatus utilized computer software to output the data from each test to a text file. Each data file had two columns, one column with the load on the specimen and another with the displacement of the specimen. Most tensile strength data for geogrid reinforcement is published as the "wide width tensile strength" or WWTS. To obtain WWTS from the output data, the force on the geogrid specimen was divided by the width of the specimen. If one geogrid rib was tested,

$$WWTS = \frac{F}{H} \tag{3.1}$$

where F = force on geogrid specimen

H = distance between geogrid junctions

and if two geogrid ribs were tested,

$$WWTS = \frac{F}{2H} \tag{3.2}$$

The displacement from the output file was used to find the strain of the geogrid rib:

$$\varepsilon = \frac{\Delta L}{L_0} \tag{3.3}$$

where ΔL = displacement

L_0 = initial length

The WWTS versus strain plots of the first ten geogrid specimens tested are shown in Figures 3.4-3.8, and the plots for the other twenty tests can be found in Appendix A.

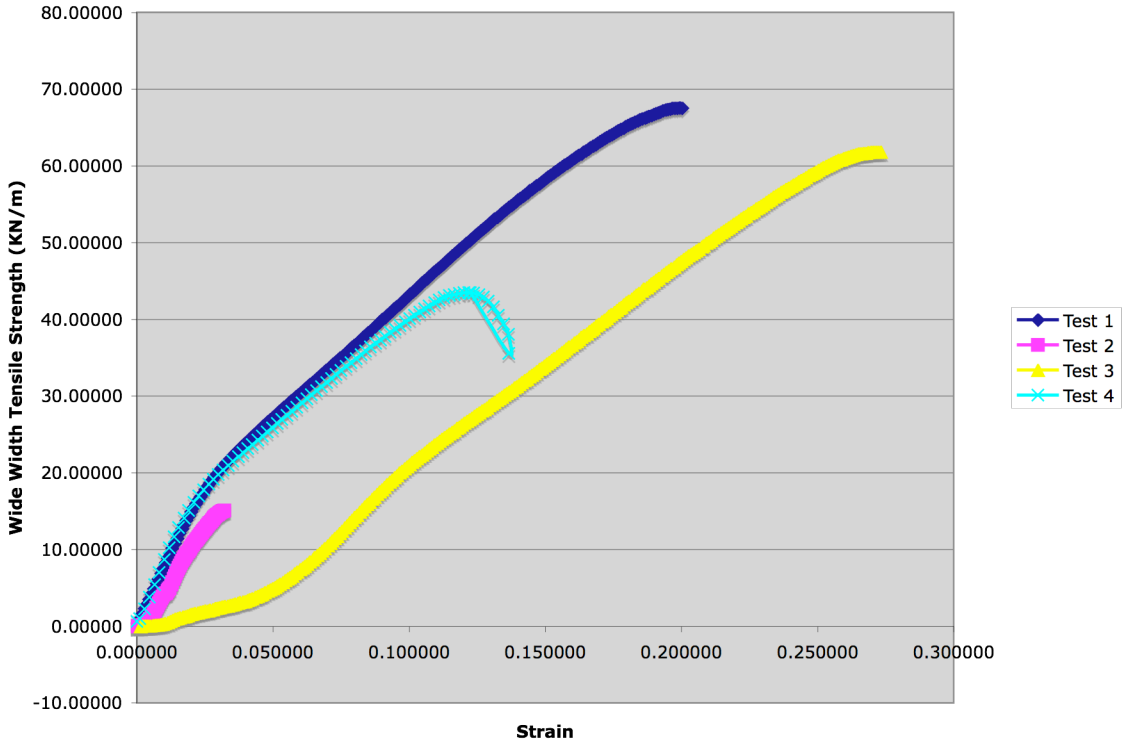

Figure 3.4 Fortrac 110 Machine Direction

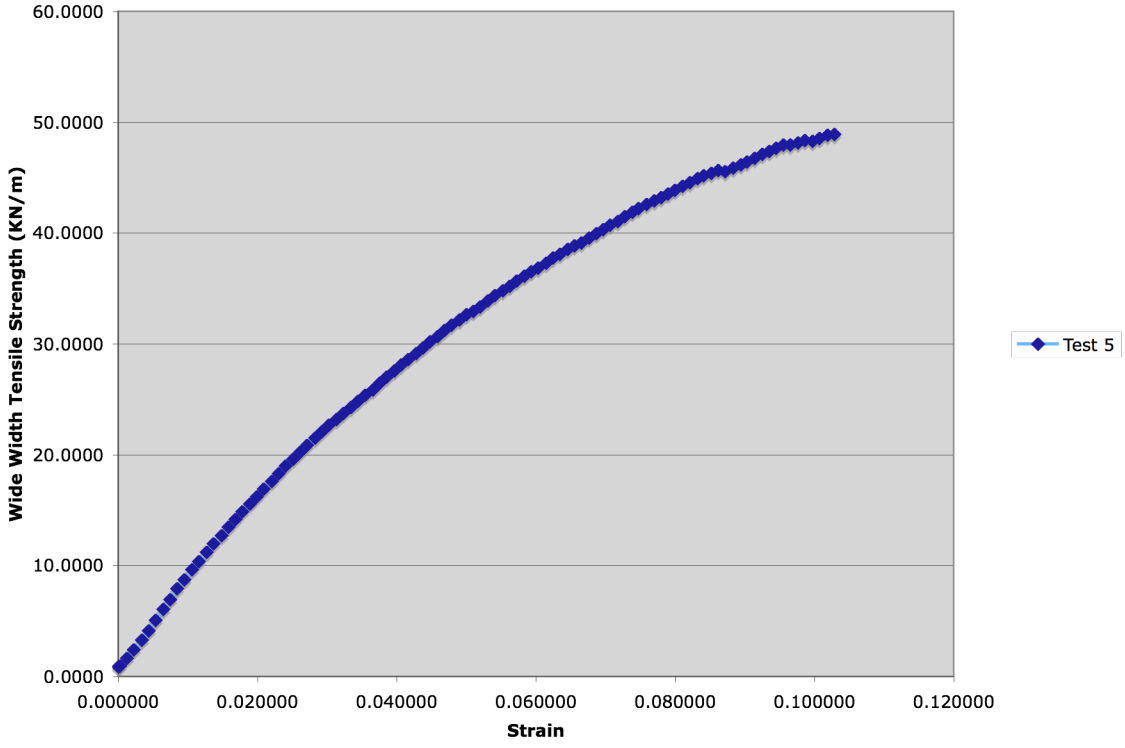

Figure 3.5 Fornit 30 Machine Direction

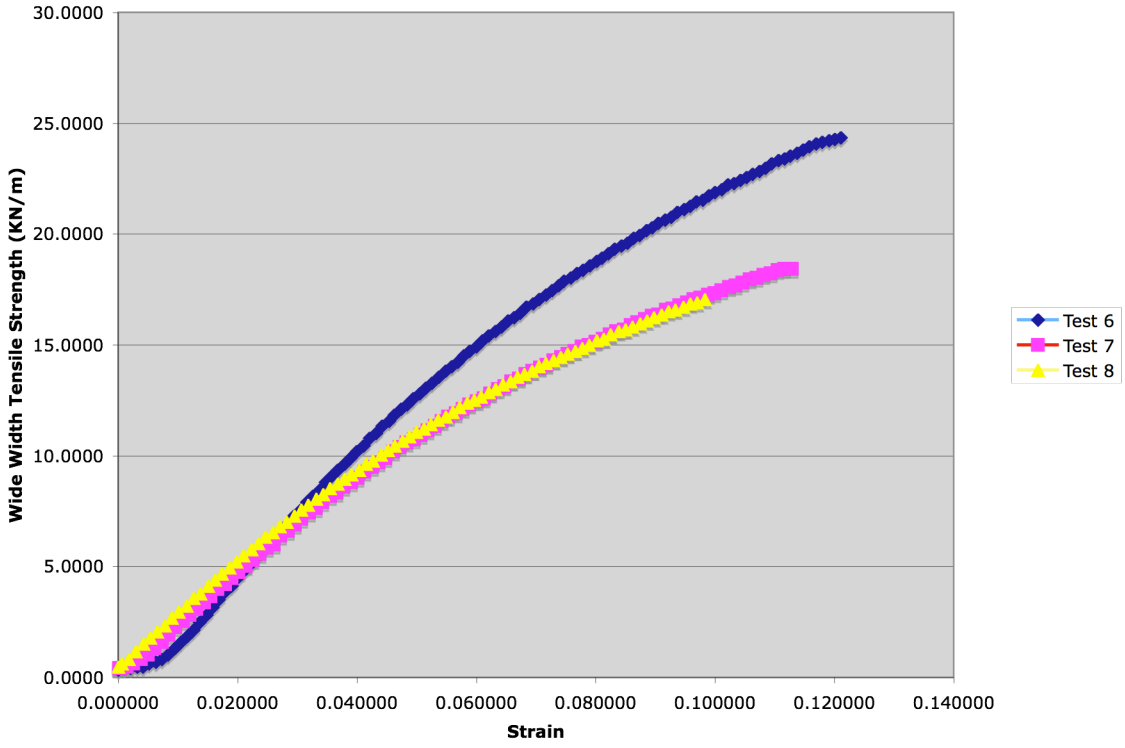

Figure 3.6 Fornit 20 Machine Direction

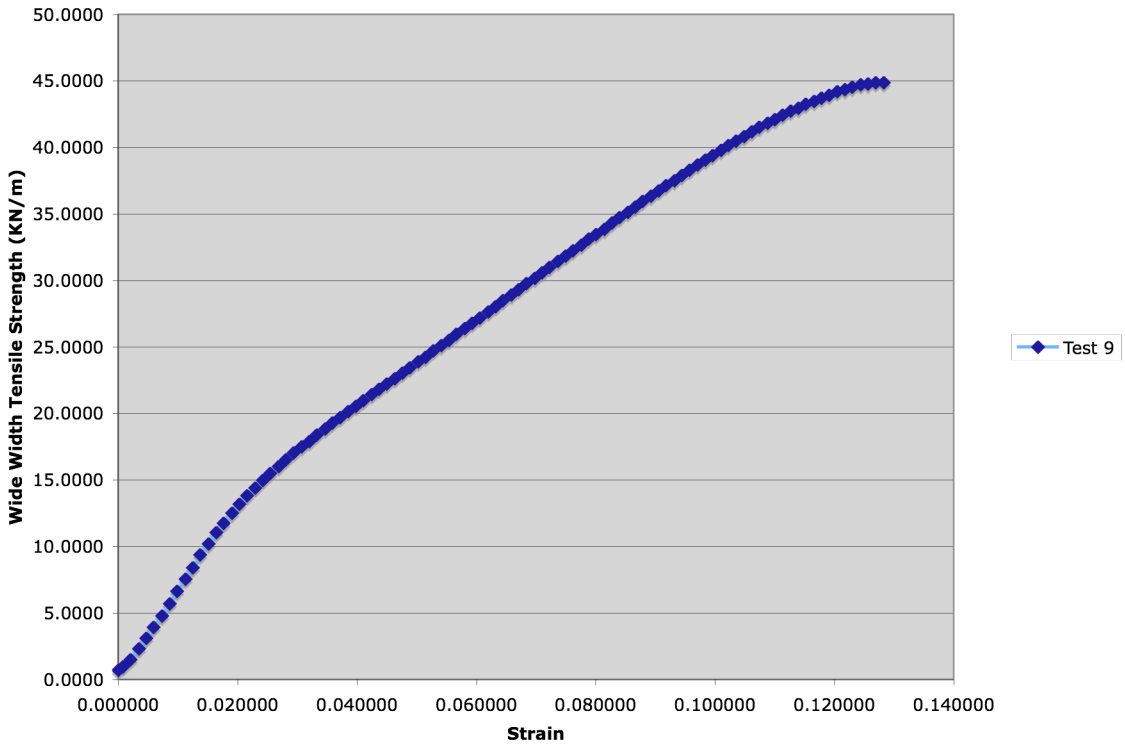

Figure 3.7 Fortrac 80 Machine Direction

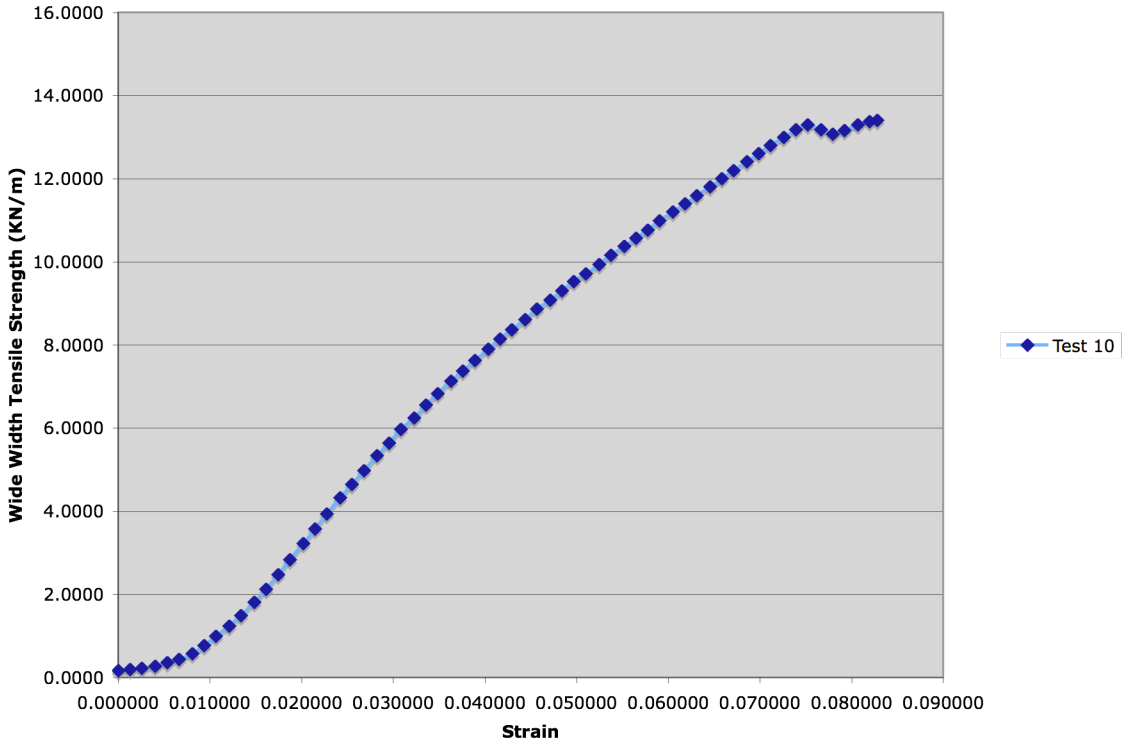

Figure 3.8 Fortrac 80 Cross Direction

3.3 Analysis

The ultimate strengths obtained from these tests were less than expected when compared to the ultimate strength reported in the Huesker product literature. When the manufacturer tested these geogrid products, they were cut into one-foot widths and pulled apart with a large circular roller. As noted above, most failures of the described tests occurred at junctions and at locations near the clamps. The manufacturer's testing apparatus had no clamps, so the load was evenly distributed throughout many junctions. This difference in testing could be the reason for the small discrepancies in ultimate strength.

The WWTS-strain relationship of each Fortrac geogrid was two consecutive linear lines, one that ended at a strain of approximately 0.02 and another that continued until the rupture strength. The Fornit geogrid material had a WWTS-strain relationship that resembled a gentle curve and could be easily approximated as a linear line. Both relationships could be conservatively approximated as a linear line.

The test results proved that it was reasonable to approximate the WWTS-strain relationship of polyester and polypropylene yarns, used in Huesker Geogrids, as a straight line. Therefore, for this thesis, the material that geogrids are manufactured from was assumed to be linearly elastic.

Most of the geogrid specimens that were tested ruptured at approximately 10 percent strain, and some geogrid specimens withstood even larger deformations. These large deformations and the flat position in which geogrid reinforcement is initially placed made it necessary to use a geometrically non-linear analysis to analyze a geogrid-reinforced structure.

Chapter 4
Minimization of Energy Method and Verification Studies

Several shape-finding methodologies have been developed for cable net structures. A minimization of energy approach was chosen after investigating several established methods. Mathematica Version 5.2 was used to implement this approach and to obtain values for the displacements, strains, and tensile forces. Mathematica is an elaborate program with many capabilities. In particular, Mathematica's nearly unlimited solving capability and its built-in minimization functions make it well suited to solve cable net structures. A Mathematica algorithm was developed to solve cable net structures, and it was used to solve several well-established example problems to verify the validity of the process.

4.1 Shape-Finding Methodologies

Three shape-finding methodologies, the stiffness matrix, dynamic relaxation, and minimization of energy method, were investigated to find the one best suited for this model. All three of these algorithms rely on the minimum energy principle to locate the final displacements of each node for a set of cables. The minimum energy principle states that a system is in stable equilibrium when the total potential energy of the system is minimum. When the total potential energy is minimum, the gradient of the total potential energy is zero.

Cable structures are unique in comparison to typical structures because cables frequently experience large deformations that lead to highly geometric non-linear behavior. The geometric non-linearity of cable structures requires analysis methods that use iterative procedures to approach a reasonable solution. The large deformation that causes geometric non-linear behavior is known as the membrane effect in geogrid reinforcement.

The stiffness matrix method uses the equilibrium equation to find the displacement vector through an iterative process:

$$D = K^{-1}R \tag{4.1}$$

K is the global stiffness matrix and R is the vector of residual forces on the structure. The K matrix must be updated after each iteration because it is dependent on the displacement of each node.

The dynamic relaxation method uses the dynamic equilibrium equation to find when the system is in static equilibrium:

$$P(t) = m\ddot{d} + c\dot{d} + kd \tag{4.2}$$

P is the external load acting on the system as a function of time, m is the imaginary mass of the system, c is the imaginary damping coefficient, k is the stiffness of the cables, d is the displacement, $\dot{d}$ is the first derivative of the displacement (i.e., the velocity) and $\ddot{d}$ is the second derivative of displacement (i.e., the acceleration). The residual force R is equal to

$$R = m\ddot{d} + c\dot{d} \tag{4.3}$$

The goal of this method is to obtain the static equilibrium equation. To do this, the method iterates on a time interval until the residual force approaches zero. The time interval and damping coefficient are dependent on the mass of the system. A detailed description on how to choose a reasonable value for mass can be found in Lewis (1984).

The first step of the minimum energy method is to use the following equation to obtain the total potential energy of the system:

$$\Pi = \Phi + \Omega \tag{4.4}$$

Π is the total potential energy, Φ is the strain energy, and Ω is the potential energy of the loading. As previously stated, if the gradient of the total potential energy is equal to zero, then the equilibrium equations are obtained. The point where the equilibrium equation approaches zero can be found by using several algorithms. Three that were investigated

were: the method of steepest descent, the conjugate gradient method, and the Newton-Raphson method. Each approach can be simply described with the following equation:

$$x_{k+1} = x_k + S_k v_k \tag{4.5}$$

x is the displacement, S is the step length, v is the descent vector, and the subscript k is the iteration number. The goal is to move along the descent vector using a step length until the system is in equilibrium or as close to equilibrium as desired. After each iteration the descent vector should be changed.

4.2 Minimization of Energy

Before the appropriate method was chosen, it was first necessary to revisit the problem. One square interior cell was examined that consisted of a continuous geogrid resting flatly on soil and supported by four corner piles. The geogrid was modeled as a linearly elastic cable grid and the soil was modeled as linearly elastic springs. The geogrid didn't rest directly on the piles, so the soil above the pile locations was modeled as stiffer springs. When this problem was first investigated, the soil element was neglected and a flat cable grid with no pretension was loaded vertically with fixed end supports on the perimeter. With these boundary conditions the stiffness method was virtually impossible because the initial cable grid shape allowed no stiffness in the direction of the load. However, when springs were added to this model, the problem became solvable.

The full-scale model of this system consisted of many nodes, typically over a thousand. At least three-fourths of the nodes were redundant, and if the stiffness of the cable grid was isotropic, then seven-eighths were redundant. Even when constraints were used on the redundant nodes, a large quantity of non-redundant nodes remain to be solved. A matrix of over one thousand rows and columns would have been a realistic result if the stiffness method was used, and the storage of such a matrix would have required too much memory. Also, solving these large matrices would have been time consuming.

It was decided that the minimization of energy method would be the best use of a personal computer's capabilities. The conjugate gradient method is the preferable algorithm for use with this method because it is compact and is ideal for handling cable grids with a large number of nodes (Buchholdt 1985, Stefanou 1992). At first the conjugate gradient method was programmed into Mathematica to understand how the method worked. Mathematica has a built-in minimization solver, and the results from the built-in solver were compared to the programmed conjugate gradient method. The solutions were exactly the same.

The formulation of the energy equation of this system was split into elements: the energy stored in the soil, the strain energy stored in the geogrid, and the potential energy of the loading. The soil was represented as compression springs applied at each node, and the strain energy stored in a vertical spring was mathematically represented as

$$\Phi_s = \frac{1}{2} T_s W^2 \tag{4.6}$$

where T_s = stiffness of the soil
W = vertical displacement

The strain energy stored in the length of cable between two nodes was mathematically represented as

$$\Phi_c = \frac{EA}{2H_0} e^2 \tag{4.7}$$

where E = modulus of elasticity of the cable
A = cross-sectional area of a cable
H_0 = initial length from node to node of the cable
e = elongation of the cable between nodes

The derivation of the elongation equation in terms of displacement began with the final length of a cable element:

$$H = H_0 + e \tag{4.8}$$

where H = final length of the cable

The Pythagorean theorem was used to write the final length between two nodes, i and j, as a function of the displacements:

$$H^2 = (X_i - X_j + U_i - U_j)^2 + (Y_i - Y_j + V_i - V_j)^2 + (Z_i - Z_j + W_i - W_j)^2 \tag{4.9}$$

where X = initial location with respect to the X-axis in a rectangular coordinate system
Y = initial location with respect to the Y-axis in a rectangular coordinate system
Z = initial location with respect to the Z-axis in a rectangular coordinate system
U = displacement with respect to the X-axis
V = displacement with respect to the Y-axis
W = displacement with respect to the Z-axis

Equations 4.8 and 4.9 were combined to form

$$(H_0 + e)^2 = (X_i - X_j + U_i - U_j)^2 + (Y_i - Y_j + V_i - V_j)^2 + (Z_i - Z_j + W_i - W_j)^2 \tag{4.10}$$

and then the equation was solved for the elongation:

$$e = \sqrt{(X_i - X_j + U_i - U_j)^2 + (Y_i - Y_j + V_i - V_j)^2 + (Z_i - Z_j + W_i - W_j)^2} - H_0 \tag{4.11}$$

The potential energy of a vertical point load on a junction was the negative of the work done by the load:

$$\Omega_p = -PW \tag{4.12}$$

where P = point load that acted on the junction

The total potential energy was found by summing the energy in each spring, cable, and point load:

$$\Pi = \sum \Phi_s + \sum \Phi_c + \sum \Omega_p \tag{4.13}$$

4.3 Examples

Lewis (1989) solved four examples using the dynamic relaxation method, including one example that was also solved by Buchholdt (1985) using the minimization of energy method. These four examples were solved to determine the validity of the minimization of energy method algorithm developed for Mathematica.

4.3.1 Example 1: Simple Flat Cable Net

This example consisted of one free node supported by four fully constrained nodes along the perimeter (Figure 4.1). The cross-sectional area of each of the four cables was 0.785 mm^2, the modulus of elasticity was 124,800 N/mm^2, the pretension in each of the four cables was 200 N, and the downward point load on the free node was 15 N.

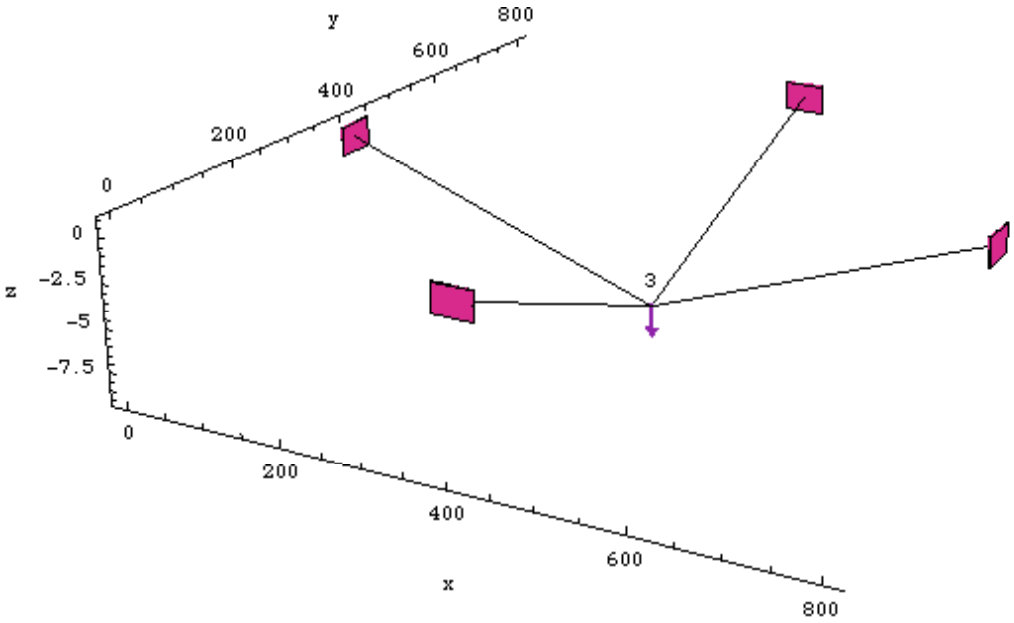

Figure 4.1 Example 1

4.3.2 Example 2: 2 x 1 Flat Cable Net

Both Lewis (1989) and Buchholdt (1985) had solved this example. The flat net consisted of two free nodes supported by fully constrained nodes along the perimeter (Figure 4.2). The cross-sectional area of each of the seven cables was 2 mm^2, the modulus of elasticity was 110,000 N/mm^2, the pretension in each cable was 500 N, and the downward point load on both free nodes was 200 N.

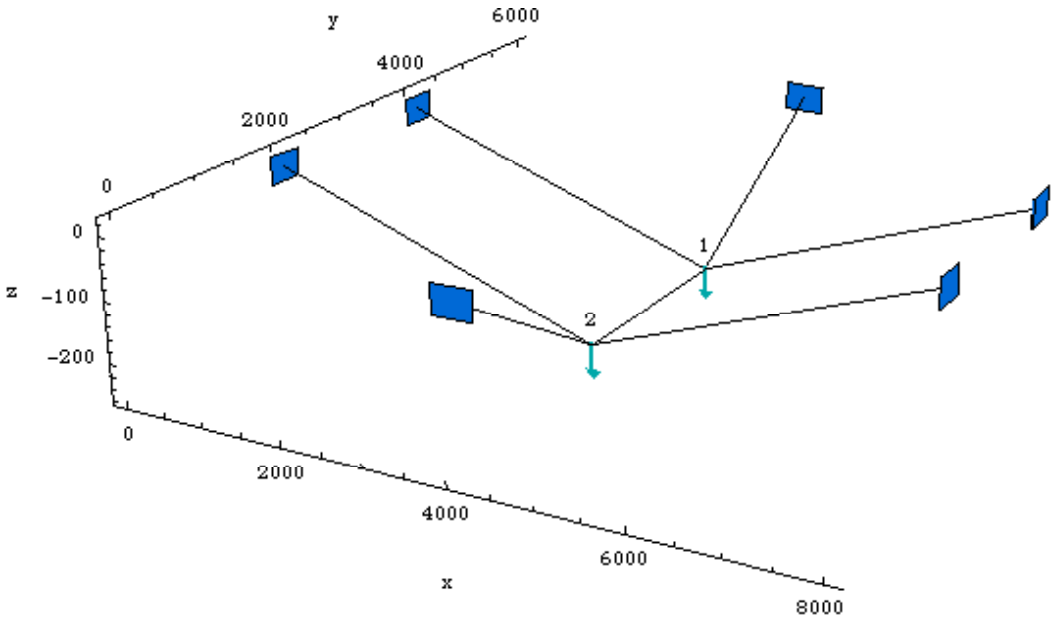

Figure 4.2 Example 2

4.3.3 Example 3: 2 x 2 Flat Cable Net

The flat net consisted of four free nodes supported by fully constrained nodes along the perimeter (Figure 4.3). The cross-sectional area of each of the 12 cables was 0.785 mm^2, the modulus of elasticity was 124,800 N/mm^2, and the pretension was 200 N. There was a 15 N downward point load on only three of the four nodes, nodes 4, 5, and 8. This example had a unique solution that verified that the method was accurate at predicting the consistency of the system.

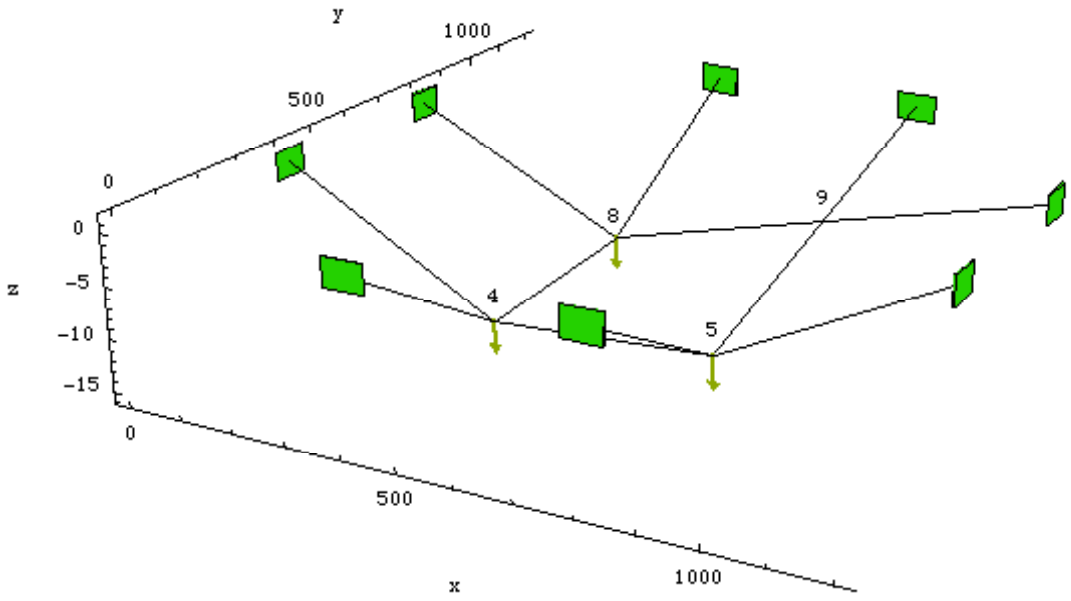

Figure 4.3 Example 3

4.3.4 Example 4: Hyper Net

The hyper net consisted of a 4 x 5 net that had one corner elevated 450 mm. There were 12 free nodes supported by fully constrained nodes along the perimeter (Figure 4.4). The cross-sectional area of each of the 31 cables was 0.785 mm^2, the modulus of elasticity was 124,800 N/mm^2, and the pretension was 200 N. There was a 15.7 N load on all of the free nodes with the exception of nodes 17, 21, and 22.

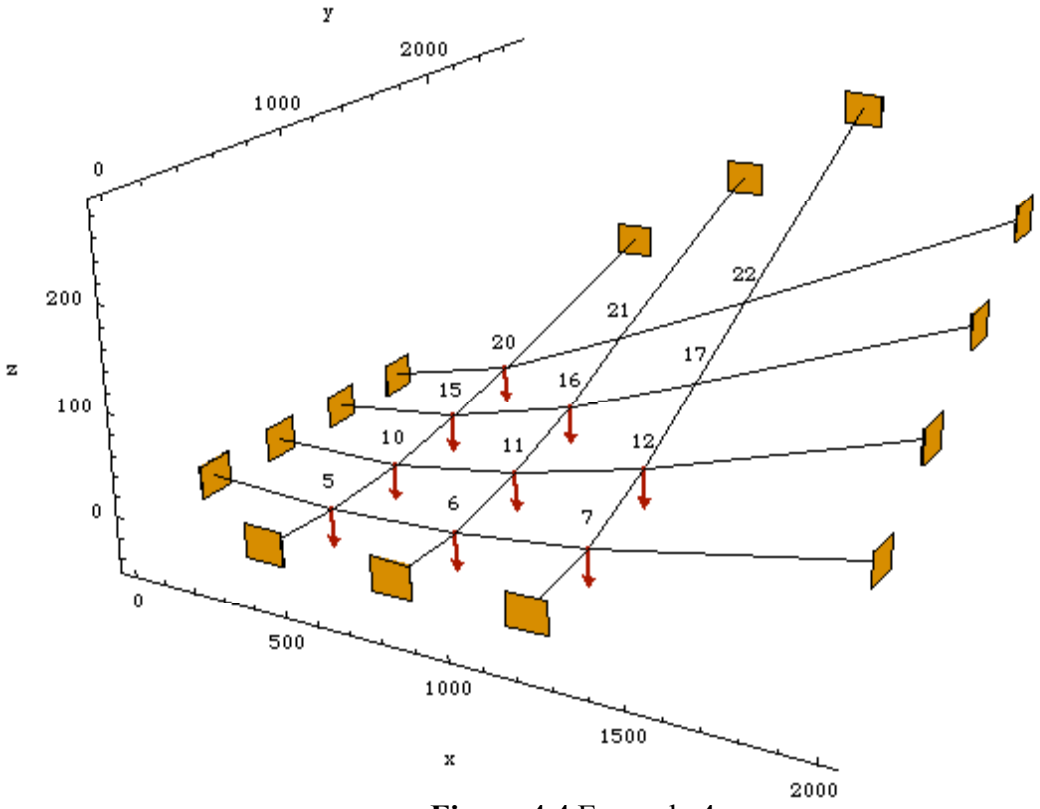

Figure 4.4 Example **4**

4.4 Comparison of Results

Table 4.1 presents a comparison of the displacements (in millimeters) from Lewis (1989), an experiment she performed using the hyper net layout, and the Mathematica algorithm. Buchholdt (1985) used the same method that the Mathematica algorithm implemented, and the results for Example 2 were exactly the same for both. A good agreement exists between Lewis's results using the dynamic relaxation method, the experimental results, and the results from the minimization of energy method. These examples verified the validity of the Mathematica algorithm for solving a diverse selection of cable net problems. The Mathematica algorithm and the solutions of the example problems can be found in Appendix B.

Table 4.1 Comparison of Result

Table 4.1 Comparison of Results											
Example	**Node**	**Lewis (1989)**			**Exp.**	**Min. Eng. (MathCAD)**			**% Diff.**		
Axis (mm)		**U**	**V**	**W**	**W**	**U**	**V**	**W**	**U**	**V**	**W**
1.1x1 Net	3	0	0	6.97		0	0	6.98	0	0	0
2 2x1 Net	1	0	-3.3	199.7		0	-3.3	199.7	0	0	0
	2	0	3.3	199.7		0	3.3	199.7	0	0	0
3. 2x2 Net	4	0.07	-0.07	12.2		0.07	-0.07	12.2	0	0	0
	5	0.04	0.08	11.2		0.04	0.08	11.2	0	0	0
	9	-0.04	-0.04	5.61		-0.04	-0.04	5.59	0	0	0
4. Hyper Net	5	-0.95	-1.01	19.39	19.5	-0.29	-0.35	19.6	69	65	1
	11	-2.56	-2.77	33.8	33.6	-1.75	-1.94	34.2	32	30	1
	15	-3.1	-1.04	25.42	25.2	-0.67	-1	25.9	78	4	2
	17	-2.34	-2.48	21.27	21	-1.73	-1.92	21.5	26	23	1
	21	-2.9	-1.63	19.77	19.8	-1	-2.38	20.1	66	46	2
	22	-2.06	-1.7	14.27	14.2	-1.13	-1.75	14.5	45	-3	2

Chapter 5
Implementation of the Minimization of Energy Method using Mathematica

The examples described in Chapter 4 verified the minimization of energy approach as an accurate method of determining the deflected shape of cable structures. After the validity of the method was established, the next challenge was implementing a series of routines to form the energy equation of the entire system. Several matrices were assembled to store the node position coordinates, cable end nodes, nodal displacements (degrees of freedom), cable stiffness, spring stiffness, and point loads. Once the system's energy equation was assembled, the program used a built-in minimization subroutine to find the displacements needed for the system to be at rest. Several modifications were made to the Mathematica program so that a model with anisotropic geogrid properties, multiple overlaying geogrids, a rotated geogrid orientation, and a geogrid with rotational stiffness could also be analyzed.

5.1 Outline of the Mathematica Program

The model was reduced to a one-eighth triangular slice of the unit cell to eliminate redundant degrees of freedom and reduce computer computation time. This section describes the purpose of the matrices that were needed to store the constants used in the energy equation, and the programming that was necessary to build the matrices.

5.1.1 Input

The dimensional parameter values served as the input to the Mathematica program. At the top of the Mathematica worksheet, the variables B, L, Q_P, Q_S, K_P, K_S, EA, and n were initialized and the values were input into the Kernel, Mathematica's memory and solver. The dimensional parameter H was found using the following relationship:

$$H = \frac{L}{2 \cdot n} \tag{5.1}$$

where n = the number of cables on the edge of the one-eighth reduced unit cell (i.e., half the number of cables on the edge of the full unit cell).

Non-dimensional parameter values were used in the analysis of the model, so Equations 2.1-2.6 had to be utilized to convert the dimensional values to non-dimensional values.

5.1.2 Location and Connectivity Matrices

The initial position matrix, designated as xyz in the program, was used to describe the location of each node in non-dimensional rectangular coordinates. The Mathematica Table function was used with two increasing variables, i and j, to create this matrix:

$$xyz_{i,j} = (i \cdot h \quad j \cdot h \quad 0) \tag{5.2}$$

where i = j, j+1,…, n

j = 0, 1,…, n

The Table function increased the variable i, beginning at the current value of j until i reached n. When i reached n, j increased by 1 and i began again at the new value of j. After the xyz matrix was built, it was flattened so that there were three columns and as many rows as there were nodes. The first column listed the position with respect to the x-axis, the second column listed the position with respect to the y-axis, and the third column listed the position with respect to the z-axis. All the nodes and cables in this model were numbered beginning from left to right and then from bottom to top. In this manner, the coordinates for each node of the reduced one-eighth grid were stored in a matrix. Figure 5.1 shows an image of the one-eighth grid in two dimensions. Since the z coordinate was initially zero for all the nodes, the z-axis is not shown in the figure.

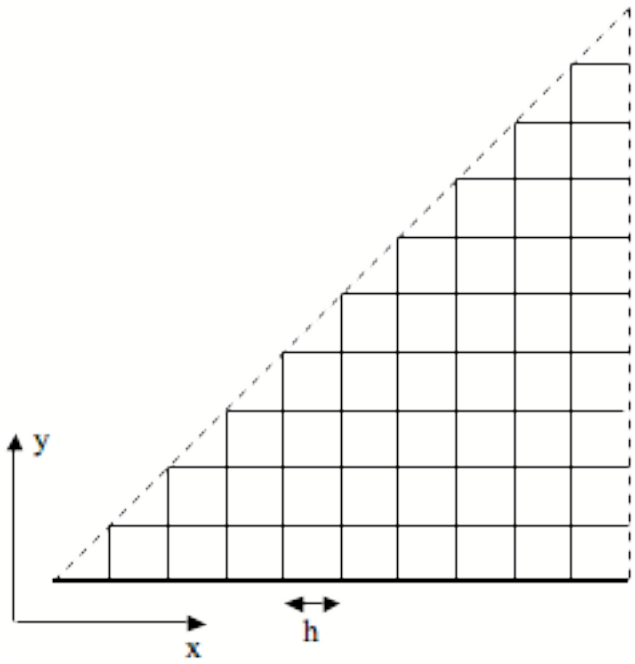

Figure 5.1 One-Eighth Reduced Grid

The connectivity matrix, designated as Con in the program, was used to store the node numbers of the nodes that were connected by cables. Two Table functions were utilized and the Join function was used to combine the two. The first table was used to store the node numbers of all the cables parallel to the x-axis. The table used two dependently increasing variables a1 and a2, and two independently increasing variables i and j:

$$Con_{i,j} = (a1 \quad a1+1) \tag{5.3}$$

where i = j, j+1,…, n

j = 1, 2,…,n

a1 = 1 and increased by 1 IF j ≠ a2 and increased by 2 IF j = a2

a2 = 2 and increased by 1 IF j = a2

The variable a2 monitored j, because when j increased the node number assigned to the left end node of the cable, al, increased again to skip over the node at the very end of the row of cables.

The second table was built to store the node numbers that made up the cables parallel to the y-axis. The table used three dependently increasing variables b1, b2, and b3, and two independently increasing variables i and j:

$$Con_{i,j} = (b1 \quad b3) \tag{5.4}$$

where I = 1, 2,…, n

j = i, i+1,…, n

b1 = 1 and increased by 1 IF j ≠ a2 and increased by 2 IF j = a2

b2 = 2 and increased by 1 IF j = a2

b3 = n+2 and increased by 1

The variables b1 and b2 served similar functions to a1 and a2 in the first table. In Figure 5.1 the y-axis was vertical, b1 was the node number assigned to the bottom end, and b3 was the node number assigned to the top end of the cable. The i and j variables had opposite values in this table because i was used to represent points along the x-axis and j was used to represent points along the y-axis. Both tables were joined and flattened to create one matrix made up of two columns and as many rows as there were cables in the reduced grid. In this manner all the node numbers that made up each cable were stored in a matrix.

5.1.3 Displacement Matrix

The displacement matrix, designated as uvw in the Mathematica program, was a matrix that corresponded directly to the xyz matrix and was made up of zeros or variables. A zero meant that the corresponding node number in the xyz matrix was fixed along the axis to which the column corresponded. If the axis was free at that node, then a variable x[a] was entered into that position in the matrix. The variable, a, was initialized and set equal to 1. The final value of a was equal to the total number of degrees of freedom in the system. A table function was used to create this matrix:

$$uvw_{i,1} = 0 \text{ IF } xyz_{i,1} = ½ \text{ or } xyz_{i,1} = 0 \quad (5.5)$$

$$\text{Otherwise } uvw_{i,1} = x[a{+}{+}]$$

$$uvw_{i,2} = 0 \text{ IF } xyz_{i,2} = ½ \text{ or } xyz_{i,2} = 0$$

$$uvw_{i,2} = x[a] \text{ IF } xyz_{i,1} = xyz_{i,2}$$

$$\text{Otherwise } uvw_{i,2} = x[a{+}{+}]$$

$$uvw_{i,3} = x[a{+}{+}]$$

where $i = 0, 1, \ldots, tn$

tn = total number of nodes in the one-eighth reduced grid

a = the degree of freedom number

a++ = a is increased by one before being used

x = degree of freedom vector where each column represented a degree of freedom

Several IF statements constrained nodes on the edge of the cell so that these nodes stayed fixed to the edge. If the node was located on the diagonal edge, then the second column had the same degree of freedom as the first column because the nodes along the diagonal edge had equal displacements along the x and y axes. These constraints and slaving constraints were necessary for the reduced grid to behave like a full grid. The third column, the displacement along the z-axis, was always free for this model.

5.1.4 Cable, Point Load, and Spring Stiffness Matrices

The cable vector stored the factor that the energy of each cable in the model would be multiplied by. The point load and spring vectors stored the point load and spring stiffness that bore on each node in the model. A constant point load and spring stiffness couldn't be used for all the nodes in the model because there were specific nodes that did not support the same tributary area as the others. The cable vector was also necessary because there were certain cables that supported tributary areas that were not included in the model, so a constant cable energy could not be used for all the cables.

The cable vector consisted of two joined tables. The first table stored the factors of the cables parallel to the x-axis, and the second stored the factors of the cables parallel to the y-axis. The cables on the edge of the reduced grid were shared with another cell or another slice of the reduced grid that combined to form the full grid. The factor of an edge cable that was shared by another reduced grid was halved because half of the cable supported the load in the reduced grid that was modeled, while the other half supported load that was not included. Figure 5.2 illustrates the cables that had reduced cable factors. The numbers inside the circles are the factors that were used for each cable location. The first table was built using an IF statement:

$$CoC_i = \frac{1}{2} \text{ IF } i \le n \qquad (5.6)$$

$$\text{Otherwise } CoC_i = 1$$

where i = 1, 2,…, tcx

tcx = total number of cables parallel to the x-axis

CoC = cable vector

and the second table was also built using an IF statement:

$$CoC_i = \frac{1}{2} \text{ IF } i = c1 \qquad (5.7)$$

$$\text{Otherwise } CoC_i = 1$$

where i = 1, 2,…, tcy

tcy = total number of cables parallel to the y-axis

c1 = n and IF i = c1 then c1 = c1 + c2

c2 = n and IF c1 = n then c2 = c2 – 1

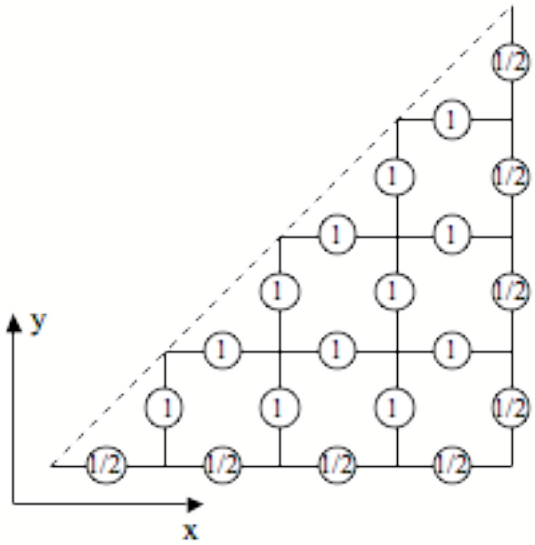

Figure 5.2 Cable Stiffness Factors

Both the point load and spring stiffness values for each node were modified by the same factor, since the modified values were both dependent on the node's tributary area. A factor vector was built to store the fraction of the tributary area that each node supported relative to a node located in the central region. The point loads and spring stiffnesses were multiplied by the factor vector to obtain the point load and spring stiffness acting on each node. Figure 5.3 illustrates the factor values in relation to the node's position on the reduced cable net. The factor vector used a table function with an increasing variable i and three IF statements:

$$Fac_i = \frac{1}{8} \text{ IF } xyz_{i,1} = xyz_{i,2} = 0 \text{ or } xyz_{i,1} = xyz_{i,2} = \frac{1}{2} \tag{5.8}$$

$$Fac_i = \frac{1}{4} \text{ IF } xyz_{i,1} = \frac{1}{2} \text{ and } xyz_{i,2} = 0$$

$$\text{Otherwise } Fac_i = \frac{1}{2} \text{ IF } xyz_{i,1} = \frac{1}{2}, xyz_{i,2} = 0 \text{ or } xyz_{i,1} = xyz_{i,2}$$

$$\text{Otherwise } Fac_i = 1$$

where i = 1, 2,…, tn

Fac = factor vector

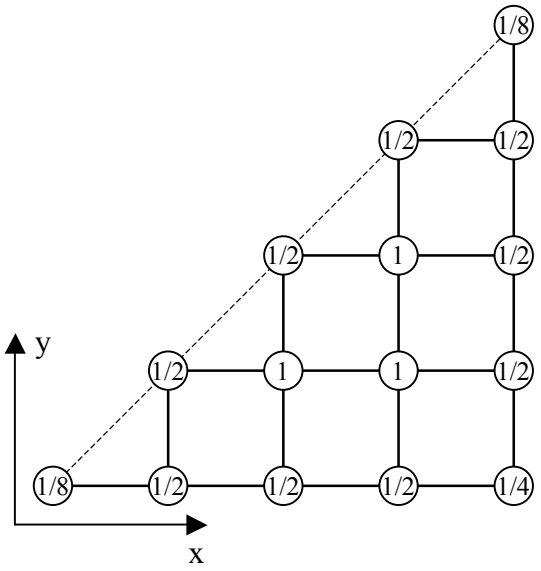

Figure 5.3 Point Load and Spring Stiffness Factors

The point load vector and spring stiffness vector differentiated between pile and soil nodes because it included the following IF statement:

$$p_i = p_p \text{ IF } xyz_{i,1} \le b \tag{5.9}$$

Otherwise $p_i = p_s$

and

$$t_i = t_p \text{ IF } xyz_{i,1} \le b \tag{5.10}$$

Otherwise $t_i = t_s$

where i = 1, 2,…, tn

p = point load as defined in equation 2.5

t = spring stiffness as defined in equation 2.6

b = non-dimensional width of the pile

The cell and pile are both square, so the one-eighth reduced grid was a right triangle and any node with an initial x-coordinate that was less than or equal to b was on top of the pile. Any node with an x-coordinate greater than b was on top of the soft soil. The point

load and spring stiffness vectors were built using these routines.

5.1.5 Elongation Matrix

The elongation vector stored the elongation equation for each cable in the reduced model. The vector used Equation 4.11 to find the elongation, and the order used in building the connectivity matrix was maintained in the elongation matrix. This organization allowed many terms in Equation 4.11 to drop out because before the load was applied, all the cables were parallel to an axis:

$$e_i = \sqrt{(h + uvw_{c1,1} - uvw_{c2,1})^2 + (uvw_{c1,2} - uvw_{c2,2})^2 + (uvw_{c1,3} - uvw_{c2,3})^2} - h \text{ IF } i \le tcx \tag{5.11}$$

Otherwise

$$e_i = \sqrt{(uvw_{c1,1} - uvw_{c2,1})^2 + (h + uvw_{c1,2} - uvw_{c2,2})^2 + (uvw_{c1,3} - uvw_{c2,3})^2} - h \tag{5.12}$$

where c1 = $Con_{i,1}$

c2 = $Con_{i,2}$

i = 1, 2,…,tn

All the rows in the vector up to the value of tcx were parallel to the x-axis, so the initial distance between cable ends, h, was added to the x-axis displacement of the cable ends and all the other initial distances dropped out. The rows of the vector greater than tcx represented cables parallel to the y-axis, so h was added to the y-axis displacement of the cable ends.

5.1.6 Summation of Energy

The energies of the springs, cables, and point loads were found using the non-dimensional forms of Equations 4.6, 4.7, and 4.12, respectively. The strain energy of the soil springs in non-dimensional form is

$$\Phi_s = \frac{1}{2}\sum_{i=1}^{tn} t_i \cdot uvw_{i,3}^2 \tag{5.13}$$

The strain energy of the cables in non-dimensional form is

$$\Phi_c = \sum_{j=1}^{tc} \frac{CoC_j \cdot e_j^2}{2h} \tag{5.14}$$

The potential energy of the point loads is

$$\Omega_p = \sum_{k=1}^{tn} p_k \cdot uvw_{k,3} \tag{5.15}$$

The total energy was found by summing the energies of all the components of the model using Equation 4.13 and the matrices discussed above.

5.1.7 Solving the Energy Equation

The Mathematica function NMinimize was used to find the degree of freedom vector x for which the energy was minimized.

5.1.8 Solution

The values of the degree of freedom vector were put into the displacement matrix and the elongation vector. Both vectors were saved to a text file so that the data was backed up. A final position matrix was built by adding the initial position matrix xyz and the displacement matrix uvw.

5.2 Modifications to the Mathematica Program

The routines described above that were used to build the matrices necessary to analyze the original model were modified to run the analysis of several case studies that had slightly different models.

5.2.1 45-degree Orientation

A model was analyzed that used the same loading and spring stiffness as the original system, but instead of a cable net with cables parallel to the x and y axes, the cable net was rotated 45-degrees. Four matrices had to be modified to perform the analysis: the initial position matrix, the connectivity matrix, the cable stiffness vector, and the elongation vector. The initial position matrix had to be modified so that every other row of nodes was offset by h divided by the square root of 2. Figure 5.4 illustrates the reduced grid in relation to the coordinate system. The routine that found the modified initial position matrix was

$$xyz_{i,j} = (i \cdot \frac{2h}{\sqrt{2}} \quad j \cdot \frac{h}{\sqrt{2}} \quad 0) \tag{5.16}$$

where i = j/2, j/2+1,…, n

j = 0, 1,…, 2n

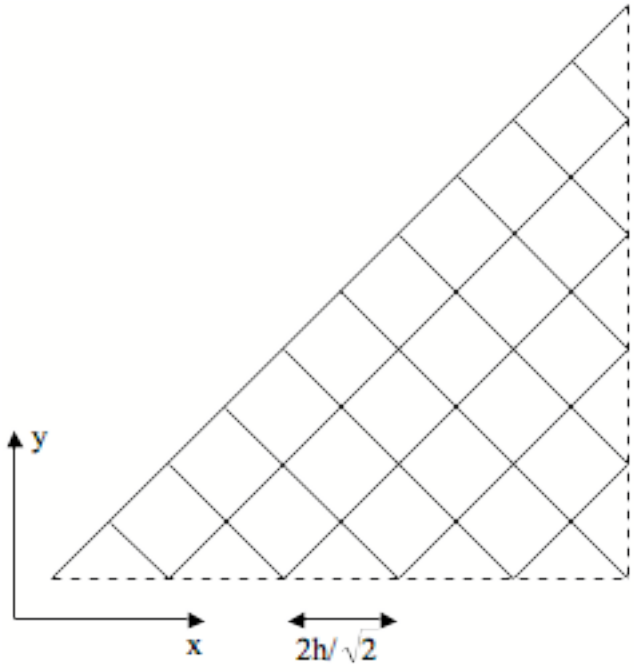

Figure 5.4 The Reduced Grid Rotated 45-Degrees

The initial position matrix was flattened so that it had three columns. The connectivity matrix had a different numbering scheme than the original model. The cable numbering alternated between cables positioned in one direction and cables perpendicular to the last. The routine used for formulating this matrix was

$$Con_{i,j} = (a1 \quad b1) \text{ IF i was odd} \tag{5.17}$$

$$\text{Otherwise } Con_{i,j} = (b1 \quad a1+1)$$

where i = 1, 2,…, 2n-j

j = 0, 1,…, 2n

a1 = 0 and increased by 2 IF i = 1 and j was odd or increased by 1 IF i was odd

b1 = n+1 and increased by 1 IF i was odd

The table used two dependently increasing variables a1 and b1, and two independently increasing variables i and j. The connectivity matrix was flattened so that it had 2 columns.

The cable matrix was modified so that the cable factor was one-half for the cables on the diagonal edge. Figure 5.5 illustrates the cable factors that were used for different locations on the reduced grid.

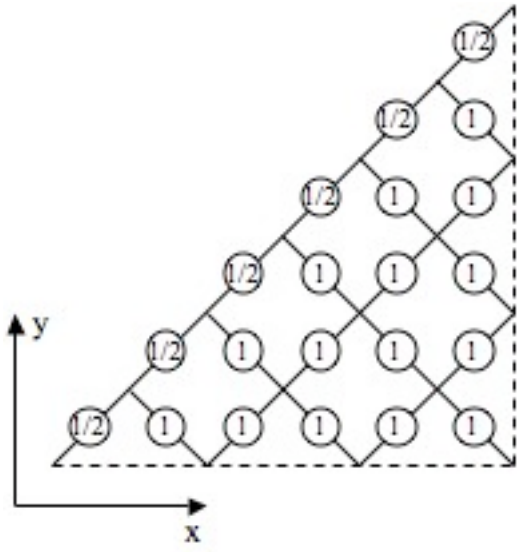

Figure 5.5 Cable Stiffness Factors

The elongation matrix had to be modified because the cables were no longer parallel to the axes. The initial distance between the ends of a cable along the x and y axes was h divided by the square root of two. Also, it was sometimes necessary to switch the sign of the initial displacement between the ends of the cable because the order of the connectivity matrix alternated between cables oriented in the positive y direction and the negative y direction. For the case $c1 < c2$

$$e_i = \sqrt{(\frac{h}{\sqrt{2}} + uvw_{c1,1} - uvw_{c2,1})^2 + (\frac{h}{\sqrt{2}} + uvw_{c1,2} - uvw_{c2,2})^2 + (uvw_{c1,3} - uvw_{c2,3})^2} - h \quad (5.18)$$

Otherwise

$$e_i = \sqrt{(\frac{h}{\sqrt{2}} + uvw_{c1,1} - uvw_{c2,1})^2 + (-\frac{h}{\sqrt{2}} + uvw_{c1,2} - uvw_{c2,2})^2 + (uvw_{c1,3} - uvw_{c2,3})^2} - h \quad (5.19)$$

where c1 = $Con_{i,1}$

c2 = $Con_{i,2}$

i = 1, 2,…, tn

With the exception of these four matrices, the analysis was performed in the same manner as the other models.

5.2.2 Anisotropic Orientation

A model was analyzed that had an anisotropic cable grid. Instead of h being uniform along the x and y axes, two variables were used, h_x and h_y, so that the spacing between cables could be independently varied. h_x was the length of the cable parallel to the x-axis and h_y was the length of the cable parallel to the y-axis. The one-eighth reduced grid couldn't be used because the cables parallel to the x-axis and y-axis were no longer interchangeable. Figure 5.4 shows the square, one-fourth slice of the full grid used as the reduced grid in this case study. All of the original matrices had to be modified slightly to accommodate this reduced grid.

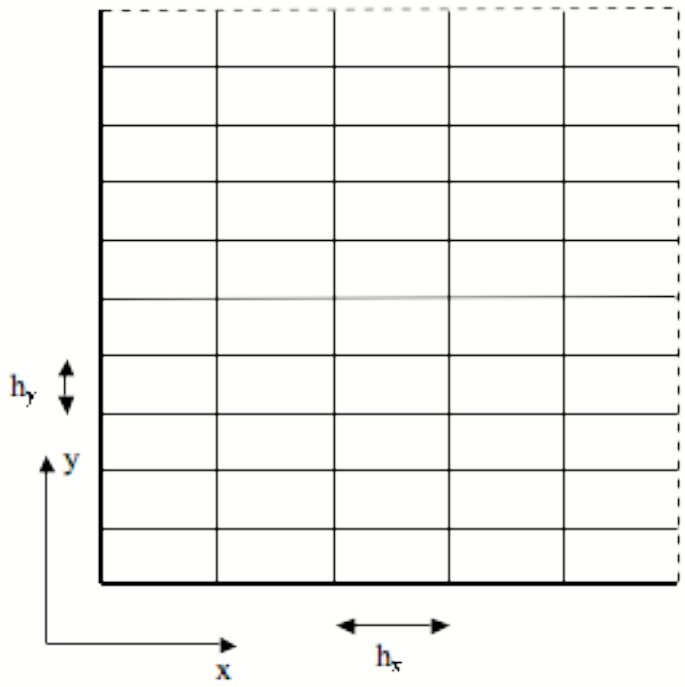

Figure 5.6 One-Fourth Grid

The initial position matrix stored the coordinates of each node in the reduced cable grid. The xyz matrix was built using a table function and the following routine:

$$xyz_{i,j} = (i \cdot h_x \quad j \cdot h_y \quad 0) \tag{5.20}$$

where i = 0, 1,…, n_x

j = 0, 1,…, n_y

n_x = half the number of cables along the x-axis

n_y = half the number of cables along the y-axis

The matrix was flattened so that there were only three columns and as many rows as there were nodes.

The connectivity matrix was built by joining two tables. The first table used two dependently increasing variables a1 and a2, and two independently increasing variables i and j:

$$Con_{i,j} = (a1 \quad a1 + 1) \tag{5.21}$$

where $i = 1, 2, \ldots, n_x$

$j = 1, 2, \ldots, n_y+1$

a1 = 1 and increased by 1 IF $j \neq a2$ and increased by 2 IF $j = a2$

a2 = 2 and increased by 1 IF $j = a2$

The first table was built to store the node numbers of the cable ends that made up the cables parallel to the x-axis, and the second stored the node numbers of the cable ends that made up the cables parallel to the y-axis. The table used two dependently increasing variables b1 and b2, and two independently increasing variables i and j:

$$Con_{i,j} = (b1 \quad b2) \tag{5.22}$$

where $i = 1, 2, \ldots, n_y$

$j = 1, 2, \ldots, n_x+1$

b1 = 1 and increased by 1 when i changed values

$b2 = n_x+2$ and increased by 1 when i changed values

The cable vector was modified so that the factors of all the cables on the edge of the one-fourth reduced grid were halved. Figure 5.7 illustrates the factors of the cables in relation to position.

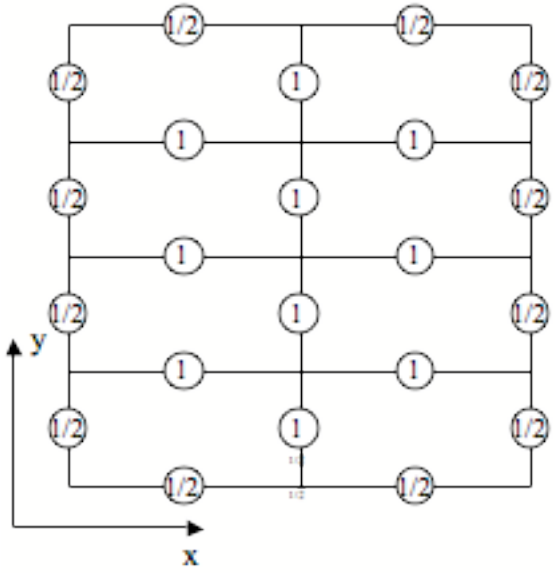

Figure 5.7 Cable Stiffness Factors

The spring stiffness and point load factors were also modified so that the edge nodes were halved and the corner nodes had a factor equal to a quarter. The factors were directly derived from the tributary areas that acted on each edge node. An illustration of these factors is shown in Figure 5.8. Equations 2.5 and 2.6 were used to obtain the p and t variables used in the spring stiffness and point load matrices, but for this model they had to be slightly modified to account for the change in tributary area:

$$p = q \bullet h_x \bullet h_y \tag{5.23}$$

$$t = k \bullet h_x \bullet h_y \tag{5.24}$$

The routines used to create the two factor matrices for the anisotropic model were very similar to the original factor matrices and were not explained in this thesis. The exact routine is found in the anisotropic algorithm in Appendix E.

The routine that built the displacement matrix uvw was similar to the original routine but the IF statement that accounted for the case $xyz_{i,1} = xyz_{i,2}$ was excluded. On an anisotropic cable net, the y-displacement was not necessarily equal to the x-displacement.

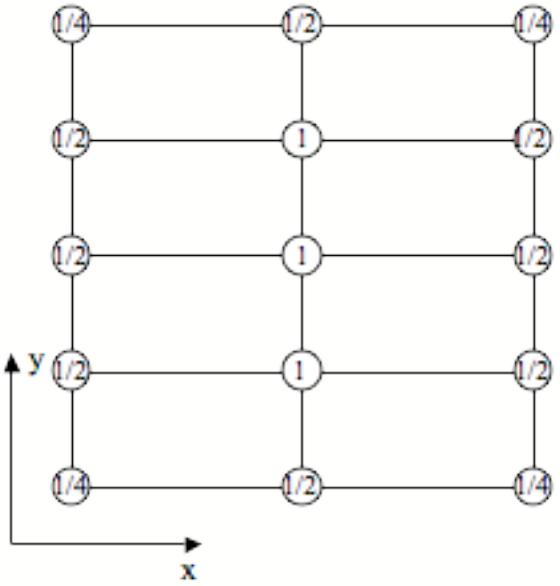

Figure 5.8 Spring Stiffness and Point Load Factors

The same elongation routine was used for this case study but the elongation equation for cables along the x-axis was replaced by

$$e_i = \sqrt{(h_x + uvw_{c1,1} - uvw_{c2,1})^2 + (uvw_{c1,2} - uvw_{c2,2})^2 + (uvw_{c1,3} - uvw_{c2,3})^2} - h_x \qquad (5.25)$$

and the elongation of a cable parallel to the y-axis was

$$e_i = \sqrt{(uvw_{c1,1} - uvw_{c2,1})^2 + (h_y + uvw_{c1,2} - uvw_{c2,2})^2 + (uvw_{c1,3} - uvw_{c2,3})^2} - h_y \qquad (5.26)$$

It was also necessary to replace h with h_x or h_y when the energy absorbed by the elongation of the cables was calculated.

5.2.3 Rotational Springs

A Mathematica worksheet was created to model the system with a cable grid that had a joint stiffness. This cable net was modeled with rotational springs at each node that resisted rotation in the x-z and y-z planes. Figure 5.9 illustrates a rotational spring acting on a joint and two adjacent cables in deformation. No rotational spring was added to the grid in the x-y plane because the x and y displacements were considerably smaller in magnitude than the z-displacements.

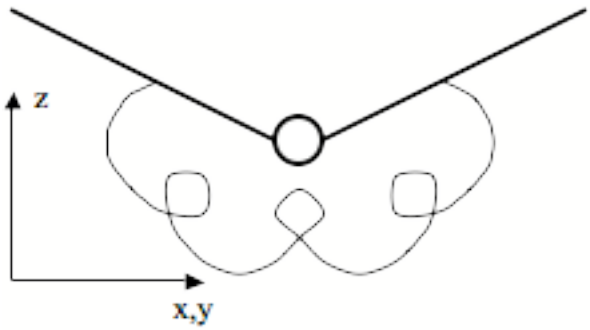

Figure 5.9 Rotational Spring in Plane View

It was not necessary to modify any of the original matrices, but a routine was added to build a matrix that computed the angles at each joint. This model was based on the assumption that the loads could be applied to the nodes, and the cables remained straight between nodes. The angles of each node were dependent on the adjacent nodes. The angle routine found the angle of the node in the x-z plane and stored it in the first column, then found the angle of the node in the y-z plane and stored it in the second column. This routine found the angles using six IF statements. Figure 5.10 illustrates each of the 6 locations that required special consideration because of the shape of the reduced cable net. The numbering of each node on the figure was added to aid the reader in understanding the purpose of each IF statement.

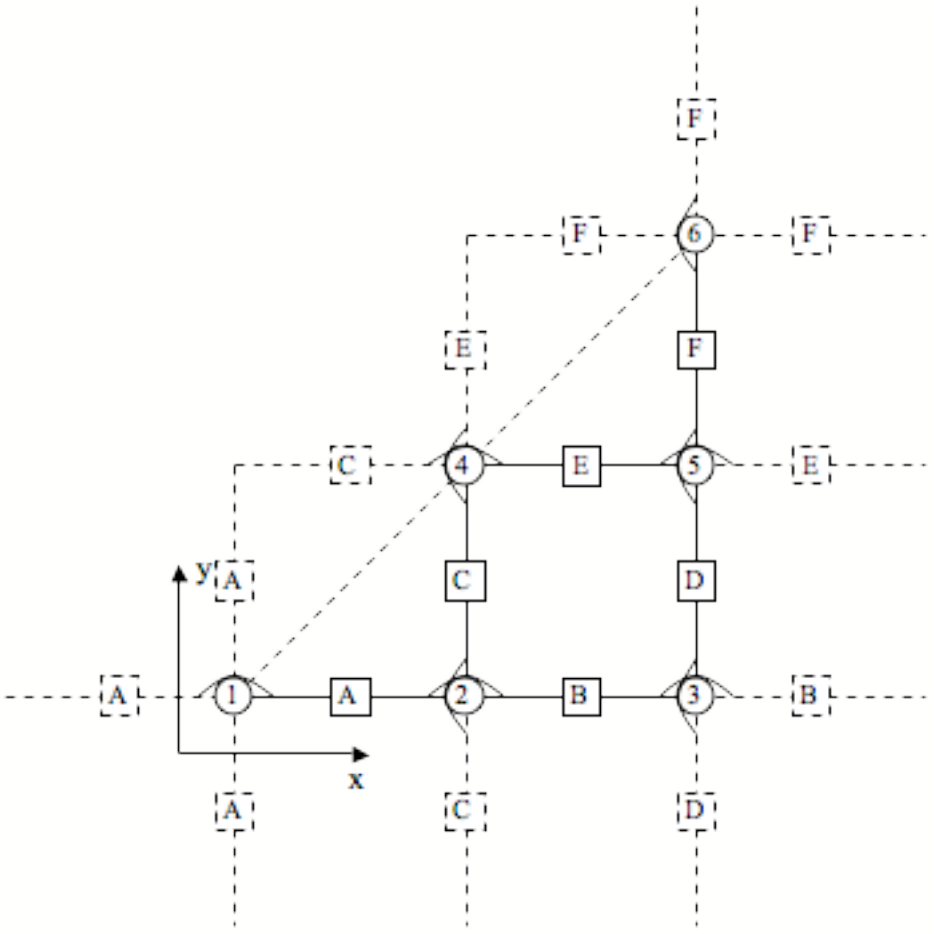

Figure 5.10 Node Angles

The first IF statement found the angle of the bottom left corner node (1 in Figure 5.10):

$$\phi_i = (2 \bullet ArcTan[\frac{uvw_{i,3} + uvw_{i+1,3}}{h}] \quad 0) \text{ IF } i = 0 \tag{5. 27}$$

where ArcTan = Arc Tangent function

i = 0, 1,…, tn

The bottom left node was not connected to a cable parallel to the y-axis, so the angle was 0. The equation used in this routine assumed that the angle was small. The denominator of the fraction was simply h, although it would be more accurate to add u. Figure 5.10 shows that the angle in the x-z plane was dependent on two nodes that had the same vertical displacement because of symmetry; this was why the angle was doubled.

The second IF statement found the angle of the top right corner node (6 in Figure 5.10):

$$\phi_i = (0 \quad 2 \bullet ArcTan[\frac{uvw_{i,3} + uvw_{b32,3}}{h}]) \text{ IF i = tn} \tag{5. 28}$$

where b30 = n+1 and decreased by 1 IF i = b31+1

b31 = n+1 and increased by b30 IF i = b31+1

b32 = 0 and added 1 IF i > n+1 unless i = b31+1 or tn, then added 2

b33 = n and increased 1 unless i = b31+1 or tn

The top right node was not connected to a cable parallel to the x-axis, so the angle was 0. The angle was doubled because symmetry caused the vertical displacement at the adjacent nodes to be equal. The dependent variable b30 was used to monitor the number of nodes in a row of the triangular reduced cable grid; b30 was added to b31 to find the node number of the right edge of the each row, b32 was used to keep track of the adjacent bottom node, and b33 was used to keep track of the adjacent top node. The third IF statement was used to find the angles of the bottom right corner node (3 in Figure 5.10):

$$\phi_i = (2 \bullet ArcTan[\frac{uvw_{i,3} + uvw_{i-1,3}}{h} \quad 2 \bullet ArcTan[\frac{uvw_{i,3} + uvw_{b33,3}}{h}]) \text{ IF i = n+1} \tag{5.29}$$

Both angles were doubled because of symmetry.

The fourth statement IF statement found the angles of the nodes on the bottom edge of the reduced grid (2 in Figure 5.10):

$$\phi_i = (ArcTan[\frac{uvw_{i,3} + uvw_{i-1,3}}{h} + ArcTan[\frac{uvw_{i,3} + uvw_{i+1,3}}{h} \quad 2 \bullet ArcTan[\frac{uvw_{i,3} + uvw_{b33,3}}{h}]) \text{ IF } i \le n \quad (5.30)$$

The angle of the node in the y-z plane was doubled because of symmetry. The fifth IF statement was used to find the angles of the nodes on the diagonal edge (4 in Figure 5.10):

$$\phi_i = (ArcTan[\frac{uvw_{i,3} + uvw_{i+1,3}}{h} + ArcTan[\frac{uvw_{i,3} + uvw_{b32,3}}{h} \quad ArcTan[\frac{uvw_{i,3} + uvw_{i+1,3}}{h} + ArcTan[\frac{uvw_{i,3} + uvw_{b32,3}}{h})$$

$$\text{IF i = b31 +1} \quad (5.31)$$

The angles of the nodes on the diagonal edge in the x-z and y-z plane were the same because the symmetry of the cell caused the node to be adjacent to the same two cables in each direction. The final IF statement was used to find the angles of the nodes on the right edge (5 in Figure 5.10):

$$\phi_i = (2 \bullet ArcTan[\frac{uvw_{i,3} + uvw_{i-1,3}}{h} \quad ArcTan[\frac{uvw_{i,3} + uvw_{32,3}}{h} + ArcTan[\frac{uvw_{i,3} + uvw_{b33,3}}{h}) \quad \text{IF i = b31}$$

$$(5.32)$$

The angle of the node in the x-z plane was doubled because of symmetry. The angles of a central node in the reduced grid (not shown in Figure 5.10) was

$$\phi_i = (ArcTan[\frac{uvw_{i,3} + uvw_{i-1,3}}{h} + ArcTan[\frac{uvw_{i,3} + uvw_{i+1,3}}{h} \quad ArcTan[\frac{uvw_{i,3} + uvw_{b32,3}}{h} + ArcTan[\frac{uvw_{i,3} + uvw_{b33,3}}{h})$$

In addition to an angle matrix, a rotational angle constant vector was built. This vector was built by multiplying the rotational spring constant C_R by the factor vector created for

the spring stiffness and point load vectors. The values assigned to the lower left and upper right corner nodes (the corner and center nodes in full grid terms) were doubled. By doubling these factors, the constants become one-fourth. Figure 5.10 illustrates that the bottom left and upper right corner nodes on the reduced grid only had one rotational spring, while the other nodes had two making it necessary to double the nodes' factors. This factor vector could be used for the rotational spring constant because the same rules of symmetry applied to the rotational springs as to the stiffness and point loads.

The energy of the rotational springs was:

$$\Phi_R = \frac{1}{2}\sum_{i=1}^{tn}\sum_{j=1}^{2} Cr_i \cdot \phi_{i,j}{}^2 \tag{5.33}$$

The sum of the energy in each rotational spring was added to the total energy of the model. The rotational spring constant of a node was equal for rotational springs in both directions, so the constant was multiplied by the square of both the angle in the x-z plane and the angle in the y-z plane, for each node.

5.2.4 Two Layers of Geogrid

A model was analyzed with two overlapping cable nets connected at their respective nodes by linear springs. The linear springs represented the soil between the geogrids. A few routines were modified, and a new variable d was added to the original algorithm to create a new algorithm that could analyze two cable nets. The variable d was the initial distance between the cable nets, hence d was also the initial length of the linear spring. For this algorithm, two displacement matrices were built, uvw1 and uvw2, using the same routine that the original algorithm used. The matrix uvw1 stored the displacements of the top geogrid and uvw2 stored the displacements of the bottom geogrid. The degree of freedom vector, x[a], for uvw2 began where uvw1 left off; in other words, x[a] began at

$$x[1]$$

for the uvw1 matrix and x[a] began at

$$x[1 + nDOF1]$$

where nDOF1 = number of degrees of freedom in the top geogrid

Multiplying the factor vector, used for the spring constant and point load vector, by the spring constant t_g, an additional spring constant vector tg was built for the springs between the geogrids. This factor vector could be used for the springs between the cable nets because the springs between the geogrids had the same tributary areas as the springs below the geogrid. Two elongation matrices were made, e1 and e2. They were based on the original elongation routine, but e1 was a function of the displacement matrix uvw1 and e2 was a function of the displacement matrix uvw2.

The total energy was the same as in the original system, but the cable energy for both cable nets and the energy in the additional springs also needed to be included in the summation. The energy in the linear springs between the geogrids was

$$\Phi_g = \sum_{i=1}^{tn} \frac{1}{2} tg_i (uvw1_{1,3} - uvw2_{i,3})^2 \tag{5.31}$$

The potential energy caused by the point loads was a function of the vertical displacement of the top cable net, and the strain energy caused by the soil springs was a function of the lower cable net's vertical displacement. In this manner, the linear springs transferred the load from the higher cable net to the lower cable net.

Restrictions were imposed on the vertical displacements of the top cable net so that it could not pass through the bottom cable net at any point. This restriction was justified because it was impossible for a geogrid layer to penetrate through another geogrid layer. A potential energy minimum was found that met these restrictions.

5.2.5 No In-Plane Displacement

Only one modification was needed to perform the analysis of a cable grid with no in-plane displacement. The first two columns of the displacement matrix were set equal to zero because no nodal displacements were allowed along the x or y axes.

Chapter 6
Parametric Study

The Mathematica algorithm described in Chapter 5 implemented the minimization of energy method to solve the geogrid-reinforced pile-supported foundation model proposed in Chapter 2. The values of the non-dimensional parameters q_P, q_S, k_P, k_S, b, and h used with the model were determined with the help of Professor Filz and his previous research (Stewart 2005). Each parameter was varied separately to determine the effect of the parameter on the deformed shape and the strain in the ribs of the geogrid. In addition, several other case studies were conducted using modified Mathematica algorithms. A study was performed using a grid rotated 45 degrees, another study used an anisotropic geogrid and varied the h parameter along only one axis, a case study was performed to test the effect of joint stiffness on the results, another study determined if the in-plane displacements could be ignored, and finally an analysis was performed that had two layers of geogrid. Plots of the non-dimensional vertical displacements w and strain ε that result from the analysis of each case study are shown. Also, the maximum dimensional vertical displacements W, tension T, ε, differential settlement D_S, and load distribution ratio LDR are shown in tables for each case study. The maximum dimensional results for the standard case were compared to a model that represented the geosynthetic reinforcement as a membrane, and to the strain calculated from a design formula.

6.1 Results in Dimensionless Form

As mentioned above, a standard case of dimensional parameter values was chosen with the help of Professor Filz: B = 0.6 m, L = 3 m, H = 0.03 m, EA = 21,900 N, Q_P = -14.6 N/cm^2, Q_S = -3.06 N/cm^2, K_P = 29.2 N/cm^3, K_S = 0.16 N/cm^3.

Inputting dimensional parameters into the Mathematica program initialized each case. These dimensional parameter values were then converted to non-dimensional parameters using Equations 2.1-2.6. The dimensional parameter values used for the standard case

were as follows in non-dimensional form: b = 0.2, h = 0.01, q_P = -60, q_S = -12.6, k_P = 36,000, k_S = 197.

Case studies were performed by varying all six non-dimensional parameters separately. Each parameter was varied twice and, with the exception of b, all the standard parameter values were varied by multiplying and dividing by two. The parameter b was analyzed for the values 0.15 and 0.25.

Additional case studies were conducted to test other geogrid and model properties. Using a modified Mathematica algorithm that had a different node and cable orientation, a geogrid rotated 45 degrees was tested. An anisotropic geogrid was analyzed by varying the value of h along the x-axis while keeping the value of h along the y-axis constant at 0.01. The parameter h_x was analyzed for the same non-standard values as h, 0.02 and 0.005. A geogrid joint stiffness case study was conducted by choosing the rotational stiffness constant c to be: 100, 1,000, and 10,000. A preliminary analysis was performed to determine if in-plane displacements could be neglected. To do this, geogrid junctions were only allowed one degree of freedom, W, or in non-dimensional form w. Another case study used a model that had two layers of geogrid reinforcement with linear springs that connected both geogrids at the overlaying junctions. The spring constant of the springs between the geogrids was equal to t_g and the non-dimensional, initial, vertical distance between the two geogrids was equal to 0.05.

Several graphical representations have been used to convey the results in a logical manner. The Mathematica Graphics 3D functions were used to construct a three-dimensional rendering of the displaced shape of the geogrid. First, the reduced final position matrix of the one-eighth portion of the grid had to be transformed into a full final position matrix. Two methods were developed to do this task. The first method used a table function and eight IF statements; each IF statement accounted for one of the eight portions of the grid that a node on the reduced grid could represent. The table function used two increasing variables, i and j; i was the position of the node on the x-axis and j was the position of the node on the y-axis. Each node met the requirements of one of the

eight IF statements and each IF statement associated the full node with a reduced node. Depending on the position of the node and the portion of the grid in which the node was located, the x and/or y displacement had to be modified as shown in Figure 6.1. In this manner, a matrix of the final position coordinates was created in which the rows were ordered by node number and the columns were ordered by the node's final position coordinates with respect to the x, y, and z axes. The nodes were numbered from left to right and then down to up with respect to Figure 6.1. This method was difficult to develop because it was hard to find the equation that would equate the i and j position variables with the correct reduced node number. These equations were complex and a discussion of how they work is not included in this thesis. The program and the associated comments can be found in Appendix C.

While programming the 45-degree orientation, an easier method was developed. This method utilized two Mathematica functions: the table and sort functions. The table function used two increasing variables, i and j, to represent the position of each node on the reduced grid, and IF statements were used to differentiate between edge nodes, diagonal nodes, and interior nodes. The interior node IF statement transformed the reduced grid displacements to full grid displacements eight different times according to Figure 6.1.

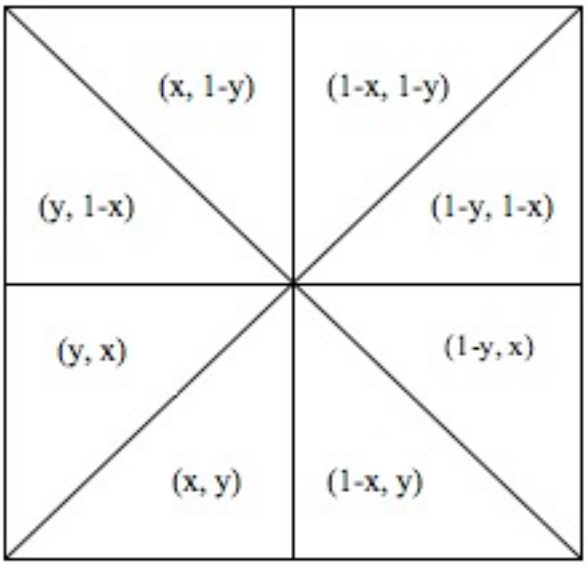

Figure 6.1 The Reduced to Full Grid Displacement Transformations

Edge and diagonal node displacements were transformed four times, because each portion of the full grid shared a common diagonal and edge. The matrix had to have five columns; the first column was the initial position along the x-axis, the second column was the initial position along the y-axis, and the final three columns were the final positions of the node. The first two columns were included so that the sort function would sort the matrix according to the initial position. The sort function organizes the rows by the value of the first column, and if two columns are equal, then the second column will decide the order of the row. The nodes of the grid had to be organized consistently for each run because the connectivity matrix was dependent on the row position of the final position matrix. In other words, the organization of the final position matrix had to be independent of the final displacements.

The full connectivity matrix, the matrix of the full grid node numbers that were connected by cables, also had to be generated. The matrix was made up of two joined tables; the first table was comprised of the cables that were parallel to the x-axis and the second table was comprised of the cables that were parallel to the y-axis. The first table used one independently increasing variable i and two increasing dependent variables, a21 and a22:

$$CON_{i,1} = a21 \tag{6.1}$$

$$CON_{i,2} = a21 + 1 \tag{6.2}$$

where CON = full connectivity matrix

i = 1, 2,…, cx

a21 = 1 and increased by 2 IF i = a22(2n)+1 otherwise increased by 1

a22 = 1 and increased by 1 IF i = a22(2n)+1

cx = number of cables parallel to the x-axis

The second table used one independently increasing variable i,

$$CON_{i,1} = i \tag{6.3}$$

$$CON_{i,2} = i + 2n + 1 \tag{6.4}$$

where i = 1, 2,…cy

cy = number of cables parallel to the y-axis

The graphics primitive “line” was used to show a 3D representation of the final displaced shape of the geogrid. The line primitive draws a line from two points, each point is indicated by two sets of rectangular coordinates. The coordinates were found by looking up the two row numbers that were indicated in each column of the connectivity matrix, in the full final position matrix. Using these tables, the entire grid was drawn in Mathematica.

The following results are shown in non-dimensional form so to convert the results on the non-dimensional plots to dimensional form the following equations can be used:

$$X = x \cdot L \tag{6.5}$$

$$Y = y \cdot L \tag{6.6}$$

$$Z = z \cdot L \tag{6.7}$$

$$W = w \cdot L \tag{6.8}$$

where x = non-dimensional location with respect to the x-axis

y = non-dimensional location with respect to the y-axis

z = non-dimensional location with respect to the z-axis

w = non-dimensional displacement with respect to the z-axis

Contour plots were drawn using a built-in Mathematica plot function. The Contour function only accepts z values, and there was a small amount of inaccuracy in each contour plot because the x and y displacements were not accounted for. Considering that the largest in-plane displacement was small and was less than h, the distance from node to node, this plot was still a useful visualization tool.

Several two-dimensional plots were generated so that effective comparisons could be made between each run of a case study. Two-dimensional cross-sections of the geogrid at

y = 0, y = 0.25, and y = 0.5 were generated by utilizing the first and third columns of the full final position matrix. The position matrix was organized by rows parallel to the x-axis, so it was a simple matter of knowing the first node number of the y = 0, y = 0.25, and y = 0.5 rows and then including all of the consecutive nodes up to the last node of the row. The first node was found using the equation

$$y \bullet 2n(2n+1) \tag{6.9}$$

and the last node was

$$y \bullet 2n(2n+2)+1 \tag{6.10}$$

where y was equal to 0, 0.25, and 0.5 for the two-dimensional plots.

The strain values were more difficult to obtain because the elongation values were organized in the same manner as the reduced connectivity matrix. The elongation values had to be transformed from the reduced grid to the full grid, and a table function and eight IF statements were used to implement this transformation. Each IF statement considered one of the eight portions of the grid that a cable on the reduced grid could represent. The table function used two increasing variables i and j. The variable i was the position of the cable on the x-axis and j was the position of the cable on the y-axis. Each node met the requirements of one of the eight IF statements and each IF statement associated the full cable with the reduced cable. The pattern that was used was too complicated to discuss in this thesis, but the program and comments can be found in Appendix C. To determine the strain, the following equation was used:

$$\varepsilon = \frac{e}{h} \tag{6.11}$$

where ε = strain

e = elongation

Once the full elongation matrix was assembled, the first row of the strain vector used in the plots of y = 0, y = 0.25, and y = 0.5 was found using the equation

$$y \bullet 4n^2 + 1 \tag{6.12}$$

and the last node was found using the equation

$$y \bullet 4n^2 + 2n \tag{6.13}$$

The tension of a single strand in the geogrid was directly proportional to the strain because the modulus of elasticity and cross-sectional area were the same for each strand in the geogrid:

$$T = EA\varepsilon \tag{6.14}$$

So a plot of the strain had the same shape as a plot of the tension. In fact, if a non-dimensional tension parameter was used, it would have been equal to the strain.

6.1.1 Standard Case

A three-dimensional rendering of the displaced shape of a geogrid subjected to the standard case parameter values is shown in Figure 6.1. Figure 6.2 shows a three-dimensional plot and Figure 6.3 shows of a contour plot of the vertical displacement w, with respect to x and y. Figure 6.4 shows a plot of the vertical displacement versus x. A solid line designates values parallel to the x-axis and along the edge, where y = 0. A dashed line designates values parallel to the x-axis and along a line one-quarter of the length of the grid, where y = 0.25. A dash-dot line designates values parallel to the x-axis and along the centerline, where y = 0.5. These line relationships hold true for all the plots in this chapter. For the standard case and all of the other case studies except the anisotropic grid, the displacement profile along the x-axis is equal to the displacement profile along the y-axis. In other words, the x and y axes are interchangeable in these plots due to symmetry. Figure 6.5 shows a three-dimensional plot and Figure 6.6 shows a contour plot of the strain ε_x with respect to x and y. Figure 6.7 shows a plot of the strain versus x. Again, a solid line designates strain values along the edge, a dashed line designates values along a line midway between the edge and the centerline, and a dash-dot line designates values along the centerline. These line relationships also hold true for

all two-dimensional strain plots throughout this chapter, and the x and y axes are also interchangeable for strain values as they were for displacement values.

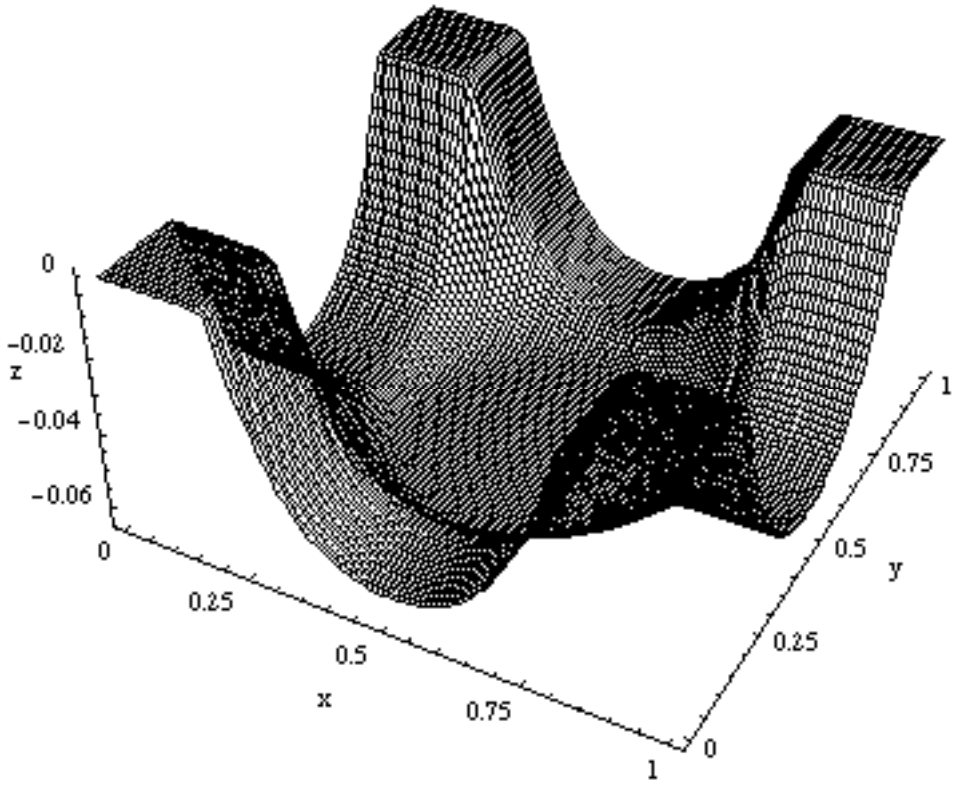

Figure 6.2 Three-Dimensional Plot for the Standard Case

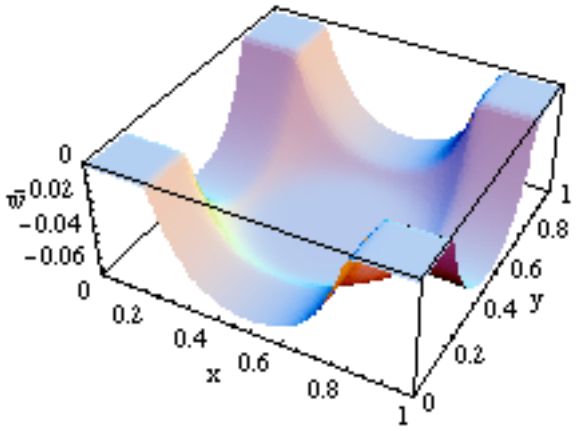

Figure 6.3 Three-Dimensional Plot of w for the Standard Case

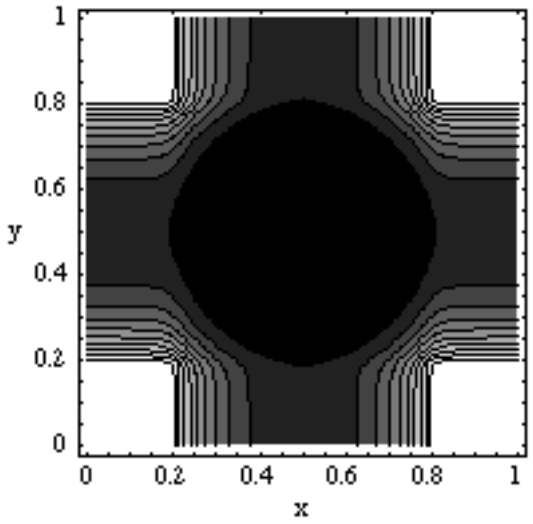

Figure 6.4 Contour Plot of w for the Standard Case

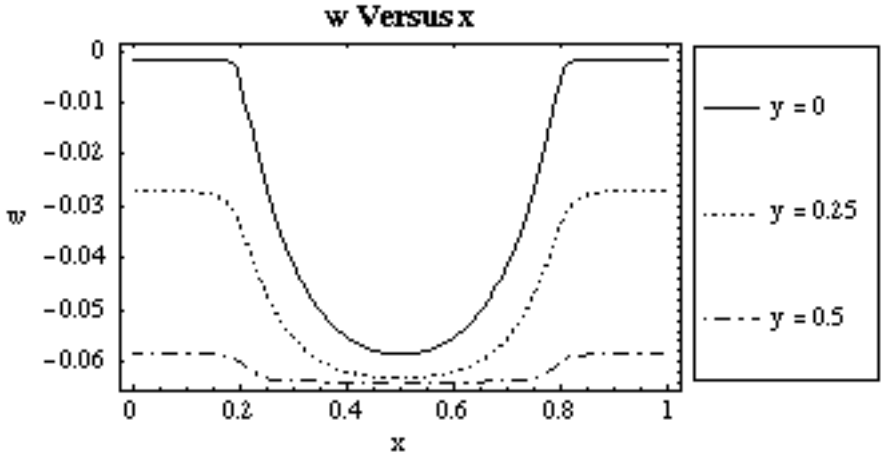

Figure 6.5 Plot of w vs. x along the Edge, Quarter, and Center for the Standard Case

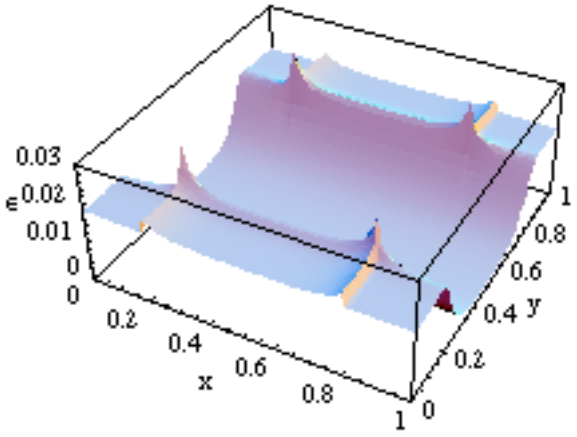

Figure 6.6 Three-Dimensional Plot of ε_x for the Standard Case

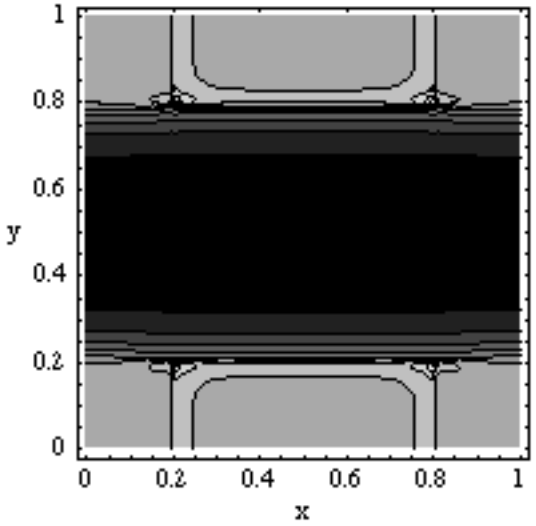

Figure 6.7 Contour Plot of ε_x for the Standard Case

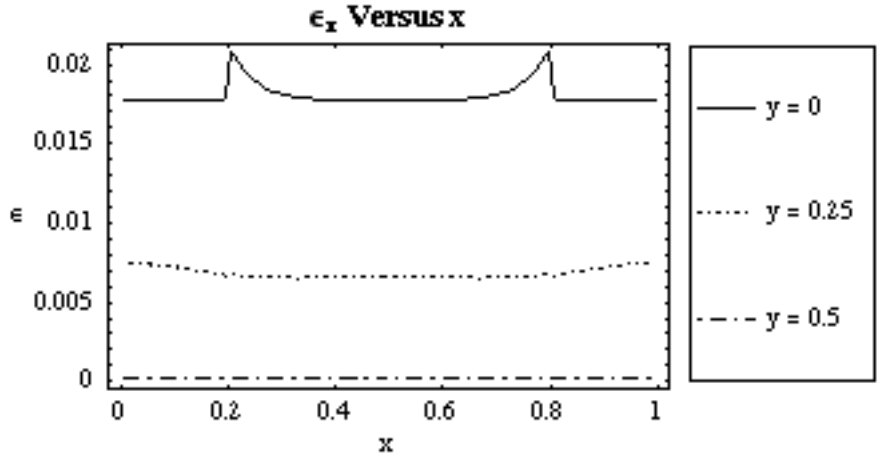

Figure 6.8 Plot of ε_x vs. x along the Edge, Quarter, and Center for the Standard Case

A few observations were made after examining the plots of the standard case. The largest vertical displacements occurred in the center, while the largest strains occurred at the pile edge. The deformed shape of the geogrid was at its highest elevation above the pile locations and it descended until the geogrid flattened out in the center of the cell. The vertical displacement was significantly larger at the center of the cell than at the center of the edge, because the geogrid was supported directly by the pile at the edges. The central region of the deformed geogrid was flat. In other words, the maximum displacement didn't occur at just one geogrid joint but at many, over a large area.

The maximum vertical displacement was constrained by the result of the following limiting displacement equation:

$$w = \frac{q}{k} \tag{6.15}$$

where w = non-dimensional vertical displacement
q = non-dimensional pressure
k = non-dimensional stiffness per unit area

The limiting displacement may also be referred to as the "unreinforced settlement" because the limiting displacement is the displacement of the soil that would occur if there were no geogrid. The result of the limiting displacement equation will be referred to as the unreinforced settlement for the remainder of this thesis. If the standard case pressure and stiffness values of the soft soil, -12.6 and 197, respectively, are used with Equation 6.11, the unreinforced settlement is 0.0637. The vertical displacement of the grid resting on the pile is subject to such a relationship as well, and using the pile pressure and stiffness values, -60 and 36,000, respectively, in Equation 6.11, the displacement above the pile is 0.00167.

The strain was largest at the pile edge because the spring stiffness and pressure values changed at these locations. The strain "spiked" at the corner of the pile where the change in slope of the geogrid was largest, and it appeared that a relationship existed between the change in slope shown on the contour and the strain plots. The strain was large if the change in slope was large and the strain was small if the change in slope was small.

6.1.2 Variation of Non-Dimensional Parameter b

The parameter b defined in Equation 2.3 is the ratio of half of the rectangular pile width to the center-to-center distance between each pile. The parameter was analyzed at 0.15 and 0.25 while all the other parameters were kept constant and equal to the standard case values.

Plots of the case with b = 0.15 are shown in Figures 6.8-6.14, and plots of the case with b = 0.25 are shown in Figures 6.15-6.21. The results of each analysis were used to compare the effect of the non-dimensional parameter b on the vertical displacement, w, and the strain, ε_x, at the edge, quarter line, and centerline of the geogrid. A solid line designates values for b = 0.15, a dashed line designates values for the standard case b = 0.2, and a dash-dot line designates values for b = 0.25. Figure 6.22 shows the vertical displacement values at the edge of the grid. Figure 6.23 shows the vertical displacement values mid-way between the edge and the center of the grid. Figure 6.24 shows the vertical displacement values along the centerline of the grid. Figures 6.25-6.27 show the strain values at the edge, midway between the edge and centerline, and at the centerline of the grid, respectively.

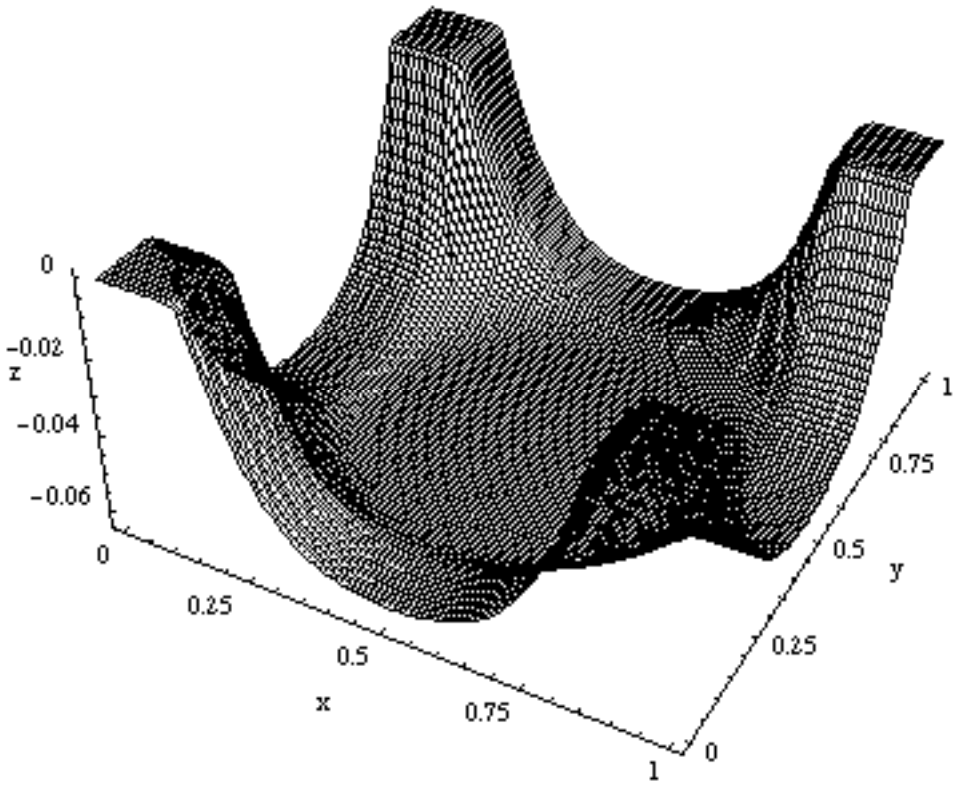

Figure 6.9 Three-Dimensional Plot for b=0.15

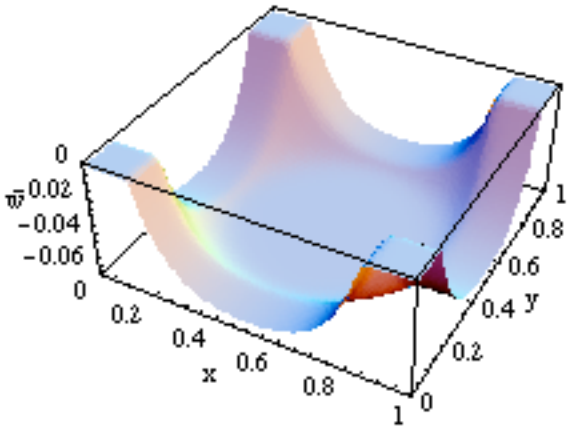

Figure 6.10 Three-Dimensional Plot of w for b=0.15

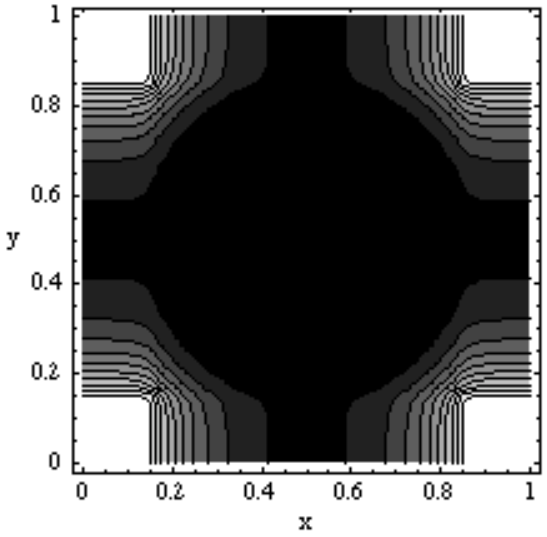

Figure 6.11 Contour Plot of w for b=0.15

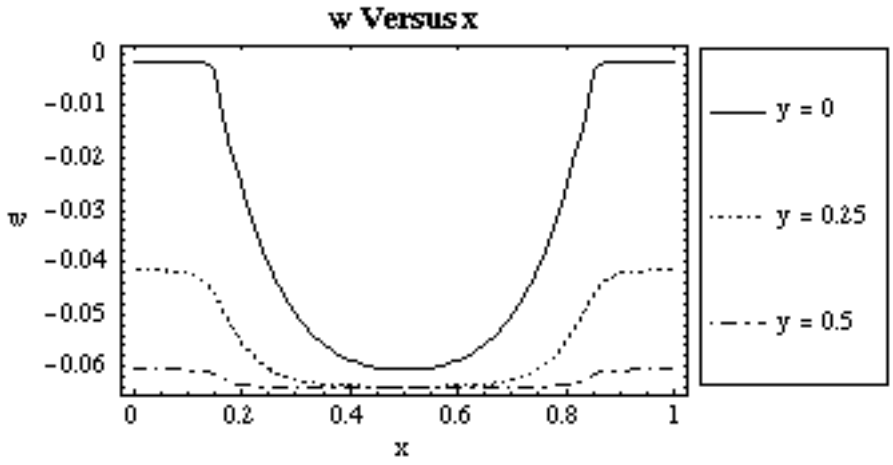

Figure 6.12 Plot of w vs. x along the Edge, Quarter and Center for b=0.15

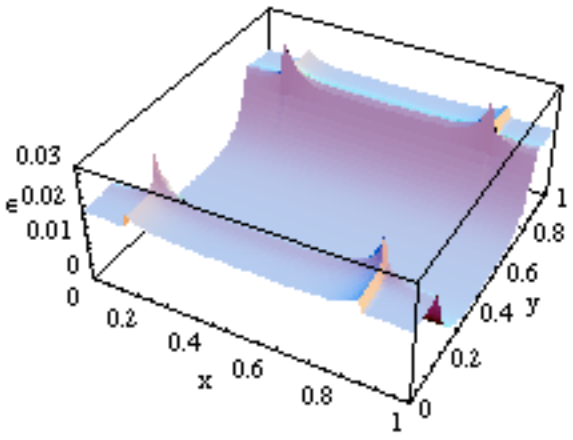

Figure 6.13 Three-Dimensional Plot of ε_x for b=0.15

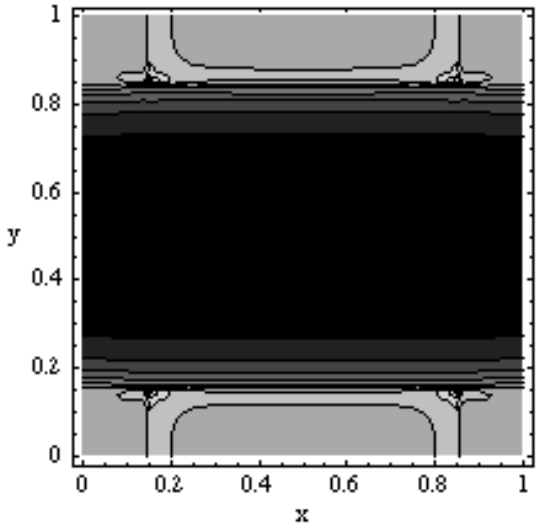

Figure 6.14 Contour Plot of ε_x for b=0.15

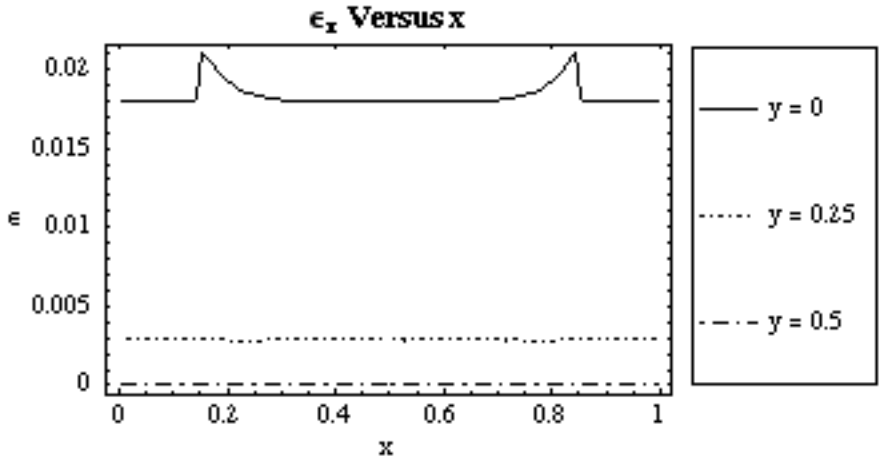

Figure 6.15 Plot of ε_x vs. x along the Edge, Quarter and Center for b=0.15

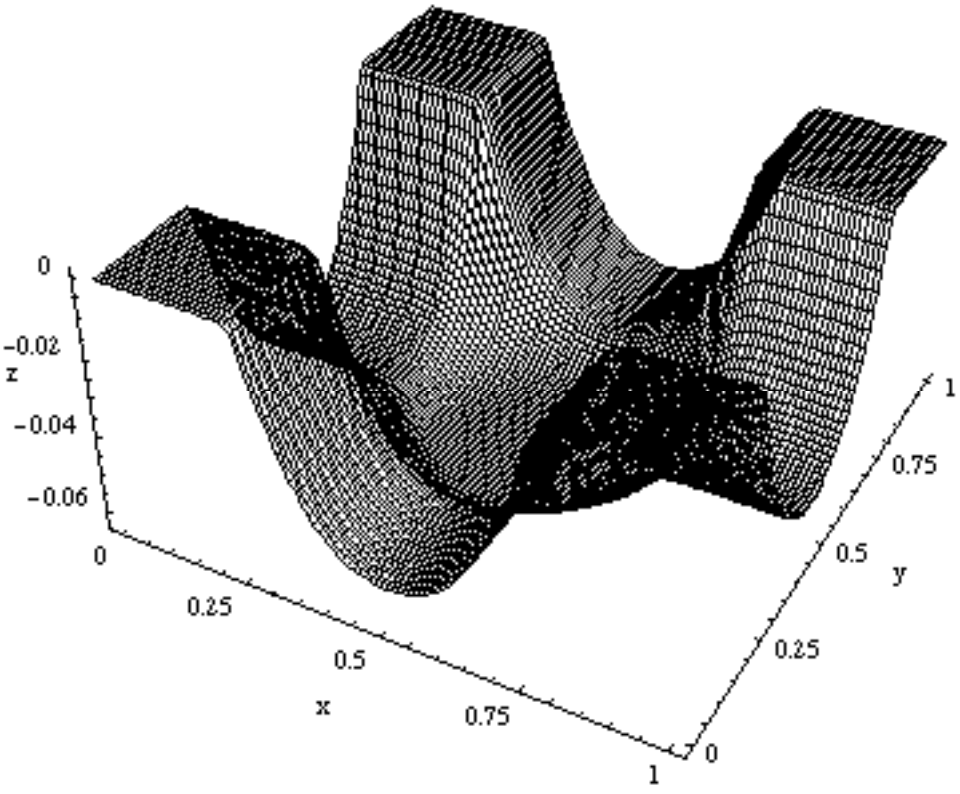

Figure 6.16 Three-Dimensional Plot for b=0.25

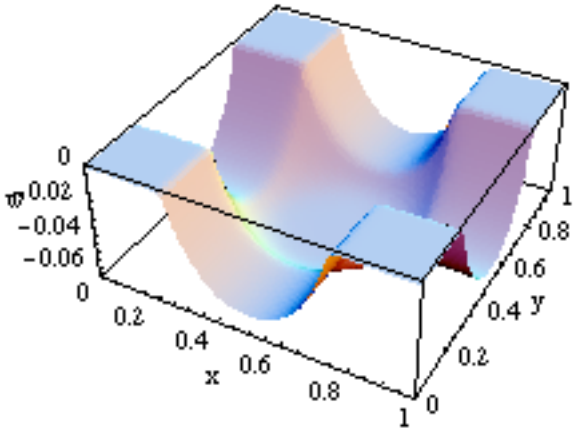

Figure 6.17 Three-Dimensional Plot of w for b=0.25

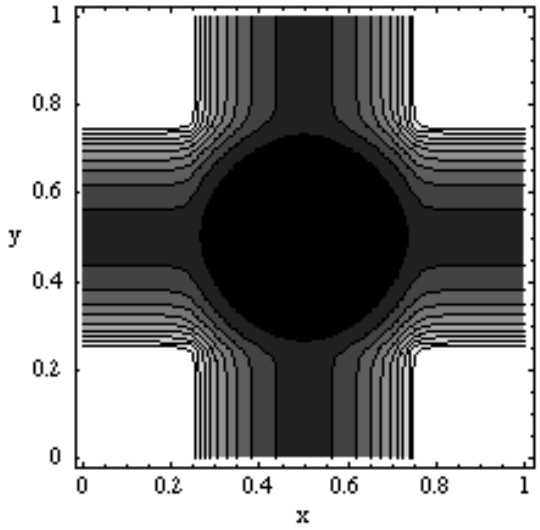

Figure 6.18 Contour Plot of w for b=0.25

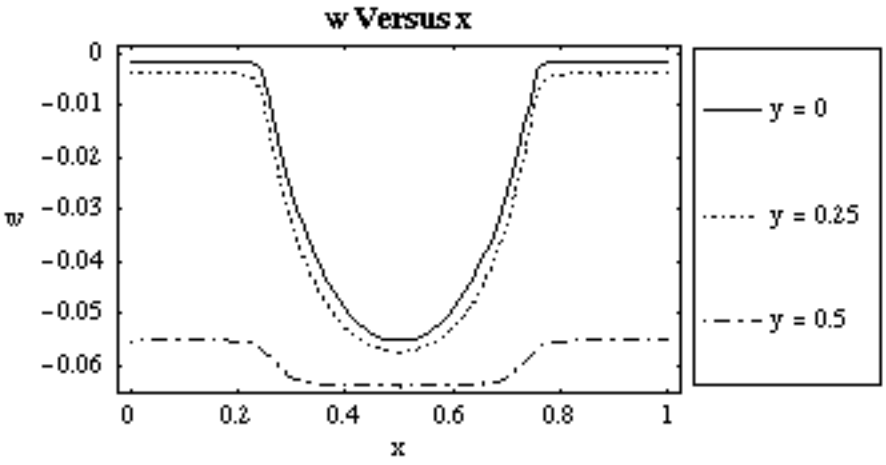

Figure 6.19 Plot of w vs. x along the Edge, Quarter and Center for b=0.25

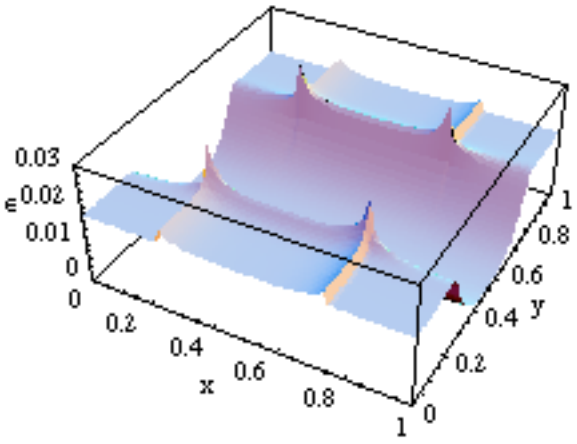

Figure 6.20 Three-Dimensional Plot of ε_x for b=0.25

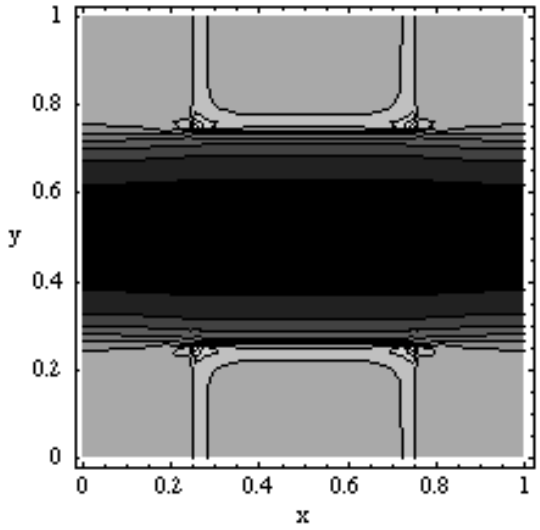

Figure 6.21 Contour Plot of ε_x for b=0.25

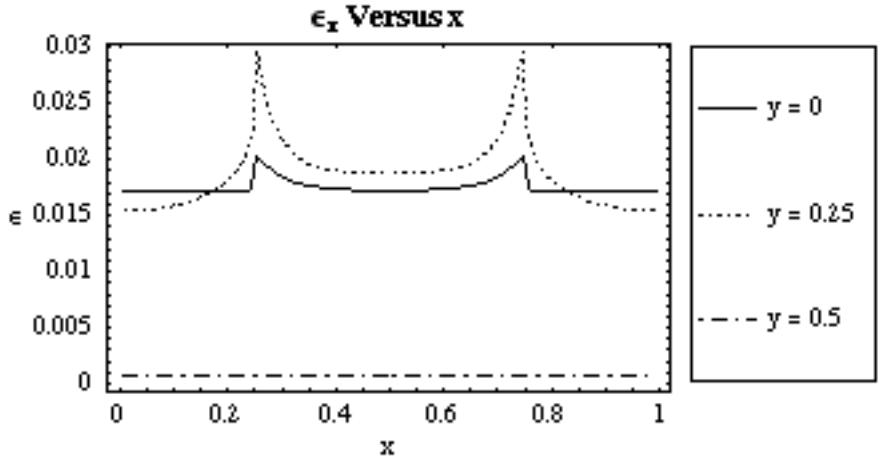

Figure 6.22 Plot of ε_x vs. x along the Edge, Quarter and Center for b=0.25

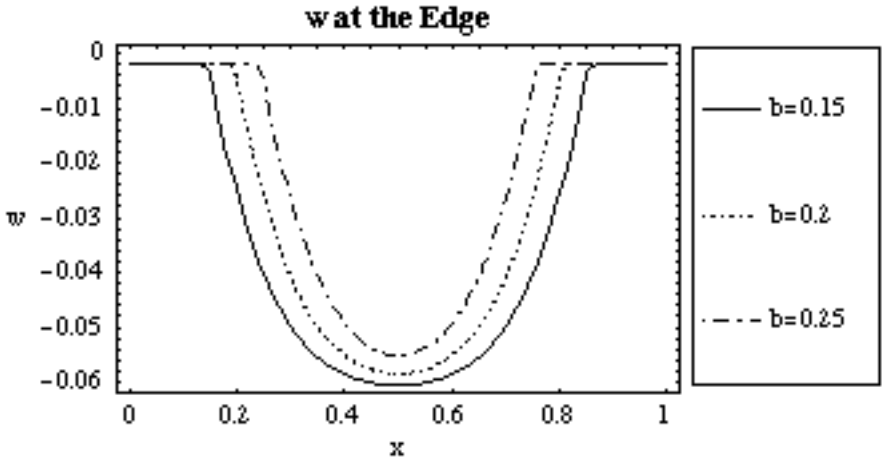

Figure 6.23 Plot of w along Edge for b = 0.15, 0.2, and 0.25

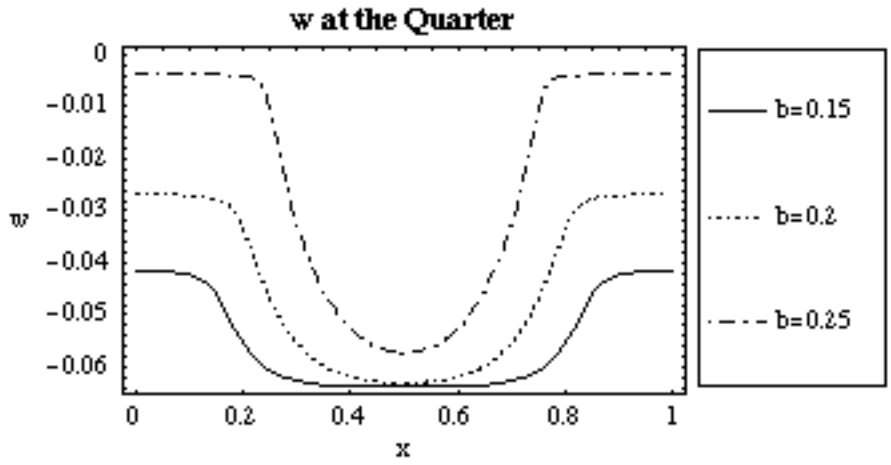

Figure 6.24 Plot of w along Quarter for b = 0.15, 0.2, and 0.25

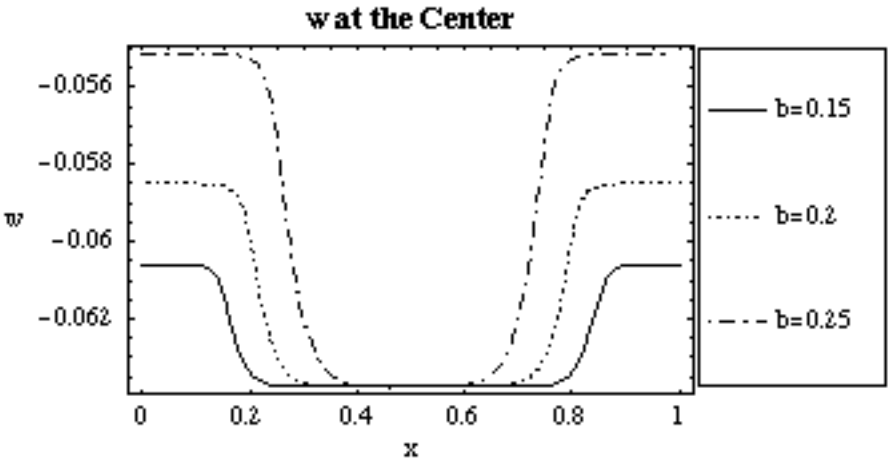

Figure 6.25 Plot of w along Center for b = 0.15, 0.2, and 0.25

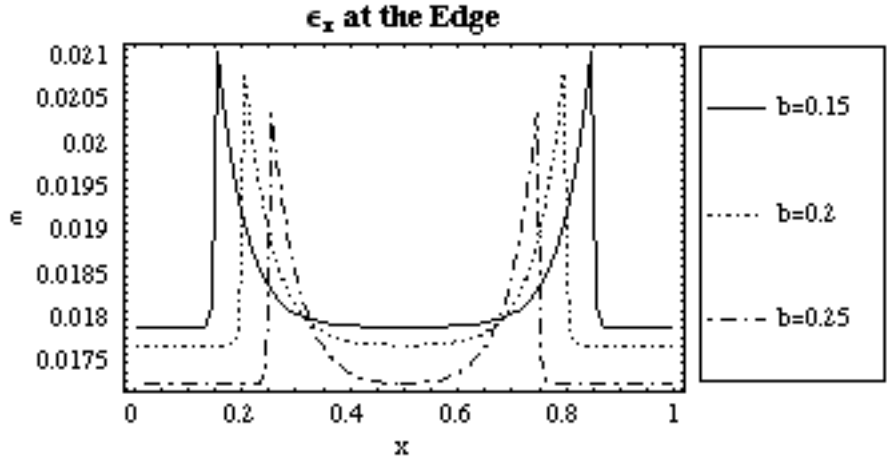

Figure 6.26 Plot of ε_x along Edge for b = 0.15, 0.2, and 0.25

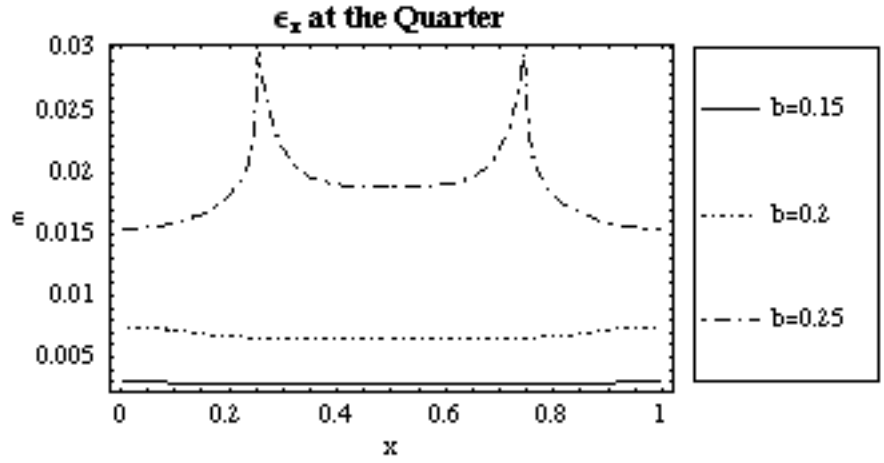

Figure 6.27 Plot of ε_x along Quarter for b = 0.15, 0.2, and 0.25

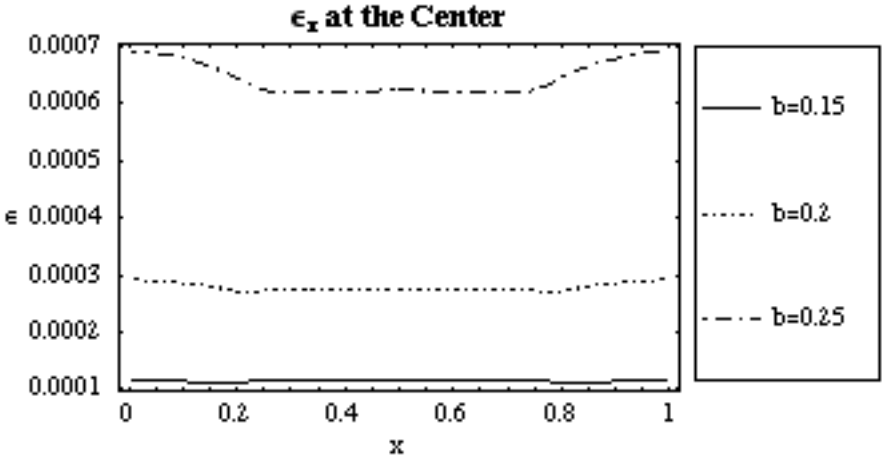

Figure 6.28 Plot of ε_x along Center for b = 0.15, 0.2, and 0.25

A few trends were observed in the plots of the pile width case study. The overall vertical displacement profile became deeper as b became smaller. The vertical displacement along the edge, Figure 6.22, best shows this trend. The displacement at the mid-point along the edge was noticeably less when b = 0.25 than when b = 0.15. If the pile width was smaller, then the geogrid rested on soft soil for a larger area, permitting the grid to have a larger area to significantly deform. All the geogrids analyzed for this case study were flat in the central region and the maximum displacements were equal to the unreinforced settlement.

Along the edge, the strain became larger as the pile width became smaller. As stated above, the largest strains occurred at the pile edge because the spring stiffness and loading changed. When b = 0.25, the strain at the quarterline was significantly larger than for the other cases because the quarterline was along the pile edge. The strain at the center was large when b was large because the displaced shape at the center was flat for a smaller region than when b was small. When the pile width was larger, the strain was larger in more strands. The differential settlement is the difference between the magnitudes of the largest vertical displacement and the smallest vertical displacement. The differential settlement was equal for each case, but if b was larger, then the geogrid under went this displacement over a smaller area, so the strands in the geogrid had to stretch more.

6.1.3 Variation of Non-Dimensional Parameter q_P

The parameter q_P is the non-dimensional pressure load that is applied to the geogrid resting above the pile, and it was derived from Equation 2.1. The model was analyzed twice, once when q_P was equal to -30 and again when q_P was equal to -120. The other parameter values were kept constant and equal to the standard case values.

Plots of the case with q_P = -30 are shown in Figures 6.28-6.34, and plots of the case with q_P = -120 are shown in Figures 6.35-6.41. The results of each analysis were used to compare the effect of the non-dimensional parameter q_P on the vertical displacement and the strain. A solid line designates values for q_P = -30, a dashed line designates values for the standard case q_P = -60, and a dash-dot line designates values for q_P = -120. Figures 6.42-6.44 show the vertical displacement values at the edge, midway between the edge and centerline, and at the centerline of the grid, respectively. Figures 6.45-6.47 show the strain values at the edge, midway between the edge and centerline, and at the centerline of the grid, respectively.

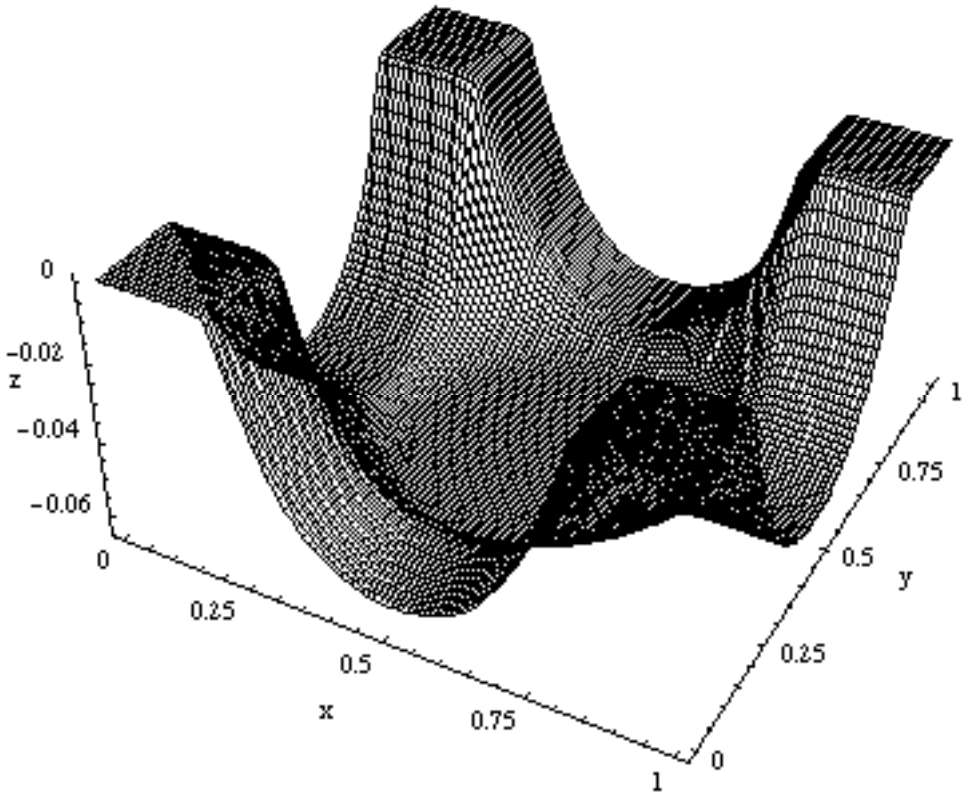

Figure 6.29 Three-Dimensional Plot for $q_P = -30$

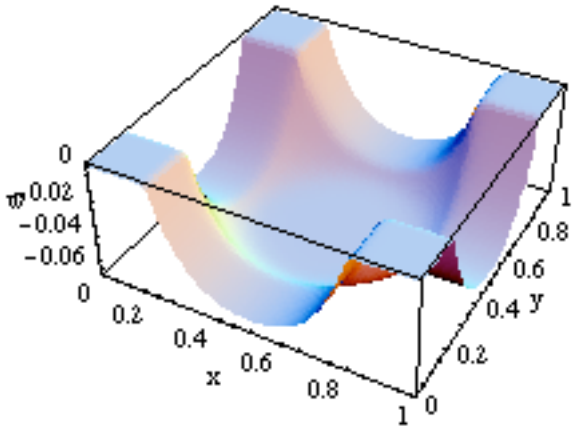

Figure 6.30 Three-Dimensional Plot of w for $q_P = -30$

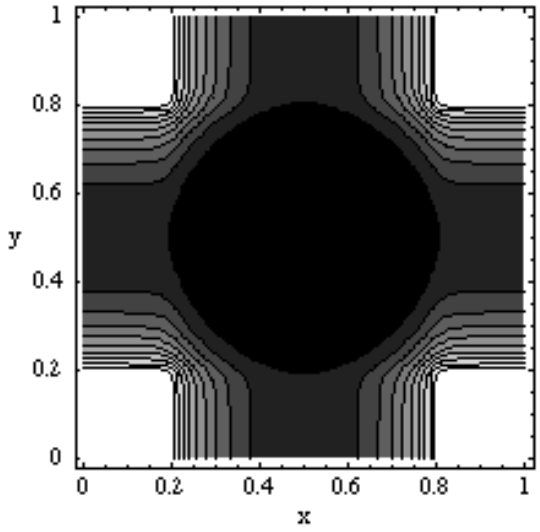

Figure 6.31 Contour Plot of w for q_P = -30

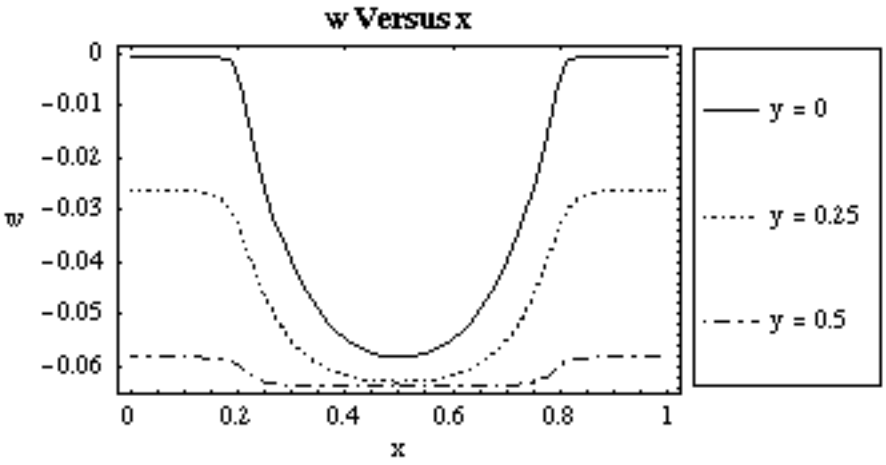

Figure 6.32 Plot of w vs. x along the Edge, Quarter and Center for q_P = -30

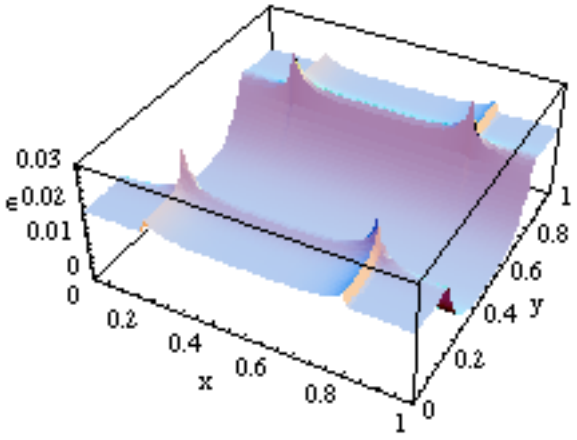

Figure 6.33 Three-Dimensional Plot of ε_x for q_P = -30

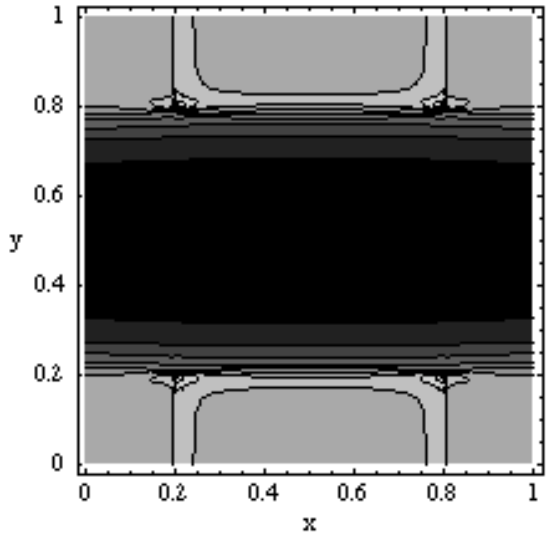

Figure 6.34 Contour Plot of ε_x for $q_P = -30$

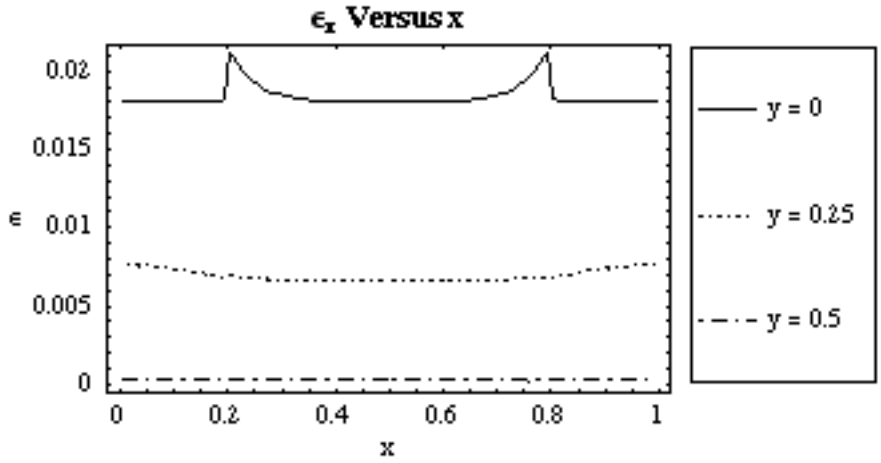

Figure 6.35 Plot of ε_x vs. x along the Edge, Quarter and Center for $q_P = -30$

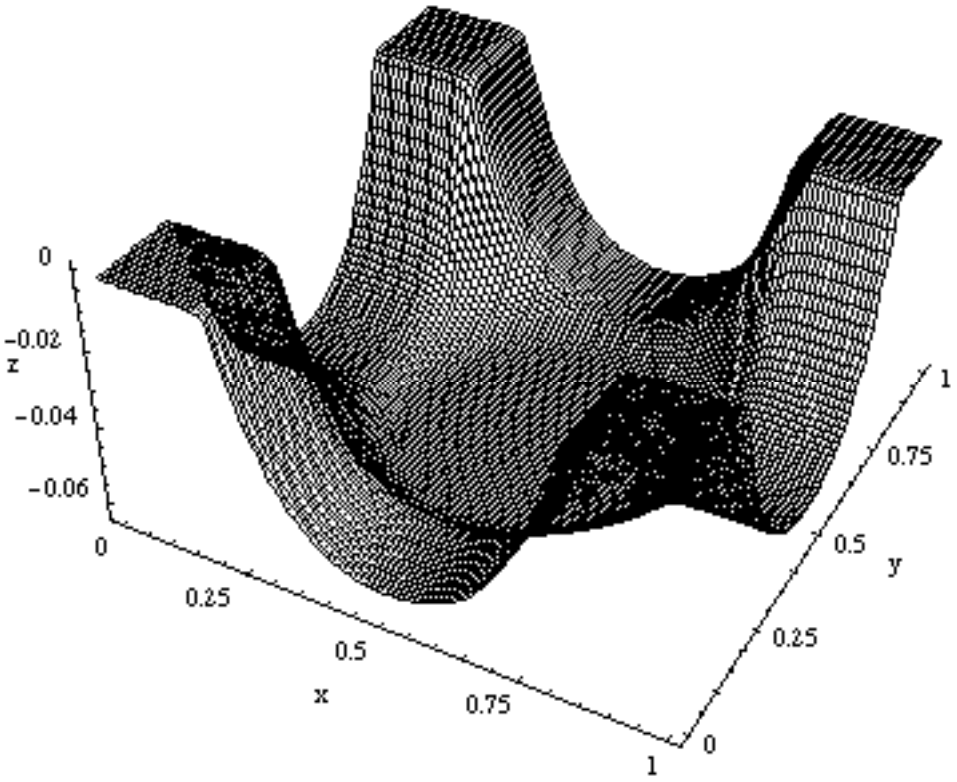

Figure 6.36 Three-Dimensional Plot for q_P = -120

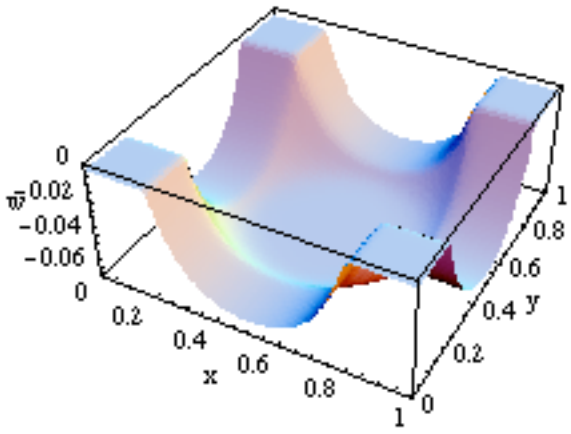

Figure 6.37 Three-Dimensional Plot of w for q_P = -120

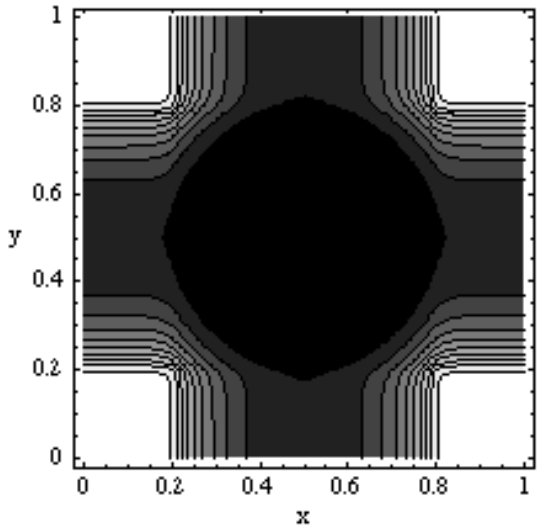

Figure 6.38 Contour Plot of w for q_P = -120

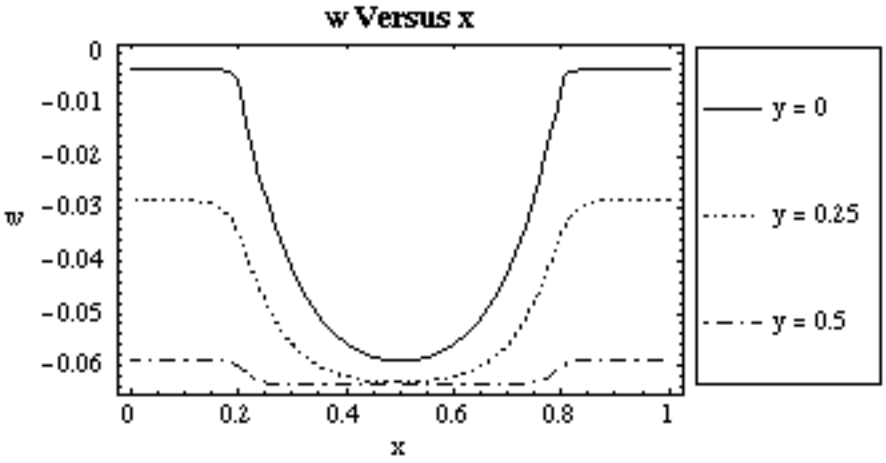

Figure 6.39 Plot of w vs. x along the Edge, Quarter and Center for q_P = -120

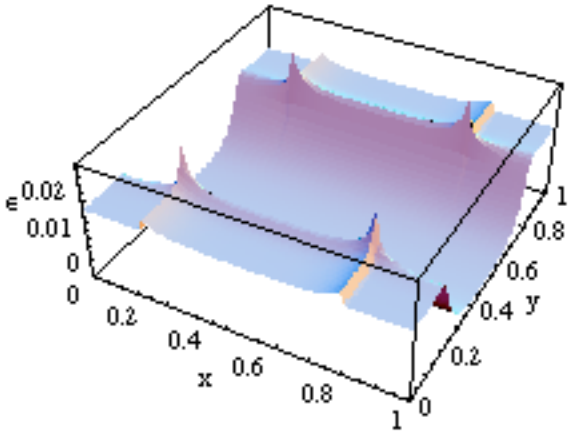

Figure 6.40 Three-Dimensional Plot of ε_x for q_P = -120

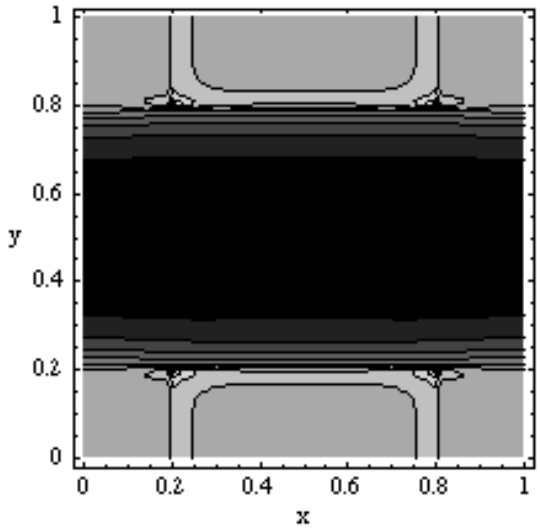

Figure 6.41 Contour Plot of ε_x for q_P = -120

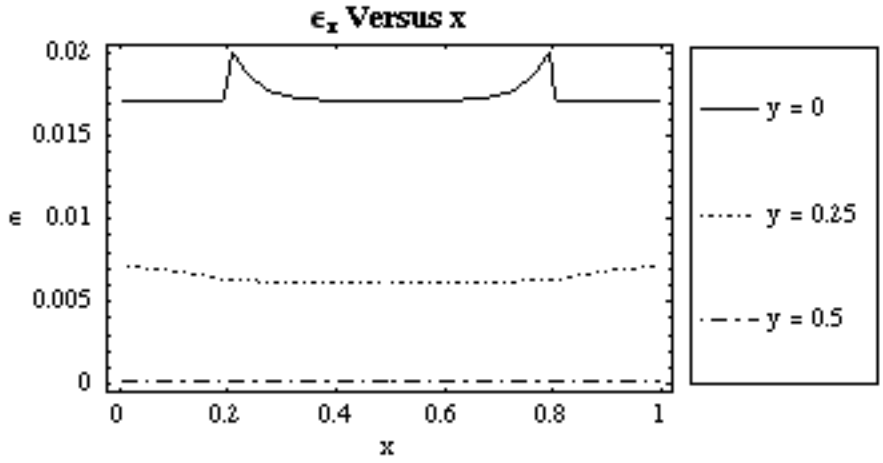

Figure 6.42 Plot of ε_x vs. x along the Edge, Quarter and Center for q_P = -120

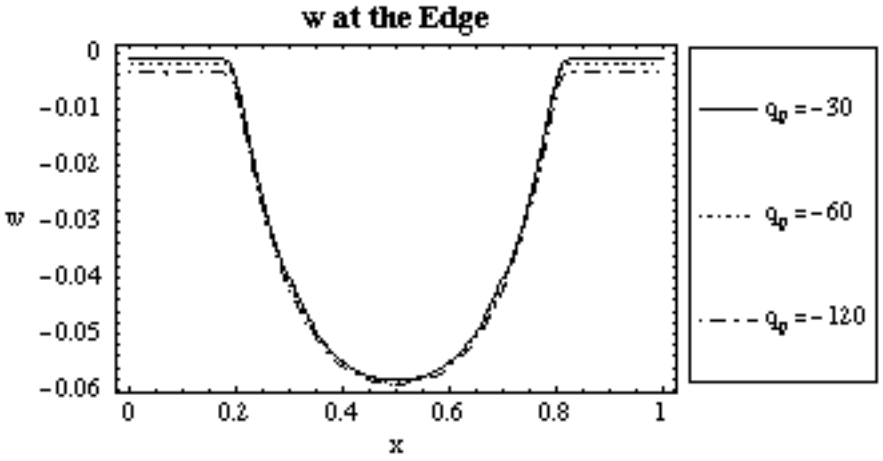

Figure 6.43 Plot of w along Edge for q_P = -30, -60, and -120

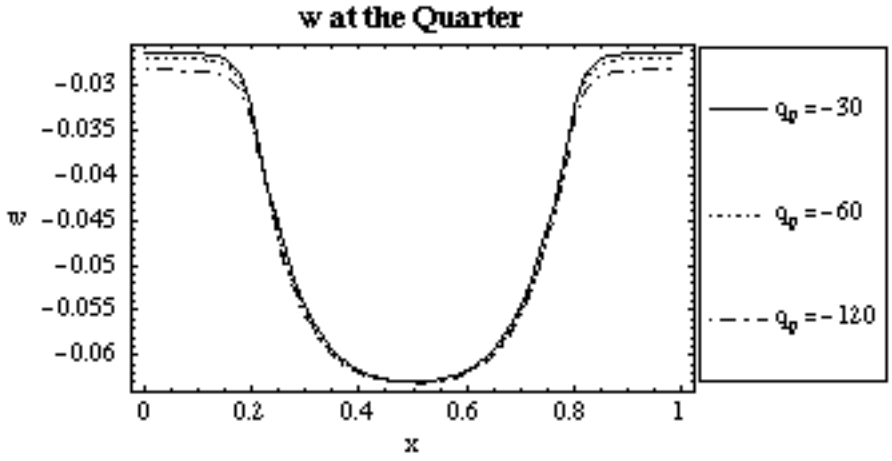

Figure 6.44 Plot of w along Quarter for q_P = -30, -60, and -120

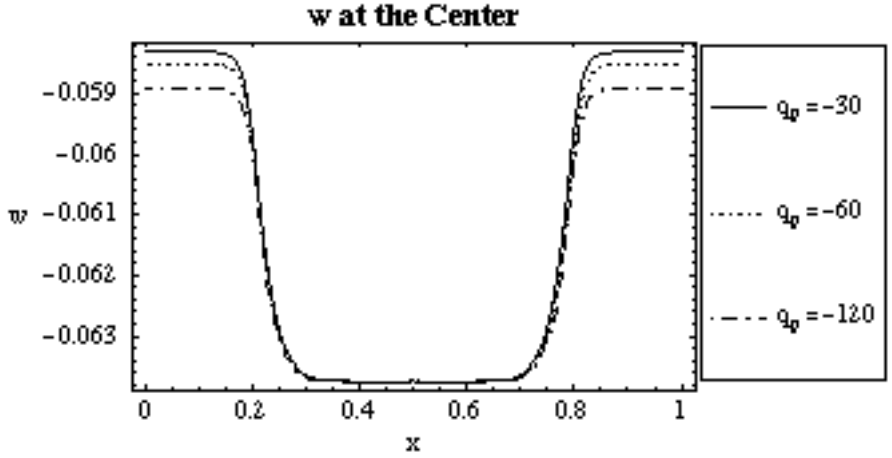

Figure 6.45 Plot of w along Center for q_P = -30, -60, and -120

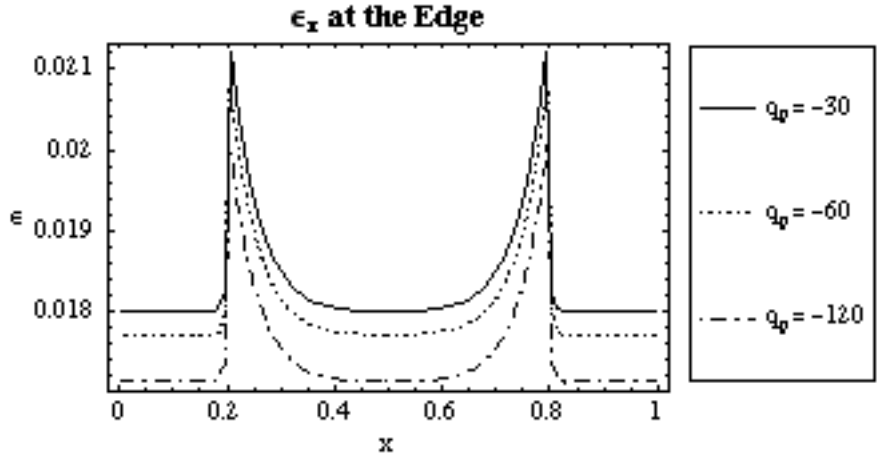

Figure 6.46 Plot of ε_x along Edge for q_P = -30, -60, and -120

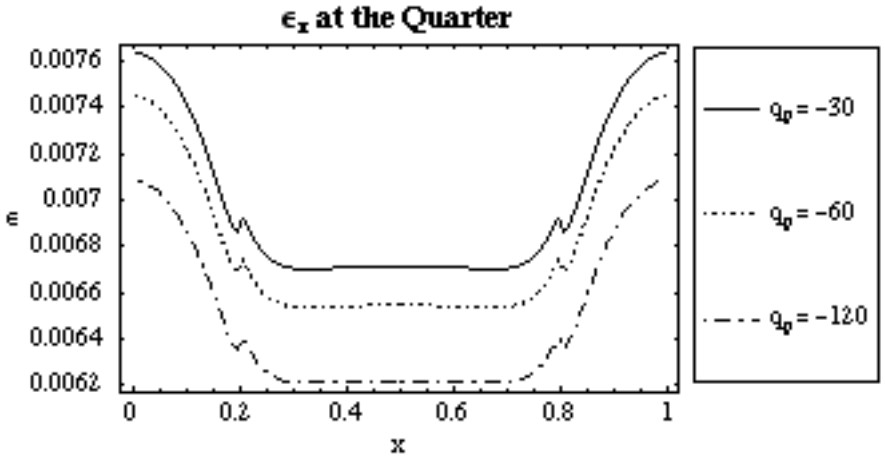

Figure 6.47 Plot of ε_x along Quarter for q_P = -30, -60, and -120

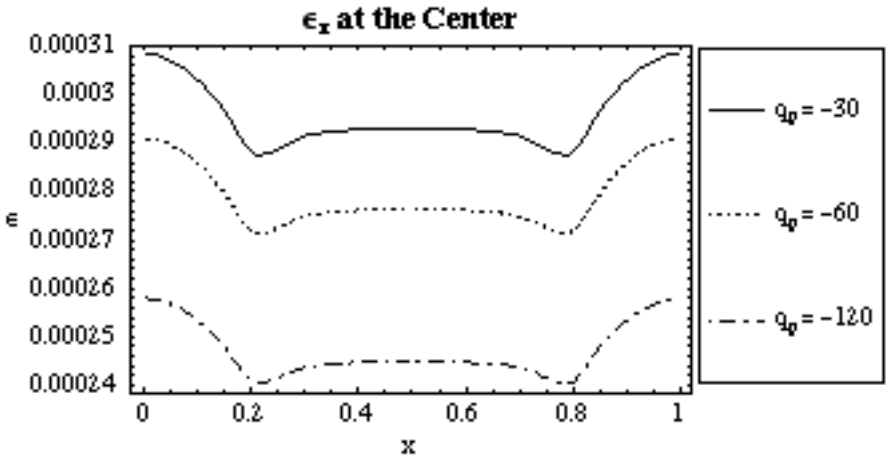

Figure 6.48 Plot of ε_x along Center for q_P = -30, -60, and -120

The displacement of the geogrid over the pile was equal to the relationship described in Equation 6.11. As a result, the vertical displacement of the geogrid was largest when q_P = -120 and smallest when q_P = -30. The displacement of the geogrid over the piles had a very small impact on the displaced shape of the geogrid region that rested on the soil. The overall deformed shape was slightly shallower if the magnitude of q_P was larger. The soil pressure and soil stiffness were unchanged, so the maximum displacement was equal to the same unreinforced settlement for all cases.

The strain increased slightly as the load over the pile increased. The differential settlement was largest when the load on the pile was least, because the elevation of the

geogrid above the pile was higher when the load was smaller. The strands of the geogrid were required to stretch more when the differential settlement was larger, so if the magnitude of q_P decreased, the strain increased.

6.1.4 Variation of Non-Dimensional Parameter q_S

The parameter q_S is the non-dimensional pressure load that is applied to the geogrid resting on the soft soil, and it is derived using Equation 2.1. Two separate analyses were carried out for this case study, once with q_S equal to -6.29 and another with q_S equal to -25.2. All other parameters remained equal to the standard case values.

Plots of the case with q_s = -6.29 are shown in Figures 6.48-6.54, and plots of the case with q_P = -25.2 are shown in Figures 6.55-6.61. The results of each analysis were used to compare the effect of the non-dimensional parameter q_S on the vertical displacement and the strain. A solid line designates values for q_S = -6.29, a dashed line designates values for the standard case q_P = -12.6, and a dash-dot line designates values for q_S = -25.2. Figures 6.62-6.64 show the vertical displacement values and Figures 6.65-6.67 show the strain values at the edge, midway between the edge and centerline, and at the centerline of the grid, respectively.

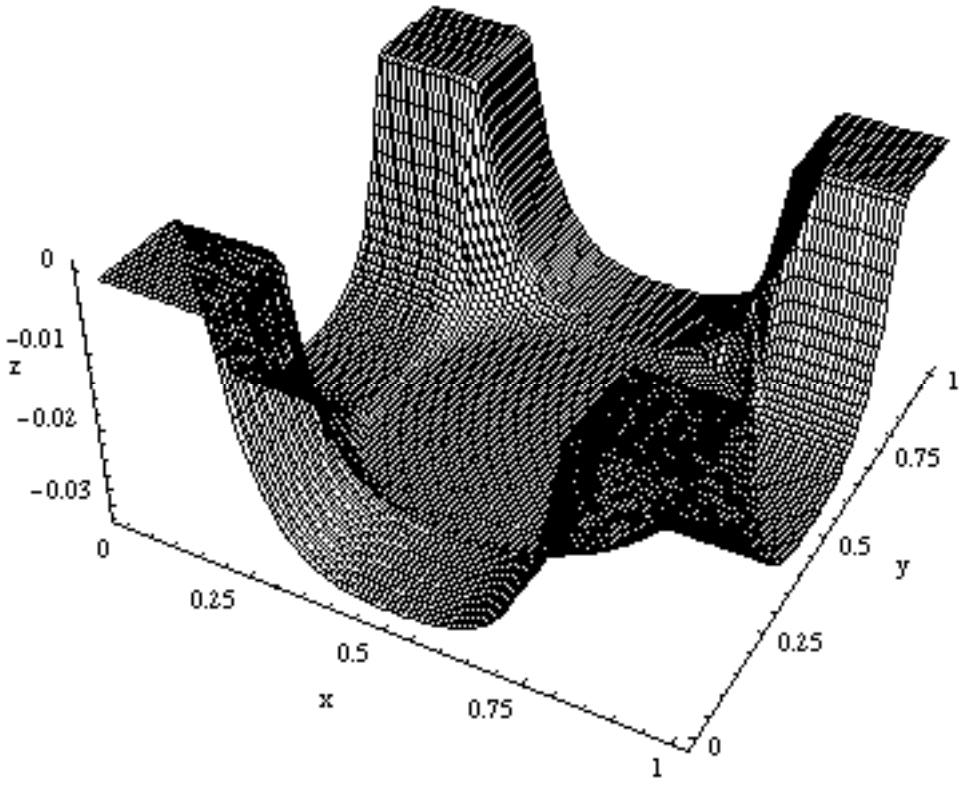

Figure 6.49 Three-Dimensional Plot for q_S = -6.29

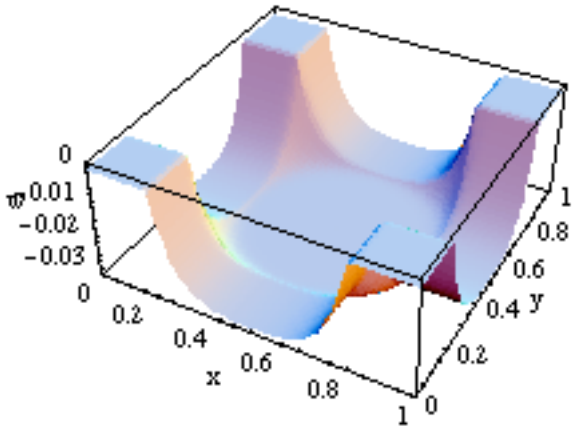

Figure 6.50 Three-Dimensional Plot of w for q_S = -6.29

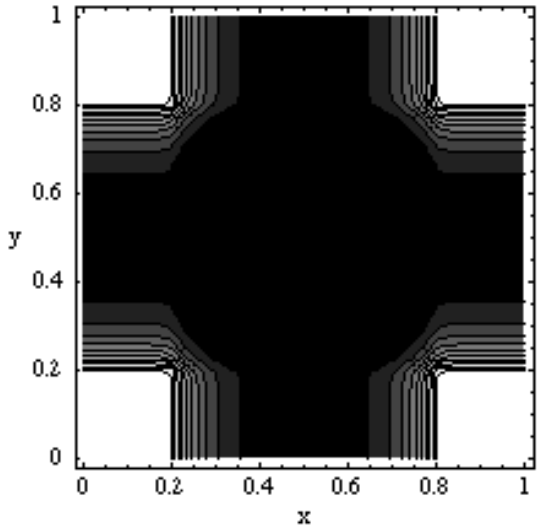

Figure 6.51 Contour Plot of w for q_S = -6.29

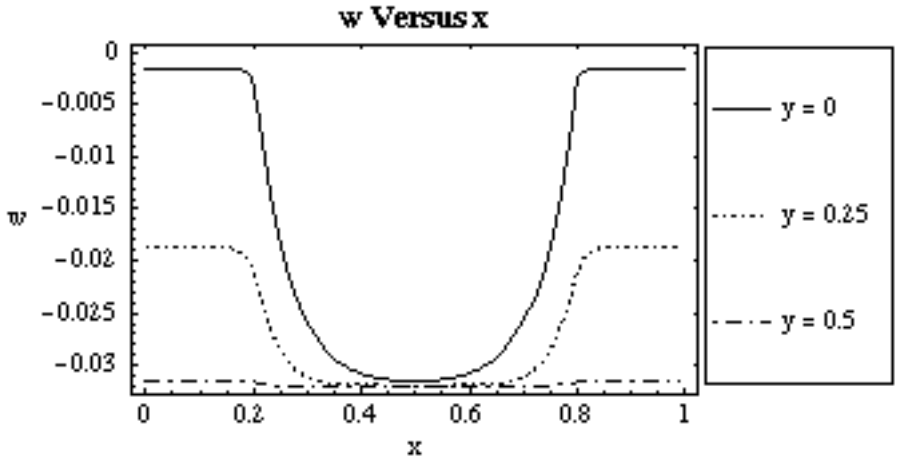

Figure 6.52 Plot of w vs. x along the Edge, Quarter and Center for q_S = -6.29

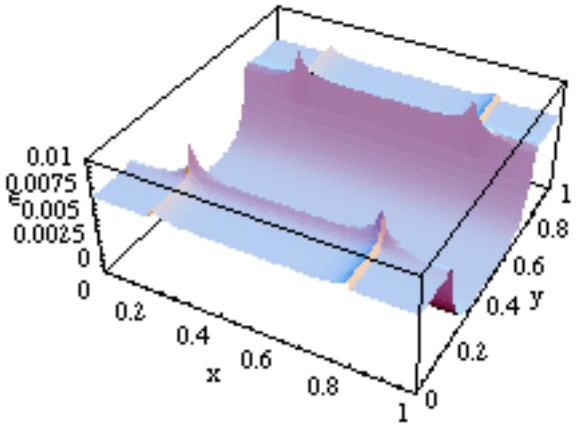

Figure 6.53 Three-Dimensional Plot of ε_x for q_S = -6.29

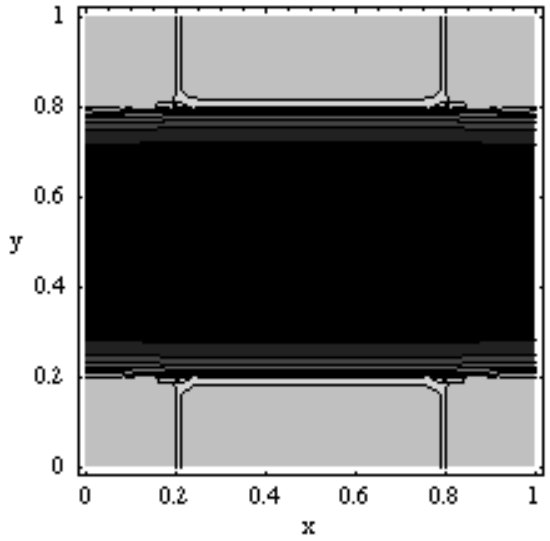

Figure 6.54 Contour Plot of ε_x for $q_S = -6.29$

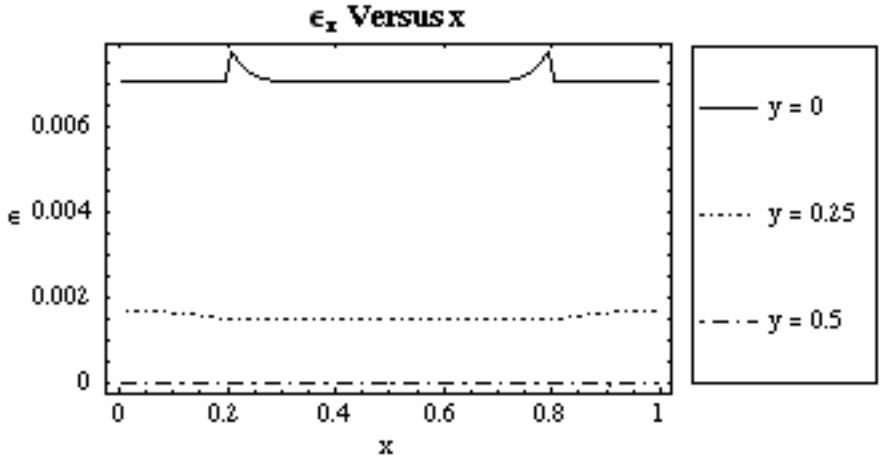

Figure 6.55 Plot of ε_x vs. x along the Edge, Quarter and Center for $q_S = -6.29$

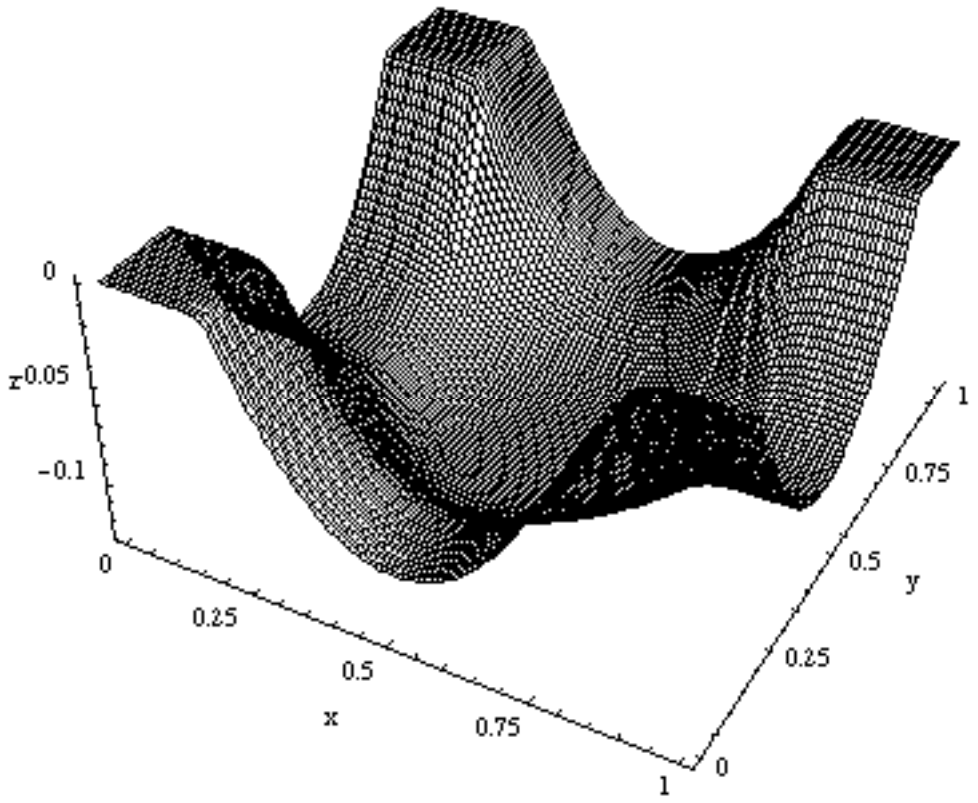

Figure 6.56 Three-Dimensional Plot for q_S = -25.2

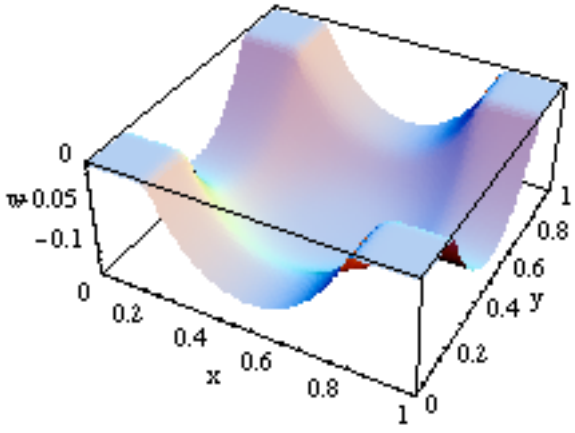

Figure 6.57 Three-Dimensional Plot of w for q_S = -25.2

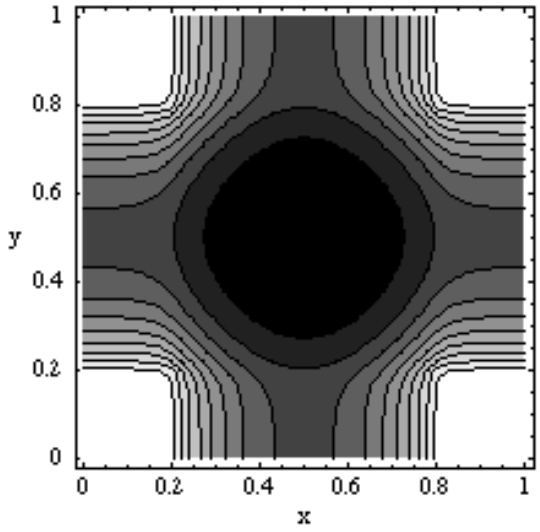

Figure 6.58 Contour Plot of w for q_S = -25.2

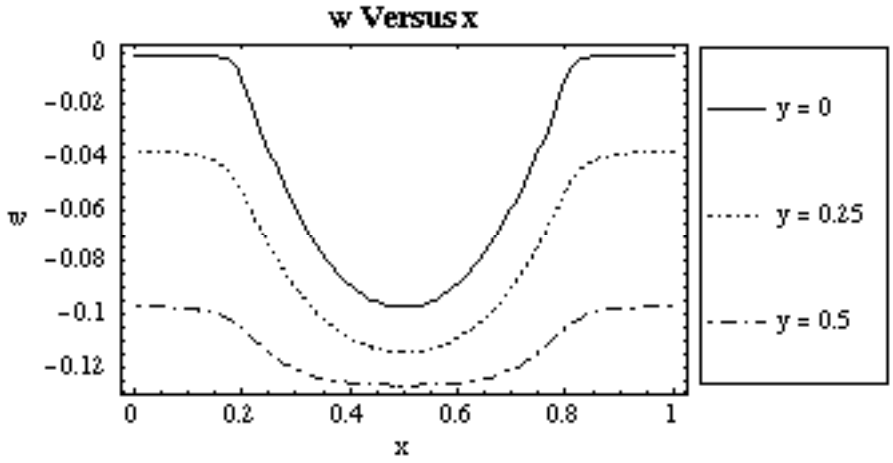

Figure 6.59 Plot of w vs. x along the Edge, Quarter and Center for q_S = -25.2

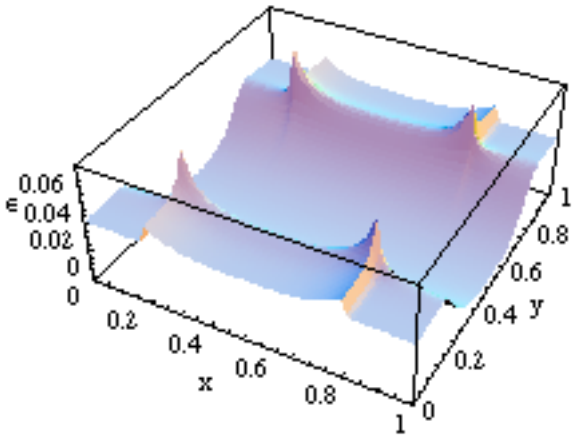

Figure 6.60 Three-Dimensional Plot of ε_x for q_S = -25.2

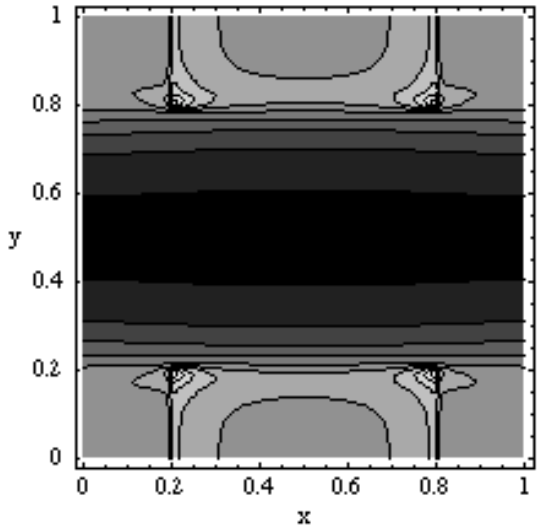

Figure 6.61 Contour Plot of ε_x for q_S = -25.2

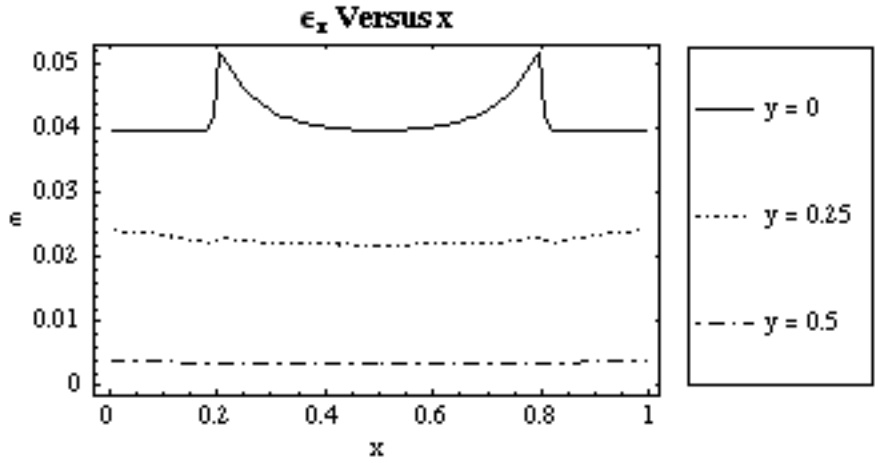

Figure 6.62 Plot of ε_x vs. x along the Edge, Quarter and Center for q_S = -25.2

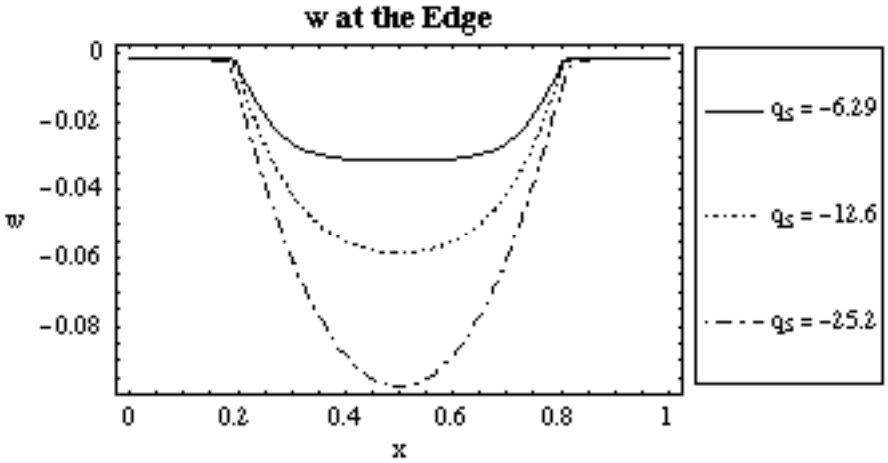

Figure 6.63 Plot of w along Edge for q_S = -6.29, -12.6, and -25.2

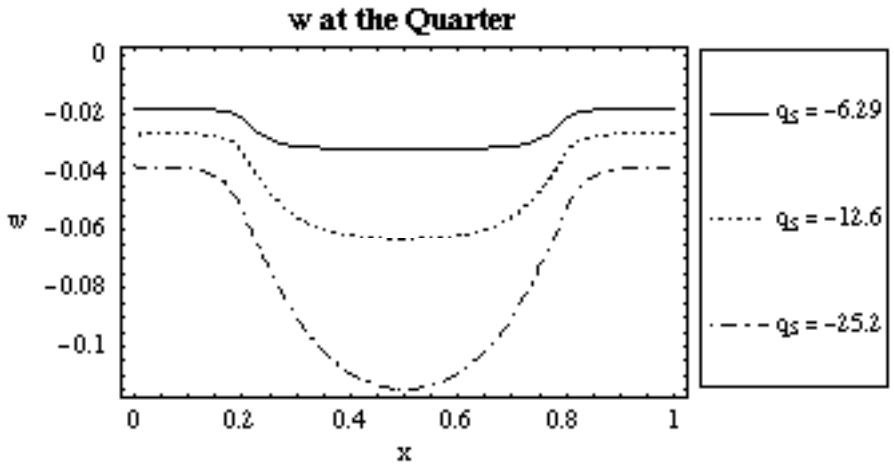

Figure 6.64 Plot of w along Quarter for q_S = -6.29, -12.6, and -25.2

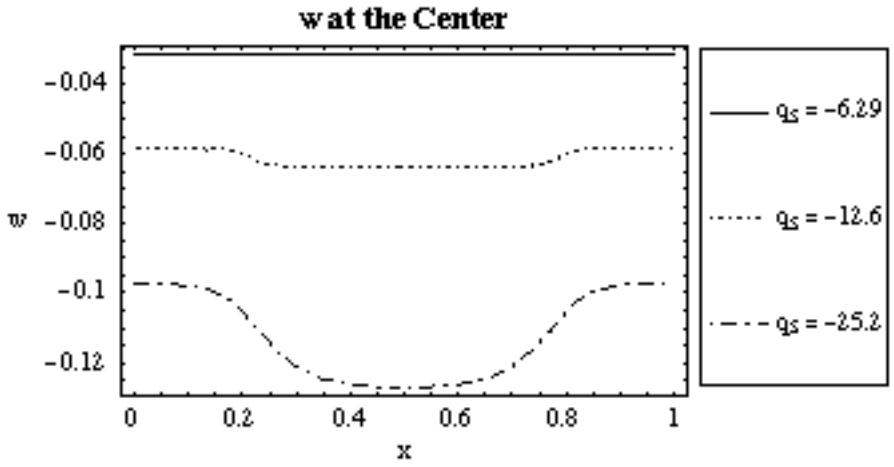

Figure 6.65 Plot of w along Center for q_S = -6.29, -12.6, and -25.2

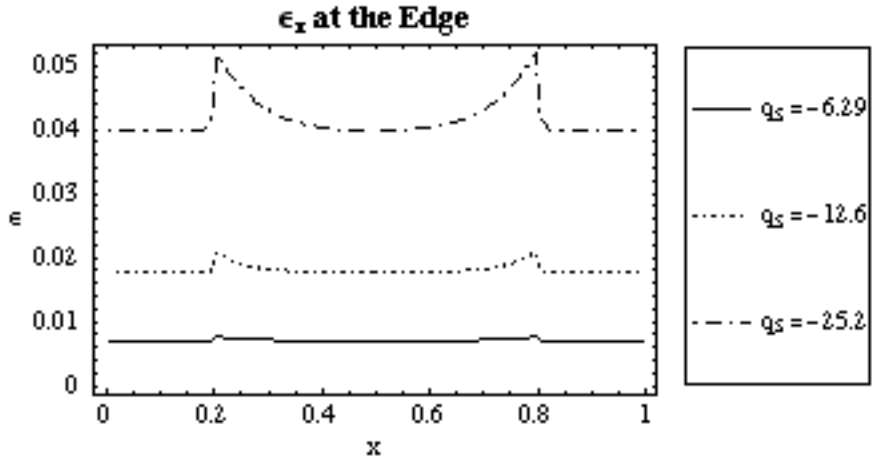

Figure 6.66 Plot of ε_x along Edge for q_S = -6.29, -12.6, and -25.2

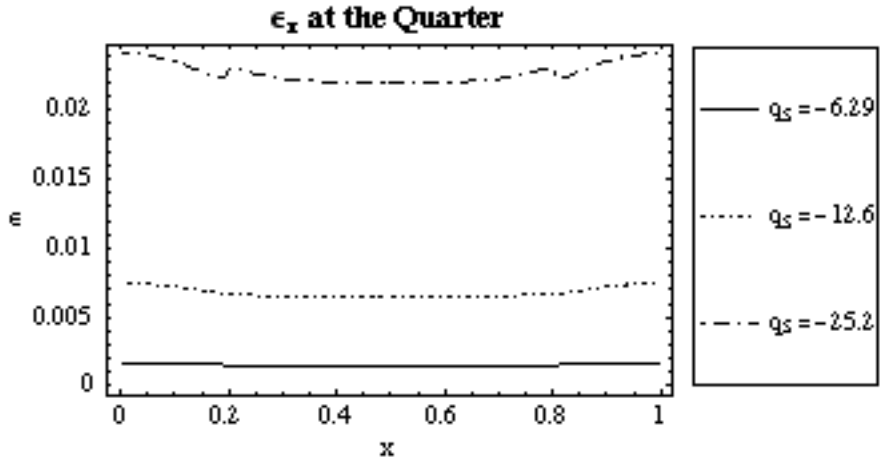

Figure 6.67 Plot of ε_x along Quarter for q_S = -6.29, -12.6, and -25.2

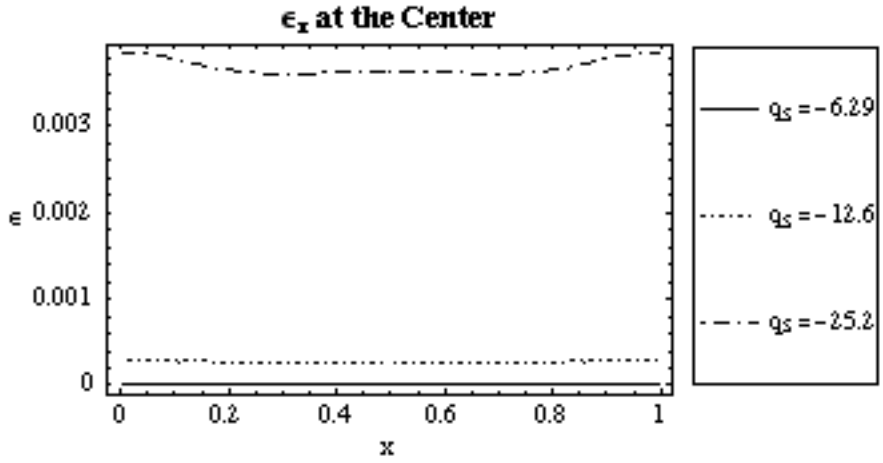

Figure 6.68 Plot of ε_x along Center for q_S = -6.29, -12.6, and -25.2

Figures 6.62-6.64 show that the vertical displacements were largest when the overlaying soil pressure was greatest. After examining the displaced shapes of the geogrids in Figures 6.48, 6.49, 6.1, 6.2, 6.55, and 6.56, it was observed that when the soil pressure was smaller in magnitude, the geogrid had a large central flat region, and when the soil pressure was larger in magnitude, the geogrid had a small central flat region or none at all. As previously stated above, in almost all cases for this model the maximum displacement was equal to the unreinforced settlement that was described by Equation 6.11, and since the maximum displacement cannot exceed the unreinforced settlement, flat regions occurred.

The maximum displacement increased when the soil pressure increased, while the displacement of the geogrid above the piles remained unchanged, so if the soil pressure increased, then the geogrid's strands had to elongate more. The larger differential settlement when q_S = -25.2 caused some of the strains in the strands to exceed 0.07. If strains were this large, it was unlikely that the strands behaved linear elastically. The limited testing that was done during the research for this thesis, and the geogrid product literature, seem to imply that the geogrid strands do not behave linear elastically for strains exceeding 0.05. This model assumed that all the strands behaved linear elastically.

6.1.5 Variation of Non-Dimensional Parameter k_P

The parameter k_P derived from Equation 2.2 is the non-dimensional stiffness per unit area of the soil above the pile. Two different k_P values were used in this case study: 18,000 and 72,000. The standard case values were used for all the other parameters in this study and they remained unchanged for each analysis.

Plots of the case with k_P = 18,000 are shown in Figures 6.68-6.74, and plots of the case with k_P = 72,000 are shown in Figures 6.75-6.81. The results of each analysis were used to compare the effect of the non-dimensional parameter k_P on the vertical displacement and the strain. A solid line designates values for k_P = 18,000, a dashed line designates values for the standard case k_P = 36,000, and a dash-dot line designates values for k_P = 72,000. Figures 6.82-6.84 show the vertical displacement values and Figures 6.85-6.87 show the strain values at the edge, midway between the edge and centerline, and at the centerline of the grid, respectively.

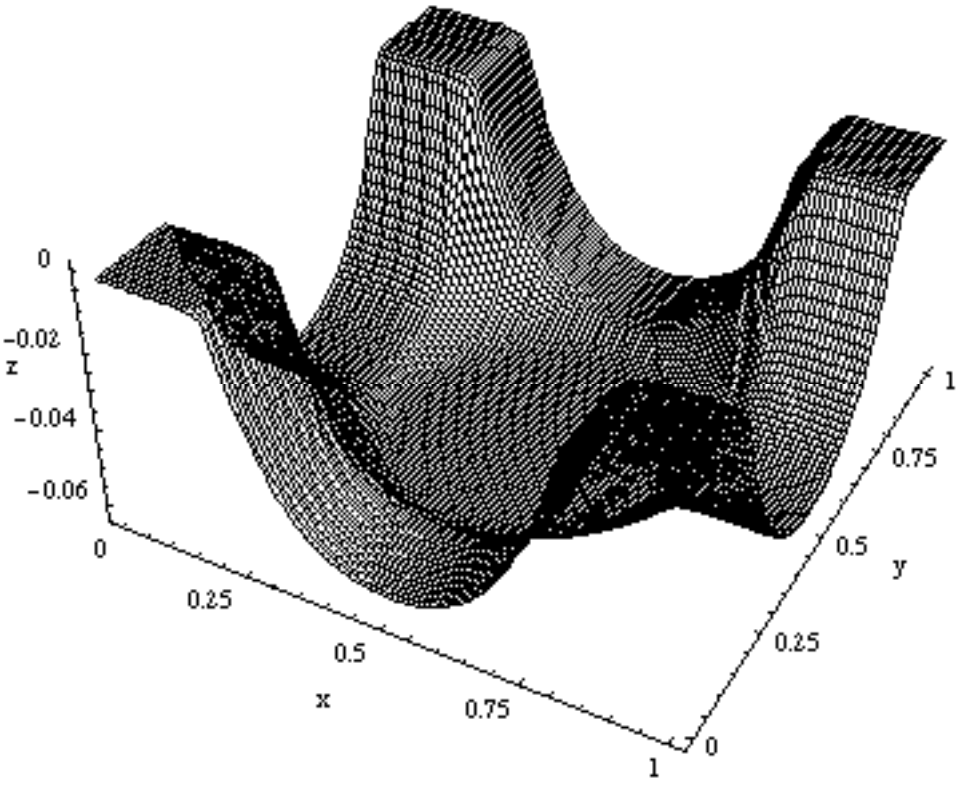

Figure 6.69 Three-Dimensional Plot for k_P = 18000

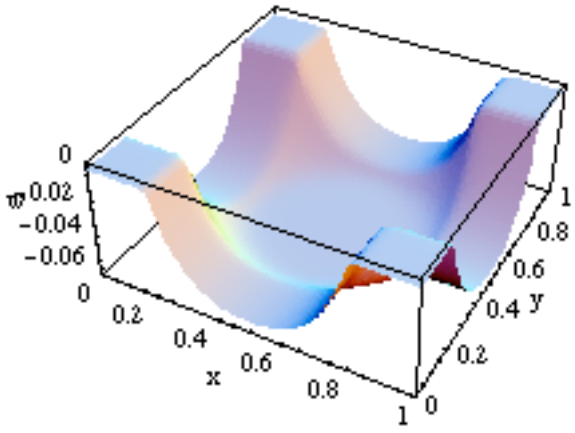

Figure 6.70 Three-Dimensional Plot of w for k_P = 18000

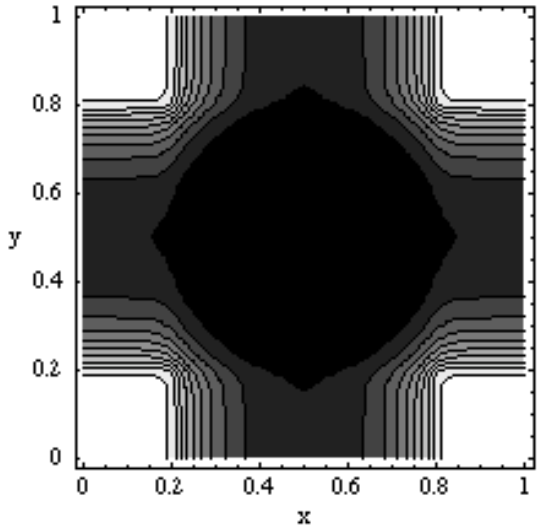

Figure 6.71 Contour Plot of w for k_P = 18000

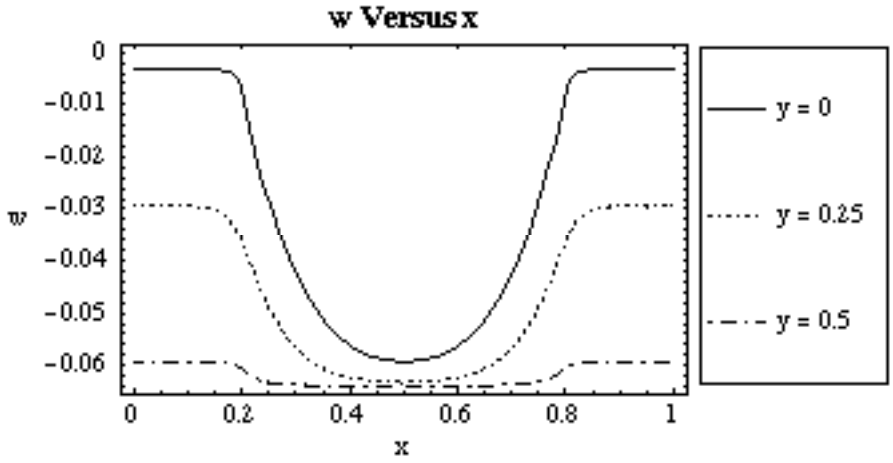

Figure 6.72 Plot of w vs. x along the Edge, Quarter and Center for k_P = 18000

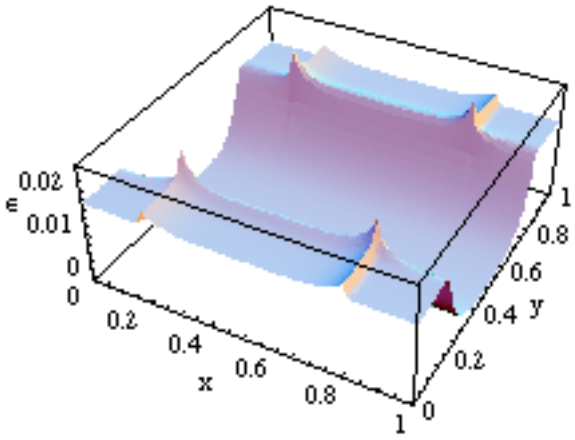

Figure 6.73 Three-Dimensional Plot of ε_x for k_P = 18000

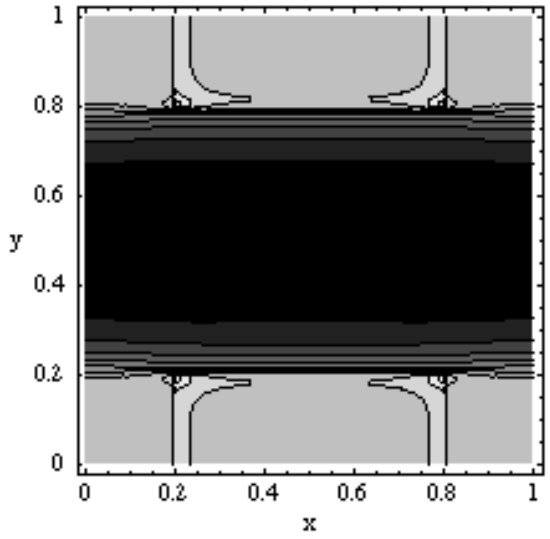

Figure 6.74 Contour Plot of ε_x for $k_P = 18000$

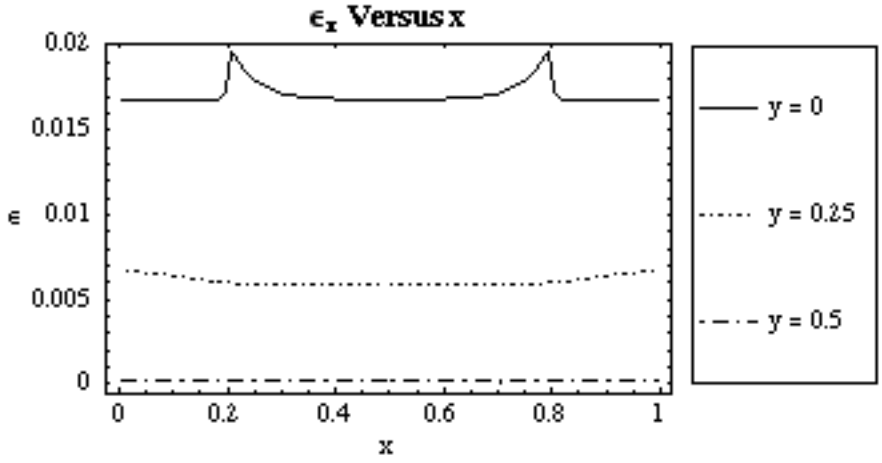

Figure 6.75 Plot of ε_x vs. x along the Edge, Quarter and Center for $k_P = 18000$

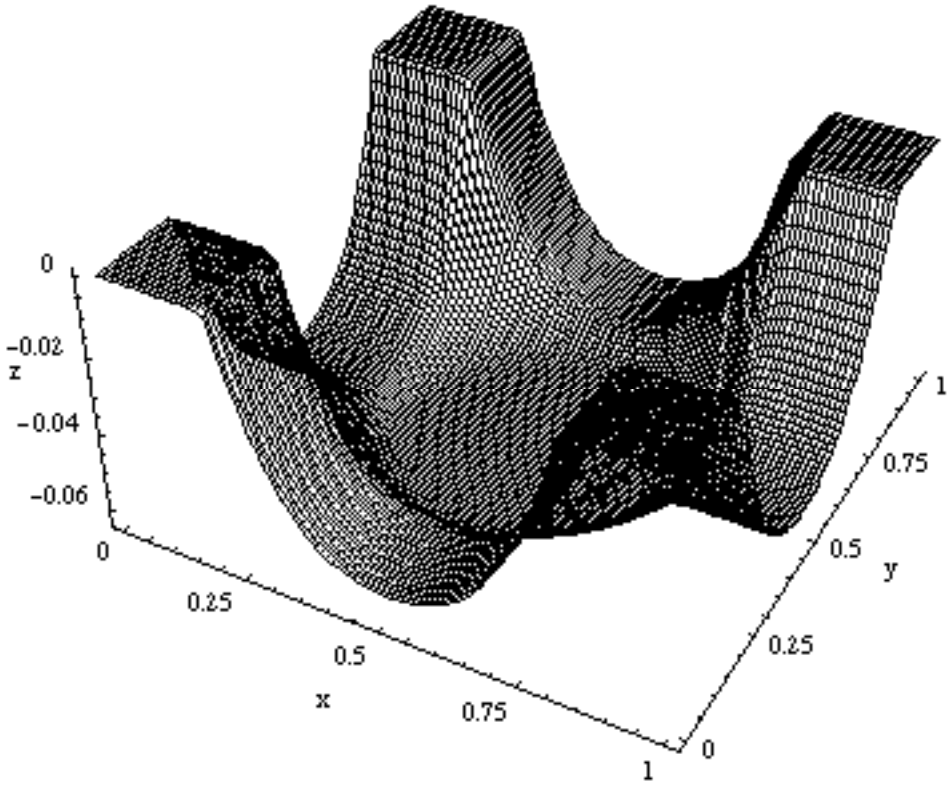

Figure 6.76 Three-Dimensional Plot for k_P = 72000

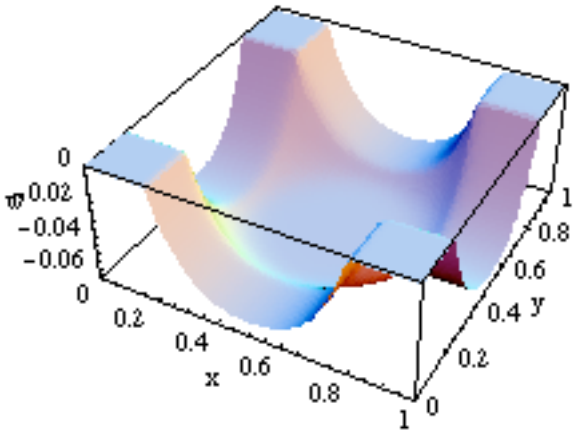

Figure 6.77 Three-Dimensional Plot of w for k_P = 72000

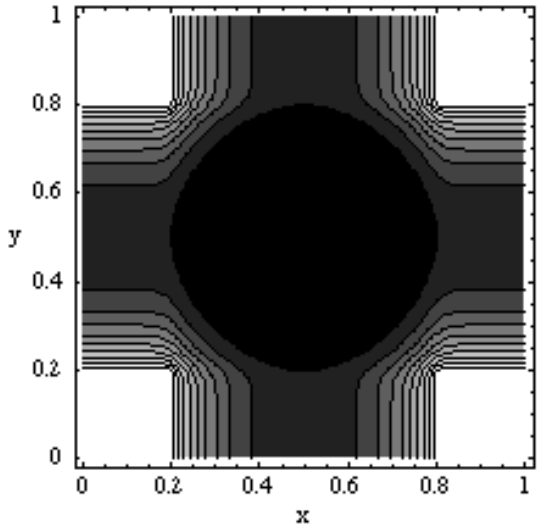

Figure 6.78 Contour Plot of w for k_P = 72000

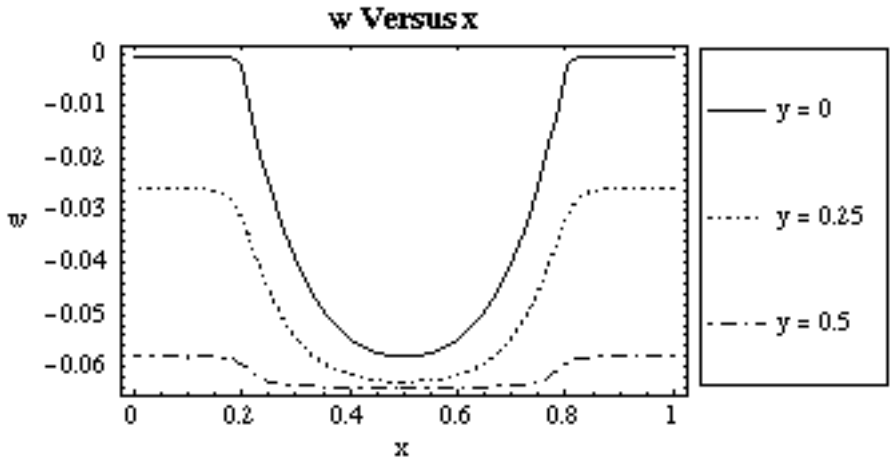

Figure 6.79 Plot of w vs. x along the Edge, Quarter and Center for k_P = 72000

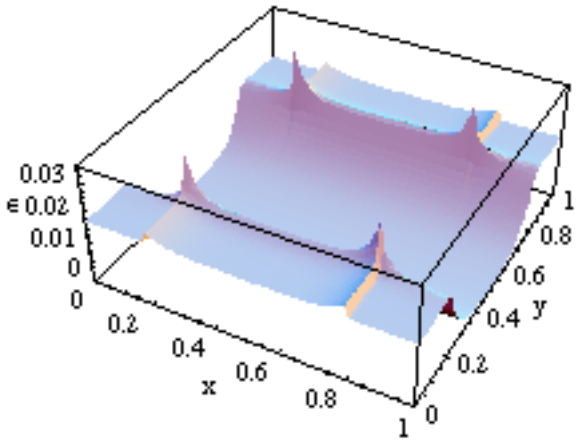

Figure 6.80 Three-Dimensional Plot of ε_x for k_P = 72000

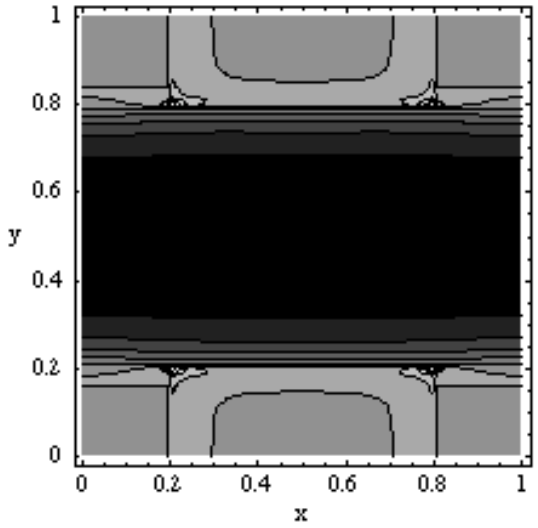

Figure 6.81 Contour Plot of ε_x for $k_P = 72000$

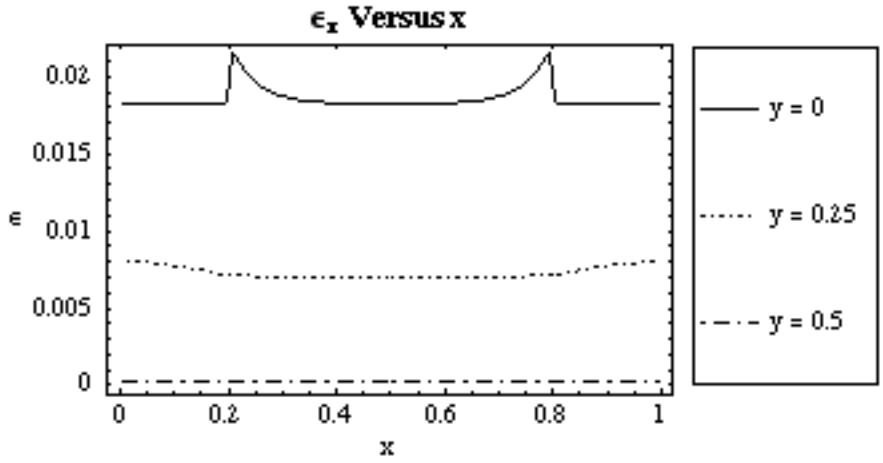

Figure 6.82 Plot of ε_x vs. x along the Edge, Quarter and Center for $k_P = 72000$

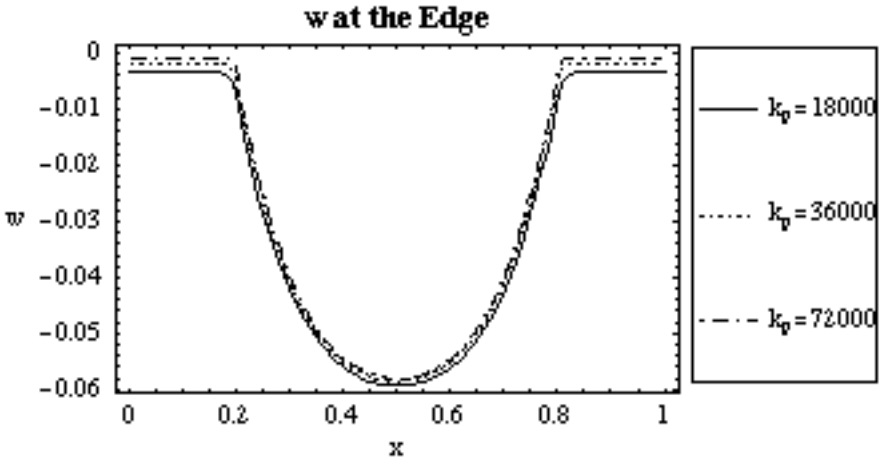

Figure 6.83 Plot of w along Edge for $k_P = 18000$, 36000, and 72000

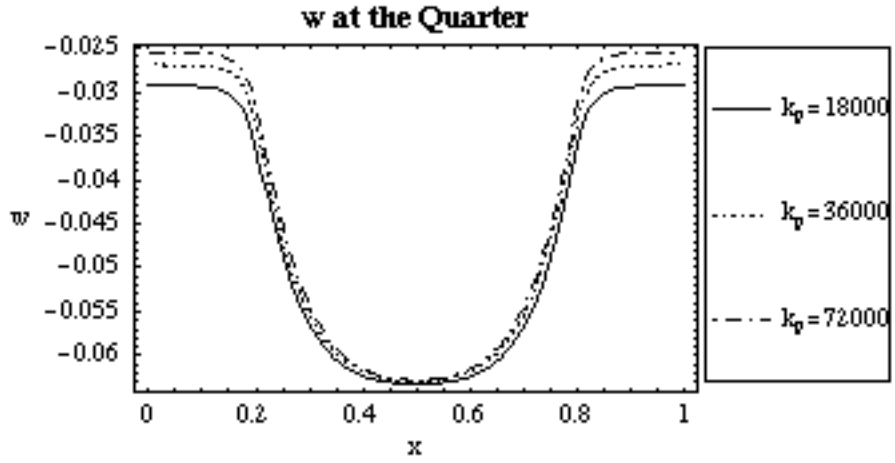

Figure 6.84 Plot of w along Quarter for k_P = 18000, 36000, and 72000

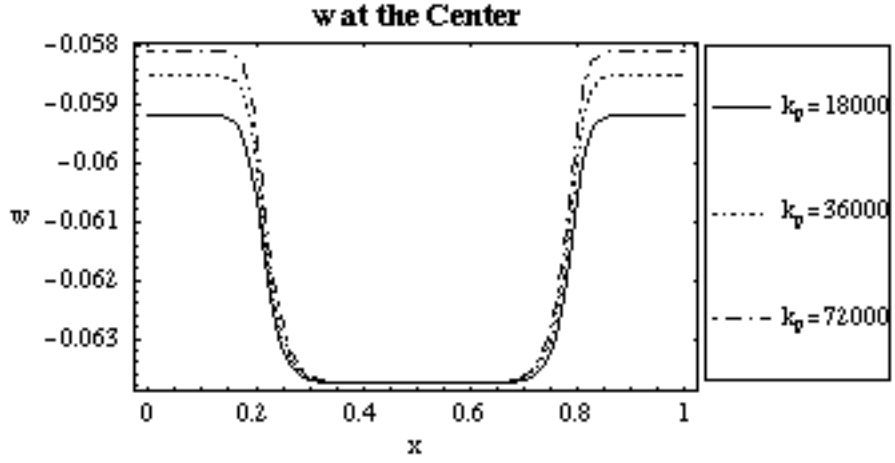

Figure 6.85 Plot of w along Center for k_P = 18000, 36000, and 72000

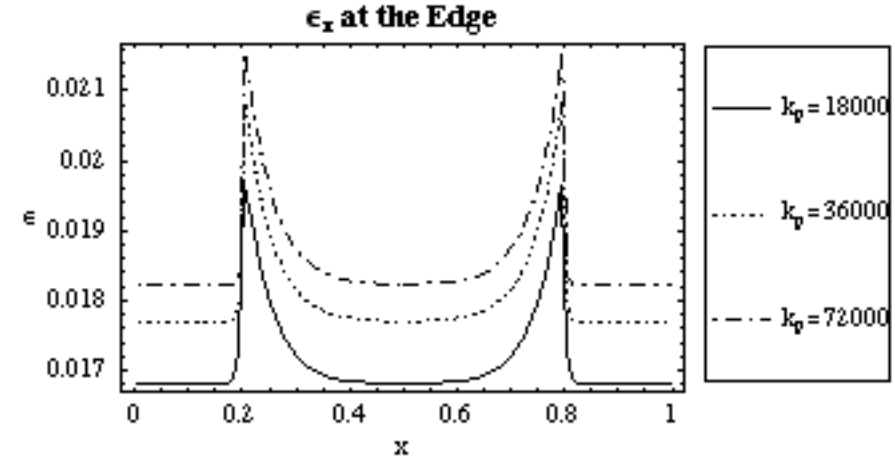

Figure 6.86 Plot of ε_x along Edge for k_P = 18000, 36000, and 72000

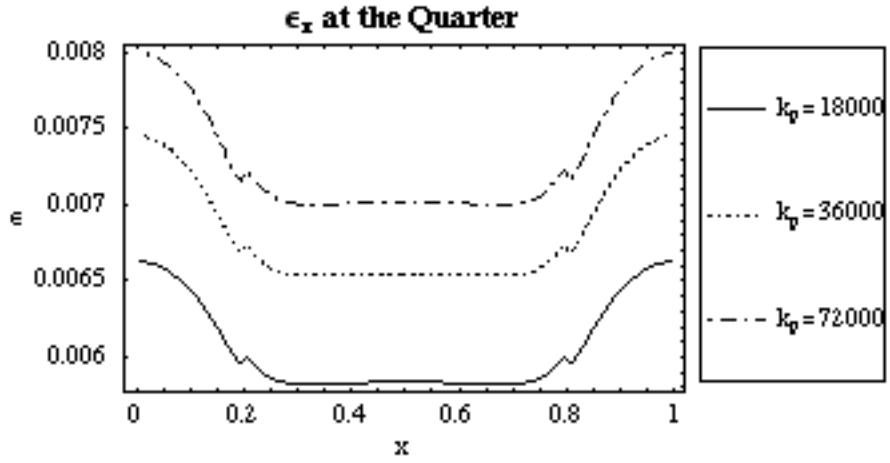

Figure 6.87 Plot of ε_x along Quarter for k_P = 18000, 36000, and 72000

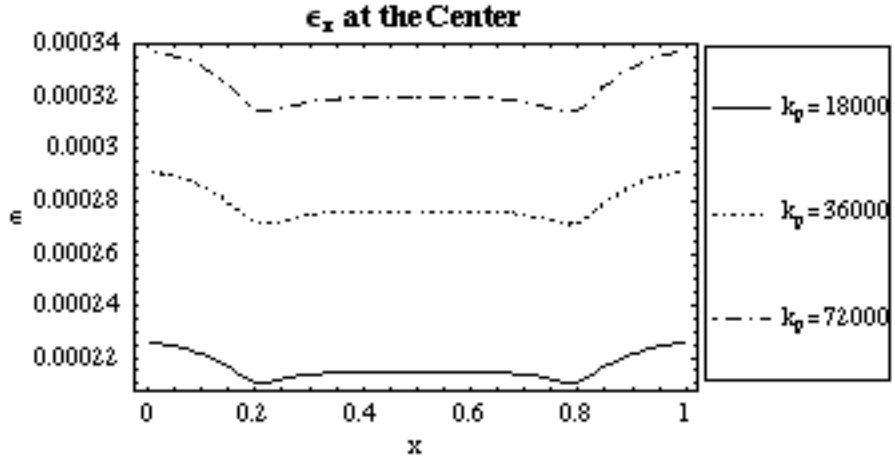

Figure 6.88 Plot of ε_x along Center for k_P = 18000, 36000, and 72000

As k_P increased in magnitude, the elevation of the geogrid over the pile rose. The elevation of the geogrid at all regions except the flat region in the center rose as k_P became large. The displaced shapes shown in the vertical displacement plots of the k_P case study were very similar to the plots of the q_P case study. The similarities were a result of the relationship shown in Equation 6.11. The displacement of the geogrid above the pile was equal to the result of this equation when k_P was twice the standard case value and q_P was half the standard case value, and when k_P was half the standard case value and q_P was twice the standard case value. The maximum strain was largest when k_P was largest because the differential settlement was also largest.

6.1.6 Variation of Non-Dimensional Parameter k_S

The parameter k_S is derived from Equation 2.2, and is the non-dimensional stiffness per unit area of the soil. The case study analyzed the model while varying the k_S value twice, once for $k_S = 98.6$ and another for $k_S = 395$. All the other parameter values were fixed and equal to the standard case values.

Plots of the case with $k_S = 98.6$ are shown in Figures 6.88-6.94, and plots of the case with $k_S = 395$ are shown in Figures 6.95-6.101. The results of each analysis were used to compare the effect of the non-dimensional parameter k_S on the vertical displacement and the strain. A solid line designates values for $k_S = 98.6$, a dashed line designates values for the standard case $k_S = 197$, and a dash-dot line designates values for $k_S = 395$. Figures 6.102-6.104 show the vertical displacement values and Figures 6.105-6.107 show the strain values at the edge, midway between the edge and centerline, and at the centerline of the grid, respectively.

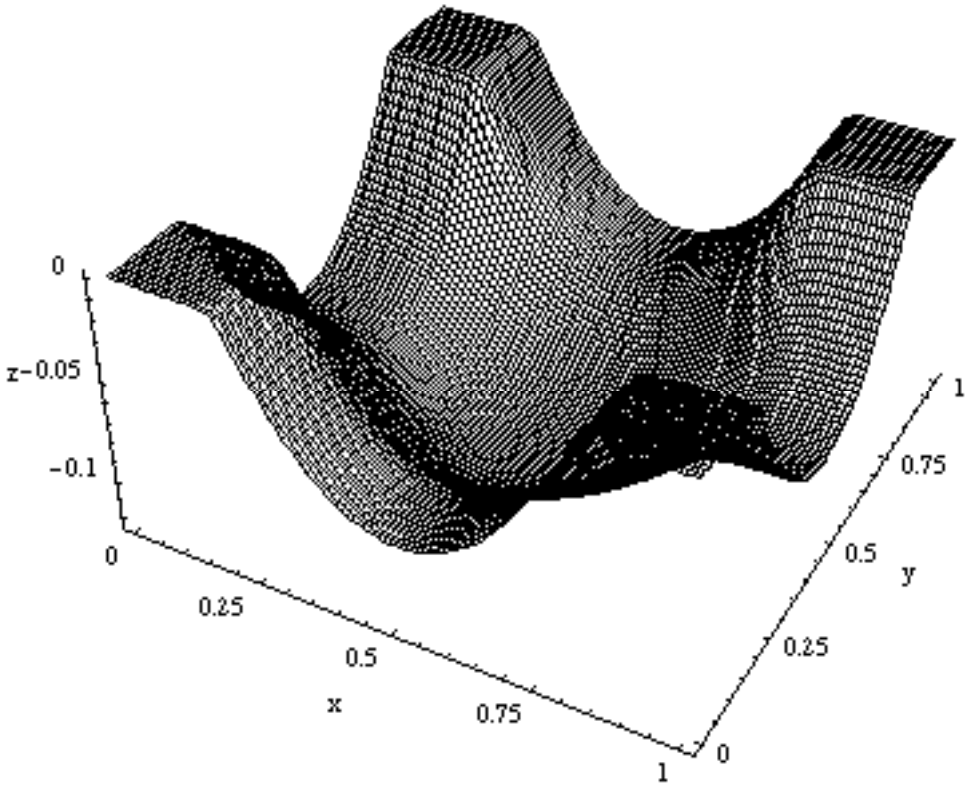

Figure 6.89 Three-Dimensional Plot for $k_S = 98.6$

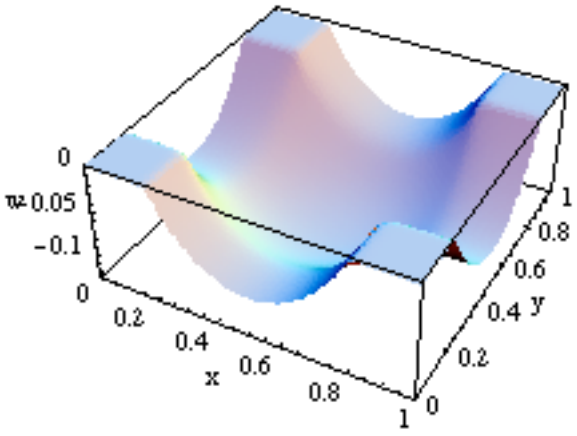

Figure 6.90 Three-Dimensional Plot of w for $k_S = 98.6$

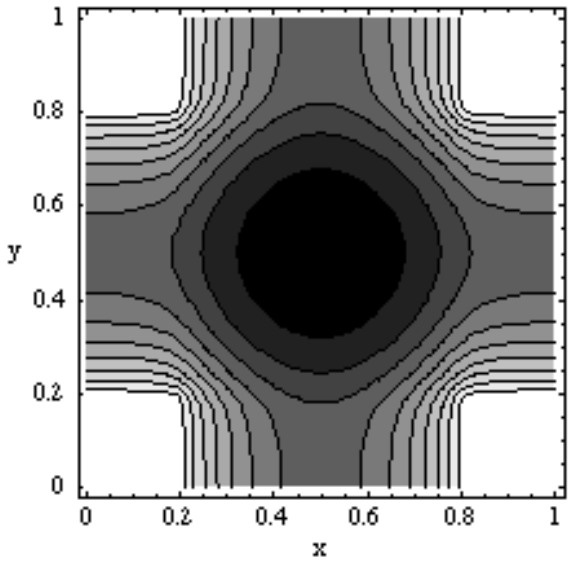

Figure 6.91 Contour Plot of w for $k_S = 98.6$

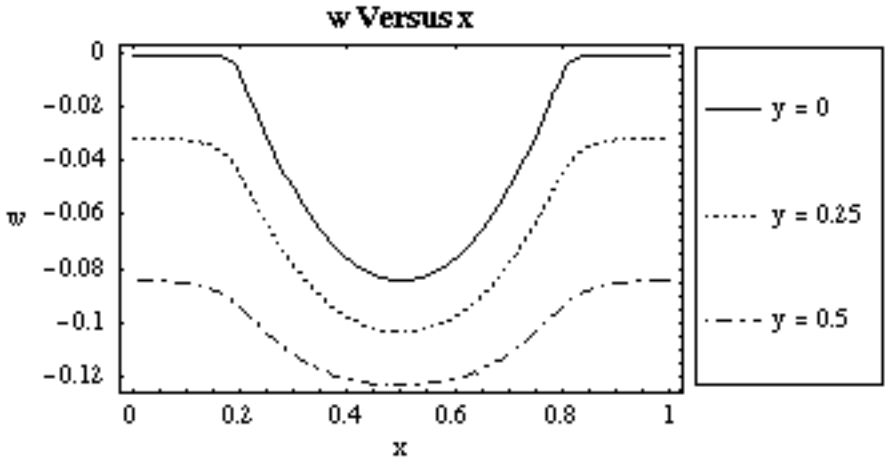

Figure 6.92 Plot of w vs. x along the Edge, Quarter and Center for $k_S = 98.6$

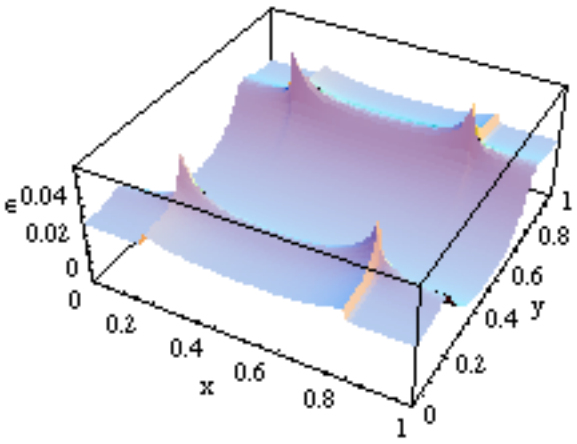

Figure 6.93 Three-Dimensional Plot of ε_x for $k_S = 98.6$

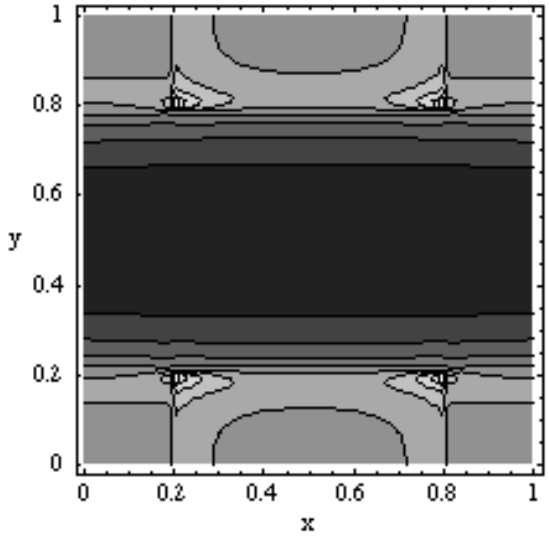

Figure 6.94 Contour Plot of ε_x for $k_S = 98.6$

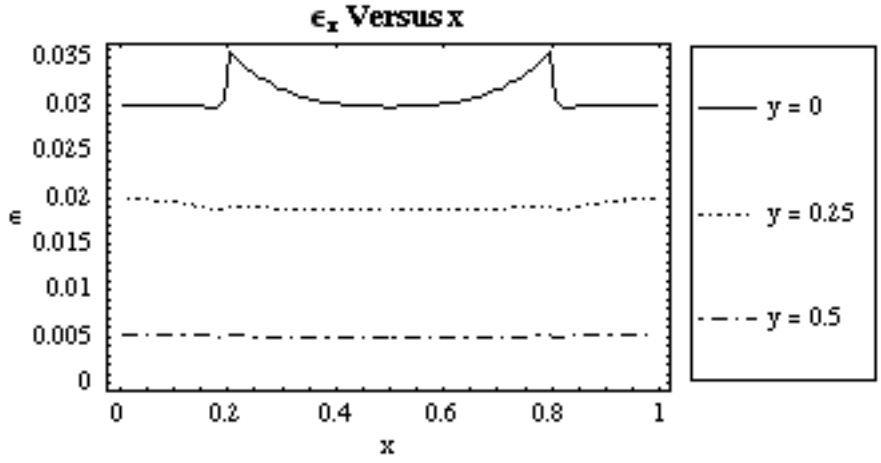

Figure 6.95 Plot of ε_x vs. x along the Edge, Quarter and Center for $k_S = 98.6$

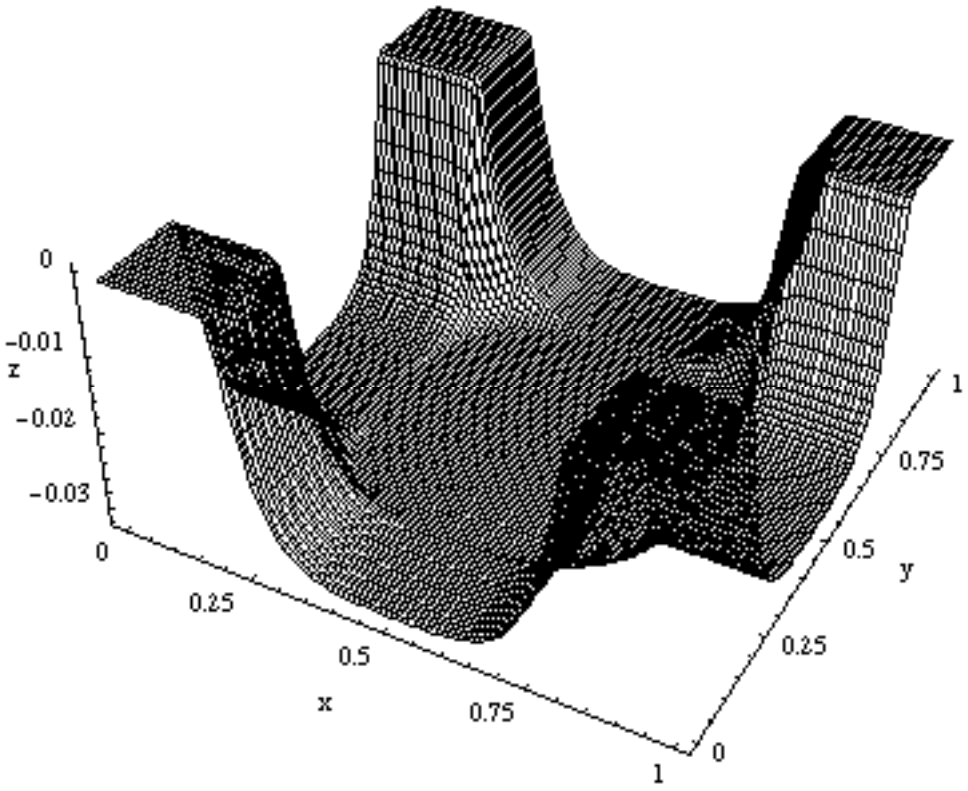

Figure 6.96 Three-Dimensional Plot for $k_S = 395$

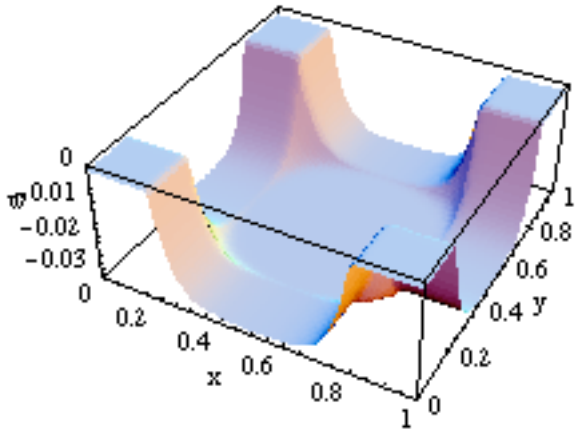

Figure 6.97 Three-Dimensional Plot of w for $k_S = 395$

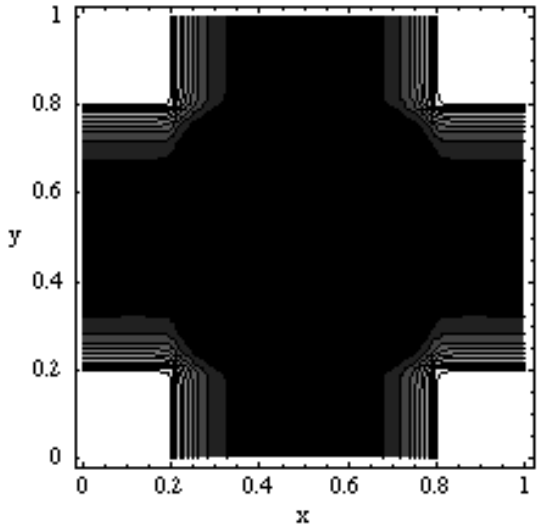

Figure 6.98 Contour Plot of w for $k_S = 395$

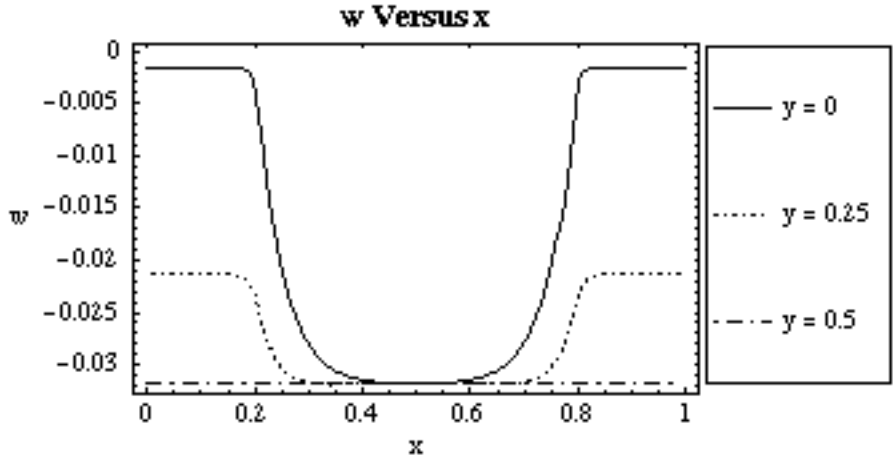

Figure 6.99 Plot of w vs. x along the Edge, Quarter and Center for $k_S = 395$

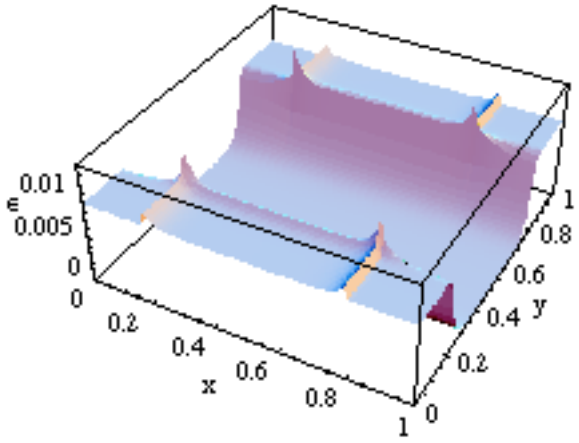

Figure 6.100 Three-Dimensional Plot of Strain for $k_S = 395$

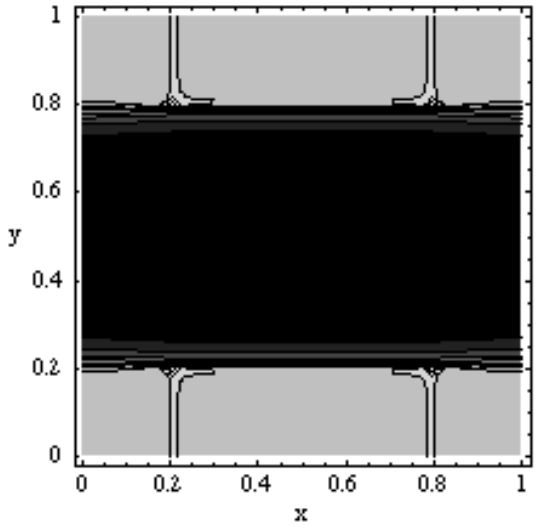

Figure 6.101 Contour Plot of Strain for $k_S = 395$

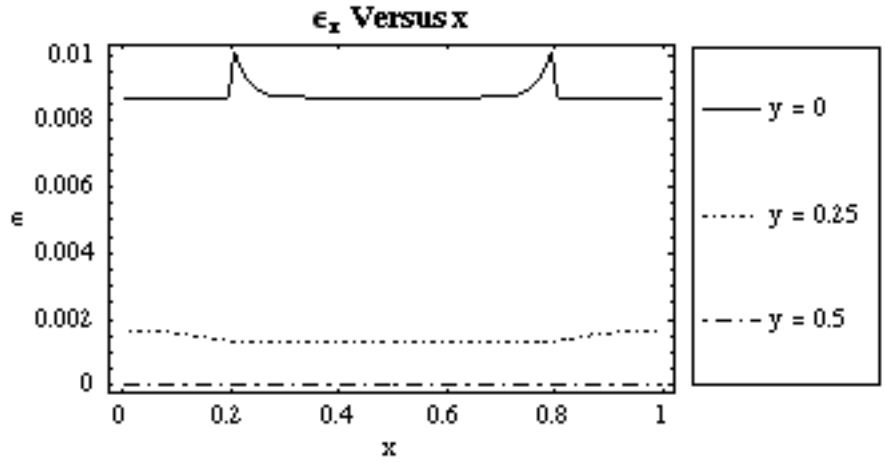

Figure 6.102 Plot of ε_x vs. x along the Edge, Quarter and Center for $k_S = 395$

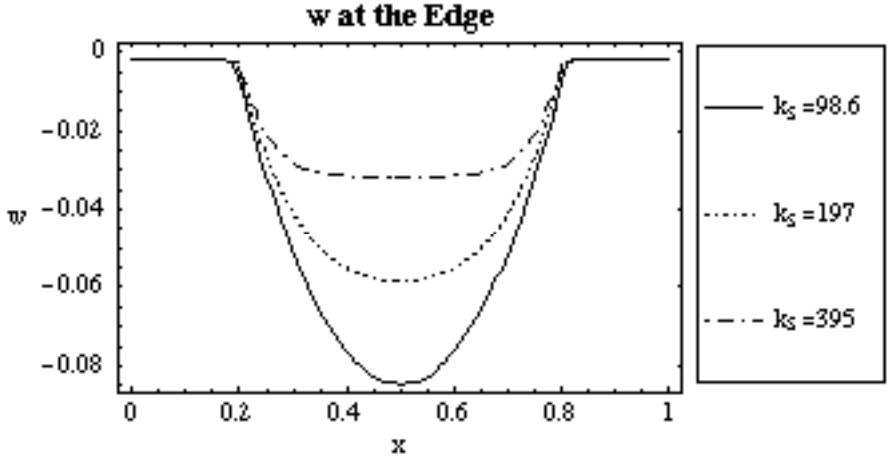

Figure 6.103 Plot of w along Edge for $k_S = 98.6$, 197, and 395

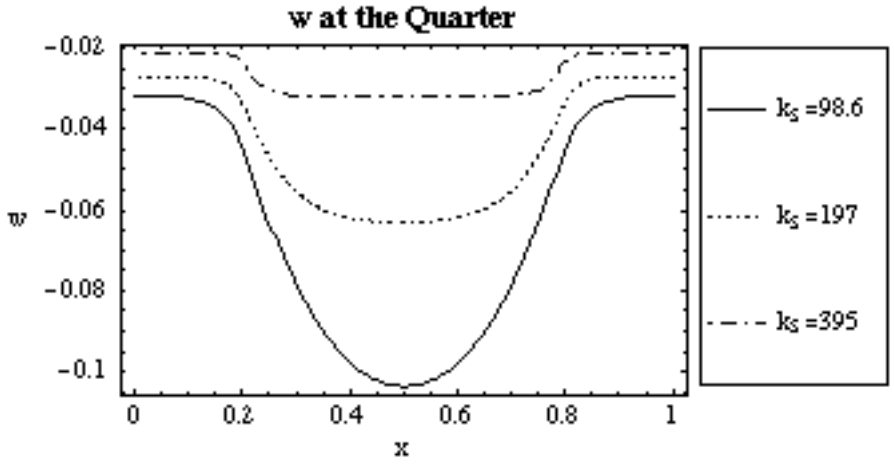

Figure 6.104 Plot of w along Quarter for k_S = 98.6, 197, and 395

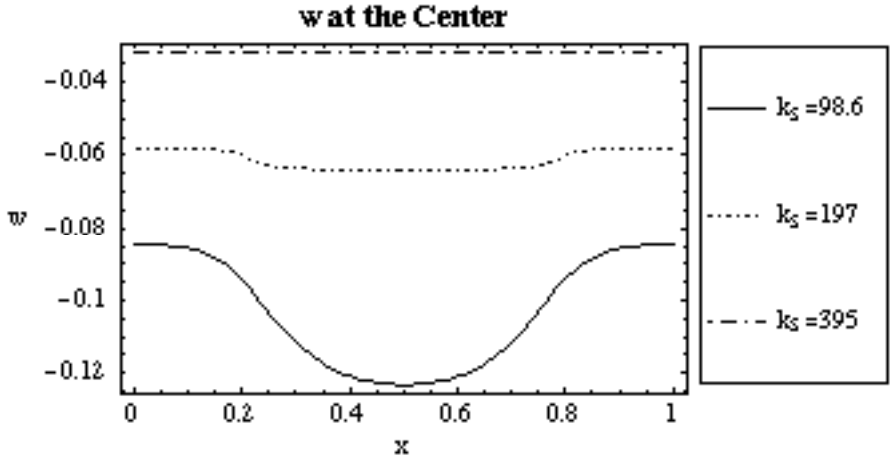

Figure 6.105 Plot of w along Center for k_S = 98.6, 197, and 395

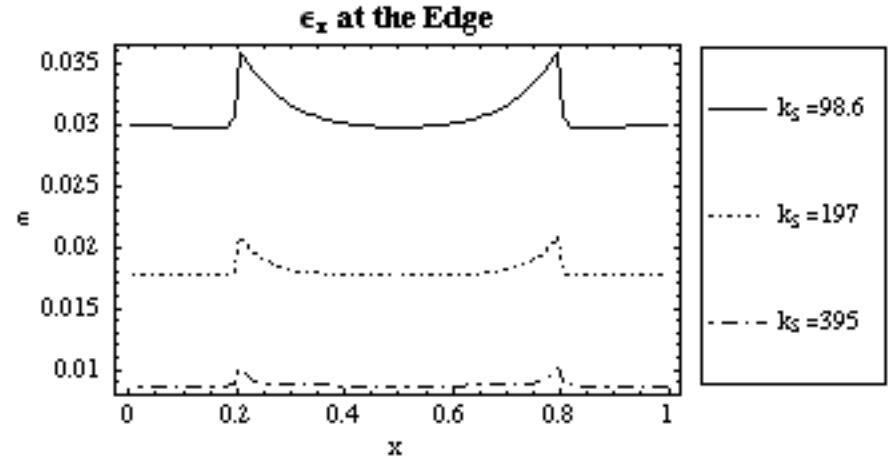

Figure 6.106 Plot of ε_x along Edge for k_S = 98.6, 197, and 395

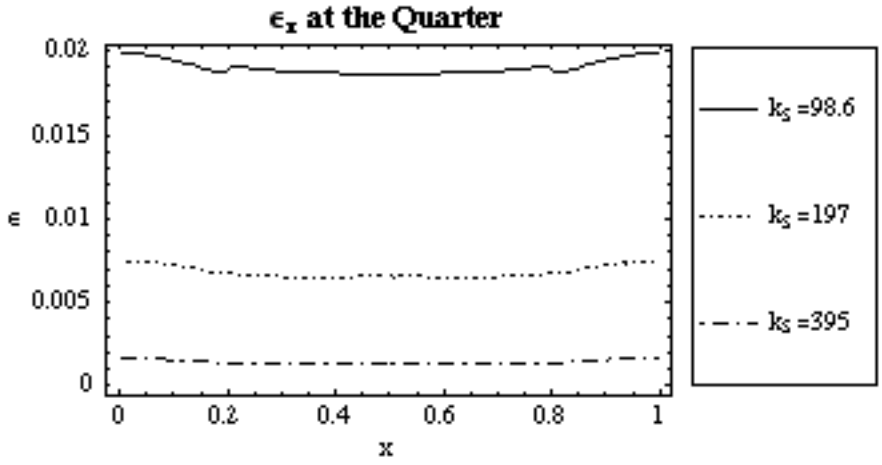

Figure 6.107 Plot of ε_x along Quarter for k_S = 98.6, 197, and 395

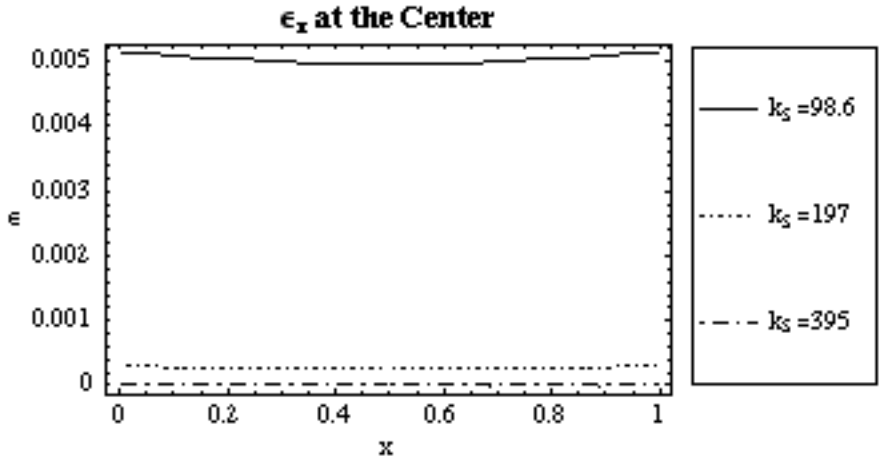

Figure 6.108 Plot of ε_x along Center for k_S = 98.6, 197, and 395

As k_S increased in magnitude, the unreinforced settlement decreased in magnitude, and as a result the displaced shape of the geogrid had a larger flat region. It was stated in the analysis of the previous case study that the k_P case study had a similar displacement profile to the q_P displacement profile when k_P was largest and the magnitude of q_P was smallest; the opposite was also true. This relationship did not hold for soil stiffness and soil pressure. After comparing Figures 6.58 and 6.91, it was evident that the slope was steeper when the magnitude of q_S was smallest than when k_S was largest. The potential energy caused by the load had a linear relationship with the vertical displacement, while the strain energy of the soil had a quadratic relationship with the vertical displacement, so it was intuitive that the displacement profiles were different. When k_S was smallest, there

was no flat region in the center because the magnitude of the maximum displacement was not as large as the unreinforced settlement calculated using Equation 6.11. The magnitude of the maximum displacement was larger if q_S was twice the value for the standard case than if k_S was half the value for the standard case.

The different displacement profiles were also evident when comparing strain values because the strain values were significantly larger when q_S was largest than when k_S was smallest. This observation was shown in Figures 6.59 and 6.92. As k_S decreased in magnitude, the strains increased because the differential settlement increased.

6.1.7 Variation of Non-Dimensional Parameter h

The non-dimensional length of a geogrid strand, h, was calculated using Equation 2.4. If h decreased, then the number of geogrid junctions increased, causing more strands to exist in the perpendicular direction. The overall stiffness of the geogrid increased as h decreased because there was more geogrid material. For this case study, the standard case values were used for all analyses with the exception of h, which was analyzed at 0.02 and 0.005.

Plots of the case with h = 0.02 are shown in Figures 6.108-6.114, and plots of the case with h = 0.005 are shown in Figures 6.115-6.121. The results of each analysis were used to compare the effect of the non-dimensional parameter h on the vertical displacement and the strain. A solid line designates values for h = 0.02, a dashed line designates values for the standard case h = 0.01, and a dash-dot line designates values for h = 0.005. Figures 6.122-6.124 show the vertical displacement values and Figures 6.125-6.127 show the strain values at the edge, midway between the edge and centerline, and at the centerline of the grid, respectively.

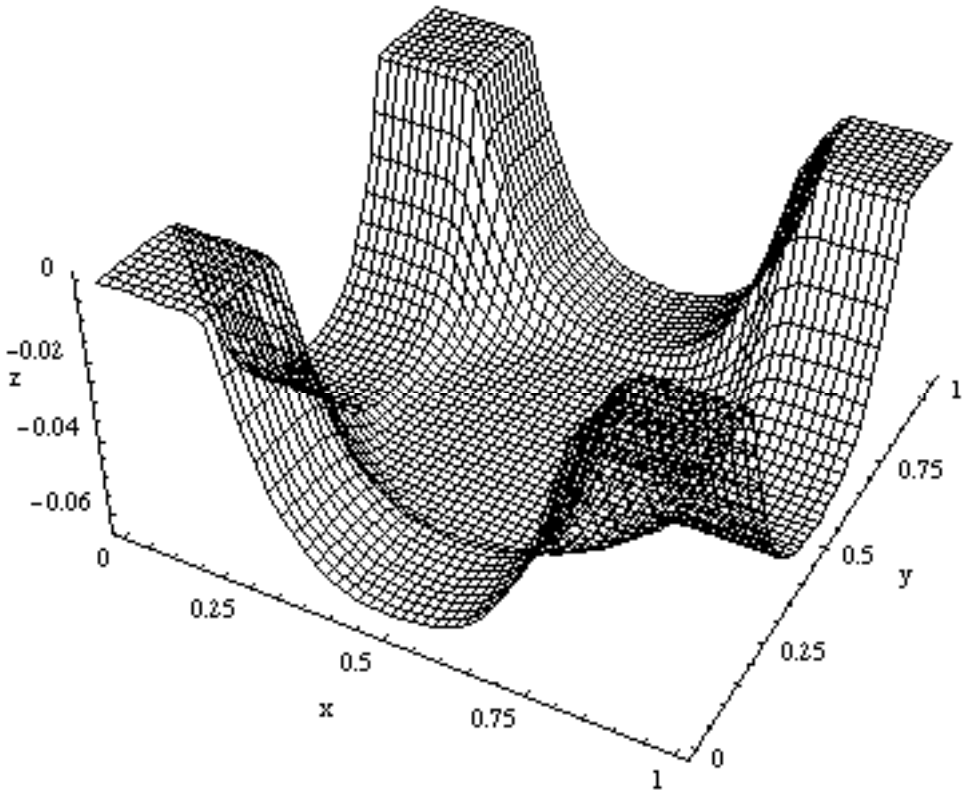

Figure 6.109 Three-Dimensional Plot for h = 0.02

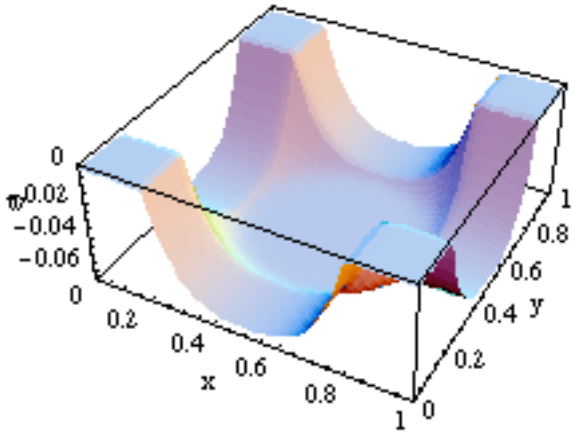

Figure 6.110 Three-Dimensional Plot of w for h = 0.02

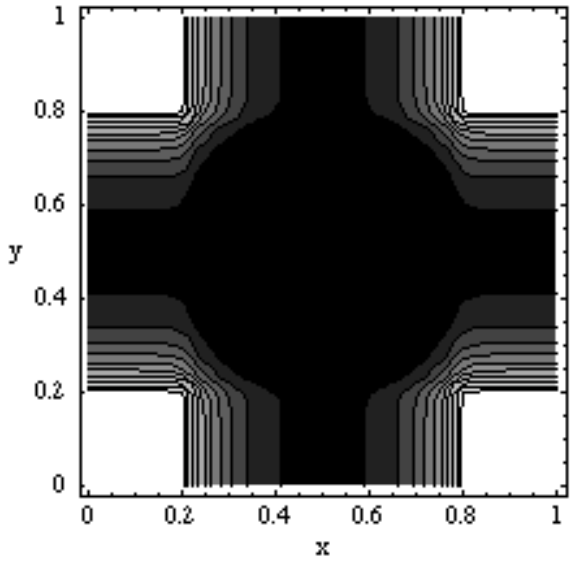

Figure 6.111 Contour Plot of w for h = 0.02

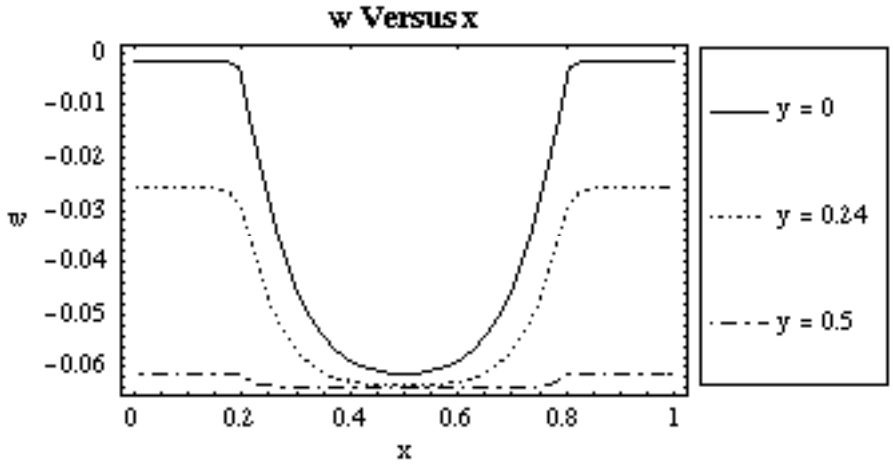

Figure 6.112 Plot of w vs. x along the Edge, Quarter and Center for h = 0.02

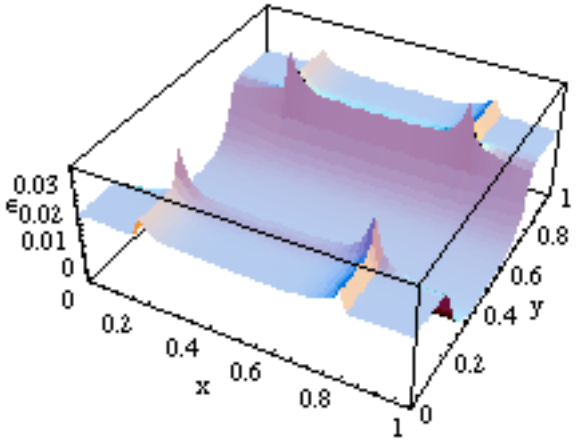

Figure 6.113 Three-Dimensional Plot of ε_x for h = 0.02

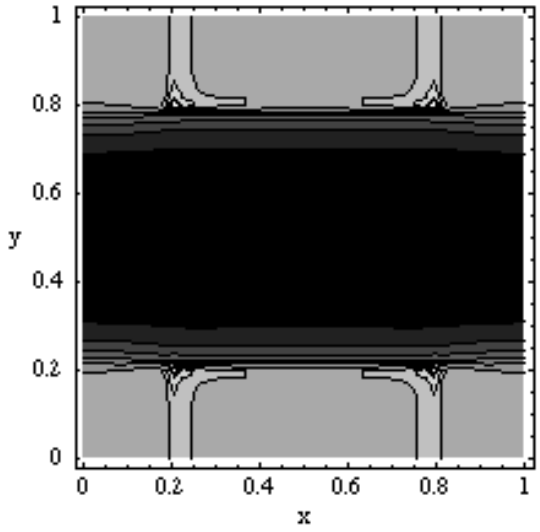

Figure 6.114 Contour Plot of ε_x for h = 0.02

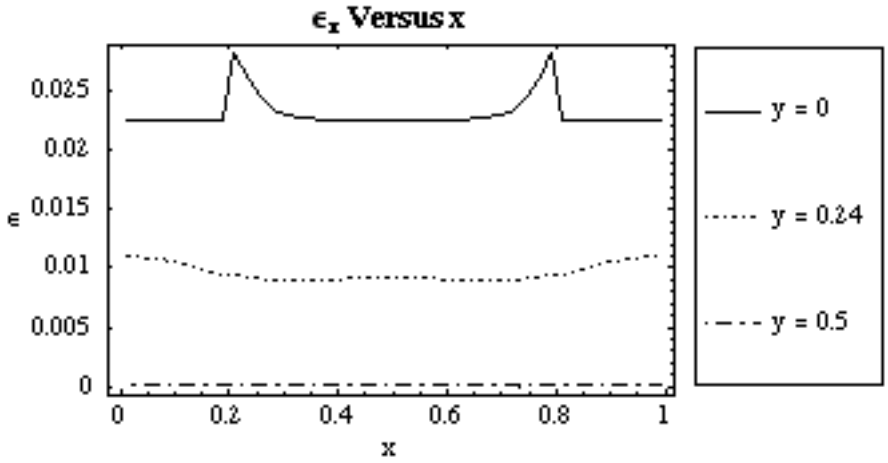

Figure 6.115 Plot of ε_x vs. x along the Edge, Quarter and Center for h = 0.02

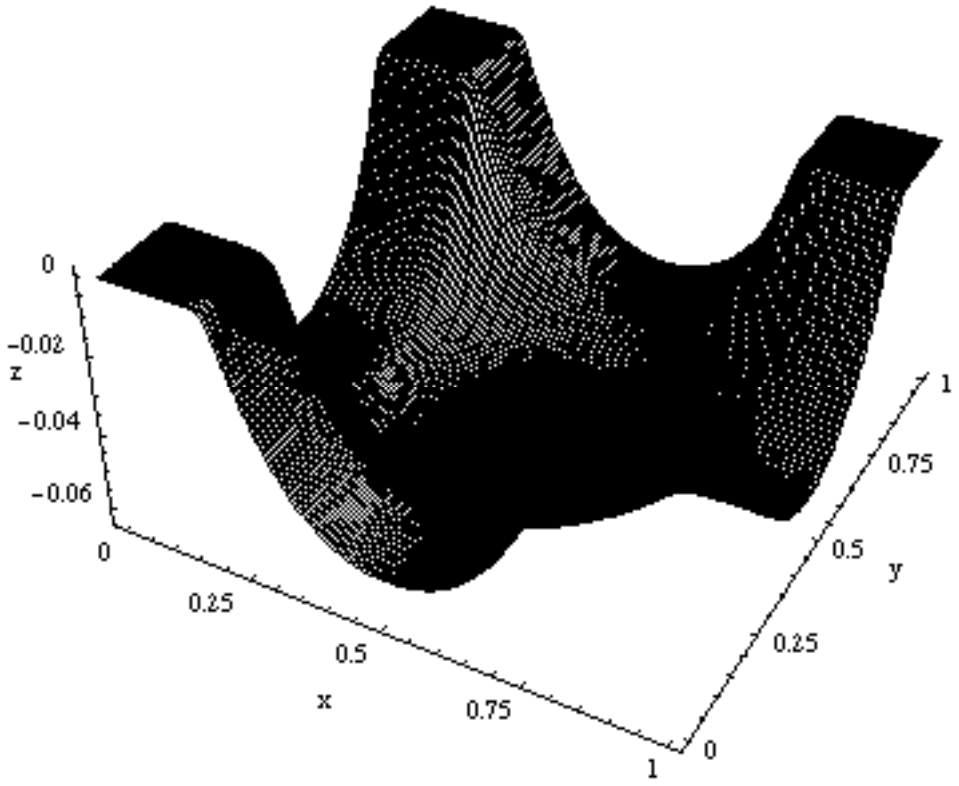

Figure 6.116 Three-Dimensional Plot for h = 0.005

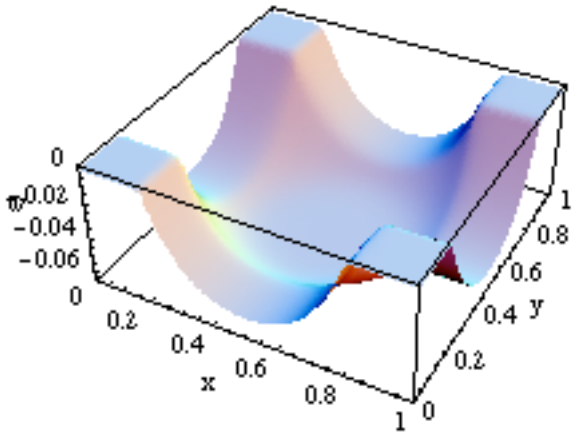

Figure 6.117 Three-Dimensional Plot of w for h = 0.005

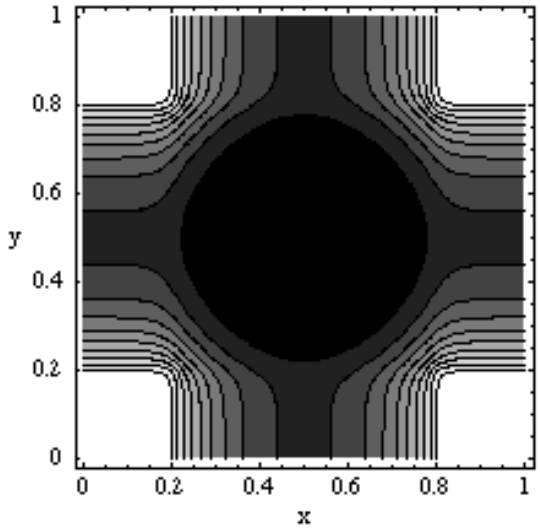

Figure 6.118 Contour Plot of w for h = 0.005

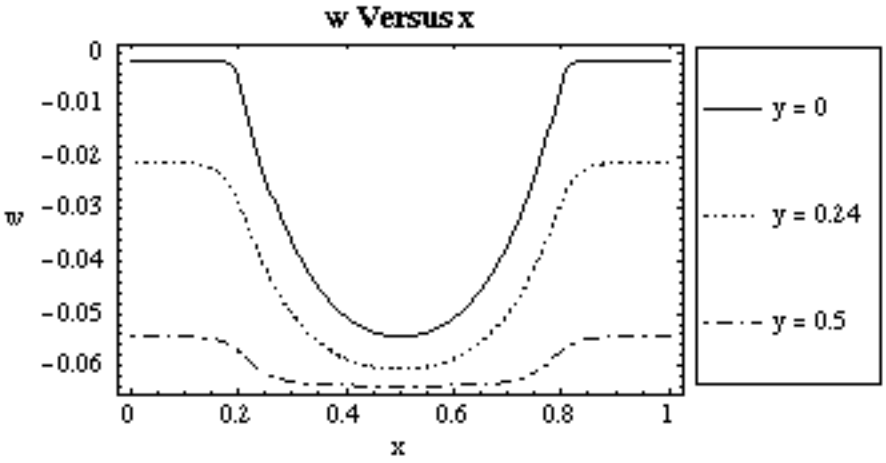

Figure 6.119 Plot of w vs. x along the Edge, Quarter and Center for h = 0.005

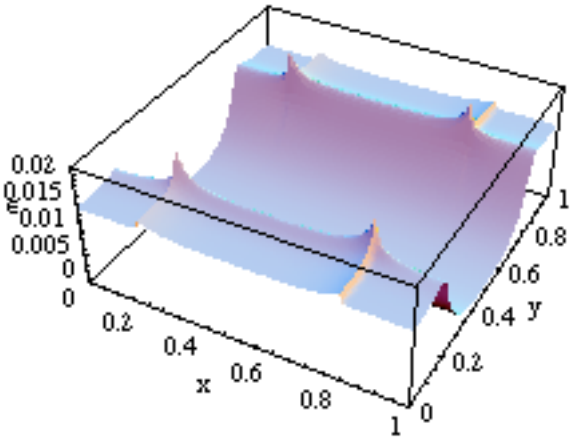

Figure 6.120 Three-Dimensional Plot of ε_x for h = 0.005

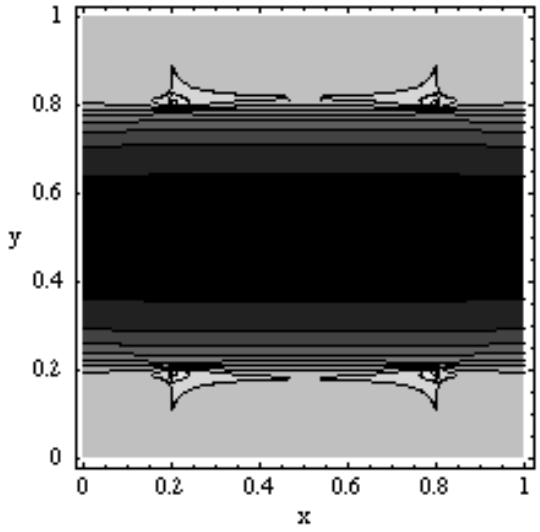

Figure 6.121 Contour Plot of ε_x for h = 0.005

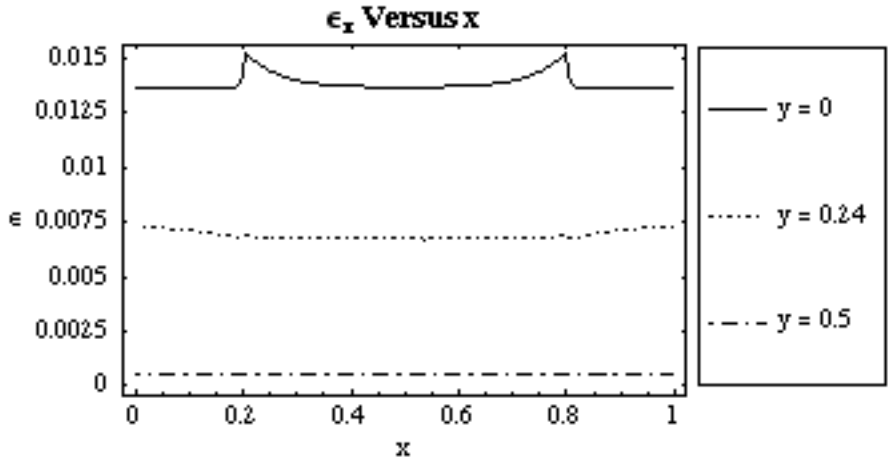

Figure 6.122 Plot of ε_x vs. x along the Edge, Quarter and Center for h = 0.005

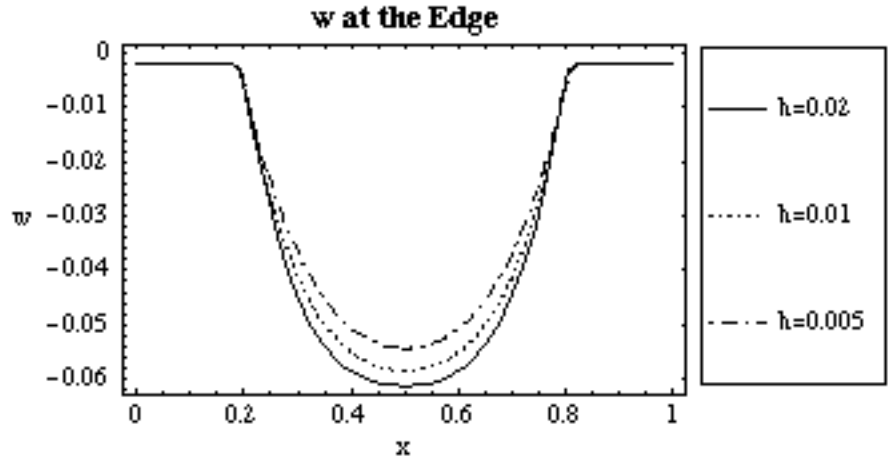

Figure 6.123 Plot of w along Edge for h = 0.02, 0.01, 0.005

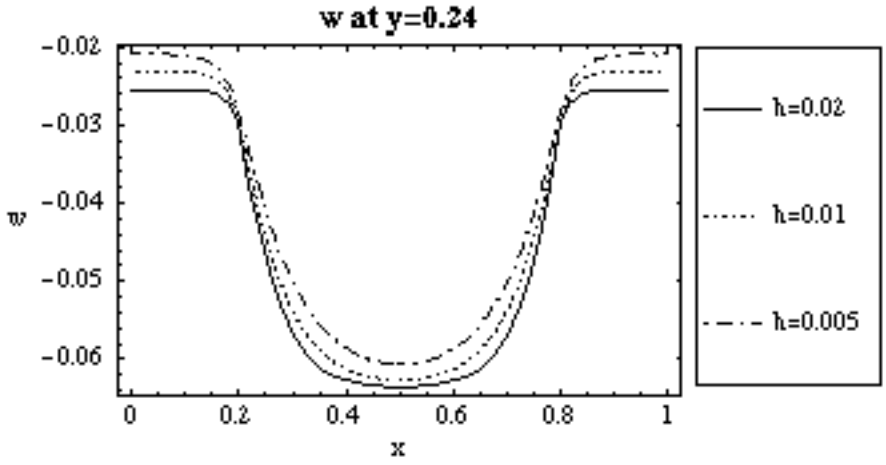

Figure 6.124 Plot of w along Quarter for h = 0.02, 0.01, 0.005

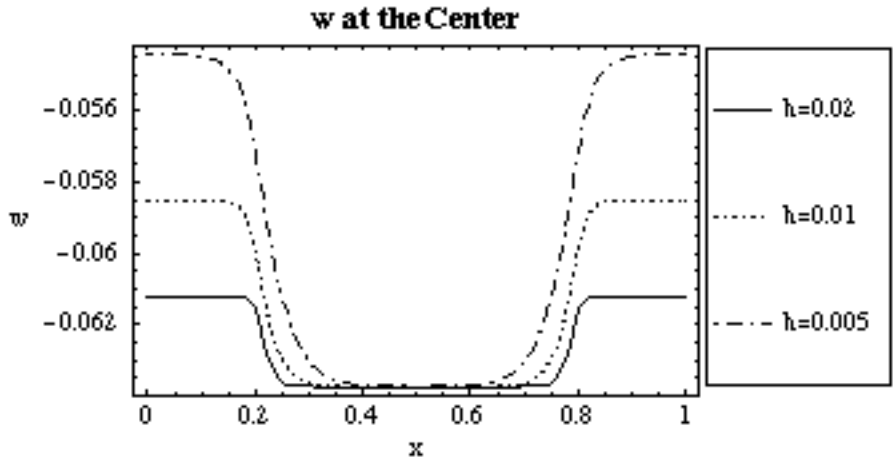

Figure 6.125 Plot of w along Center for h = 0.02, 0.01, 0.005

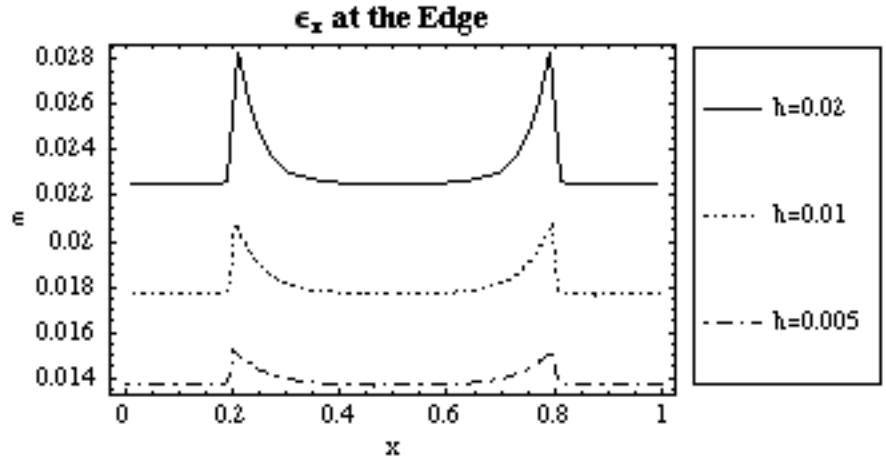

Figure 6.126 Plot of ε_x along Edge for h = 0.02, 0.01, 0.005

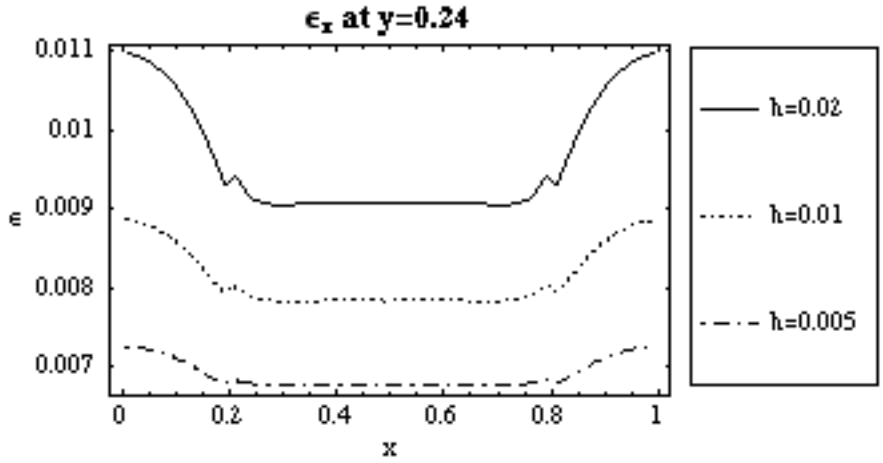

Figure 6.127 Plot of ε_x along Quarter for h = 0.02, 0.01, 0.005

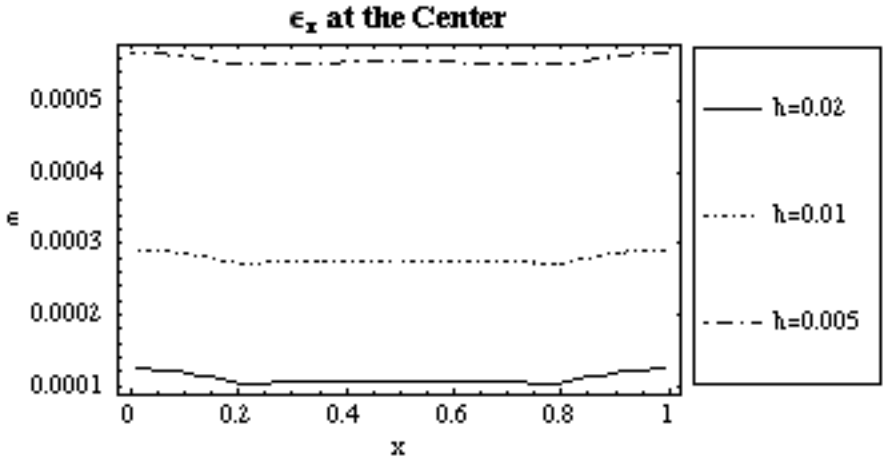

Figure 6.128 Plot of ε_x along Center for h = 0.02, 0.01, 0.005

When h = 0.02, the displacement profile was significantly deeper than the other cases at all of the cell regions except the flat region. As h increased, the area of the flat region in the center of the grid increased. Figure 6.124 shows this trend; when h = 0.005 the geogrid had the smallest flat region. The displaced shape of the geogrid was dependent on the strand length; if the strand length was reduced and the stiffness of the strand remained constant, then the overall stiffness of the geogrid increased and the overall vertical displacement was less. In this case study, the slope of the geogrid's deformed shape decreased and less of the geogrid reached the unreinforced settlement if the strand length was reduced. Figures 6.122 and 6.123 clearly show that when h was larger, the geogrid had a steeper slope.

The overall strain was larger when h was larger. The stiffness of the geogrid decreased as h increased, which caused the change in slope of the geogrid's deformed shape to increase and the strain to increase. In other words, if there were fewer strands to resist the load, then the soft soil underneath the geogrid had to support more load. The three-dimensional plots of the strain shown in Figures 6.112, 6.119, and 6.5 indicate that the spike in strain that occurred at the corner of the pile was amplified when h was greatest. The strain in a geogrid strand at the centerline decreased as h increased. Figure 6.127 shows that the largest strain occurred at the centerline when h = 0.005 and the smallest occurred when h = 0.02. The geogrid was flat for a larger region as h increased, and the flat region reduced the strain in the strands at the centerline.

6.1.8 Anisotropic Geogrid with a Variation of Non-Dimensional Parameter h_X

A model was developed that allowed the strand length parallel to the x-axis to be modified independently of the strand length parallel to the y-axis. In this model the parameter h_X was the non-dimensional strand length of strands parallel to the x-axis. All other non-dimensional standard case values did not vary throughout this case study. The non-dimensional strand length of strands parallel to the y-axis, h_Y, was also kept constant and equal to the standard case value h = 0.01. For this case study, h_X was varied twice by doubling h_Y so that $h_X = 0.02$ and by halving h_Y so that $h_X = 0.005$.

Plots of the case with $h_X = 0.02$ are shown in Figures 6.128-138, and plots of the case with $h_X = 0.005$ are shown in Figures 6.139-149. The results of each analysis were used to compare the effect of the non-dimensional parameter h_X on the vertical displacement and the strain. A solid line designates values for $h_X = 0.02$, a dashed line designates values for the standard case $h_X = 0.01$, and a dash-dot line designates values for $h_X = 0.005$. Figures 6.150-6.152 show the vertical displacement values along the x-axis and Figures 6.153-6.155 show the strain values parallel to the x-axis at the edge, midway between the edge and centerline, and at the centerline of the grid, respectively. Figures 6.156-6.158 show the vertical displacement values along the y-axis and Figures 6.159-

6.161 show the strain values parallel to the y-axis at the edge, midway between the edge and centerline, and at the centerline of the grid, respectively.

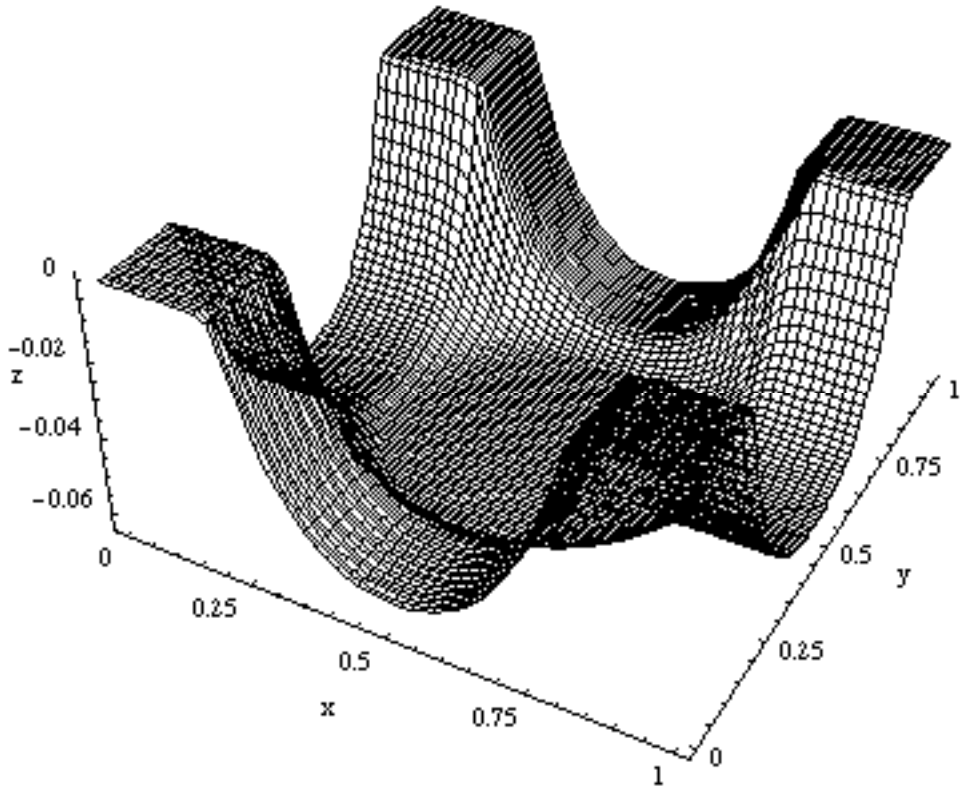

Figure 6.129 Three-Dimensional Plot for $h_X = 0.02$

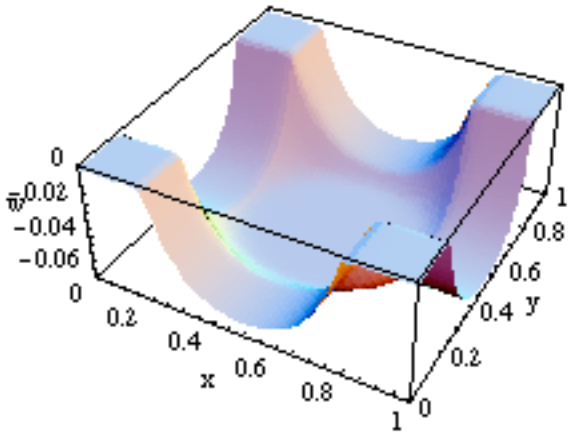

Figure 6.130 Three-Dimensional Plot of w for $h_X = 0.02$

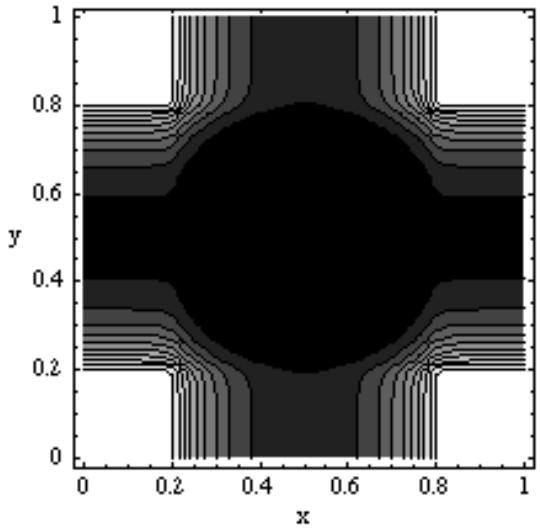

Figure 6.131 Contour Plot of w for $h_X = 0.02$

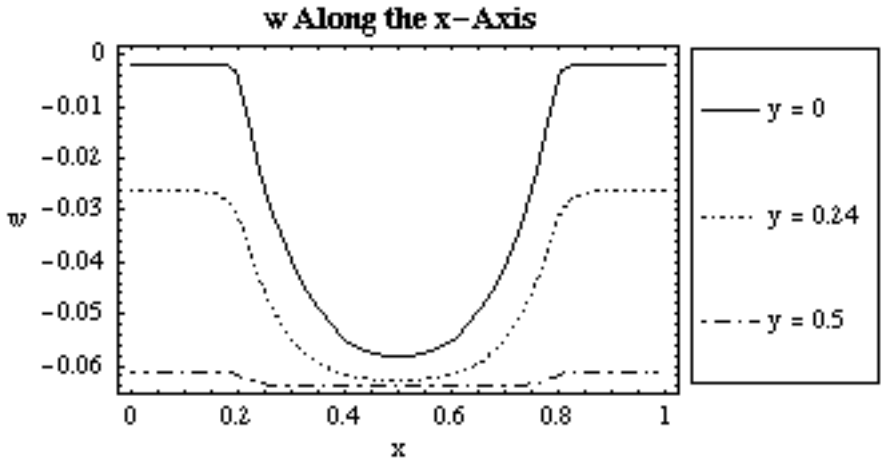

Figure 6.132 Plot of w vs. x along the Edge, Quarter and Center for $h_X = 0.02$

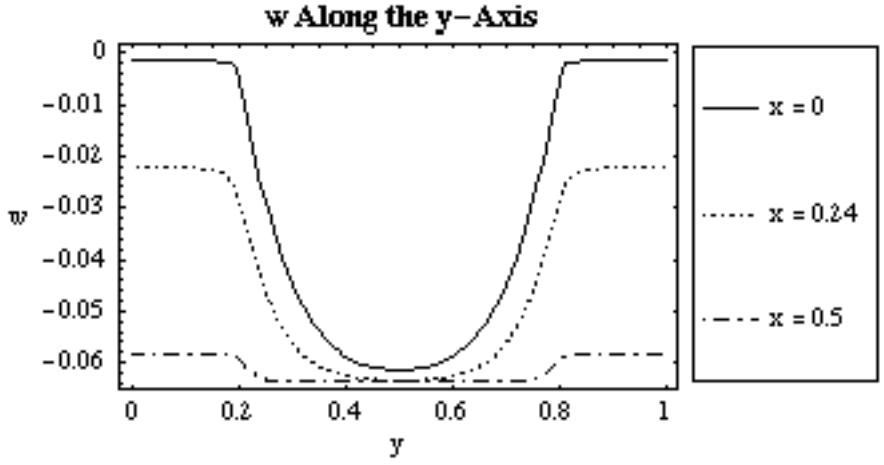

Figure 6.133 Plot of w vs. y along the Edge, Quarter and Center for $h_X = 0.02$

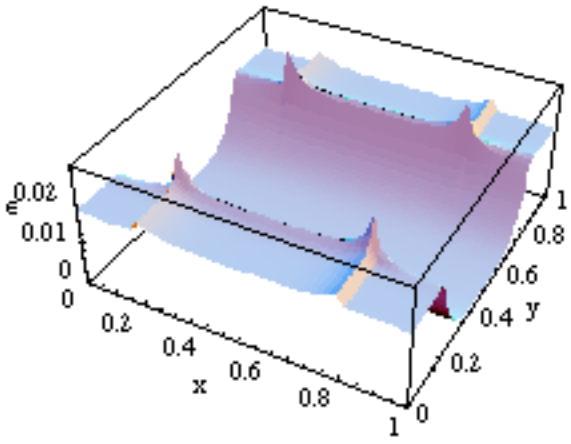

Figure 6.134 Three-Dimensional Plot of ε_x Parallel to x-axis for $h_X = 0.02$

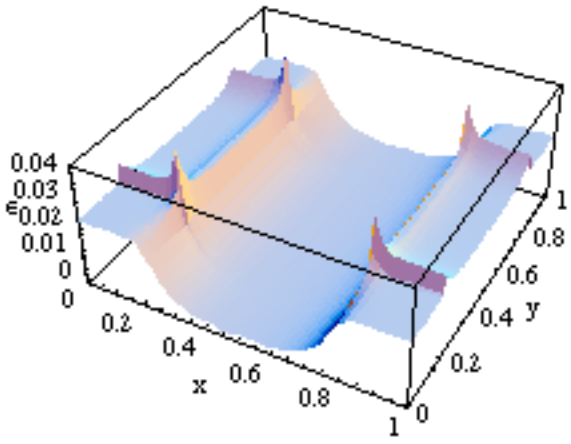

Figure 6.135 Three-Dimensional Plot of ε_x Parallel to y-axis for $h_X = 0.02$

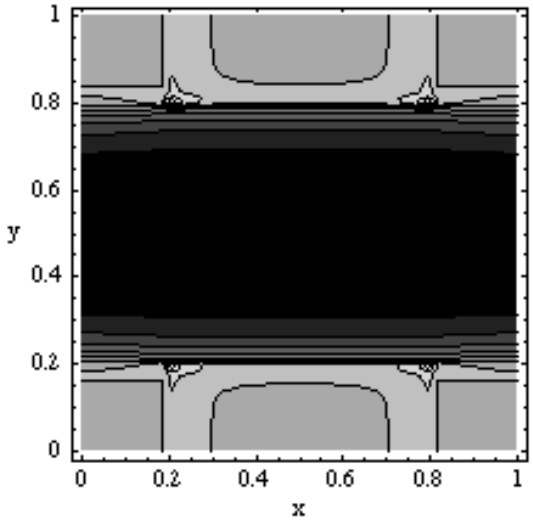

Figure 6.136 Contour Plot of ε_x Parallel to x-axis for $h_X = 0.02$

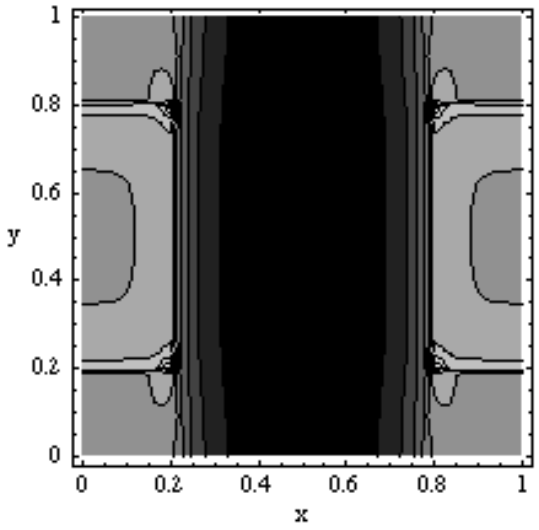

Figure 6.137 Contour Plot of ε_x Parallel to y-axis for $h_X = 0.02$

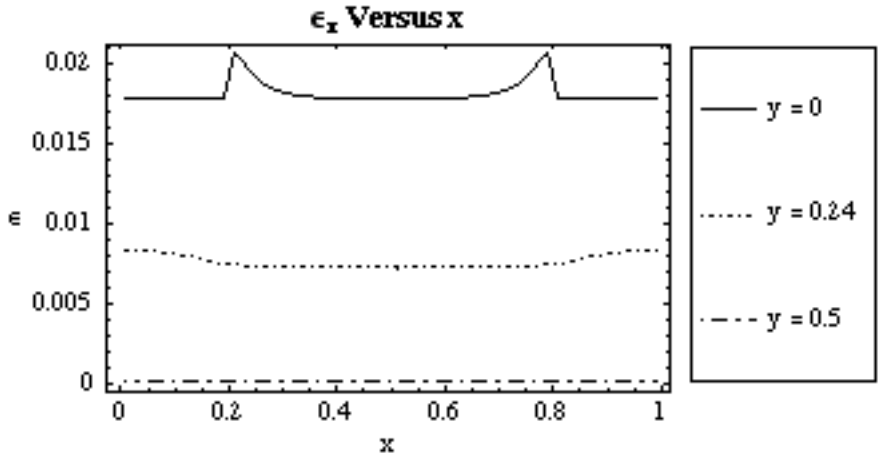

Figure 6.138 Plot of ε_x along the Edge, Quarter and Center for $h_X = 0.02$

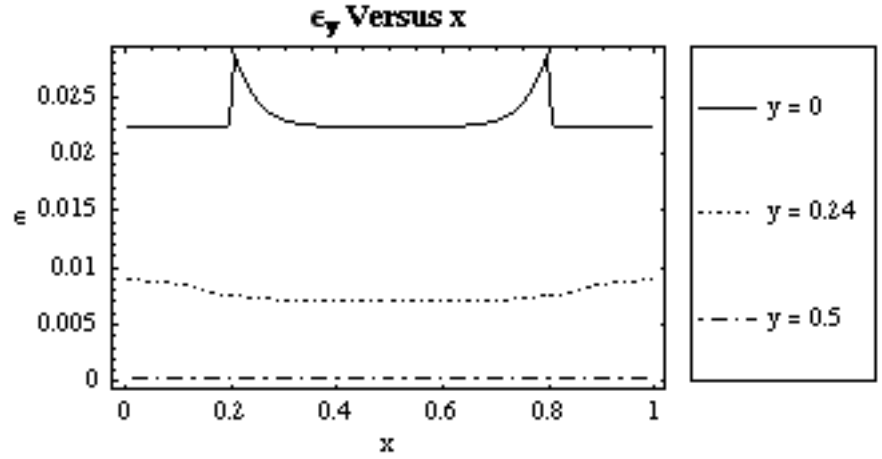

Figure 6.139 Plot of ε_y along the Edge, Quarter and Center for $h_X = 0.02$

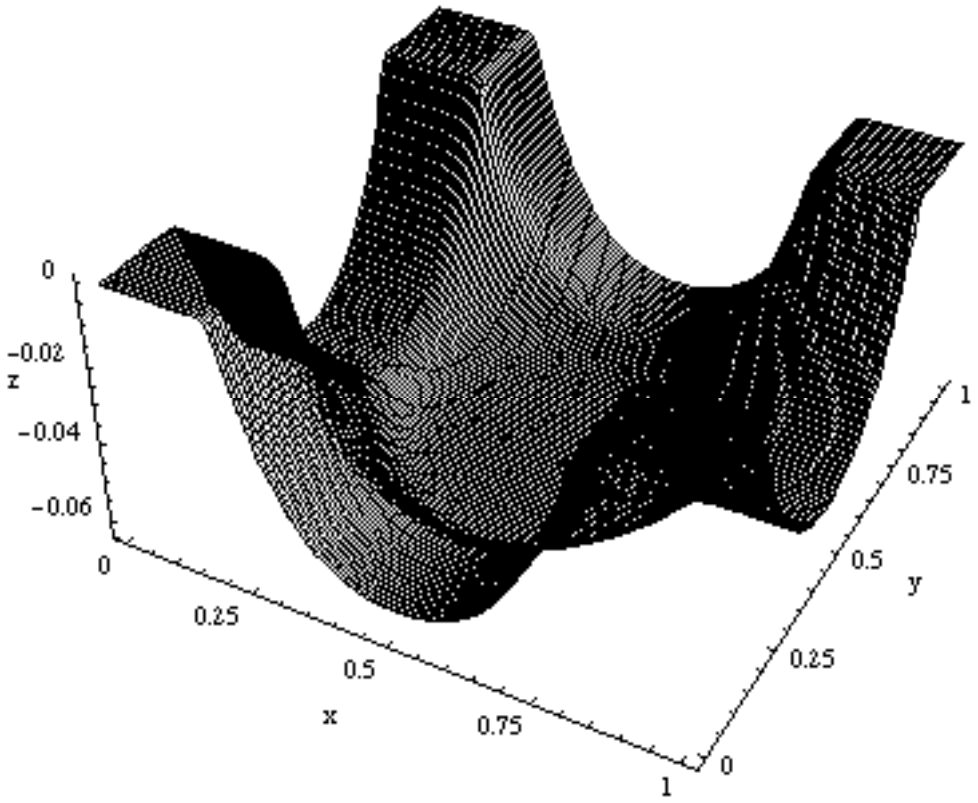

Figure 6.140 Three-Dimensional Plot for $h_X = 0.005$

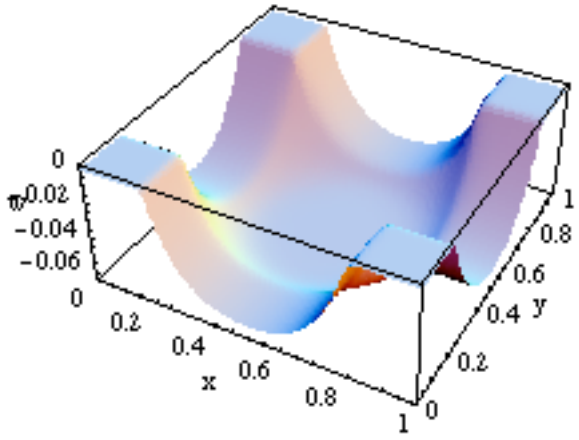

Figure 6.141 Three-Dimensional Plot of w for $h_X = 0.005$

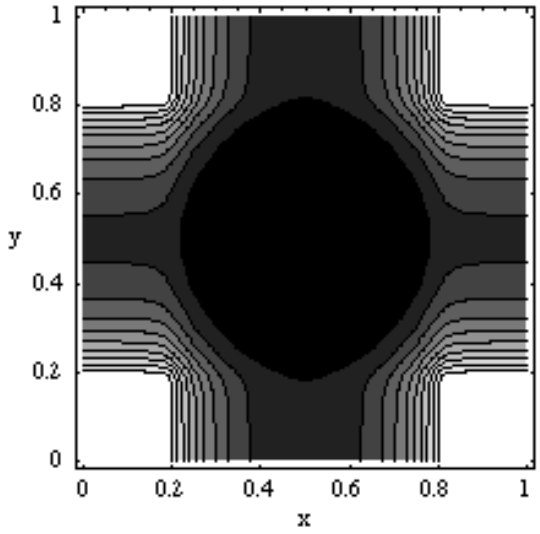

Figure 6.142 Contour Plot of w for $h_X = 0.005$

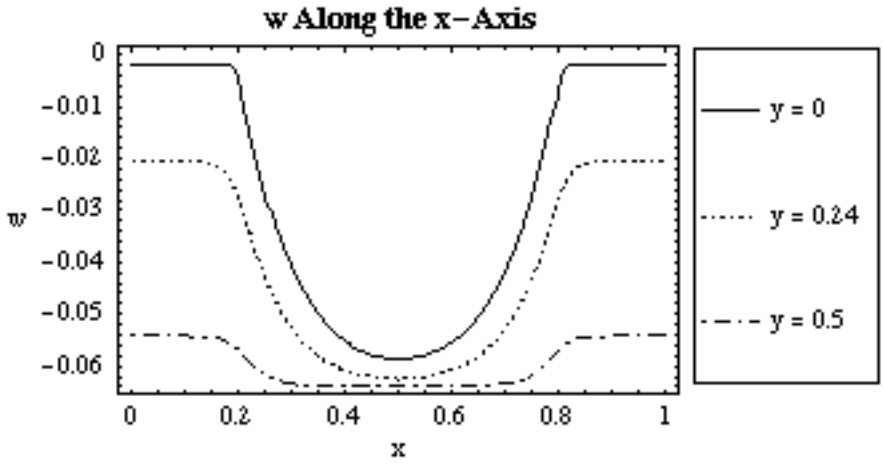

Figure 6.143 Plot of w vs. x along the Edge, Quarter and Center for $h_X = 0.005$

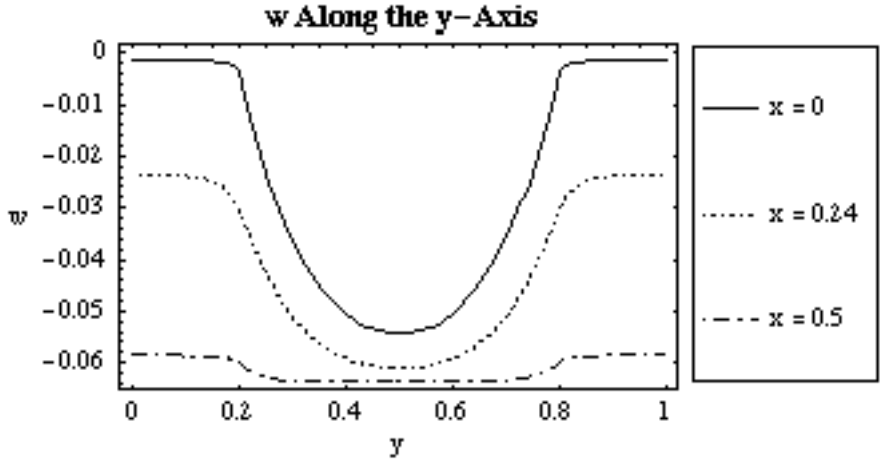

Figure 6.144 Plot of w vs. y along the Edge, Quarter and Center for $h_X = 0.005$

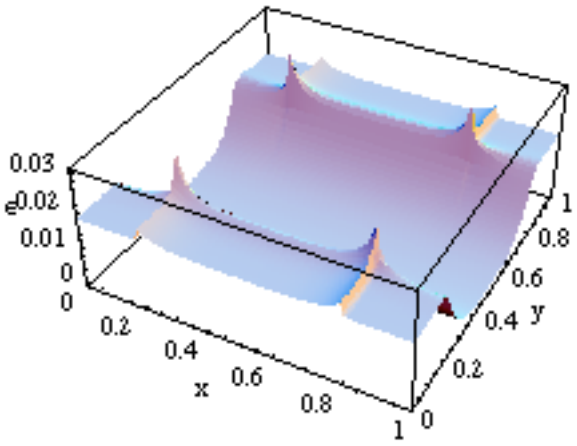

Figure 6.145 Three-Dimensional Plot of ε_x Parallel to x-axis for $h_X = 0.005$

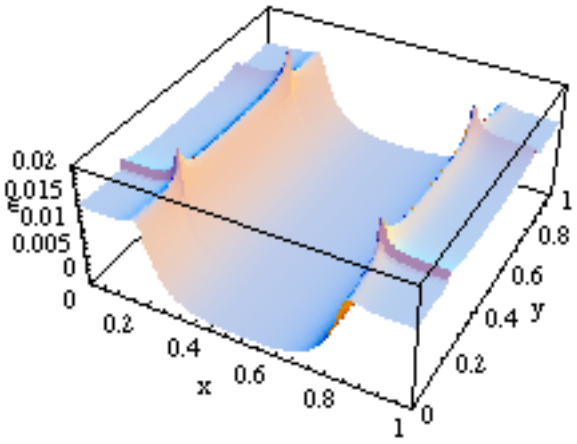

Figure 6.146 Three-Dimensional Plot of ε_x Parallel to y-axis for $h_X = 0.005$

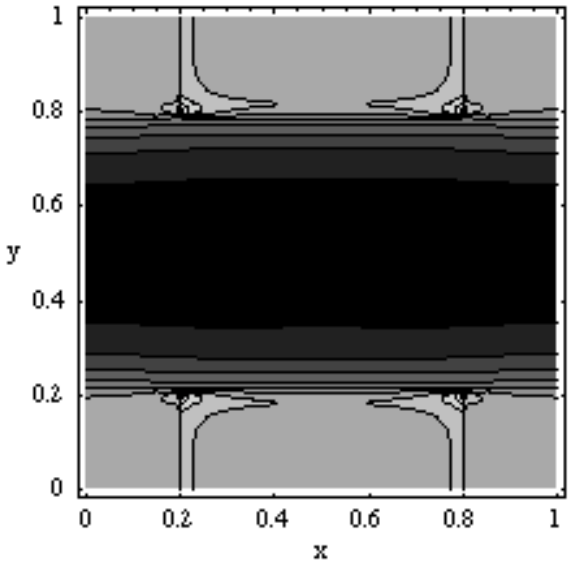

Figure 6.147 Contour Plot of ε_x Parallel to x-axis for $h_X = 0.005$

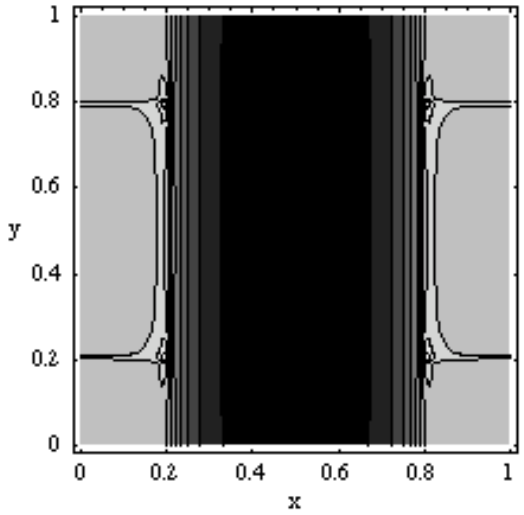

Figure 6.148 Contour Plot of ε_x Parallel to y-axis for $h_X = 0.005$

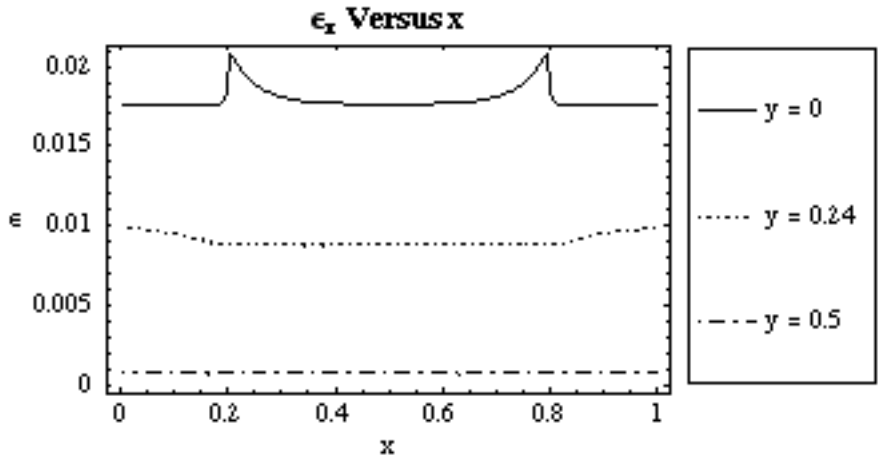

Figure 6.149 Plot of ε_x along the Edge, Quarter and Center for $h_X = 0.005$

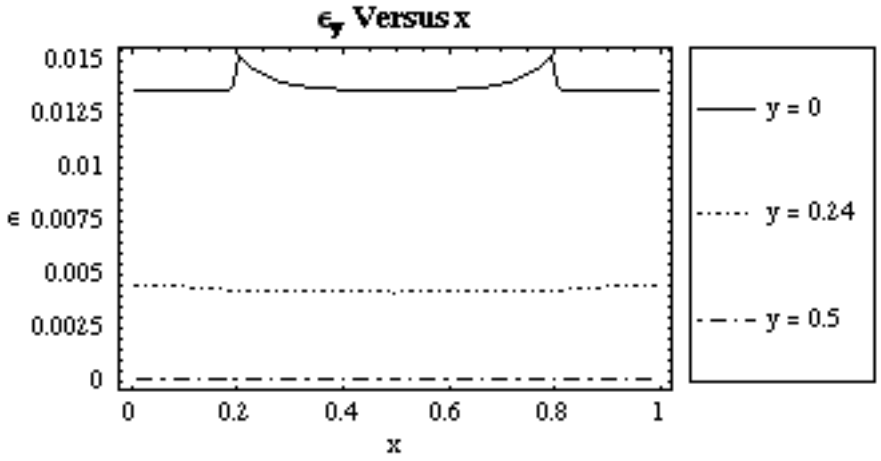

Figure 6.150 Plot of ε_y along the Edge, Quarter and Center for $h_X = 0.005$

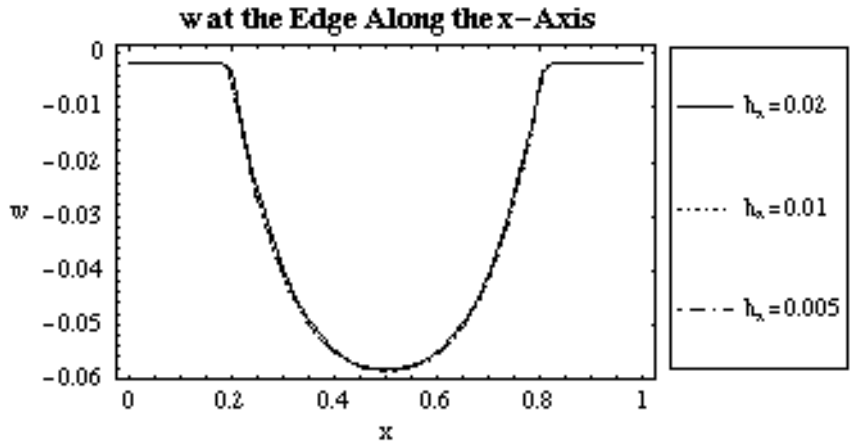

Figure 6.151 Plot of w along x-axis Edge for h_X = 0.02, 0.01, 0.005

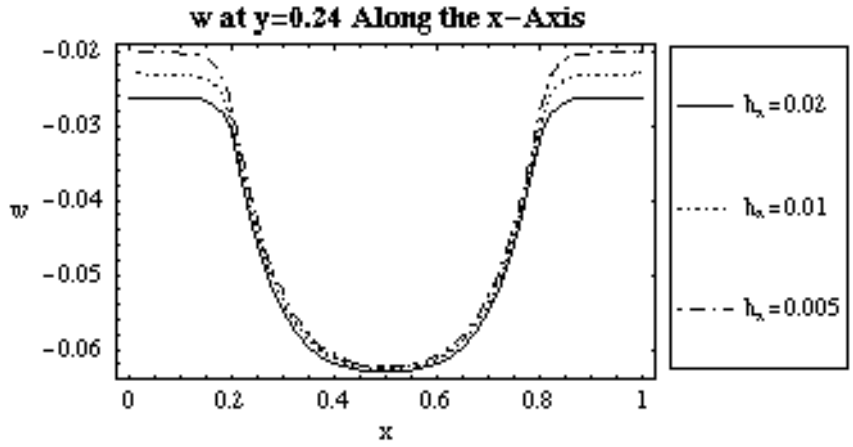

Figure 6.152 Plot of w along x-axis Quarter for h_X = 0.02, 0.01, 0.005

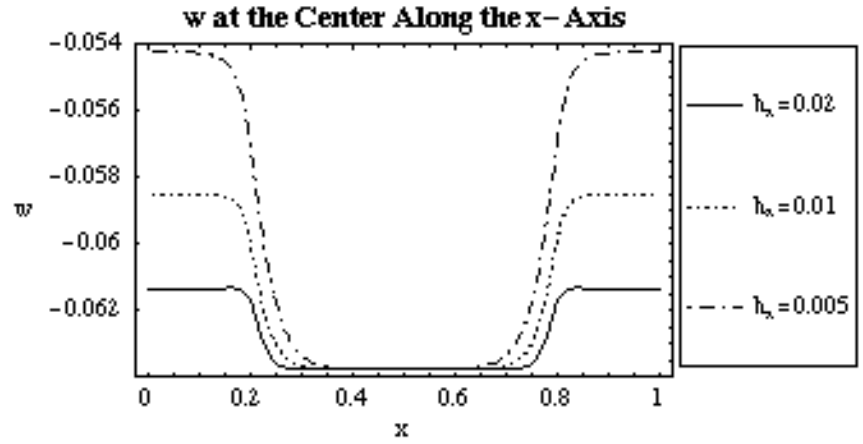

Figure 6.153 Plot of w along x-axis Center for h_X = 0.02, 0.01, 0.005

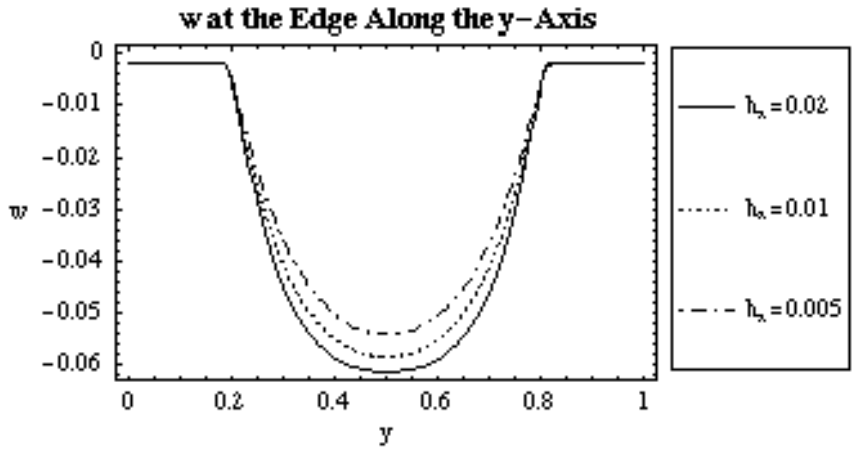

Figure 6.154 Plot of w along y-axis Edge for h_X = 0.02, 0.01, 0.005

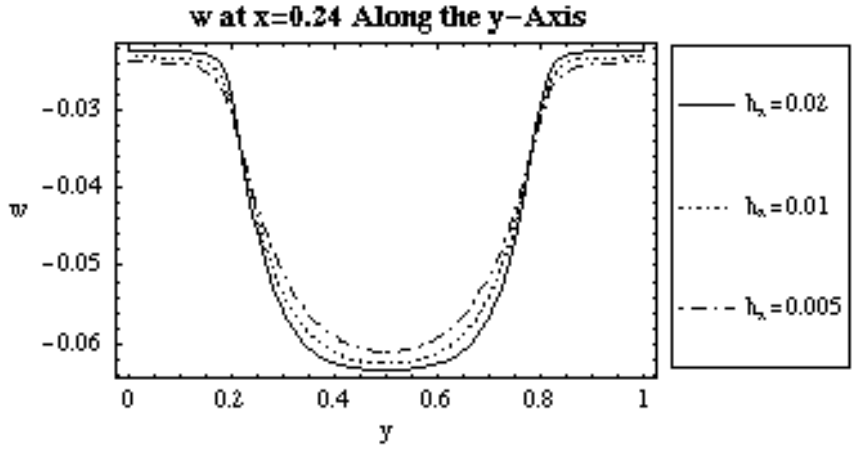

Figure 6.155 Plot of w along y-axis Quarter for h_X = 0.02, 0.01, 0.005

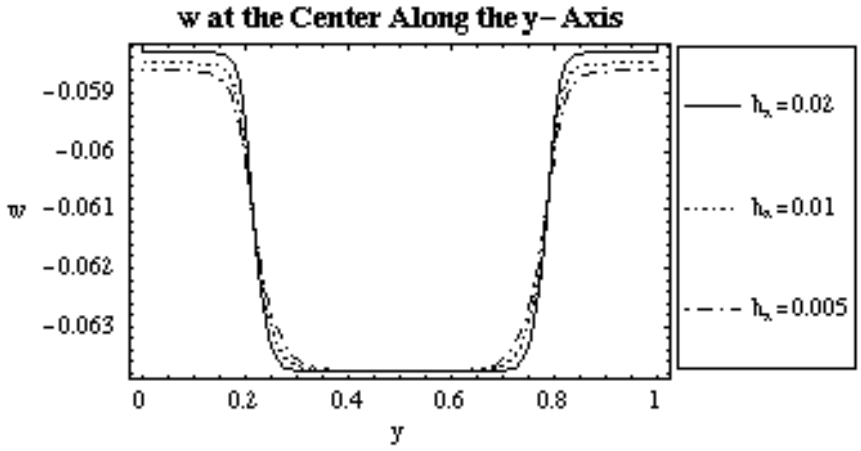

Figure 6.156 Plot of w along y-axis Center for h_X = 0.02, 0.01, 0.005

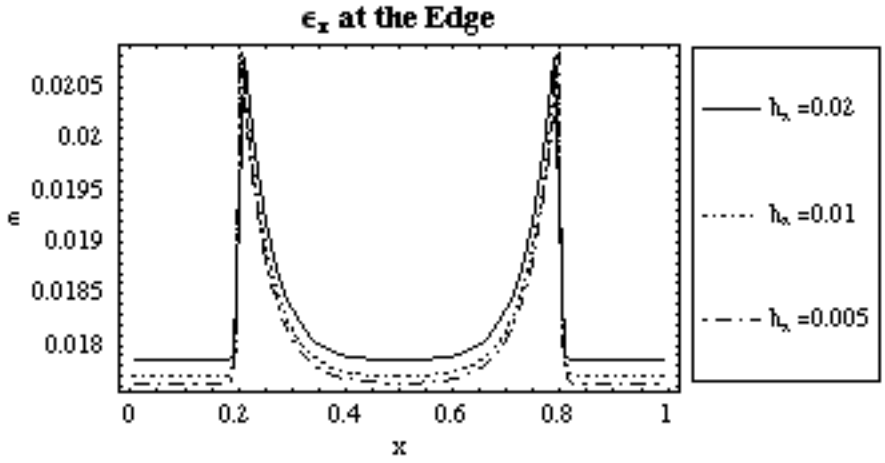

Figure 6.157 Plot of ε_x along x-axis Edge for h_X = 0.02, 0.01, 0.005

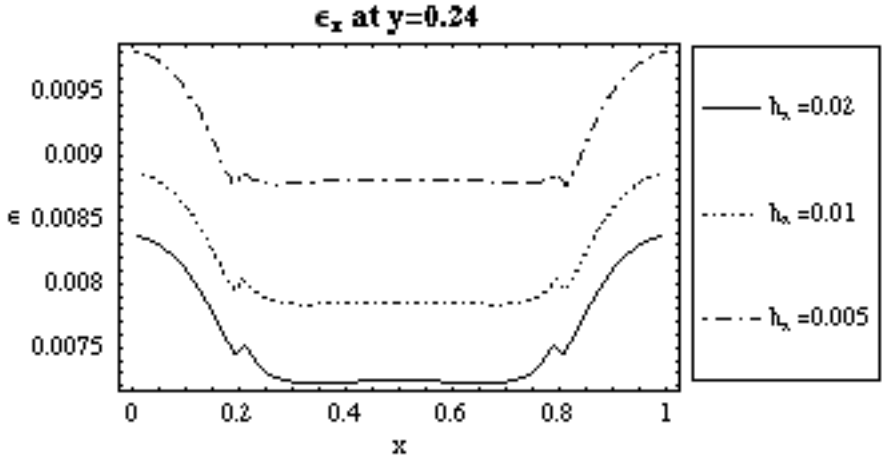

Figure 6.158 Plot of ε_x along x-axis Quarter for h_X = 0.02, 0.01, 0.005

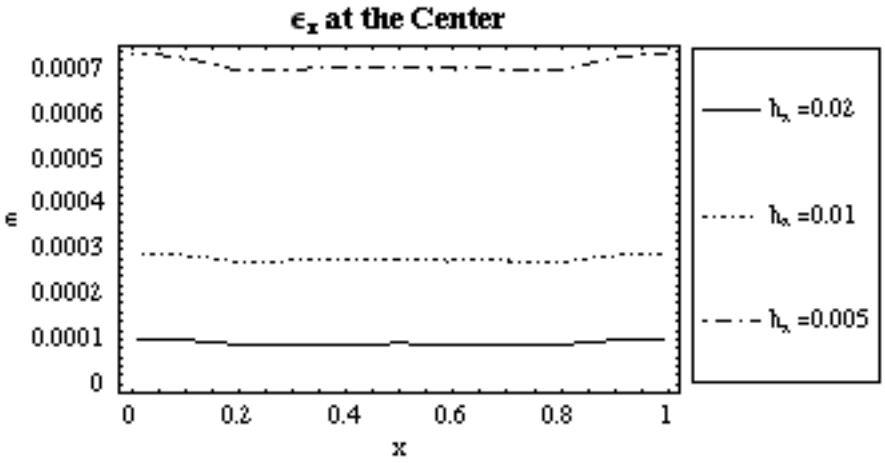

Figure 6.159 Plot of ε_x along x-axis Center for h_X = 0.02, 0.01, 0.005

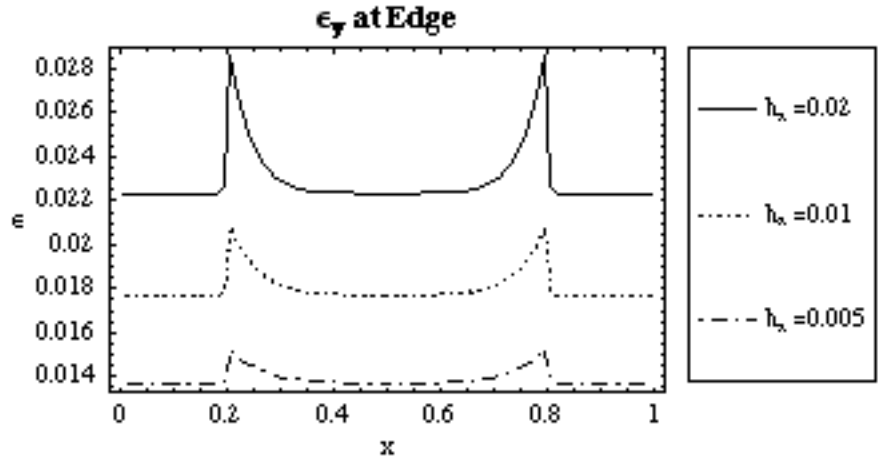

Figure 6.160 Plot of ε_x along x-axis Edge for h_X = 0.02, 0.01, 0.005

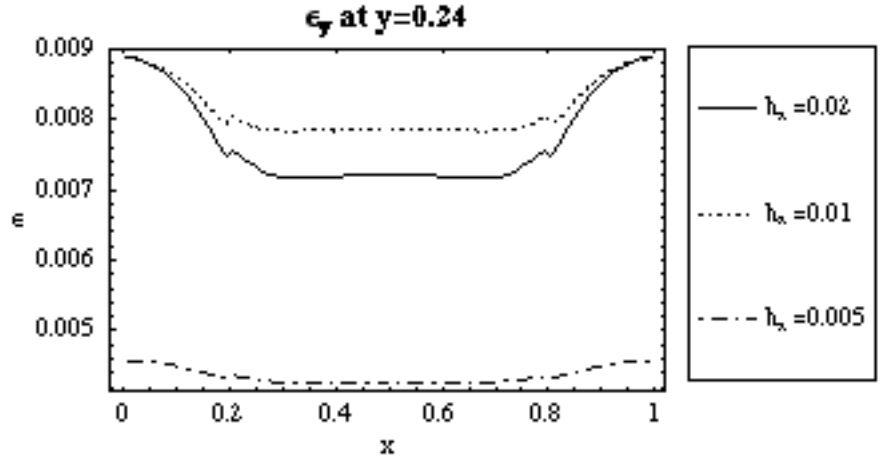

Figure 6.161 Plot of ε_x along x-axis Quarter for h_X = 0.02, 0.01, 0.005

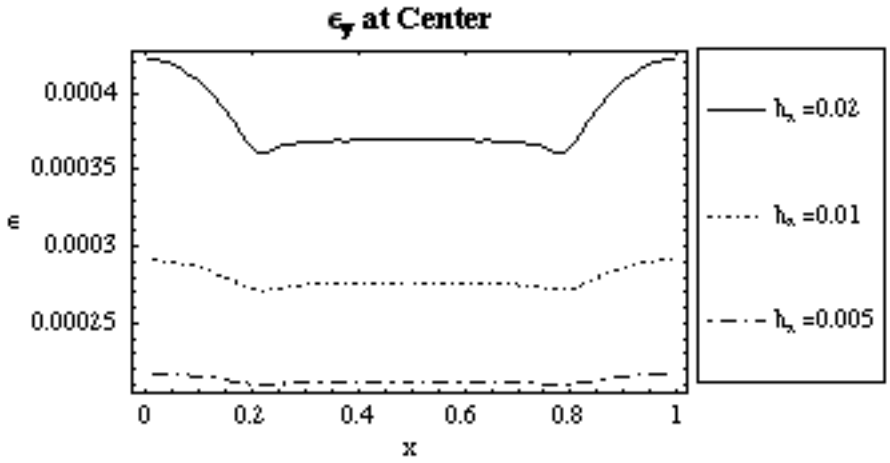

Figure 6.162 Plot of ε_x along x-axis Center for h_X = 0.02, 0.01, 0.005

The vertical displacement at the center of the y-axis edge was shallower than at the center of the x-axis edge when $h_X = 0.005$, and the y-axis was deeper than the x-axis edge when $h_X = 0.02$. This trend is shown in the vertical displacement contour plots, Figures 6.130 and 6.141. The vertical displacement decreased along the y-axis as h_X decreased, because the number of strands per width parallel to the y-axis increased when the strand length along the x-axis decreased. The cross-sectional area and modulus of elasticity were constant, so when the strand per width ratio increased, the grid stiffened. In other words, the load was distributed among more rows of strands parallel to the y-axis as h_X decreased. The strains in the strands parallel to the y-axis also decreased when h_X decreased, for the same reasoning.

The vertical displacement along the x-axis was slightly shallower in elevation when $h_X = 0.005$ than when $h_X = 0.02$. The slope of the geogrid's deformed shape was steepest when $h_X = 0.02$. Figure 6.152 depicts this trend: when $h_X = 0.02$, there was a larger region in the center that was equal to the unreinforced settlement than when $h_X = 0.005$. As h_X increased, the central region where the geogrid flattened out enlarged. Figures 6.156-6.158 show the strains in the geogrid strands increasing at the edge, and decreasing at the quarterline and at the centerline when h_X increased.

The vertical displacement profile became significantly shallower as h_X decreased. The difference was more pronounced along the y-axis: in Figures 6.153 and 6.154 the slope of the geogrid's deformed shape at the edge and the quarterline became steeper as h_X increased.

Figures 6.159-6.161 show that the strain in the strands parallel to the y-axis was largest at the edge and the center when $h_X = 0.02$, but midway between the edge and the center the largest strain values occurred when $h_X = 0.01$.

6.1.9 Variation of Orientation

A model was developed that used a geogrid rotated 45-degrees with respect to the x-axis. The model was developed to test the effect of the orientation of the geogrid on the vertical displacement and the strain in the strands. The orientation of the grid altered the non-dimensional parameters slightly because for symmetry to be maintained, the parameter h could not be 0.01 exactly. The values used for this case study were the following: b = 0.2, h = 0.0101, q_P = -59.4, q_S = -12.4, k_P = 35,600, and k_S = 195.

A three-dimensional rendering of the displaced shape of the 45-degree angled geogrid is shown in Figure 6.162. A three-dimensional plot and a contour plot of w are shown in Figures 6.163-6.164. A two-dimensional plot of the w at y is equal to 0, 0.30, and 0.5 is shown in Figure 6.165.

The results of each analysis were used to compare the effect of the geogrid orientation on the vertical displacement and the strain. A solid line designates values for the standard case orientation, and a dashed line designates values for the geogrid angled 45 degrees. Figures 6.166-6.169 show the vertical displacement values at the edge, quarterline, the centerline of the grid, and along the diagonal, respectively. Figures 6.170 shows the strain values along the diagonal.

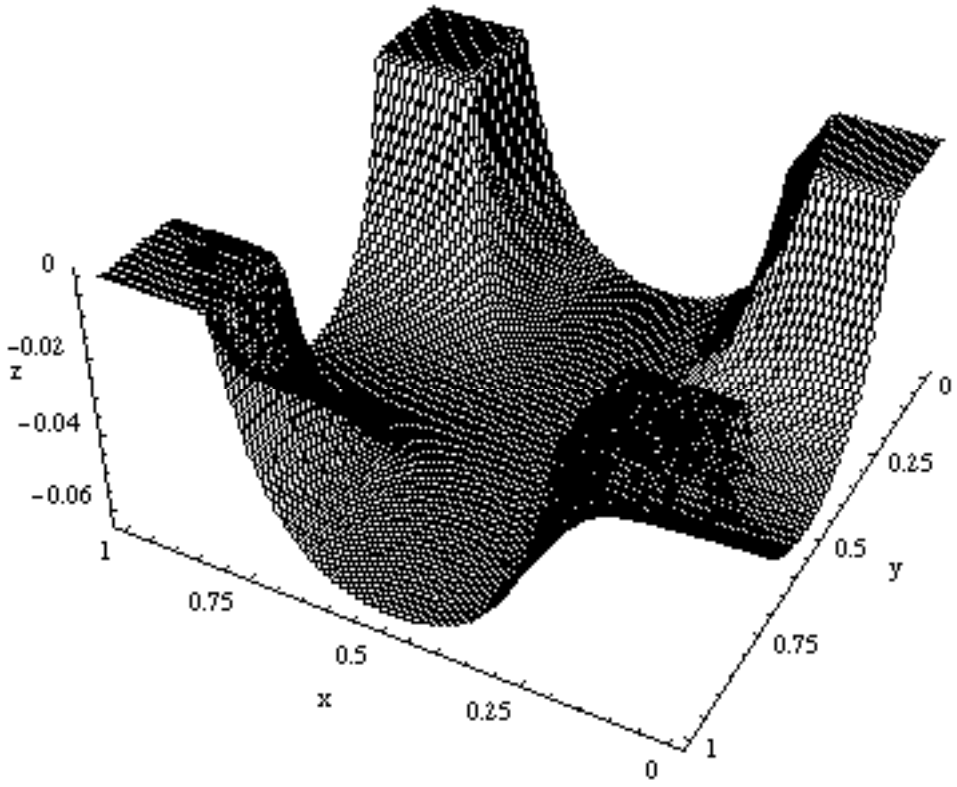

Figure 6.163 Three-Dimensional Plot for a 45° Rotation

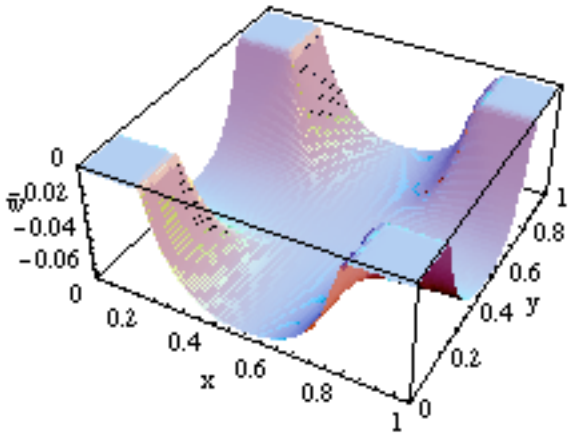

Figure 6.164 Three-Dimensional Plot of w for a 45° Rotation

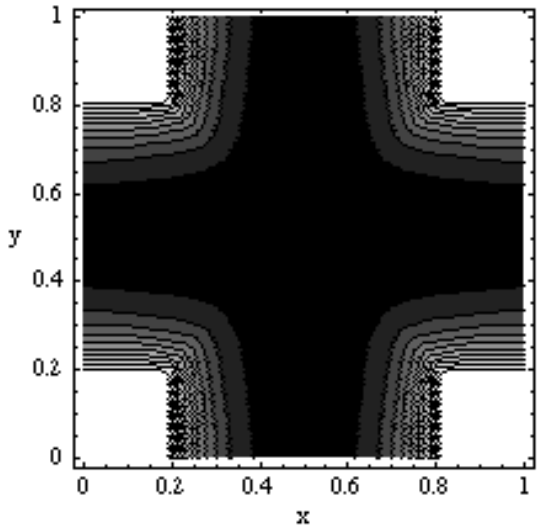

Figure 6.165 Contour Plot of w for a 45° Rotation

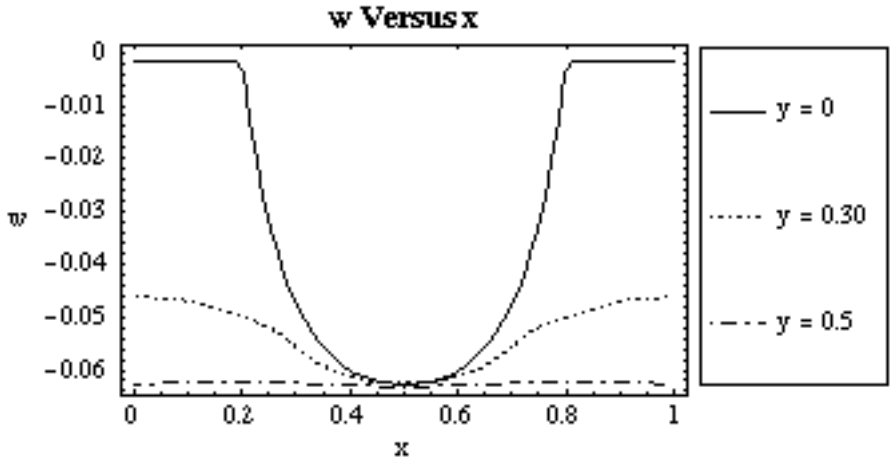

Figure 6.166 Plot of w vs. x along the Edge, Quarter and Center for a 45° Rotation

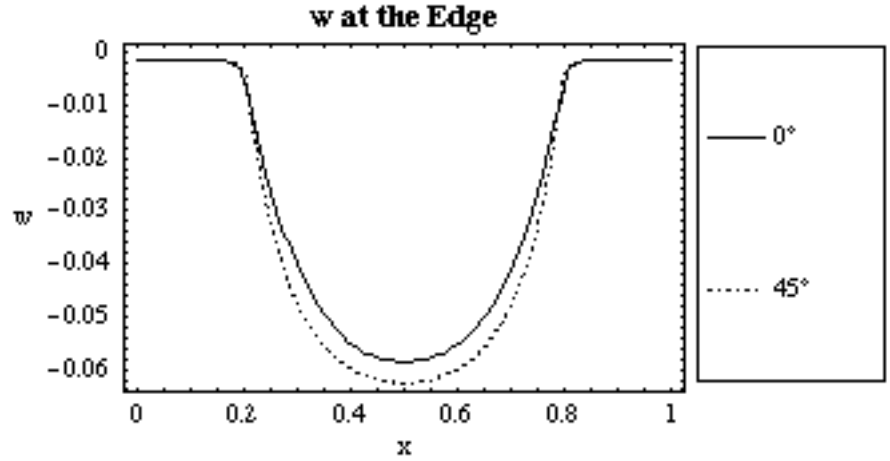

Figure 6.167 Plot of w along Edge for a 0° and 45° Rotation

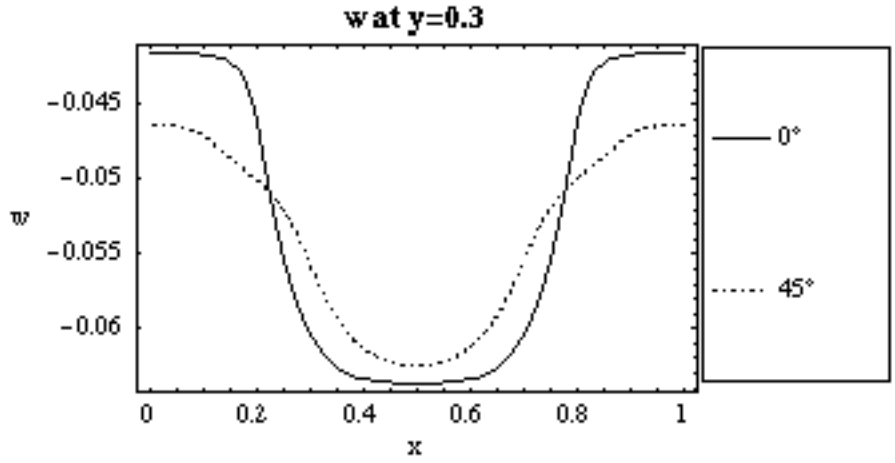

Figure 6.168 Plot of w along y=0.3 for a 0° and 45° Rotation

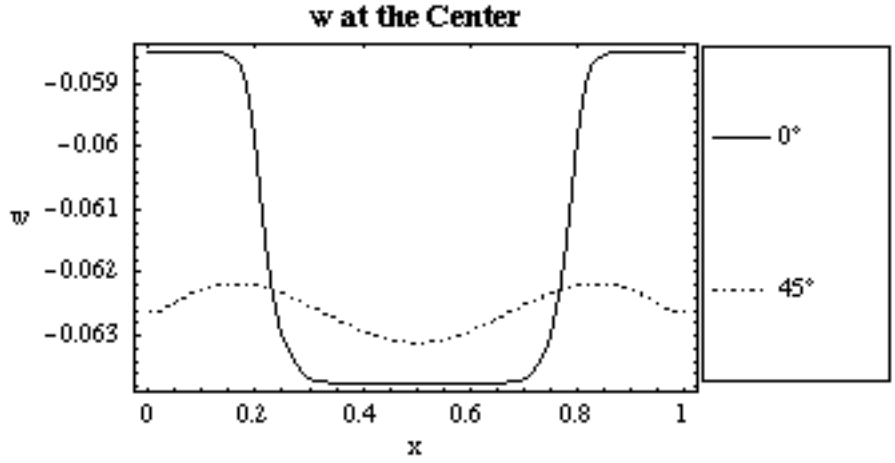

Figure 6.169 Plot of w along Center for a 0° and 45° Rotation

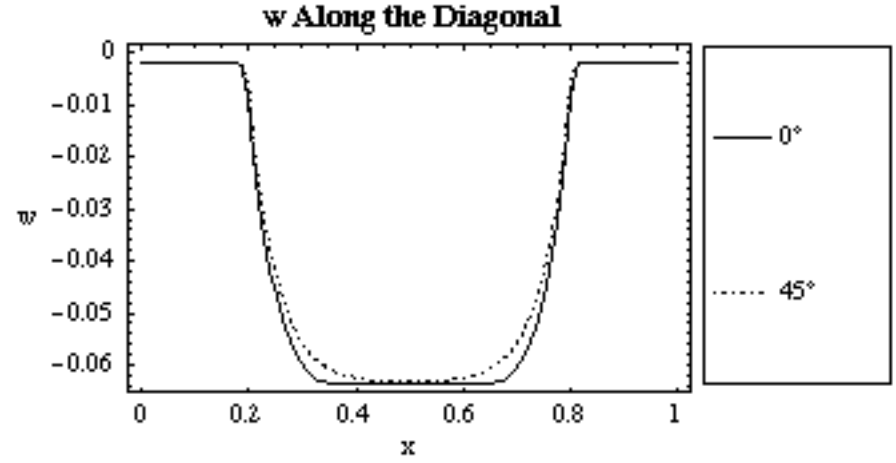

Figure 6.170 Plot of w along Diagonal for a 0° and 45° Rotation

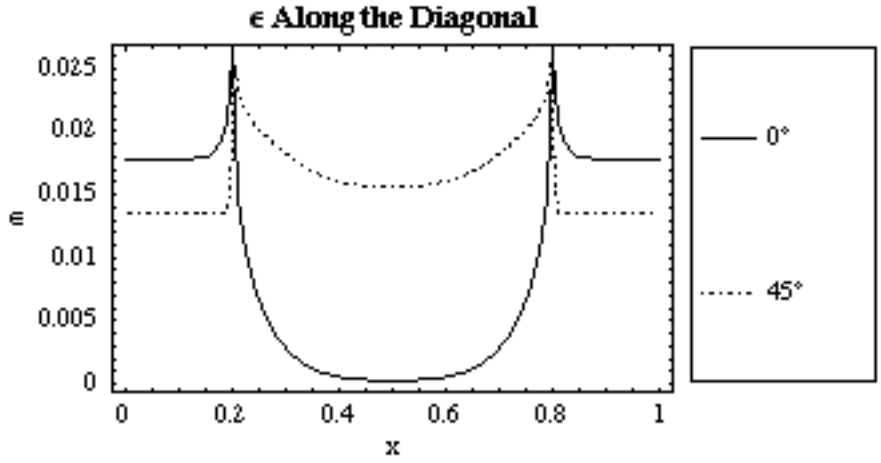

Figure 6.171 Plot of ε_x along Diagonal for a 0° and 45° Rotation

The vertical displacement at the center was shallower for the 45-degree orientation than the vertical displacement at the center for the 0-degree orientation. In other words, the vertical displacement at the center was less than the unreinforced settlement for a geogrid oriented 45-degrees to the x-axis. The maximum vertical displacement was 0.0631 for a geogrid oriented 45-degrees with respect to the x-axis, and the maximum vertical displacement was 0.0637 for a geogrid oriented so that the strands were parallel to the x-axis or y-axis. The largest span between two piles in the unit cell was the distance between two diagonal piles, and the largest vertical displacement occurred at the center of the cell where the distance from each pile was greatest. Logically, the best orientation was one that aligned the geogrid's stiffness with the largest span. The geogrid oriented 45-degrees had strands that spanned diagonally from one pile to the opposing pile, and it was the best orientation to reduce the vertical displacement at the center. The vertical displacement at the edge was almost equal to the displacement at the center. The geogrid stiffness was bilateral, and since the strands were not parallel to the cell edges, the stiffness was significantly reduced along the edge, causing larger displacements.

A 45-degree orientation had a more evenly distributed vertical displacement among the joints, and strain among the strands. A geogrid oriented so that the strands were parallel to the edges of the pile had high strain concentrations above the pile, and above the soil

near the edges of the pile; otherwise the strands had considerably smaller strains than those of the 45-degree orientation.

6.1.10 Variation of Non-Dimensional Joint Stiffness c

A model was developed that integrated rotational springs at the joints to resist bending in the vertical plane. The model was used to find the effect of joint bending stiffness on the displaced shape and the strain in the strands. The parameter c is the non-dimensional rotational stiffness at a joint relative to bending of a strand in the vertical plane. For this case study, the rotational stiffness was assumed to be equal for the strands parallel to the x-axis and strands parallel to the y-axis. All the other non-dimensional parameters were equal to the standard case values and remained constant for each analysis. A broad range of c values was used so that the effect of this parameter on the behavior of the geogrid could be understood. The variable, c, was chosen to have the following values: 100, 1,000, and 10,000. Previous results correspond to c = 0.

Plots of the case with c = 100 are shown in Figures 6.171-6.177, plots of the case with c = 1,000 are shown in Figures 6.178-6.184, and plots of the case with c = 10,000 are shown in Figures 6.185-6.191. The results of each analysis were used to compare the effect of the non-dimensional parameter c on the vertical displacement and the strain. A solid line designates values for the standard case c = 0, a dotted line designates values for c = 100, a dash-dot line designates values for c = 1,000, and a dashed line designates values for c = 10,000. Figures 6.192-6.194 show the vertical displacement values at the edge, midway between the edge and centerline, and at the centerline of the grid, respectively. Figures 6.195-6.197 show the strain values at the edge, midway between the edge and centerline, and at the centerline of the grid, respectively.

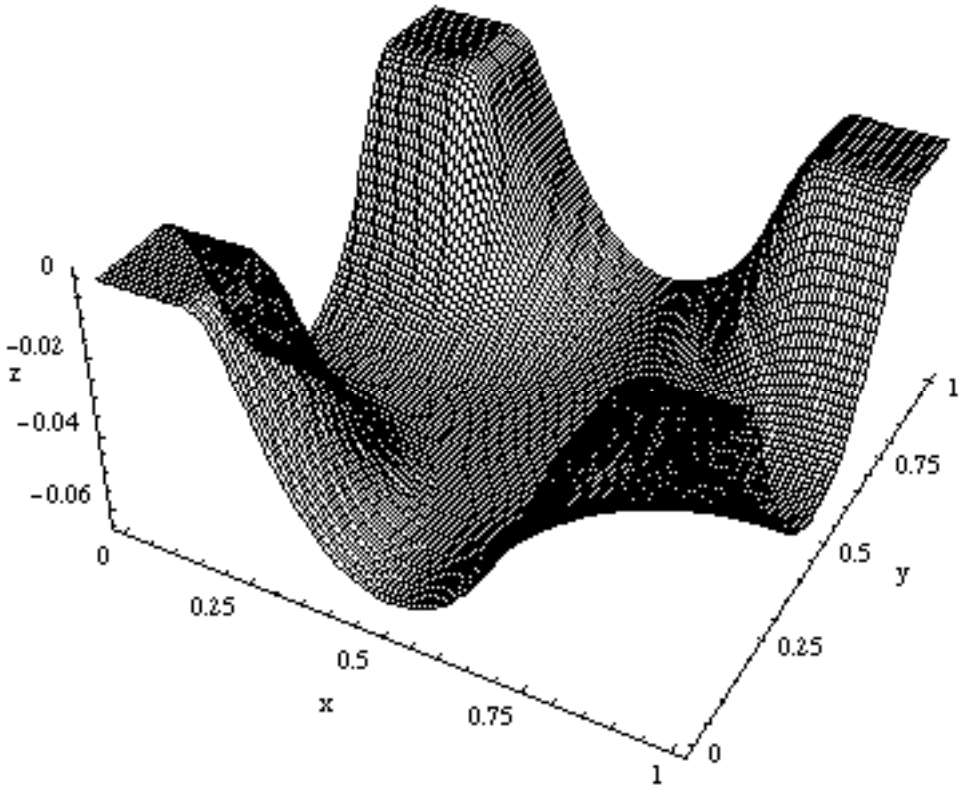

Figure 6.172 Three-Dimensional Plot for c = 100

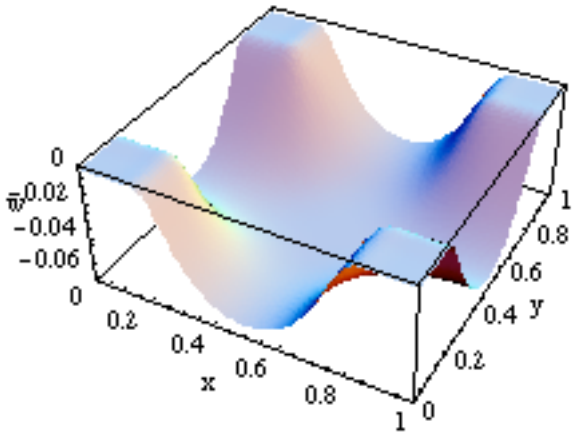

Figure 6.173 Three-Dimensional Plot of w for c = 100

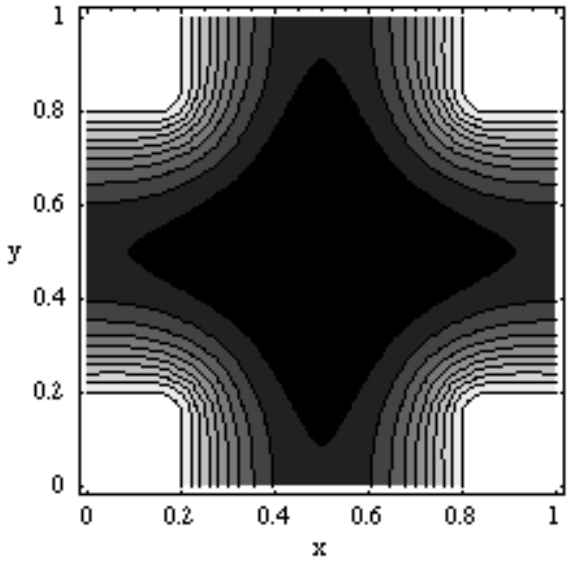

Figure 6.174 Contour Plot of w for c = 100

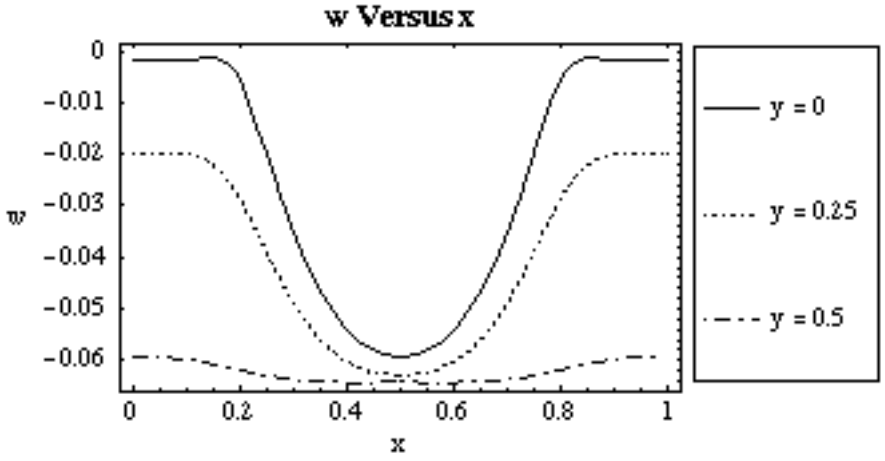

Figure 6.175 Plot of w vs. x along the Edge, Quarter and Center for c = 100

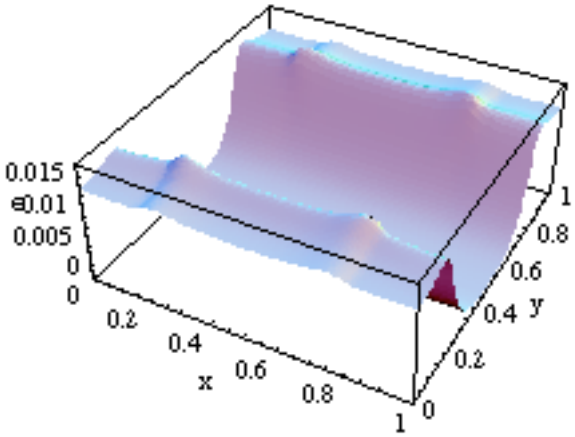

Figure 6.176 Three-Dimensional Plot of ε_x for c = 100

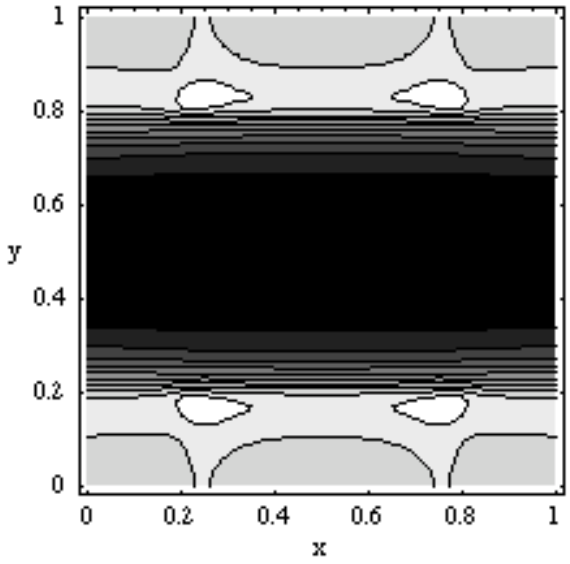

Figure 6.177 Contour Plot of ε_x for c = 100

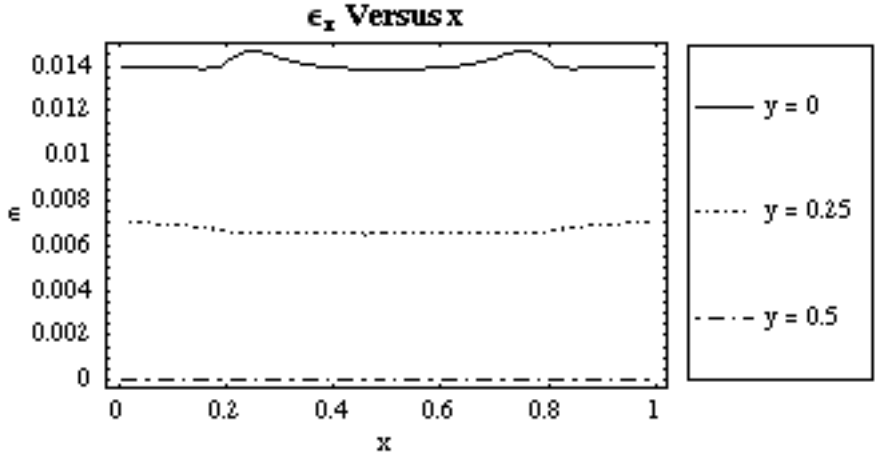

Figure 6.178 Plot of ε_x vs. x along the Edge, Quarter and Center for c = 100

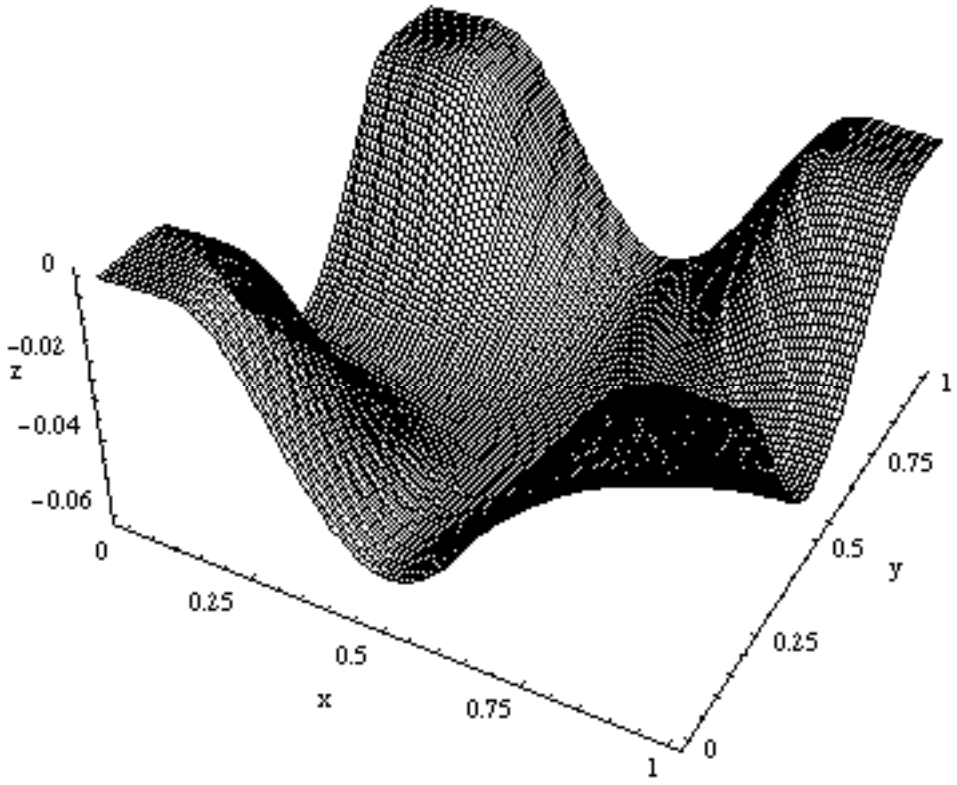

Figure 6.179 Three-Dimensional Plot for c = 1000

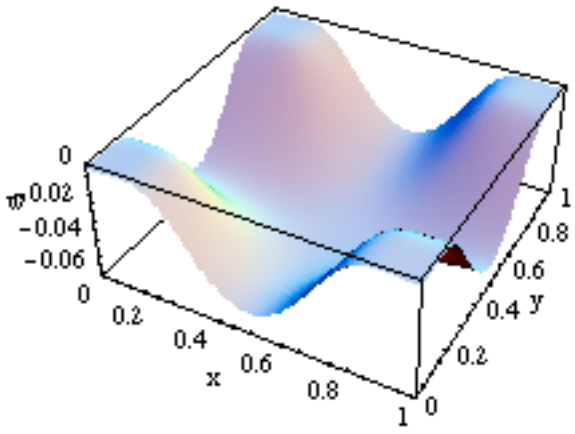

Figure 6.180 Three-Dimensional Plot of w for c = 1000

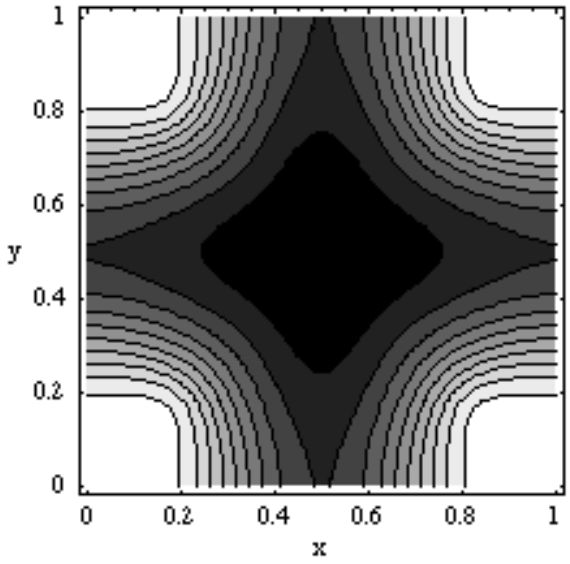

Figure 6.181 Contour Plot of w for c = 1000

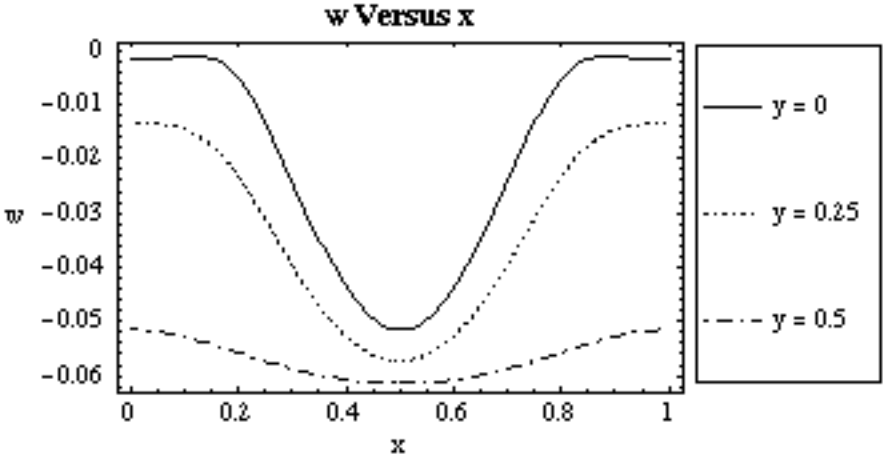

Figure 6.182 Plot of w vs. x along the Edge, Quarter and Center for c = 1000

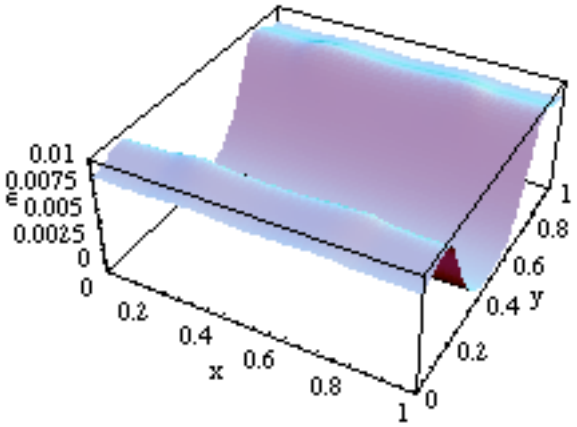

Figure 6.183 Three-Dimensional Plot of ε_x for c = 1000

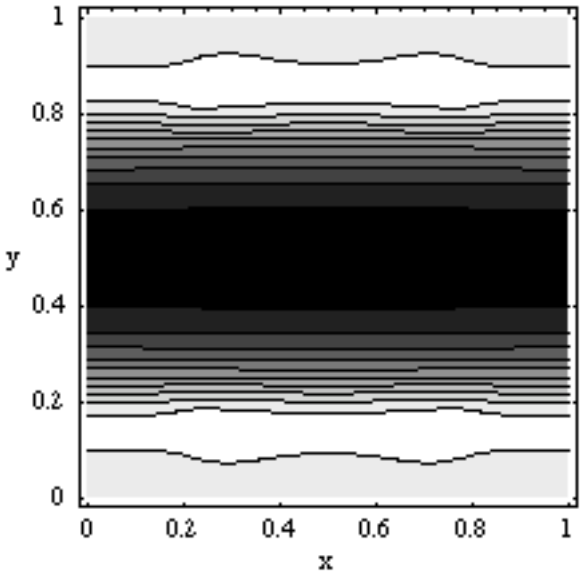

Figure 6.184 Contour Plot of ε_x for c = 1000

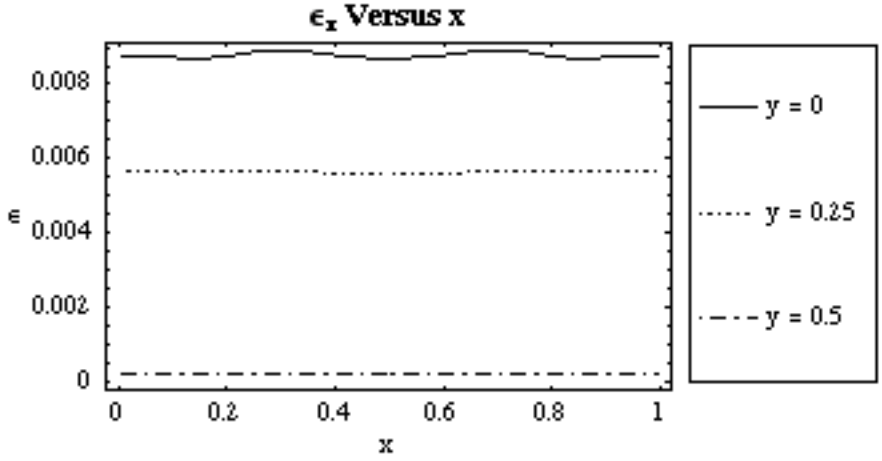

Figure 6.185 Plot of ε_x vs. x along the Edge, Quarter and Center for c = 1000

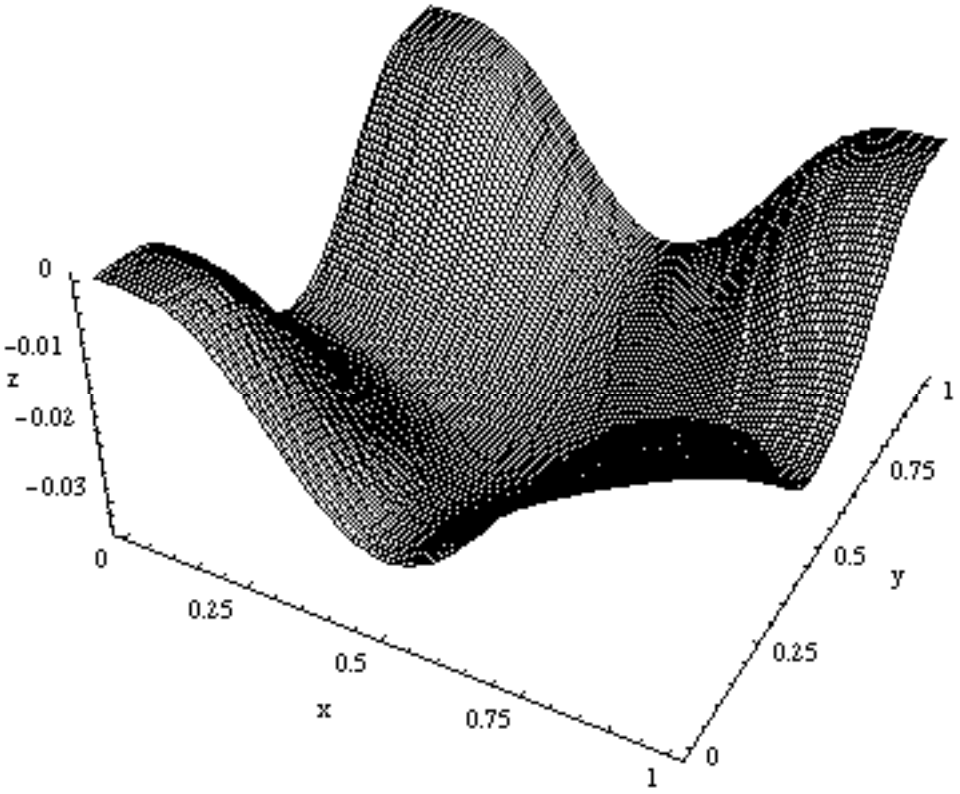

Figure 6.186 Three-Dimensional Plot for c = 10,000

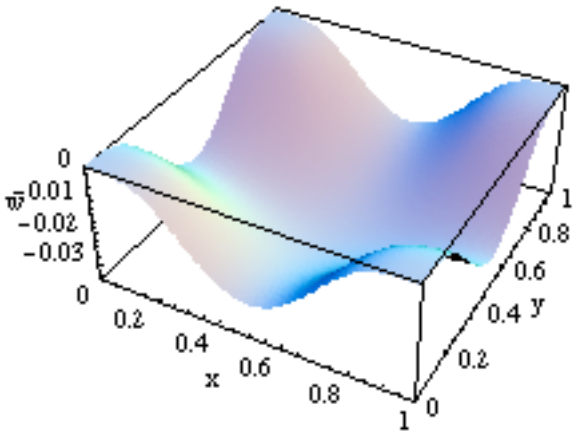

Figure 6.187 Three-Dimensional Plot of w for c = 10,000

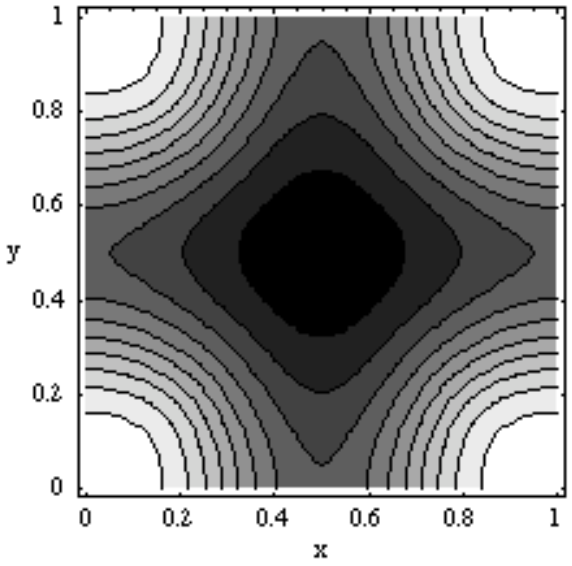

Figure 6.188 Contour Plot of w for c = 10,000

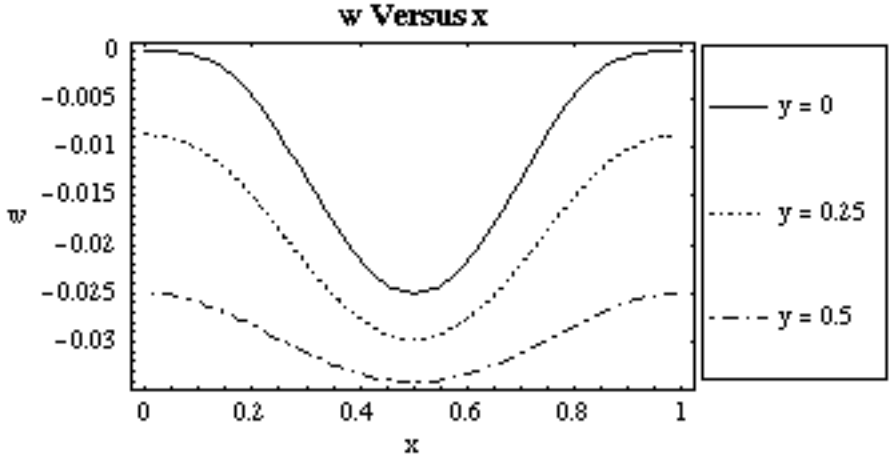

Figure 6.189 Plot of w vs. x along the Edge, Quarter and Center for c = 10,000

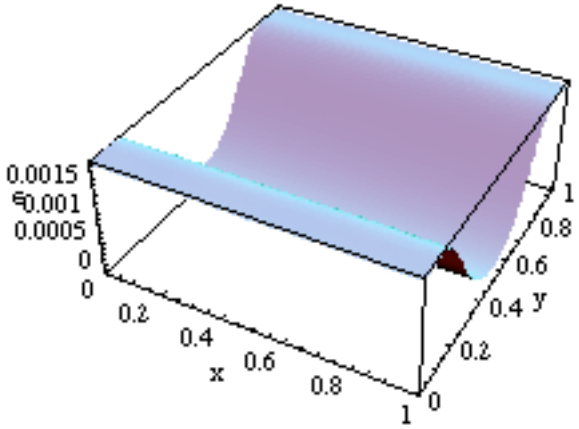

Figure 6.190 Three-Dimensional Plot of ε_x for c = 10,000

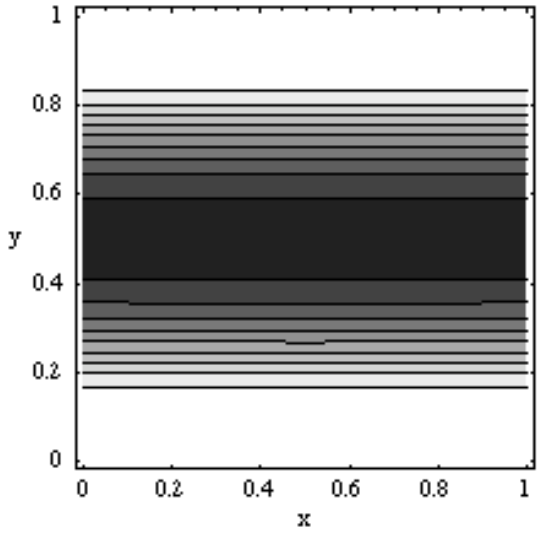

Figure 6.191 Contour Plot of ε_x for c = 10,000

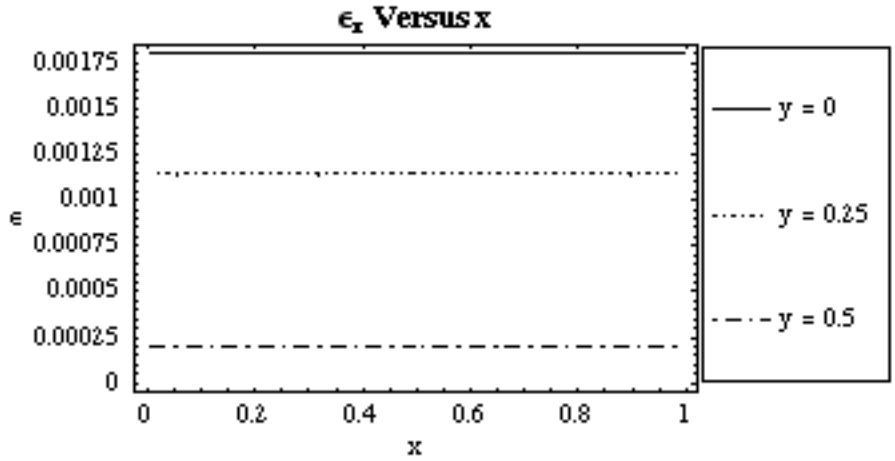

Figure 6.192 Plot of ε_x vs. x along the Edge, Quarter and Center for c = 10,000

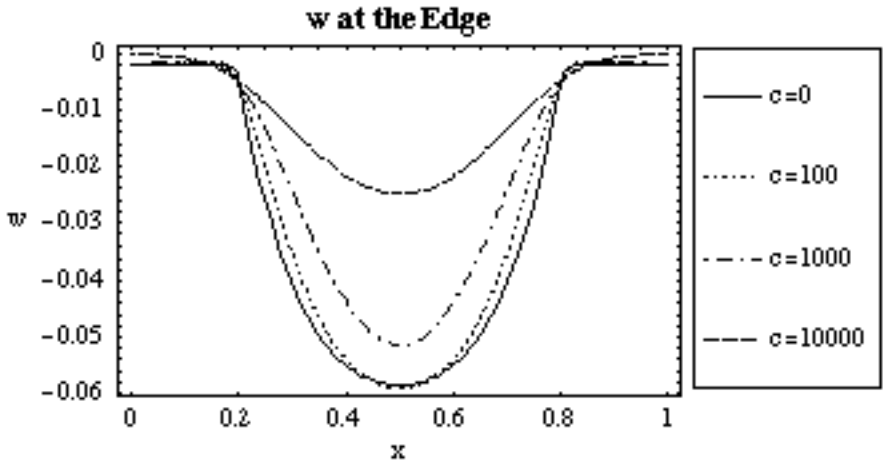

Figure 6.193 Plot of w along Edge for c = 0, 100, 1000, 10000

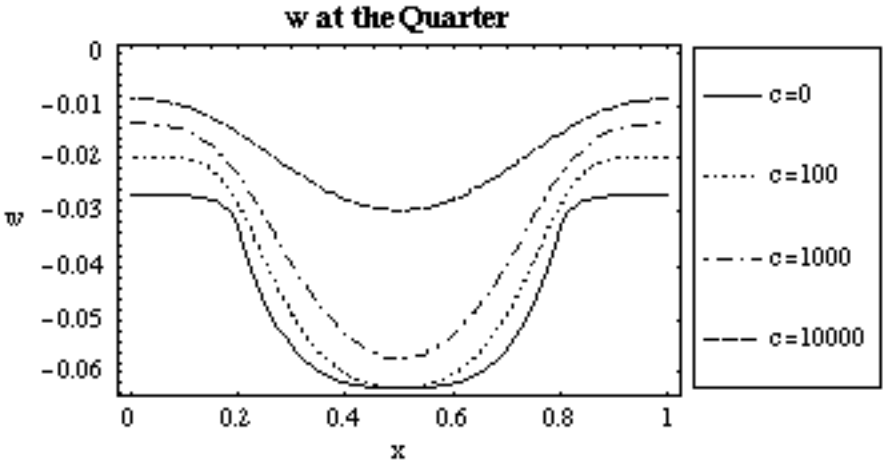

Figure 6.194 Plot of w along Quarter for c = 0, 100, 1000, 10000

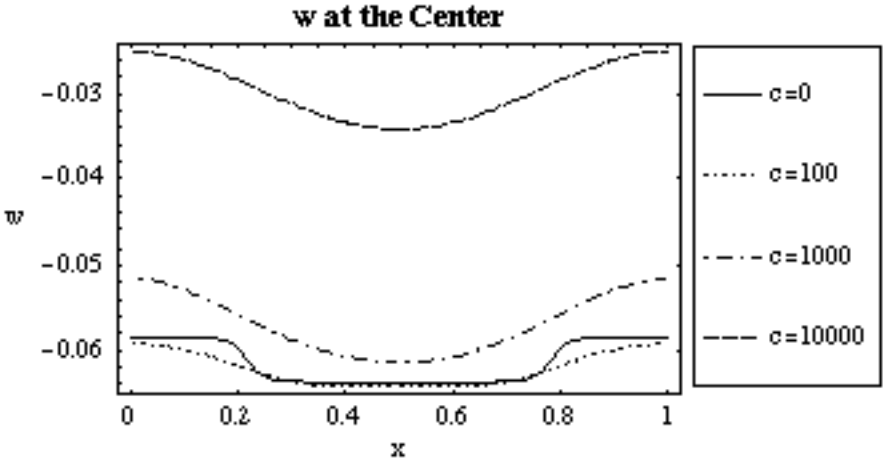

Figure 6.195 Plot of w along Center for c = 0, 100, 1000, 10000

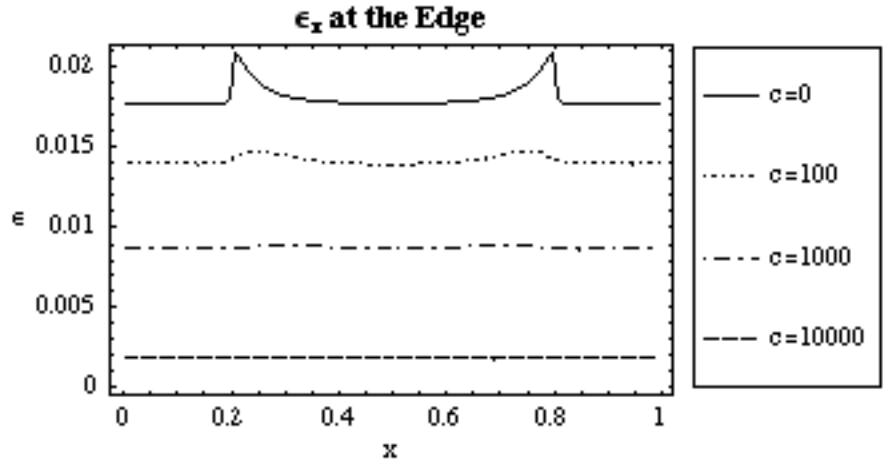

Figure 6.196 Plot of ε_x along Edge for c = 0, 100, 1000, 10000

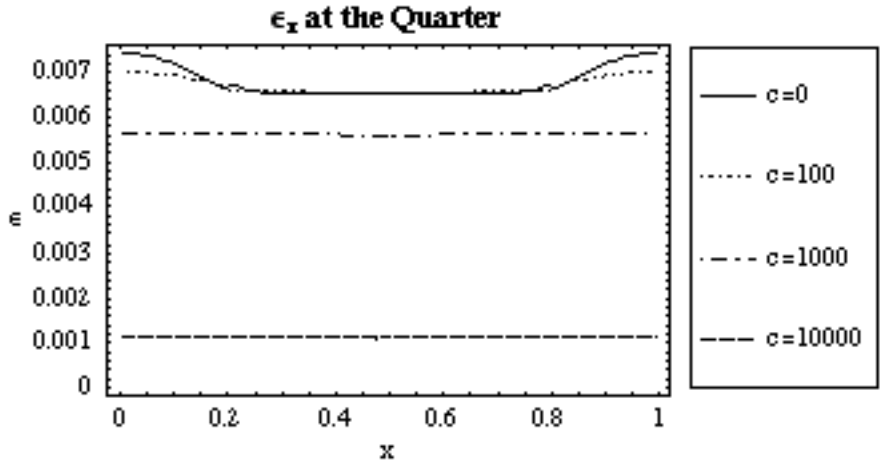

Figure 6.197 Plot of ε_x along Quarter for c = 0, 100, 1000, 10000

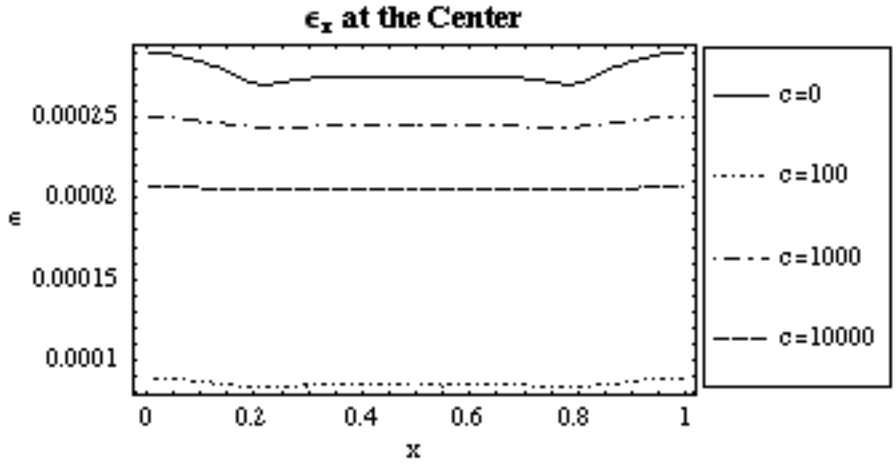

Figure 6.198 Plot of ε_x along Center for c = 0, 100, 1000, 10000

A comparison between Figures 6.171, 6.178, and 6.185 shows that the geogrid's deformed shape became round and the change in slope was more gradual as c increased. This trend was also observed in the two-dimensional vertical displacement plot shown in Figures 6.174 and 6.181 because the displacement rose slightly before the edge of the pile to alleviate the rotation of the geogrid strands at the pile edge. The contour lines shown in Figures 6.173, 6.180, and 6.187 became circular around the pile, and diamond shaped as they approached the center, as c increased. The vertical displacement profile in Figure 6.188 for c = 10,000 shows that the corner nodes had the highest elevation and the deformed shape gently sloped downwards from the corners, very similar to the behavior of a plate. The relationship found for pile displacements described in Equation 6.11 did

not apply to this case study, and the maximum vertical displacements were not equal to the unreinforced settlement. The maximum displacements at the edge, quarter, and center locations were largest when c = 100. When c was equal to 1,000 and 10,000, the joint stiffness reduced the vertical displacement. The reduction was largest when c = 10,000: the vertical displacement was almost halved. Figures 6.192-6.194 show these observations.

The joint stiffness also reduced the strain values. The three-dimensional plots of the strain in Figures 6.175, 6.182, and 6.189 show that the shape of the plot became smooth and an overall reduction in strain occurred as c increased. When c = 0, there was a sharp spike in strain at the corner of each pile, but as c increased, the spike became smooth and disappeared when c = 10,000. An observation worth mentioning is that the strain was significantly smaller at the center when c = 100. The change in slope was gentler at the center for this case, and this could be the reason for this exception. When c = 10,000 the strain was approximately constant along each row of strands, and the shape of the three-dimensional plot looked like a rolled sheet. As rotational stiffness was added to the joints, the deformed shape became circular, causing the change in slope of the geogrid strands to approach a constant value.

6.1.11 Two Layers of Geogrid Reinforcement

A model was developed that included two layers of cable nets connected by linear springs in an effort to analyze two layers of geogrid reinforcement separated by a layer of granular soil. The spring constant of the soil between the geogrids was equal to the spring constant of the soil over the pile, k_g = 36,000. This value was taken because the same granular fill is usually used between the geogrid layers as is used to separate the pile caps and the bottom geogrid layer. Also, the separation between geogrid reinforcement layers is typically of equal distance to the separation between the bottom geogrid and the pile. The stiffness of the soil resting on the pile and the soil separating the geogrid layers are approximately equal because the material properties are the same and the initial thickness of both soil layers are approximately equal. Since linear springs were used, the energy of

the model was not dependent on the initial separation of the two layers of geogrid, but for plotting purposes a non-dimensional distance of 0.05 was used, which was equivalent to 15 cm. The standard case parameter values were used for this analysis.

A three-dimensional rendering of the displaced shape of both geogrid layers is shown in Figure 6.198. The top geogrid is shown lighter than the bottom geogrid so that it is easier to differentiate between both layers. Three-dimensional plots of the vertical displacement for the top and bottom geogrids are shown in Figures 6.199 and 6.200, respectively. Contour plots of the vertical displacement for the top and bottom geogrids are shown in Figures 6.201 and 6.202, respectively. Figures 6.203-6.205 show a comparison of the vertical displacements that occur at the edge, quarterline, and centerline when one layer and two layers of geogrid reinforcement were analyzed. A solid line designates vertical displacements for the one-layer geogrid model, a dashed line designates vertical displacements for the top layer of the two-layer geogrid model, and a dash-dot line designates vertical displacements for the bottom layer of the two-layer geogrid model. Figures 6.206 and 6.207 show three-dimensional plots of the strain for the top and bottom geogrids, respectively. Figures 6.208 and 6.209 show contour plots of the strain for the top and bottom geogrids, respectively. Figures 6.210-6.212 show a comparison of the strains that occur at the edge, quarterline, and centerline when one and two layers of geogrid reinforcement are analyzed. The same line designations that are used for the vertical displacement plots are used for the strain plots.

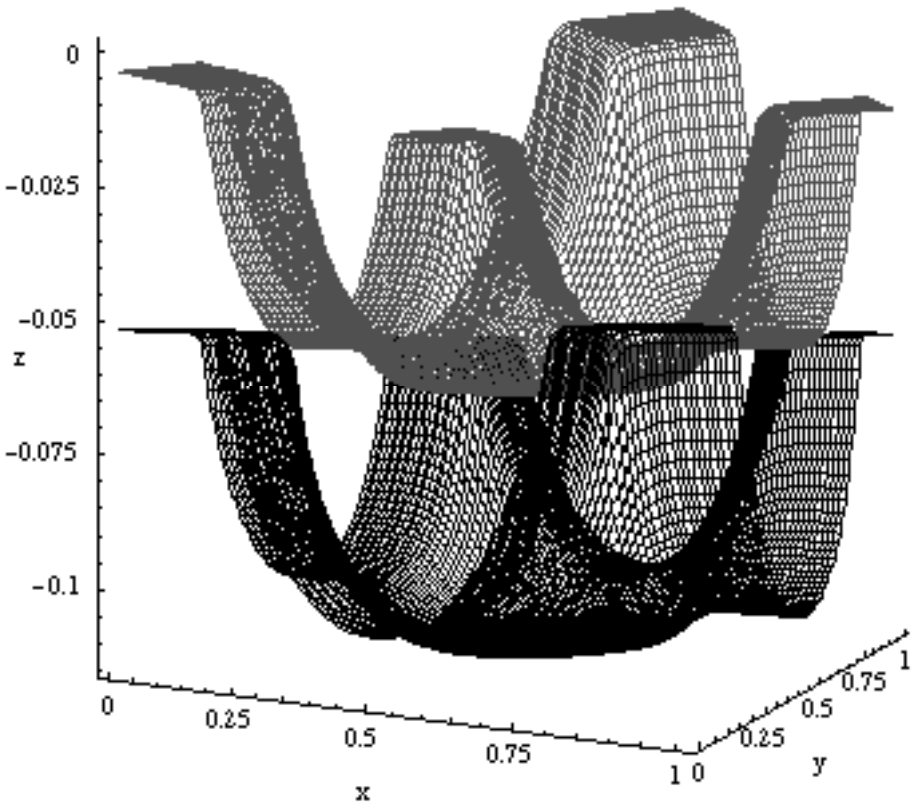

Figure 6.199 Three-Dimensional Plot of Two Geogrids

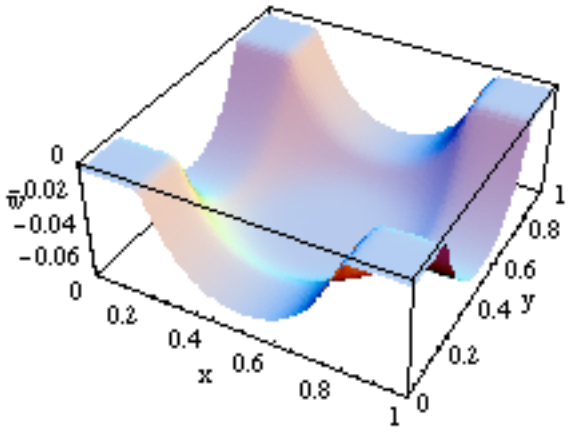

Figure 6.200 Three-Dimensional Plot of w for Top Geogrid

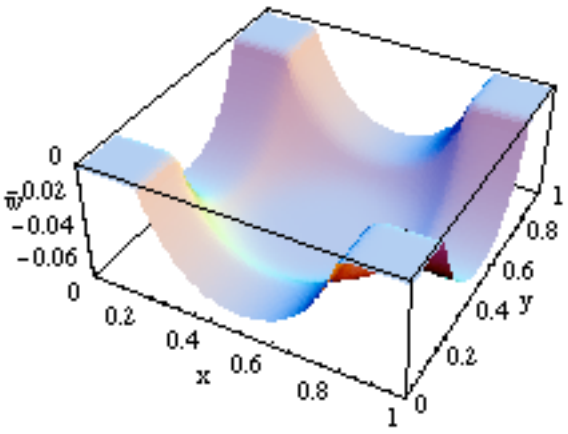

Figure 6.201 Three-Dimensional Plot of w for Bottom Geogrid

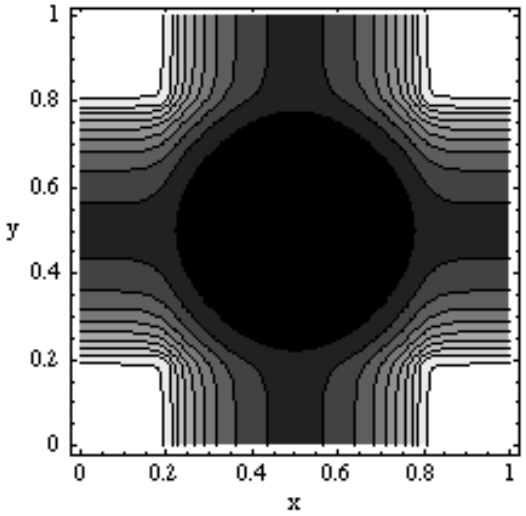

Figure 6.202 Contour Plot of w for Top Geogrid

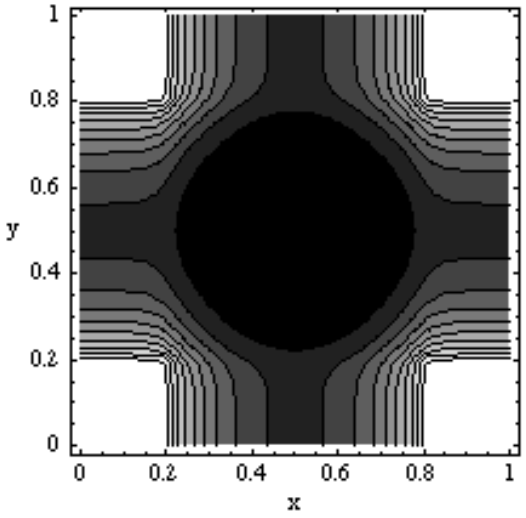

Figure 6.203 Contour Plot of w for Bottom Geogrid

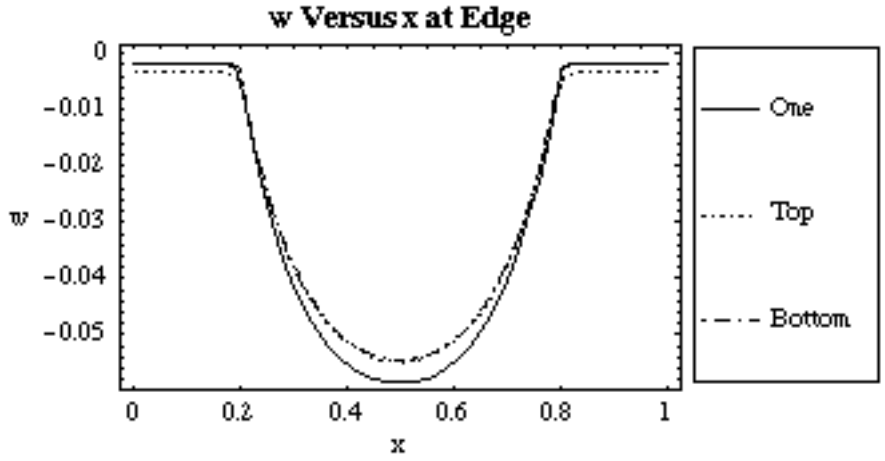

Figure 6.204 Plot of w vs. x along the Edge, Quarter and Center

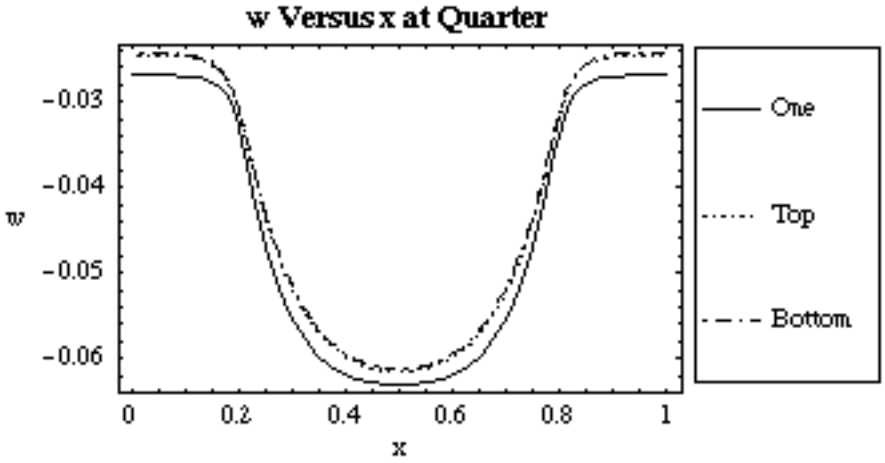

Figure 6.205 Plot of w vs. x along the Quarter

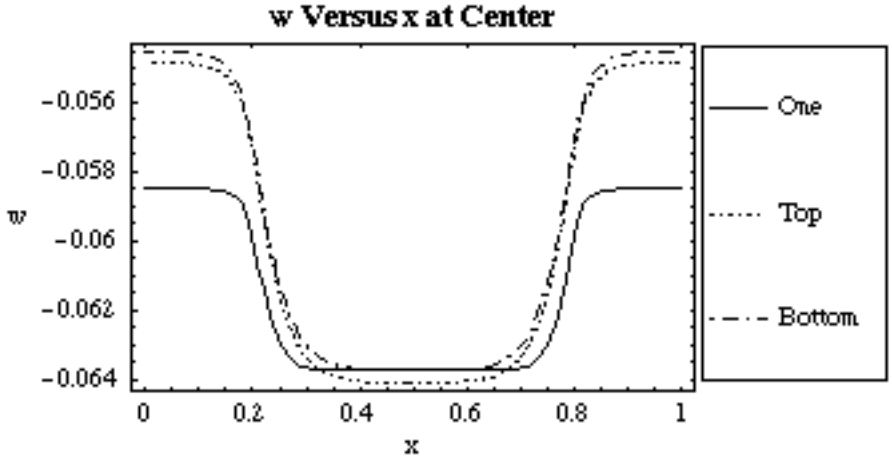

Figure 6.206 Plot of w vs. x along the Center

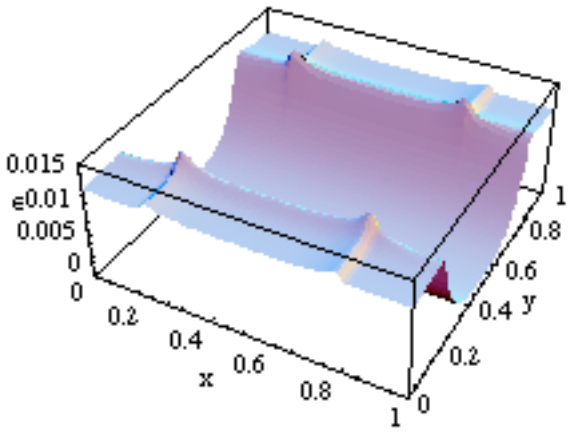

Figure 6.207 Three-Dimensional Plot of ε_x for Top Geogrid

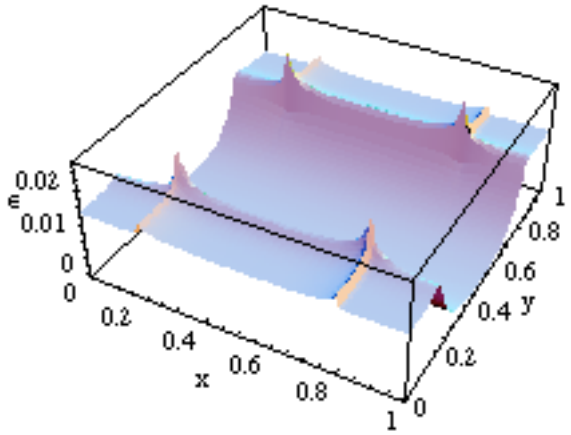

Figure 6.208 Three-Dimensional Plot of ε_x for Bottom Geogrid

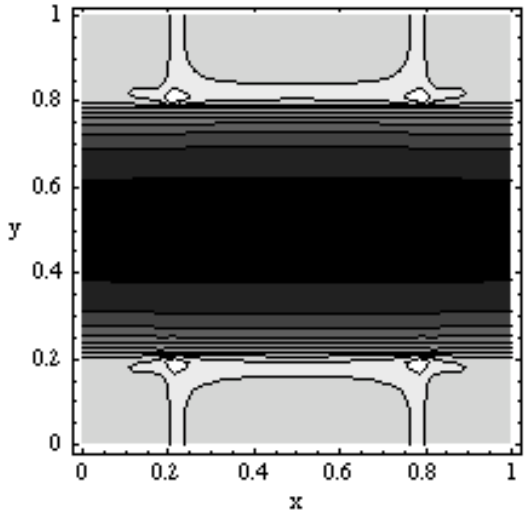

Figure 6.209 Contour Plot of ε_x for Top Geogrid

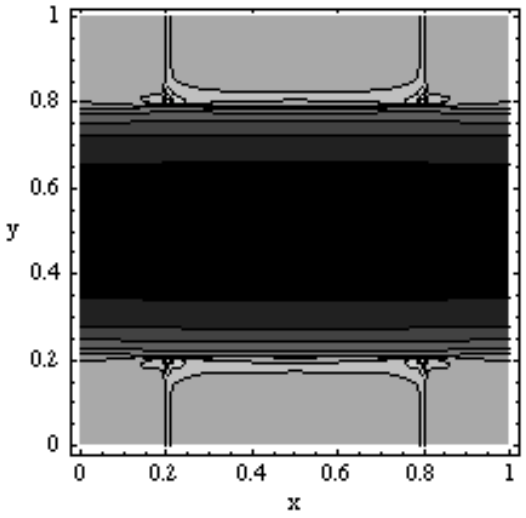

Figure 6.210 Contour Plot of ε_x for Bottom Geogrid

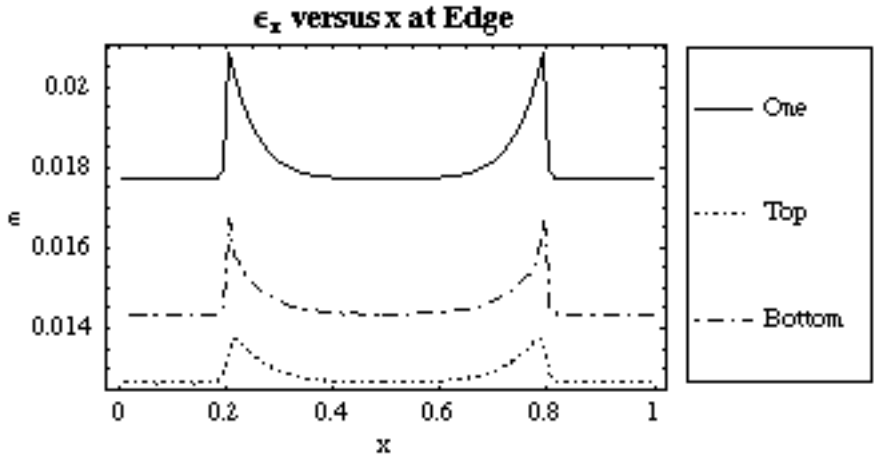

Figure 6.211 Plot of ε_x vs. x along the Edge

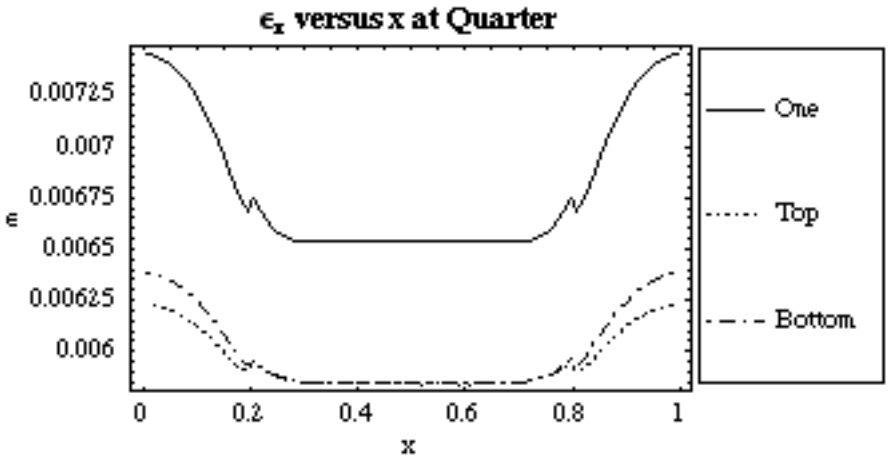

Figure 6.212 Plot of ε_x vs. x along the Quarter

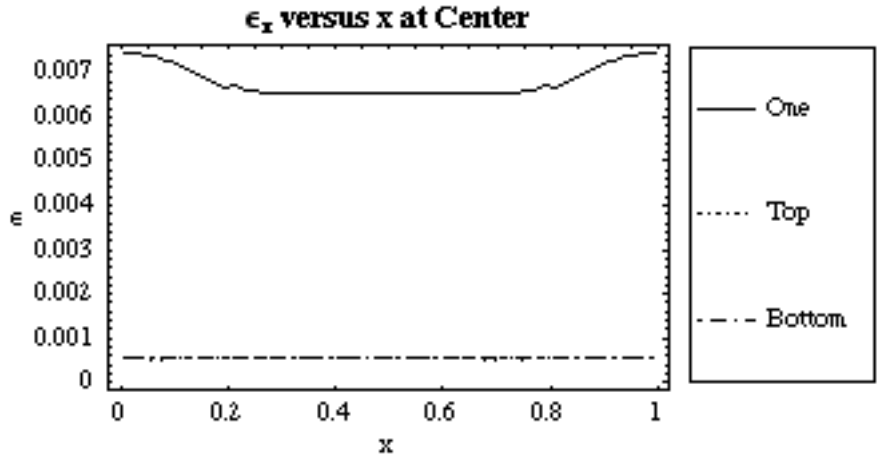

Figure 6.213 Plot of ε_x vs. x along the Center

The deformed shapes of both geogrid layers were very similar to the one-geogrid-layer model. Figures 6.203 and 6.204 show that the vertical displacements of the top and bottom geogrid layers were less at the edge and the quarterline than for a model with only one layer of geogrid. The top geogrid was not constrained by the result of the limiting displacement equation, but the bottom and single layer of reinforcement were. The single layer and bottom geogrids were supported entirely by the bottom soil springs, while the top geogrid was supported by two springs in series, the bottom soil springs and the soil springs between geogrid layers, so the top geogrid deformed more. Figure 6.205 shows that the largest maximum displacement occurred in the top layer of reinforcement.

The largest strains in the two-layer geogrid model occurred at the pile edge of the bottom geogrid. The overall strain in the bottom geogrid was larger than the strains in the top geogrid at the edge and the quarterline, as shown in Figures 6.209 and 6.210. When two layers of geogrid were used instead of one, the overall vertical displacement was slightly reduced, and strains in the geogrid strands were reduced by more than 15%.

6.1.12 No In-Plane Displacement

A model was developed to test the effect of in-plane displacement on the calculated strain and vertical displacement. Each geogrid junction was restrained for all movement except

in the vertical plane, and all of the non-dimensional parameters were equal to the standard case values.

Plots of the case with no in-plane displacement are shown in Figures 6.213-6.219. The results of each analysis were used to compare the effect in-plane displacement has on the vertical displacement and the strain. A solid line designates values for the model that excludes in-plane displacement and a dashed line designates values for the standard case model. Figures 6.220-6.222 show the vertical displacement values at the edge, midway between the edge and centerline, and at the centerline of the grid, respectively. Figures 6.223-6.225 show the strain values at the edge, midway between the edge and centerline, and at the centerline of the grid, respectively

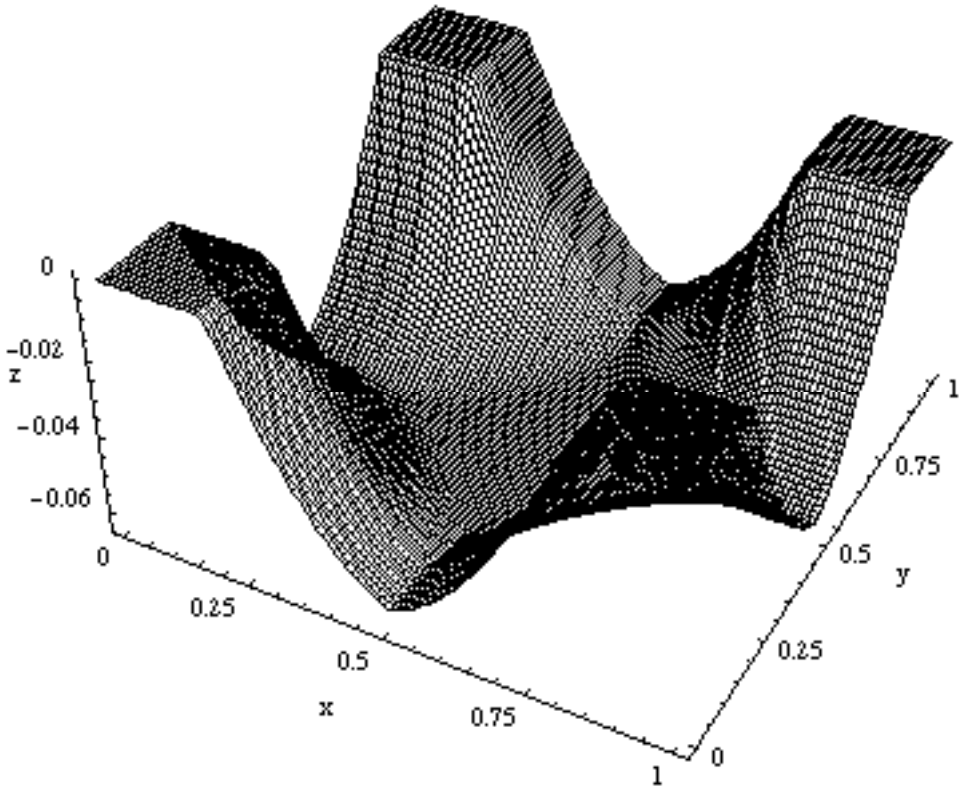

Figure 6.214 Three-Dimensional Plot

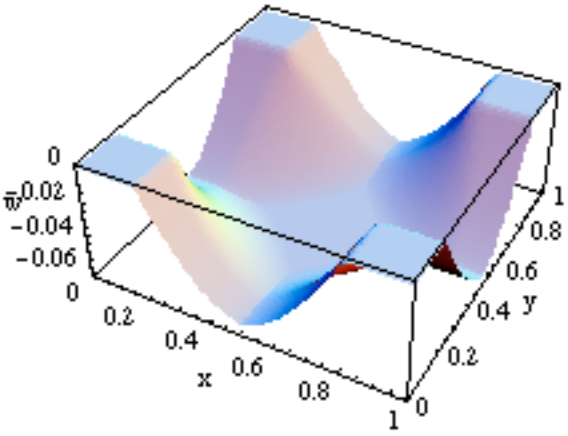

Figure 6.215 Three-Dimensional Plot of w

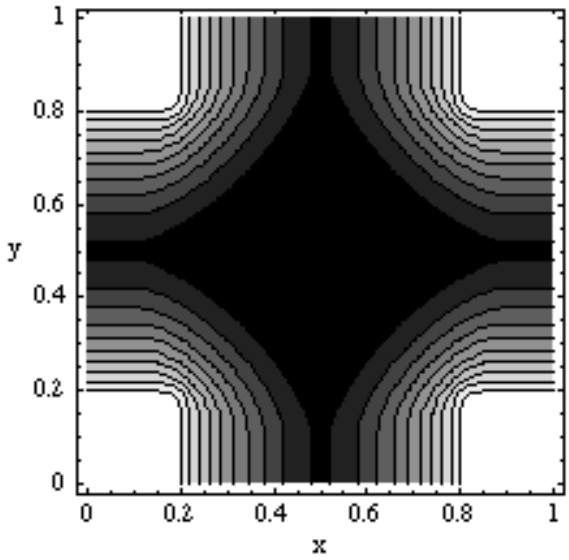

Figure 6.216 Contour Plot of w

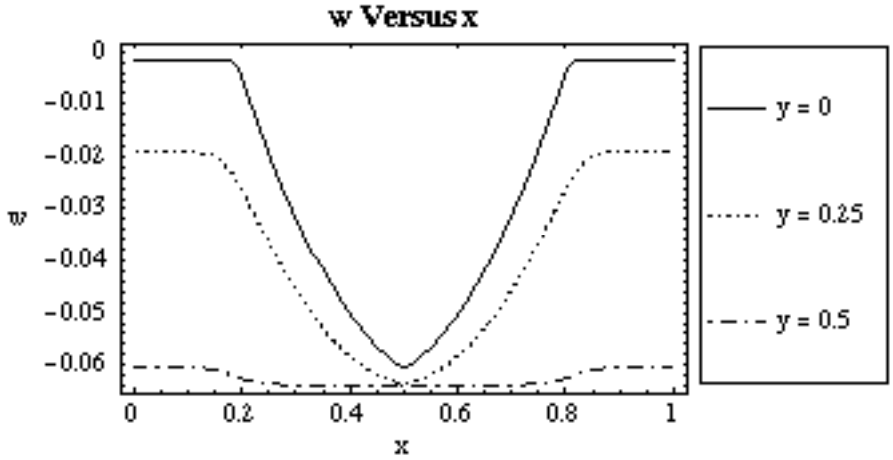

Figure 6.217 Plot of w vs. x along the Edge, Quarter and Center

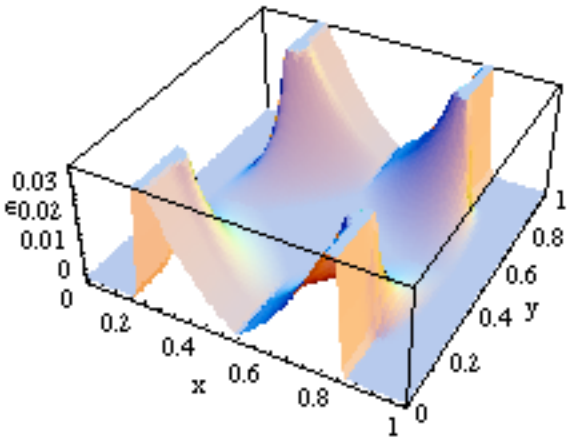

Figure 6.218 Three-Dimensional Plot of ε_x

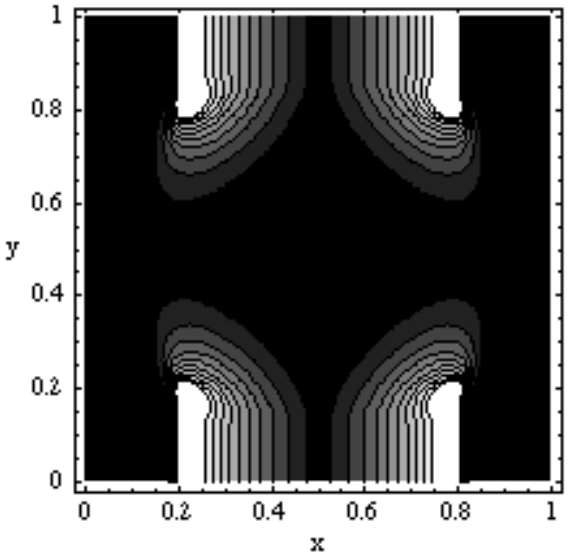

Figure 6.219 Contour Plot of ε_x

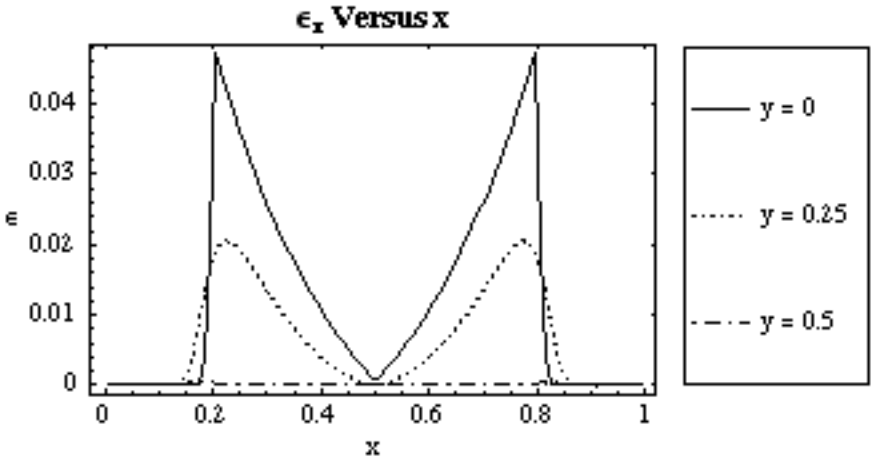

Figure 6.220 Plot of ε_x vs. x along the Edge, Quarter and Center

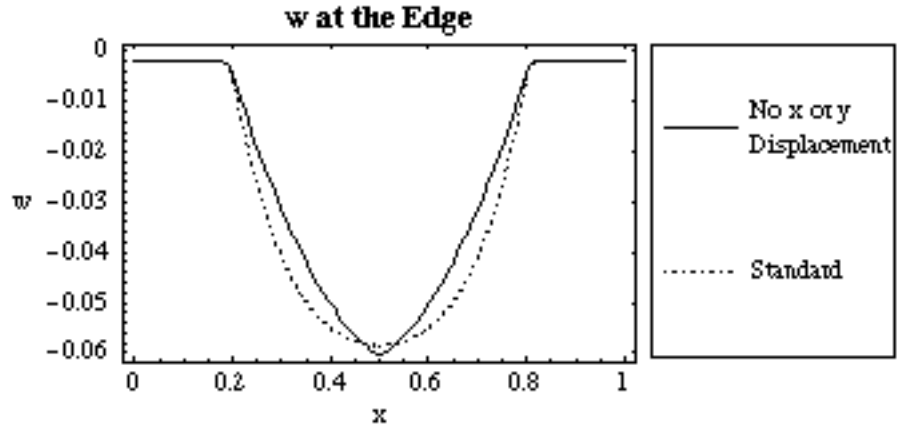

Figure 6.221 Plot of w along Edge

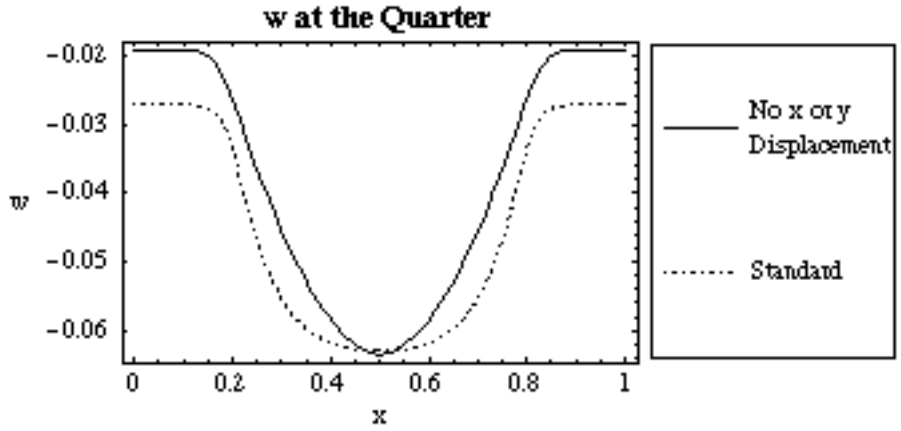

Figure 6.222 Plot of w along Quarter

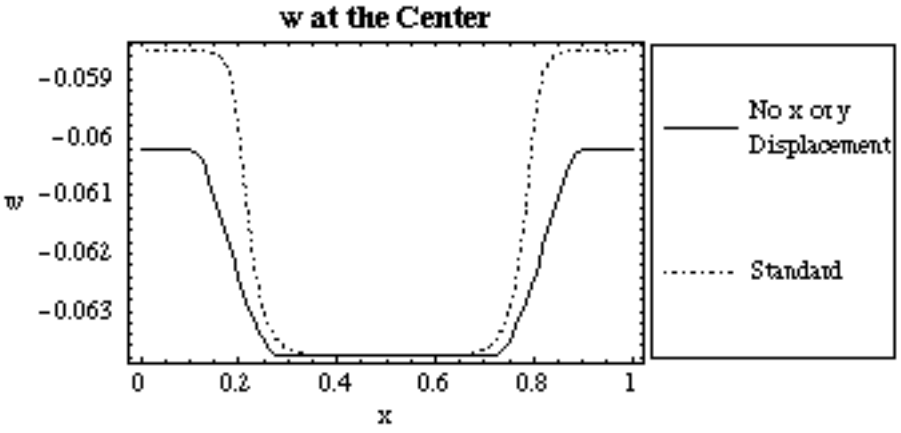

Figure 6.223 Plot of w along Center

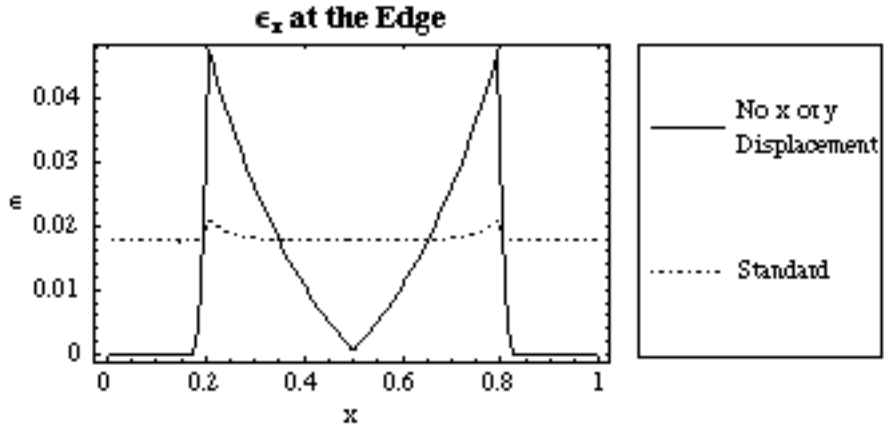

Figure 6.224 Plot of ε_x along Edge

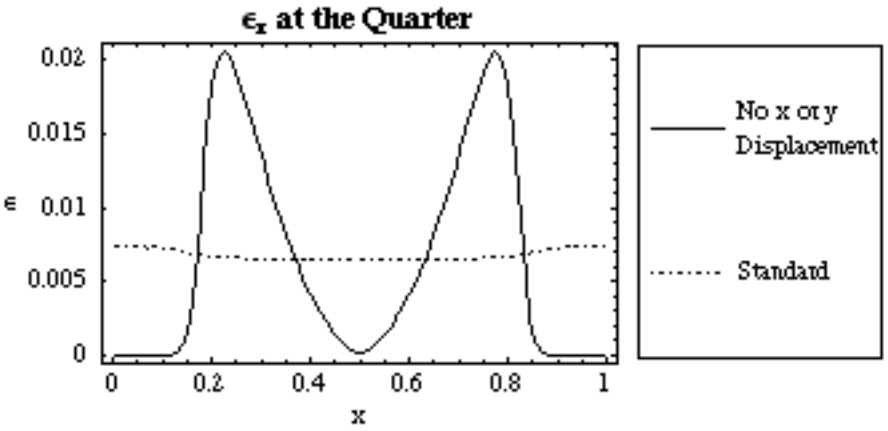

Figure 6.225 Plot of ε_x along Quarter

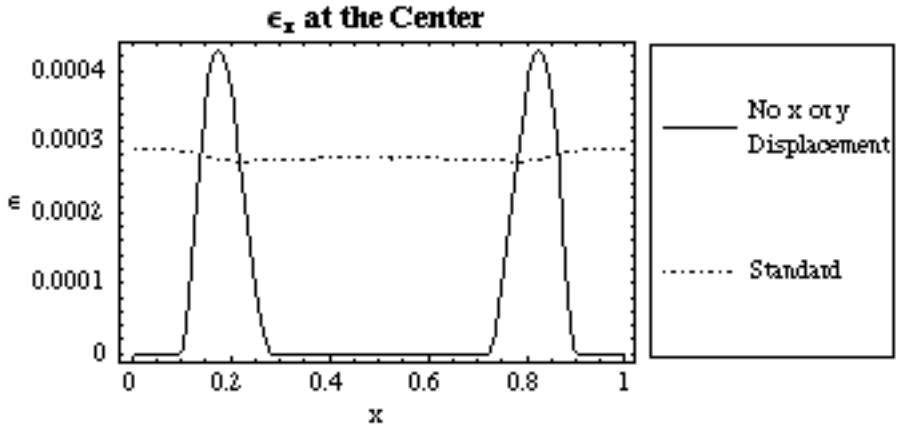

Figure 6.226 Plot of ε_x along Center

The displaced shape of the geogrid was very different when the x and y displacements were included than when they were excluded from the analysis. When in-plane

displacements were excluded, the central vertical displacement was deeper at the edge and midway between the edge and the center. Also, the slope of the deformed geogrid was more uniform. The slope of the deformed shape was not as steep near the pile, but was steeper as the strands approached the center. Both models had maximum displacements that were equal to the unreinforced settlement, but the flat region was slightly larger when the in-plane displacements were neglected.

The strain profile was very sharp at the edge of the pile and the strain value was considerably larger at this location; the magnitude was more than double. The strains directly on top of the pile and at the center of the cell were nearly zero. The strain was distributed more evenly among the strands when the in-plane displacements were included in the analysis.

Judging by the observations previously outlined, it was reasonable to assume that ignoring in-plane displacements was unsatisfactory, and was in no way accurate for the parameter values analyzed in this case study. The strain and displacement profiles were not comparable. The maximum vertical displacements were the same, but for the parameter values used, the maximum displacements could be calculated using Equation 6.11, so an analysis ignoring in-plane displacements provided no beneficial data.

6.2 Results in Dimensional Form

Thus far, the results have been shown in non-dimensional form. Non-dimensional values are very powerful in the sense that the values represent an infinite amount of dimensional case values. However, the results have to be converted to dimensional values to obtain values for actual examples. The set of dimensional parameter values used for the standard case has already been listed in the beginning of Section 6.1. With the conclusion of each analysis, the Mathematica program converted the non-dimensional results to dimensional results using Equation 6.10 and

$$T = \frac{EA\varepsilon}{H} \tag{6.15}$$

A few additional variables were used to analyze the data in this thesis. The differential settlement was discussed in the beginning of this chapter, but it was only discussed in terms of the trends seen in the vertical displacement plots. The actual equation used to find differential settlement was

$$D_S = W_{Max} - W_{Min} \tag{6.16}$$

The load distribution ratio, LDR, was used to judge the geogrid's performance in reducing the load on the soft soil between the piles. LDR is the ratio of the load supported by the soft soil to the load applied to the geogrid above the soft soil. It was found by using the following non-dimensional equation:

$$LDR = \frac{\sum_{i=1}^{ns} ts_i \cdot w_i}{\sum_{j=1}^{ns} ps_j} \tag{6.17}$$

where ns = total number of nodes on soft soil

ts = soil stiffness vector of the springs over soft soil (modified from Chapter 5)

ps = point load vector of load acting on top of soft soil (modified from Chapter 5)

If LDR =1, then the geogrid reinforcement did not support any load, and the soft soil supported the entire load between the piles. If LDR = 0, then the geogrid reinforcement supported the entire load between the piles. If there was no reinforcement, then LDR = 1, and the case LDR = 0 was impossible unless there was a soil void or no load at all.

6.2.1 Comparison of Standard Case Results to a Membrane Model

A fellow graduate student, Brent Jones, analyzed a similar model with square piles, the same standard case parameter values, and a geosynthetic reinforcement layer represented

as a membrane (Jones 2007). The membrane had some flexural rigidity, and Poisson's ratio was set equal to zero. The stiffness of the membrane was slightly larger than the stiffness of the cable net. If the stiffness per width of a membrane is equal to

$$J = E \cdot t \tag{6.18}$$

where E = modulus of elasticity

t = thickness of the membrane

and the stiffness per width of a cable net is equal to

$$J = \frac{EA}{H} \tag{6.19}$$

then J = 7.5 kN/cm for the membrane and J = 7.3 kN/cm for the cable net. Other than the stiffness of the reinforcement, all parameters were the same. Table 6.1 summarizes the results of both analyses.

Table 6.1

Comparison of Max Dimensional Values for a Cable Net and Membrane Model						
Model	**W (edge) (cm)**	**W (center) (cm)**	**T (max) (N/cm)**	**ε_X (max)**	**D_S (cm)**	**LDR**
Cable Net	-17.6	-19.1	217	0.0298	18.6	0.811
Membrane	-16.8	-19.1	296	0.0395	18.6	0.807

The maximum displacement was the same for both analyses, but the maximum displacement at the edge was larger for the cable net model. The maximum displacement of both models was equal to the unreinforced settlement. Also, the maximum tension and strain values were larger and the LDR was less for the membrane model. The overall stiffness of the membrane model was greater than that of the cable net model because of the membrane's flexural rigidity and slightly greater axial stiffness, so the overall vertical displacement of the membrane was less than that of the cable net. The larger maximum tension in the membrane may have occurred because the larger axial stiffness caused the axial force to increase. The LDR of the membrane was slightly smaller because the stiffer membrane had less overall vertical displacement. The membrane reinforcement model supported slightly more load than the cable net reinforcement model. The reinforcement supported less than 20% of the load between the piles for the membrane model and the cable net model. It was surprising that the reinforcement was that ineffective, but the

effectiveness was reduced because a large portion of the reinforcement had deformed to the unreinforced settlement. The results of both analyses were comparable, and the most significant conclusion that was obtained from this comparison was that the solutions of the two models were consistent, and were expected to be accurate.

In addition to comparing the cable net model to a membrane model, the maximum strain was compared to the approximate strain determined from a design formula. McGuire and Filz (2008) described a method of determining the approximate tension force in the geosynthetic reinforcement by utilizing two assumptions, a linearly elastic stress-strain relationship and a parabolic deflected shape that was developed for the BS8006 and Carlsson methods. The equation in terms of strain is

$$96 \cdot (J \cdot \varepsilon)^3 - 6 \cdot K_g{}^2 (J \cdot \varepsilon) - K_g{}^2 \cdot J = 0 \qquad (6.20)$$

where $K_g = \dfrac{Q_{NET}(L^2 - 4B^2)}{2B}$

$Q_{NET} = Q_S(1 - LDR)$

The standard case parameter values and K_g which was calculated to be 364 were input into Equation 6.20 and it was determined that $\varepsilon = 0.0316$. Surprisingly, the constant strain that was determined using the parabolic deflected shape was very close to the maximum strain obtained from the analysis of the cable net model.

For the following case studies, the results were organized into tables that include the maximum displacement at the edge and center, the maximum tension force per width at the edge, midway, and center of the cell, the maximum strain, the differential settlement, and LDR. Trend plots were developed for the vertical displacements and the tension forces, in which the results were plotted versus the parameter independently varied in the study.

6.2.2 Variation of Dimensional Parameter B

Table 6.2

Maximum Dimensional Values with Respect to a Variation in B								
B (cm)	W (edge) (cm)	W (center) (cm)	T (edge) (N/cm)	T (pile) (N/cm)	T (center) (N/cm)	ε_X (max)	D_S (cm)	LDR
45	-18.2	-19.1	154	217	0.848	0.0297	18.6	0.864
60	-17.6	-19.1	152	217	2.12	0.0298	18.6	0.811
75	-16.5	-19.1	149	217	5.04	0.0297	18.6	0.745

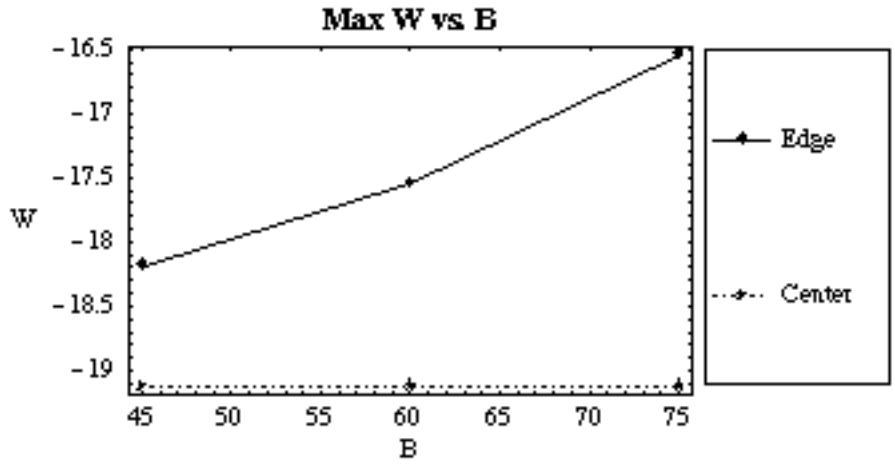

Figure 6.227 Maximum W Versus B Along the Edge and Center

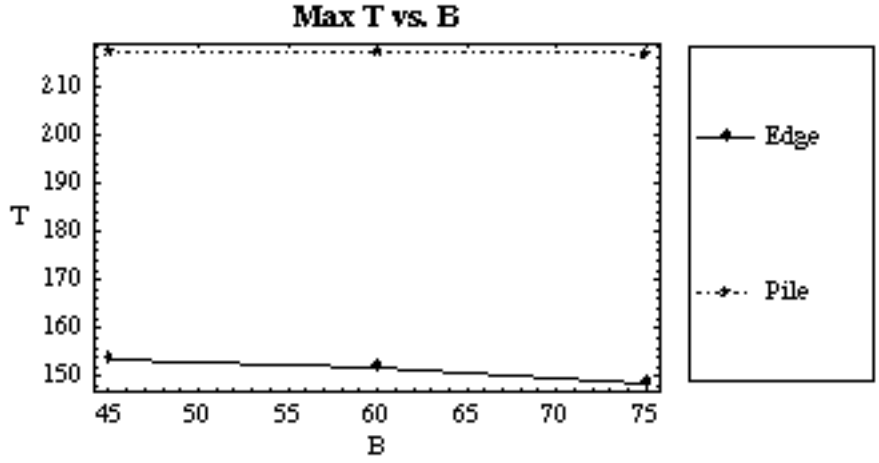

Figure 6.228 Maximum Tension Versus B Along the Edge and Pile

The maximum displacement and tension were not affected by a variation in pile width for the parameter values used in this case study. When B = 75 cm, both the maximum displacement and the maximum tension were less than one percent smaller than when B = 45 cm. These observations may have been different if the maximum displacement of all three cases were not equal to the same unreinforced settlement. The vertical displacement and the tension gradually decreased at the edge as B increased. The tension increased at the center when B increased because the flat region of the deformed shape diminished. The LDR decreased as B increased, so the geogrid reinforcement became more effective as the pile width increased. In conclusion, the maximum tension and vertical displacement were identical or almost identical for the three pile widths analyzed in this study, so it is not worthwhile to increase the pile width if a design is based solely on maximum tension and vertical displacement.

6.2.3 Variation of Dimensional Parameter Q_P

Table 6.3

Maximum Dimensional Values with Respect to a Variation in Q_P								
Q_P (N/cm^2)	W (edge) (cm)	W (center) (cm)	T (edge) (N/cm)	T (pile) (N/cm)	T (center) (N/cm)	ε_X (max)	D_S (cm)	LDR
-7.3	-17.5	-19.1	155	221	2.25	0.0303	18.9	0.807
-14.6	-17.6	-19.1	152	217	2.12	0.0298	18.6	0.811
-29.2	-17.7	-19.1	147	209	1.88	0.0286	18.1	0.820

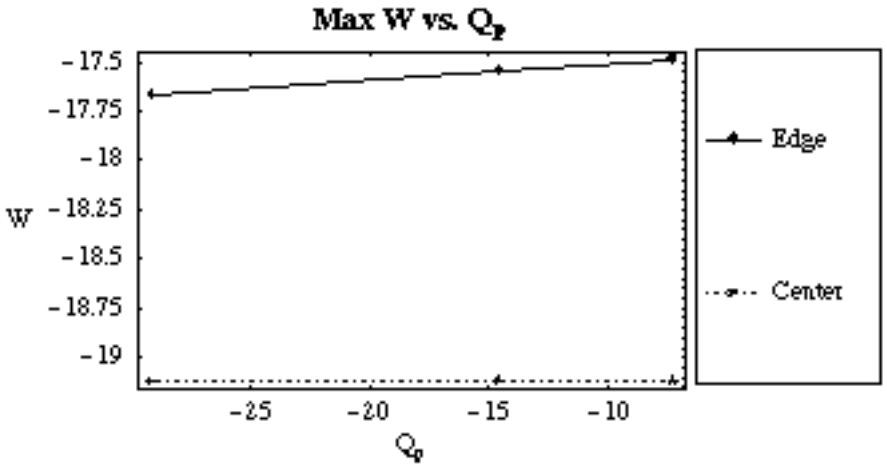

Figure 6.229 Maximum W Versus Q_P Along the Edge and Center

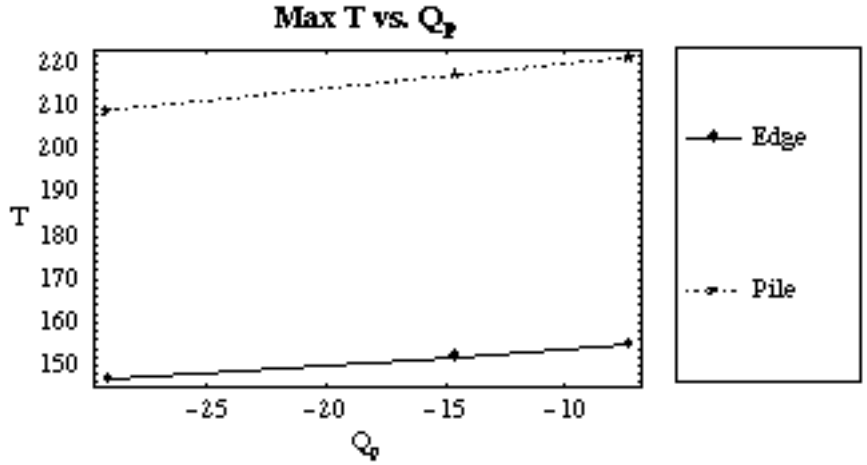

Figure 6.230 Maximum Tension Versus Q_P Along the Edge and Pile

The magnitude of the maximum displacement at the edge increased as Q_P increased. The magnitude of the displacement over the pile increased when Q_P increased, which caused the magnitude of the displacement along the edge to increase as well. The differential settlement decreased as Q_P increased because the magnitude of the displacement over the pile increased, and the difference between the two extreme displacements was reduced. As differential settlement decreased, the tension also decreased, because the strands were not required to elongate as much. The LDR increased as Q_P increased, so the geogrid became less effective as the load on the pile increased. Overall, the results did not change significantly as the load on the pile fluctuated.

6.2.4 Variation of Dimensional Parameter Q_S

Table 6.4

Maximum Dimensional Values with Respect to a Variation in Q_S								
Q_S (N/cm^2)	W (edge) (cm)	W (center) (cm)	T (edge) (N/cm)	T (pile) (N/cm)	T (center) (N/cm)	ε_X (max)	D_S (cm)	LDR
-1.53	-9.44	-9.56	57	76	0.332	0.0104	9.06	0.891
-3.06	-17.6	-19.1	152	217	2.12	0.0298	18.6	0.811
-6.12	-29.2	-38.1	379	564	28.1	0.0778	37.6	0.696

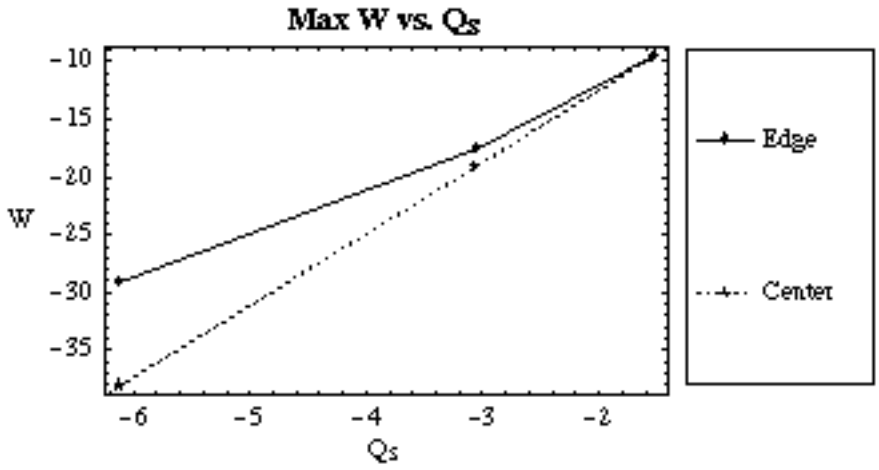

Figure 6.231 Maximum W Versus Q_S Along the Edge and Center

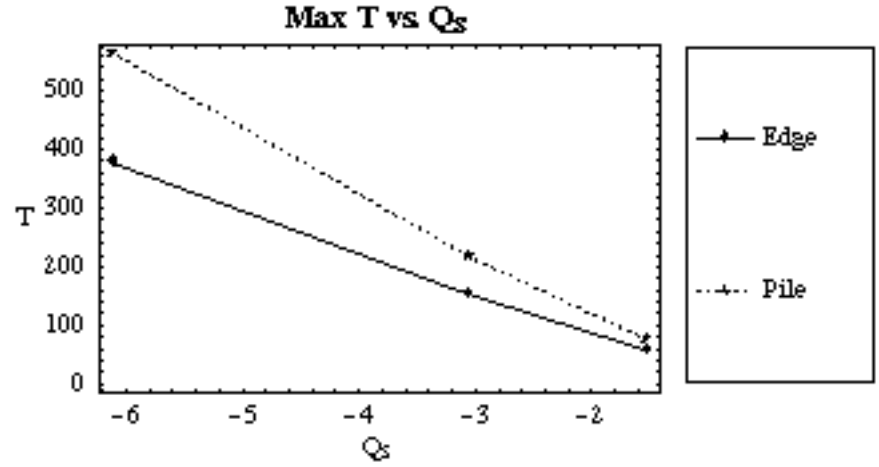

Figure 6.232 Maximum Tension Versus Q_S Along the Edge and Pile

The magnitude of the vertical displacement increased as the pressure on the soil increased. For the first two cases, Q_S = -1.53 and Q_S = -3.06, the displacements at the center were equal to the unreinforced settlement described in Equation 6.11. Since the unreinforced settlement was proportional to Q_S and the soil springs were linear, a linear relationship was expected. At Q_S = -6.12, the linear relationship did not hold; the maximum displacement fell slightly short of the unreinforced settlement. It appeared that Q_S = -6.12 was on the threshold of this linear relationship. The displacement on the edge did not reach the unreinforced settlement for any case, so the maximum displacement of the geogrid on the edge did not have a linear relationship with Q_S.

The maximum tension forces increased as Q_S increased for two reasons:

- The unreinforced settlements and the differential settlements increased which caused the strains to increase.
- The displaced shape of the geogrid had a reduced flat region as the magnitude of the unreinforced settlement increased.

The LDR decreased as the magnitude of Q_S increased, so the geogrid reinforcement supported a larger portion of the load as the load on the soil increased.

6.2.5 Variation of Dimensional Parameter K_P

Table 6.5

Maximum Dimensional Values with Respect to a Variation in K_P								
K_P (N/cm^3)	W (edge) (cm)	W (center) (cm)	T (edge) (N/cm)	T (pile) (N/cm)	T (center) (N/cm)	ε_X (max)	D_S (cm)	LDR
14.6	-17.7	-19.1	143	182	1.65	0.0249	18.1	0.828
29.2	-17.6	-19.1	152	217	2.12	0.0298	18.6	0.811
58.4	-17.4	-19.1	157	250	2.46	0.0343	18.9	0.800

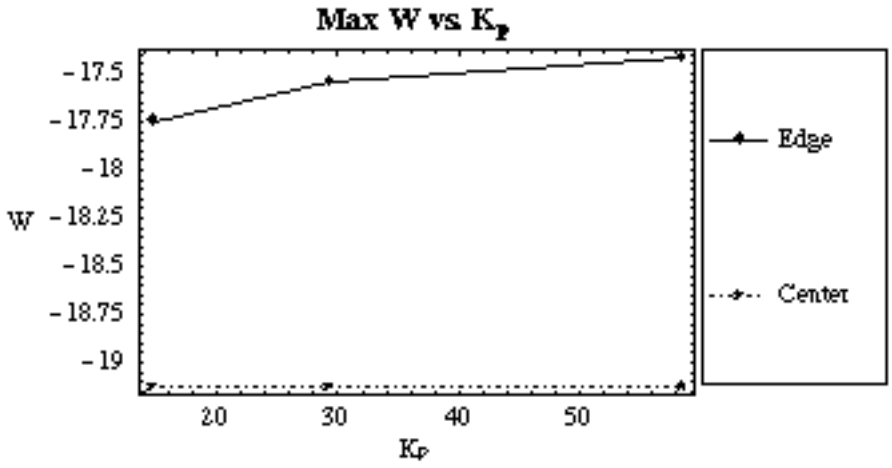

Figure 6.233 Maximum W Versus K_P Along the Edge and Center

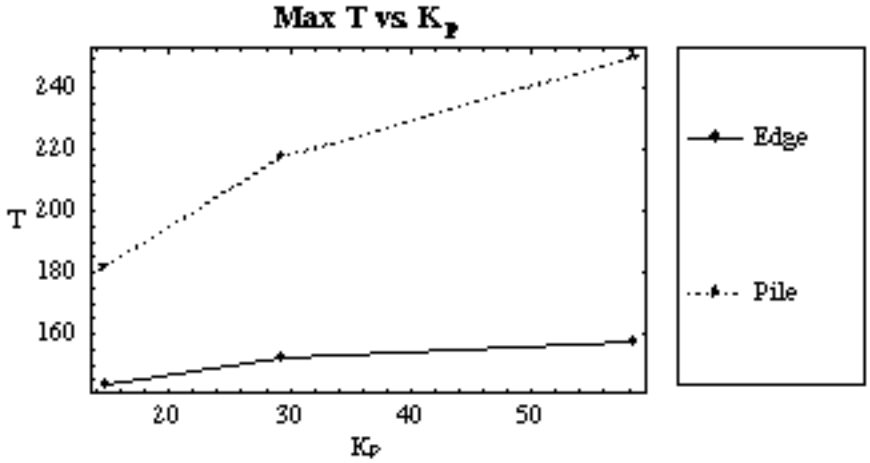

Figure 6.234 Maximum Tension Versus K_P Along the Edge and Pile

The magnitude of the maximum displacement at the edge decreased as K_P increased. When the stiffness of the soil on the pile increased, the vertical displacement of the geogrid over the pile rose and caused the differential settlement to increase. The

maximum displacement was constant throughout this case study and the displacement over the pile rose, so the difference in maximum and minimum displacements increased. The maximum tension increased as K_P increased, and once again there was a correlation between an increasing maximum tension and an increasing differential settlement.

6.2.6 Variation of Dimensional Parameter K_S

Table 6.6

Maximum Dimensional Values with Respect to a Variation in K_S								
K_S (N/cm^3)	W (edge) (cm)	W (center) (cm)	T (edge) (N/cm)	T (pile) (N/cm)	T (center) (N/cm)	ε_X (max)	D_S (cm)	LDR
0.08	-25.4	-36.9	261	427	37.4	0.0585	36.4	0.621
0.16	-17.6	-19.1	152	217	2.12	0.0298	18.6	0.811
0.32	-9.53	-9.56	73.5	93.6	0.332	0.0128	9.06	0.918

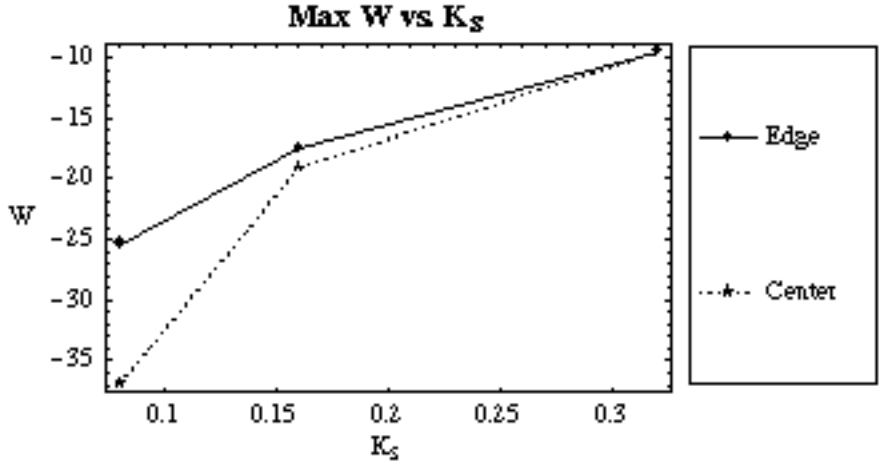

Figure 6.235 Maximum W Versus K_S Along the Edge and Center

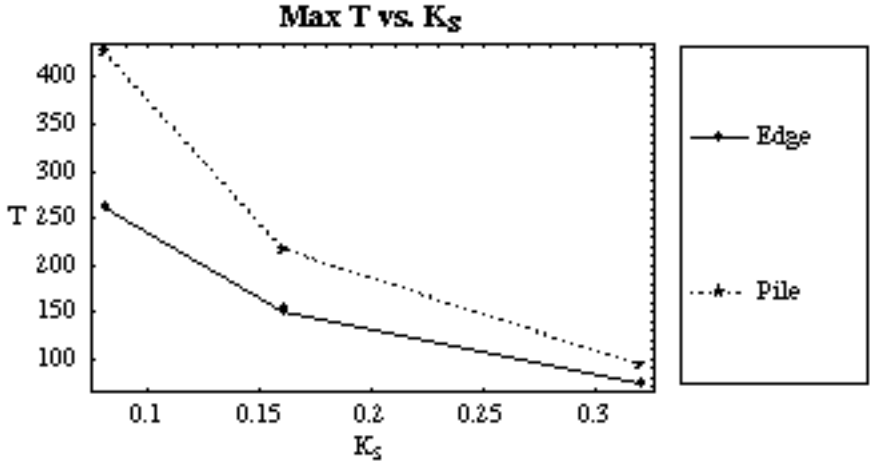

Figure 6.236 Maximum Tension Versus K_S Along the Edge and Pile

The magnitude of the maximum vertical displacement increased as the stiffness of the soil decreased. For the first two cases, $K_S = 0.08$ and $K_S = 0.16$, the displacement at the center was equal to the unreinforced settlement described in Equation 6.11. Since the unreinforced settlement was dependent on K_S and the soil springs were linear, a linear relationship was expected. When $K_S = 0.32$, the linear relationship did not hold, because the maximum displacement fell short of the unreinforced settlement. It seemed evident that this soil stiffness value was near the threshold of the linear relationship. The maximum displacement on the edges did not reach the unreinforced settlement for any case, so the maximum displacements at the edges and K_S did not have a linear relationship.

The maximum tension force and differential settlement increased when K_S decreased. Large tensile forces were caused when the strands stretched more to make up for an increase in differential settlement.

As K_S decreased the LDR decreased, so the geogrid reinforcement supported more load as the soil stiffness decreased. There was a significant improvement in geogrid performance when $K_S = 0.08$, and it could be attributed to the maximum displacement being less than the unreinforced settlement.

6.2.7 Variation of Dimensional Parameter H

Table 6.7

Maximum Dimensional Values with Respect to a Variation in H								
H (cm)	W (edge) (cm)	W (center) (cm)	T (edge) (N)	T (pile) (N)	T (center) (N)	ε_X (max)	D_S (cm)	LDR
1.5	-16.3	-19.1	223	295	8.29	0.0210	18.6	0.766
3	-17.6	-19.1	152	217	2.12	0.0298	18.6	0.811
6	-18.4	-19.1	103	141	0.452	0.0387	18.6	0.852

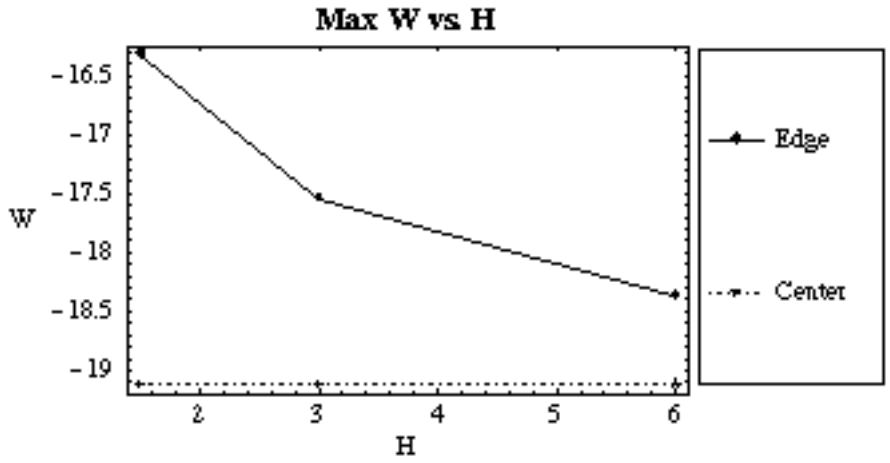

Figure 6.237 Maximum W Versus H Along the Edge and Center

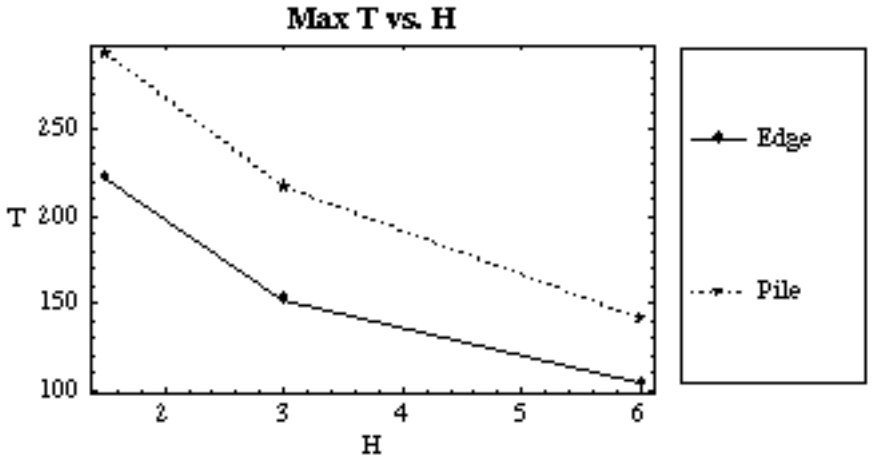

Figure 6.238 Maximum Tension Versus H Along the Edge and Pile

The stiffness of the geogrid increased as H decreased, as shown in the calculation of J: J = 14,600 N/cm if H = 1.5 cm, J = 7,300 N/cm if H = 3 cm, and J = 3,650 N/cm if H = 6 cm. The increased stiffness resulted from an increased concentration of strands as H decreased. The vertical displacement at the edge rose as H decreased. This trend was a result of an increase in stiffness. Even when H was smallest, 1.5 cm, the maximum displacement was still equal to the unreinforced settlement. The maximum tensile force decreased as H decreased, because the forces were distributed among more strands. However, the maximum tensile force increased at the center when H decreased. This occurred because the displaced shape of the geogrid had a smaller flat region when H decreased. The tensile forces at the center were so small in comparison to the maximum tensile force that this feature was hardly worth noting.

6.2.8 Anisotropic Geogrid with a Variation of Dimensional Parameter H_X

Table 6.8a

Maximum Dimensional Values w/ Respect to Variation in H_X						
H_X (cm)	W_X (edge) (cm)	W_Y (edge) (cm)	W (center) (cm)	ε_X (max)	D_S (cm)	LDR
1.5	-17.6	-16.3	-19.1	0.0306	18.6	0.789
3	-17.6	-17.6	-19.1	0.0298	18.6	0.811
6	-17.5	-18.4	-19.1	0.0411	18.6	0.832

Table 6.8b

Maximum Dimensional Values with Respect to a Variation in H_X						
H_X (cm)	T_X (edge) (N/cm)	T_Y (edge) (N/cm)	T_X (pile) (N/cm)	T_Y (pile) (N/cm)	T_X (center) (N/cm)	T_Y (center) (N/cm)
1.5	152	223	224	289	5.37	3.17
3	152	152	217	217	2.12	2.12
6	152	164	208	150	0.737	1.54

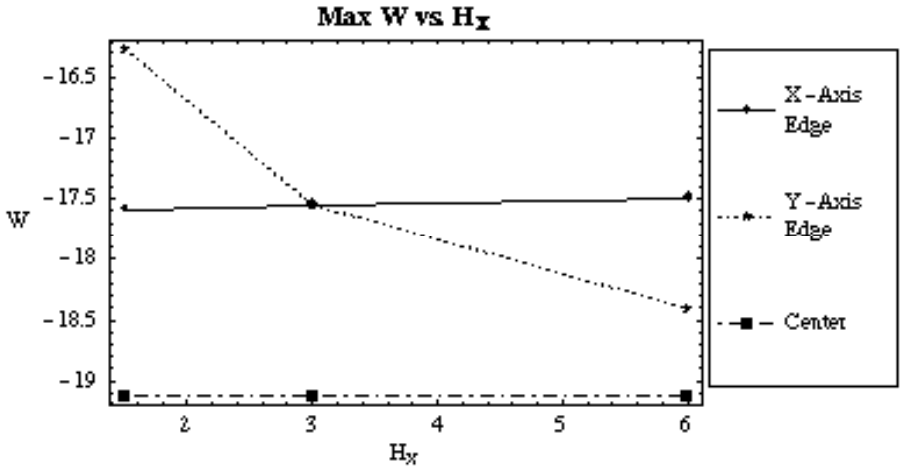

Figure 6.239 Maximum W Versus H_X Along the Edge and Center

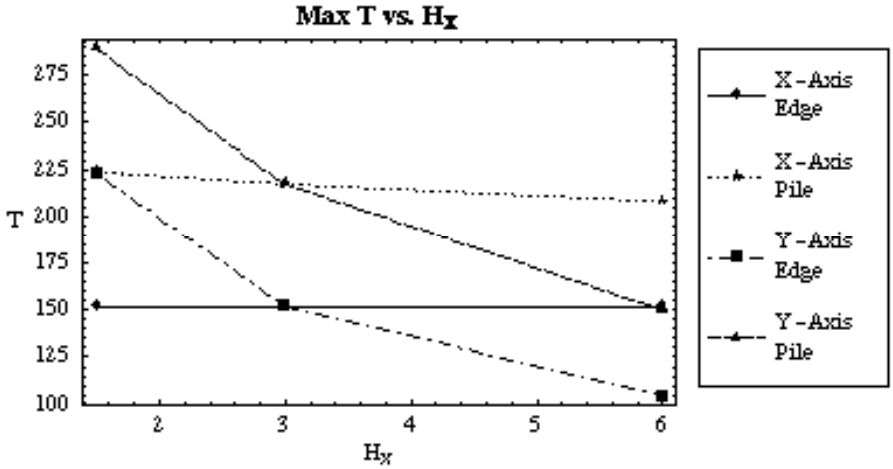

Figure 6.240 Maximum Tension Versus H_X Along the Edge and Pile

The stiffness of the geogrid along the Y-axis increased as H_X decreased, as shown in the calculation of J_Y: J_Y = 14,600 N/cm if H_X = 1.5 cm, J_Y = 7,300 N/cm if H_X = 3 cm, and J_Y = 3,650 N/cm if H_X = 6 cm, while J_X = 7,300 N/cm for all cases. There was a gentle rise in vertical displacement on the X-axis edge and a significant drop in vertical displacement on the Y-axis edge as H_X increased. The drop in displacement was expected because the stiffness along the Y-axis decreased. The maximum displacement remained equal to the unreinforced settlement for all cases in this study.

The maximum tension along the X-axis edge was relatively unchanged, and the tension along the X-axis pile edge and center decreased as H_X increased. Less load bore on each strand as the strand length decreased because the tributary area of the strand decreased. It was actually surprising that the tension force did not decrease more as H_X increased. However, as H_X increased, the strands along the X-axis became stiffer in comparison to the strands along the Y-axis, so the X-axis strands attracted more force. The maximum tension in the strands along the Y-axis decreased as H_X increased. As H_X became small, the stiffness of the geogrid along the Y-axis increased, and caused the geogrid to attract more load. This point was further proven because LDR decreased as H_X decreased.

6.2.9 Dimensional Results For a Variation of Orientation

Table 6.9

Maximum Dimensional Values with Respect to Orientation								
Angle (Degrees)	W (edge) (cm)	W (center) (cm)	T (edge) (N/cm)	T (pile) (N/cm)	T (center) (N/cm)	ε_X (max)	D_S (cm)	LDR
45	-18.8	-18.9	107.92	184.96	113.8	0.0253	18.4	0.838
0	-17.6	-19.1	152.07	217.25	2.12	0.0298	18.6	0.811

The maximum vertical displacement along the edge was significantly greater when the geogrid was oriented 45 degrees to the pile, but the maximum vertical displacement was less. The opposite was true for the tension: the maximum tension forces along the edge and pile were considerably smaller, but the maximum tension at the center was several magnitudes larger for the 45-degree orientation. These observations supported the conclusion that the 45-degree geogrid orientation distributed the displacement and force more evenly. The maximum tension was smaller when the geogrid was oriented 45 degrees to the x-axis. Angling the geogrid 45 degrees is a promising method of reducing the maximum vertical displacement and tension. However, the performance of the geogrid oriented 45 degrees to the pile edge was worse than a geogrid with ribs parallel to the pile edge, as seen in the comparison of the LDR for both cases. The overall vertical displacement of a geogrid oriented 45 degrees to the pile edge was greater than that for a geogrid with ribs parallel to the pile edge, and therefore the soil supported more load for this case.

6.2.10 Dimensional Results For a Variation of Joint Stiffness C

Table 6.10

Maximum Dimensional Values with respect to a Variation in C								
C	W (edge)	W (center)	T (edge)	T (pile)	T (center)	ε_x (max)	D_s	LDR
(MN-cm)	(cm)	(cm)	(N/cm)	(N/cm)	(N/cm)		(cm)	
0	-17.6	-19.1	152	217	2.12	0.0298	18.6	0.811
2.19	-17.8	-19.3	107	114	0.65	0.0176	19.1	0.782
21.9	-15.4	-18.4	65.1	71.1	1.83	0.0101	18.3	0.668
219	-7.5	-10.2	13.2	11.1	1.51	0.00184	10.2	0.358

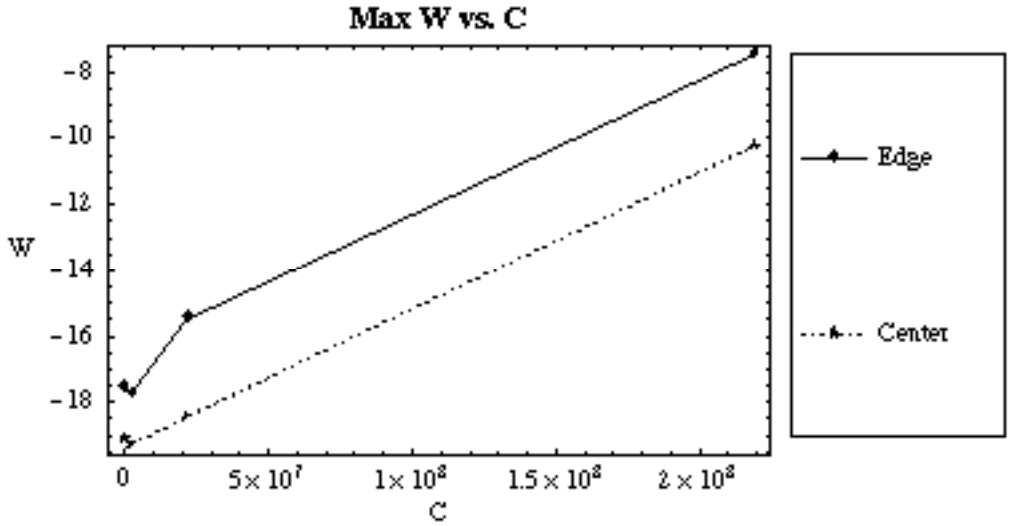

Figure 6.241 Maximum W Versus C Along the Edge and Center

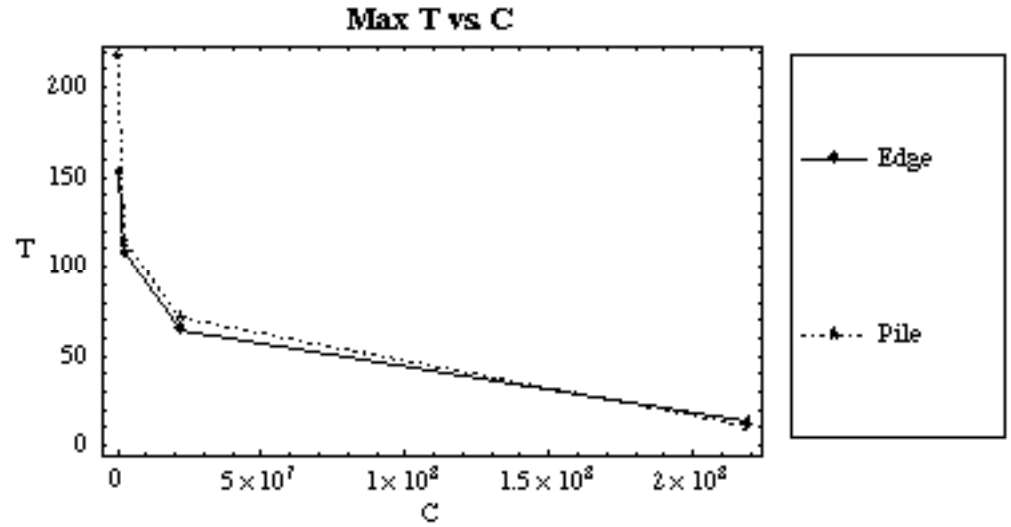

Figure 6.242 Maximum Tension Versus C Along the Edge and Pile

When C was greater than 21.9 MN-cm, the magnitude of the maximum vertical displacement decreased as C increased. When C was less than 2.19 MN-cm, the magnitude of the maximum vertical displacement increased as C increased, because large angles between adjacent strands were mitigated when rotational stiffness was added to the joints. As a result, the deformed shape had a more gradual change in slope and there was no flat region. The more gradual slope caused the maximum displacement to exceed the unreinforced settlement, and the tension decreased when C = 2.19 MN-cm. As C increased, the maximum tension in the strands decreased. The reduction in tension could be attributed to the decrease in displacement, gradual change in slope, and the large amount of energy that the joint stiffness absorbed. The LDR significantly decreased as C increased, because the overall displacement decreased. Adding rotational stiffness to a geogrid improved the performance of the geogrid.

A brief summary of a few important trends that were observed in the joint stiffness case study are listed below:

- The maximum vertical displacement was not immediately reduced as the values of C increased. If C = 2.19 MN-cm, the joint stiffness actually increased the maximum displacement, but when C = 21.9 MN-cm, the maximum vertical displacement was significantly reduced, so a value of C existed between 2.19 MN-cm and 21.9 MN-cm at which the maximum displacement was largest.
- The strain energy was reduced as C increased because energy was stored by the rotational springs.

6.2.11 Dimensional Results For Two Layers of Geogrid Reinforcement

Table 6.11

Maximum Dimensional Values of a Two Geogrid Layer Model								
Geogrid	**W (edge) (cm)**	**W (center) (cm)**	**T (edge) (N/cm)**	**T (pile) (N/cm)**	**T (center) (N/cm)**	**e_X (max)**	**D_S (cm)**	**LDR**
One	-17.6	-19.1	152	217	2.12	0.0298	18.6	0.811
Top	-16.5	-19.2	100	112	4.07	0.0162	18.2	
Bottom	-16.4	-19.1	122	185	4.07	0.0253	18.1	0.769

The maximum displacement along the edge of both geogrids in a two-layer model was less than for a one-layer model, and the maximum displacements were approximately equal. The maximum tension was reduced when two layers of geogrid were used instead of one. A two-layer geogrid model cannot exhibit beam behavior because geogrids are subject to large deformations and are not capable of resisting compression forces. However, the bottom geogrid layer in this model did have the largest tension force, similar to a simply-supported beam loaded with a downward pressure. The LDR of the two-layer model was approximately 5% less than for the one-layer model, so two layers of geogrid reinforcement had a moderate improvement in performance.

6.1.12 Dimensional Results For No In-Plane Displacement

Table 6.12

Maximum Dimensional Values with No In-Plane Displacement								
X And Y	**W (edge)**	**W (center)**	**T (edge)**	**T (pile)**	**T (center)**	**ε_X (max)**	**D_S**	**LDR**
No	-18.1	-19.1	1037	1495	9.38	0.0715	18.6	0.748
Yes	-17.6	-19.1	456	652	6.37	0.0298	18.6	0.811

The vertical displacement was considerably deeper at the edge when the in-plane displacement was neglected. The maximum tension increased at the edge, pile, and center; in fact, it doubled in magnitude along the edge and pile when the in-plane displacement was excluded. These observations support the conclusion that it was not appropriate to ignore the in-plane displacements.

Chapter 7
Summary and Conclusions

7.1 Summary

Geogrid reinforcement was first used for soil improvement in Europe almost forty years ago. The successful use of geogrid reinforcement has encouraged its widespread use for various applications in the United States. One such application has been examined, in which piles, driven into soft clay, are covered with a composite "bridging" layer made up of granular soil and one or multiple layers of geogrid. This geogrid-reinforced pile-supported (GRPS) foundation system has been constructed in several locations throughout the United States to support embankments, yet there is significant disagreement among both the professional and academic community on how to properly design it.

The purpose of this research was to investigate the performance of the geogrid in a GRPS system by utilizing a three-dimensional model in which the geogrid reinforcement was represented as a linearly elastic cable net with properties based on actual values provided in the geogrid product literature. A geogrid is made up of woven ribs that are laid at a set spacing in two orthogonal directions, so a cable net is an ideal model for a geogrid. To the author's knowledge, a geogrid has only been modeled as a cable net once, and very few three-dimensional analyses have been performed of GRPS embankment systems. The cable net was initially flat and was supported by soil that was represented as linearly elastic springs of comparable compressive stiffness to soil. Friction and the lateral soil stiffness between the geogrid and the soil were ignored for this analysis. It is common practice in the analysis of cable nets with short cable lengths to approximate a uniform pressure as point loads that act at the cable net junctions. So the uniform pressure was applied as equivalent point loads, and the soil was modeled as springs that act at each cable net junction. In addition, the loads were strengthened at the pile locations to account for arching effects, and the springs were stiffened at the pile locations because only a thin layer of soil exists between the pile cap and the geogrid.

The three-dimensional model consisted of a square unit cell with four square piles in each corner, in which the centerlines of the piles made up the edges of the cell. Only one cell was modeled because it would have been redundant to examine more. The location of this cell permited the perimeter of the cell to be restrained in the direction perpendicular to each edge.

Two different Huesker geogrid products were tested, Fornit and Fortrac, to prove that geogrids are made up of strands that behave approximately linear. Each sample was cut so that only one or two ribs would be loaded in tension. The plots of the relationship between tension per unit width and strain were determined to be approximately linear as load was increased. Also, most of the geogrid samples ruptured at a strain of approximately 10%.

A geogrid is initially flat and it may undergo large deformations under the embankment loading, so it was necessary to solve the model using a non-linear method. Several methods to analyze cable net structures were investigated, and the minimization of energy method was chosen as the best method to solve this highly geometrically non-linear structure. Mathematica Version 5.2 was used to implement this method, which was tested by solving four well-established cable net problems. A good agreement was obtained between the nodal displacements determined using this analysis and those calculated by other researchers, so the validity of the method was verified.

An algorithm was programmed in Mathematica to determine the total energy in the system by summing the strain energy of the cable net, strain energy of the springs, and potential energy of the point loads. By taking advantage of symmetry, geometrically redundant degrees of freedom were eliminated causing a reduction in computation time and computer memory. An isotropic cable net was reduced to one-eighth of its full size and an anisotropic cable net was reduced to one-fourth of its full size. Boundary conditions were required at certain nodes, in order for the displacements of the reduced cable net to be equivalent to the displacements of the full-size cable net. The algorithm

was made up of several routines that formed matrices to store the node position coordinates, cable end nodes, nodal displacements, cable stiffnesses, spring stiffnesses, and point loads. Then the algorithm was used to form the total energy equation and minimize it. The solution determined the nodal displacements required for the system to be in equilibrium, and other routines were developed to find tensile forces and strains from these displacements. Modifications were made to the Mathematica algorithm so that a model with an anisotropic geogrid, a rotated geogrid orientation, a geogrid with rotational stiffness, and multiple geogrid layers could be analyzed.

The algorithm was used to conduct a parametric study to determine important parameters, so that the design of geogrid reinforcement could be optimized. A standard case was determined based on actual parameters used in GRPS embankment design. The parametric study varied the standard case parameters B (half the pile cap width), Q_P (pressure on the geogrid over the piles), Q_S (pressure on the geogrid over the soil), K_P (stiffness of the soil on top of the piles), K_S (stiffness of the soil) and H (rib length of the geogrid), and also studied cases with an anisotropic geogrid in which the H_X (rib length along the X-axis) varied, a geogrid rotated 45-degrees, a geogrid with rotational stiffness of various values C, and two layers of geogrid reinforcement. Dimensionless results of these case studies were used to create a plot of the three-dimensional deformed shape, three-dimensional and contour plots of the vertical displacement w, two dimensional plots of w along the edge, quarterline, and centerline of the cell, three-dimensional and contour plots of the strain ε_x, and two-dimensional plots of ε_x along the edge, quarterline, and centerline of the cell. Additionally, the results were converted to dimensional form and the maximum values of W (dimensional vertical displacement), T (tension per unit width), ε_x, D_S (differential settlement) and LDR (load distribution factor) were compared to a membrane model. The maximum strain was compared to the strain calculated using a design formula with the standard case parameter values. The maximum dimensional results for each case study were presented in tables and in trend plots of W and T.

The results of the parametric study are summarized below:

- The tension per unit width and strain along the edge significantly decreased as the embankment pressure on the soil decreased, soil stiffness increased, and rotational stiffness increased.
- The tension per unit width and strain along the edge slightly decreased as the pile width increased, embankment pressure above the pile increased, and stiffness of the soil on top of the pile decreased. Also, the tension and strain decreased slightly if the geogrid was rotated 45-degrees or two layers of geogrid were used.
- The tension per unit width slightly decreased along the edge as the cable length increased, and the strain in the ribs decreased along the edge as the cable length decreased.
- The maximum tension per unit width and maximum strain significantly decreased as the embankment pressure on the soil decreased, soil stiffness increased, and rotational stiffness increased.
- The maximum tension per unit width and maximum strain decreased as the stiffness of the soil above of the pile decreased, and slightly decreased as the embankment pressure above the pile increased. Also, the maximum tension and maximum strain decreased if the geogrid was rotated 45-degrees or two layers of geogrid were used.
- The maximum tension per unit width decreased as the cable length increased, and the maximum strain in the ribs decreased as the cable length decreased.
- The vertical displacement along the edge significantly decreased as the pile width increased, embankment pressure on the soil decreased, soil stiffness increased, and cable length decreased. Also, the vertical displacement along the edge significantly decreased if two layers of geogrid were used, and increased if the geogrid was rotated 45-degrees.
- The vertical displacement along the edge slightly decreased as the embankment pressure above the pile decreased, and the stiffness of the soil on top of the pile increased.
- The vertical displacement along the edge slightly increased if C = 2.19 MN-cm and significantly decreased if C was greater than 21.9 MN-cm.

- The maximum vertical displacement significantly decreased as the embankment pressure above the soil decreased, and as the soil stiffness increased.
- The maximum vertical displacement slightly decreased if the geogrid was rotated 45-degrees.
- The maximum vertical displacement slightly increased if C = 2.19 MN-cm and significantly decreased if C was greater than 21.9 MN-cm.
- The maximum vertical displacement was not affected by a fluctuation in pile width, embankment pressure above the pile, stiffness of the soil on top of the pile, cable length, or if there were two layers of geogrid.
- The load distribution ratio significantly deceased as the embankment pressure on the soil increased, and as soil stiffness decreased.
- The load distribution ratio decreased as the pile width increased and the rib length decreased, and slightly decreased as the stiffness of the soil on top of the pile increased, and the embankment pressure above the pile decreased. Also, the load distribution ratio slightly decreased if two layers of geogrid were used, and increased if the geogrid was rotated 45-degrees.

7.2 Conclusions

A few conclusions were made from the results of the parametric study. The maximum strain and tensile force occurred at the corner of the pile edge, and the largest strains and tensile forces were concentrated on top of and around the perimeter of the pile. The maximum vertical displacement occurred in the center of the cell. A correlation between the differential settlement and the strain was apparent: if the differential settlement increased, the strands in the geogrid stretched more, causing higher strain. The load distribution ratios were very high for this model, so the soil supported the majority of the embankment pressure that was applied to the area between the piles. The high load distribution ratio was surprising and the accuracy of this model could be questioned because of this fact. Another observation that could cause questions about the accuracy of the model was that the maximum displacement was virtually unchanged for all analyses, with the exception of the soil pressure and soil stiffness case studies. For the parameter

values used in this model the geogrid did not affect the maximum vertical displacement, because the maximum vertical displacement was equal to the displacement that would have occurred if the model had no geogrid.

The three different pile widths that were analyzed using this model showed that the pile width had no effect on the maximum vertical displacement or the maximum strain. However, the geogrid reinforcement supported more of the load between the piles when the pile width was increased.

As the strand length decreased, the stiffness of the geogrid increased, causing the tension per unit width to increase, and the geogrid to support more load. However, the maximum displacement was not affected. Geogrid performance was improved by decreasing the strand length.

When the geogrid was positioned so that the ribs were 45-degrees to the pile edge the maximum displacement and tensile force was reduced, and the force and displacement was more evenly distributed amongst the entire geogrid. However, the performance of the geogrid was reduced slightly because the geogrid didn't support as much of the load between the piles.

It was difficult to interpret if adding rotational stiffness to the geogrid would decrease displacements, because the rotational stiffness constant of actual geogrid reinforcement was unknown. If the rotational stiffness is less than 2.19 MN-cm, then the maximum displacement will not decrease. However, if rotational stiffness is added to the geogrid the tension will decrease and the amount of load the geogrid supports will increase, so it is advantageous to add rotational stiffness.

Adding an additional layer of geogrid improved the performance slightly by reducing the displacements, reducing the tension, and supporting more of the load. The tension forces were larger on the bottom geogrid.

Displacements in the plane of the geogrid should always be included in the analysis because the results for a model without in-plane displacements were not similar to those for a model with in-plane displacements.

The cable net model and the membrane model (Jones 2007) yielded similar results, so the validity of both models was verified. The strain determined using an accepted design strain equation was slightly larger than the maximum strain determined using the cable net model. The design equations used to determine the strain in the geosynthetic reinforcement should be reevaluated, and new methods of determining the strains should be investigated.

7.3 Recommendations

This model could be improved in several ways. The shear stiffness of the soil could be taken into account by adding a Pasternak shear layer in addition to the vertical springs. The computation time would not be increased significantly if a Pasternak shear layer were added. Also, the soil could be better represented as non-linear springs instead of linear springs. If non-linear springs were added, the computation time would significantly increase, and it is unclear if adding non-linear springs would improve the analysis because soil parameter values are not known exactly.

The soil arching effect could be improved by adding a non-uniform pressure distribution over the soil. Soil arching does not cause a uniform pressure distribution, and the pressure in the center of the cell is different than the pressure near the pile caps.

Finally, a finite element analysis could be conducted to verify the accuracy of this model. At first, a model was attempted in SAP2000 for a cable net structure, but the geometric non-linearity of the model caused computational issues and the model was discarded. The model could be solved with patience, a large-deformation analysis, and an appropriate incremental load.

References

Agaiby, S. W. and Jones, C. J. F. P. (1995). "Design of reinforced fill systems over voids," *Canadian Geotechnical Journal*, 32, 939-945.

Buchholdt, H. A. (1985). *Introduction to Cable Roof Structures*. Cambridge University Press, Cambridge, U.K.

Burd, H. J. (1995). "Analysis of membrane action in reinforced unpaved roads," *Canadian Geotechnical Journal*, 32, 946-956.

Cao W. P., Chen Y. M., and Chen R. P. (2006). "An analytical model of piled reinforced embankments based on the principle of minimum potential energy,"*Geotechnical Special Publication 151*, *Advances in Earth Structures: Research to Practice - Proceedings of the GeoShanghai Conference 2006*, 217-224.

Chen C. F., and Yang, V. (2006). "Research on bearing capacity of geosynthetic-reinforced and pile-supported earth platforms over soft soil and analysis of its affecting factors,"*Geotechnical Special Publication 151*, *Advances in Earth Structures: Research to Practice - Proceedings of the GeoShanghai Conference 2006*, 294-301.

Deb, K., Chandra, S. and Basudhar, P. K. (2005). "Settlement response of a multilayer geosynthetic-reinforced granular fill-soft soil system,"*Geosynthetics International*, 12, 288-298.

Espinoza, R. (1993). "Soil-geotextile interaction: evaluation of membrane support," *Geotextiles and Geomembranes*, 13, 281-293.

Fakher, A. and Jones, C. J. F. P. (2001). "When the bending stiffness of geosynthetic reinforcement is important," *Geosynthetics International*, 8, 445-460.

Han, J. and Gabr, M. A. (2002). "Numerical analysis of geosynthetic-reinforced and pile-supported earth platforms over soft soil," *Journal of Geotechnical and Geoenvironmental Engineering,* 128, 44-53.

Huang, J., Han, J., and Collin, J. G. (2005). "Serviceability analysis of geomembrane in geosynthetic-reinforced column-support platforms," *Geotechnical Special Publication 151*, *Advances in Earth Structures: Research to Practice - Proceedings of the GeoShanghai Conference 2006*, 211-216.

Jones, M. B. (2007). "Three-dimensional finite difference analysis of geosynthetic reinforcement used in column-supported embankments," M.S. Thesis. Virginia Polytechnic Institute and State University, Blacksburg, Virginia.

Koerner, R. M. (2005) *Designing with Geosynthetics*. New Jersey: Prentice-Hall, Inc.

Lewis, W. J. (1987). "A comparative study of numerical methods for the solution of pretensioned cable networks," *Proceedings of the International Conference on the Design and Construction of Non-Conventional Structures,* London, England, 2, 27-33.

Lewis, W. J. (1989). "The efficiency of numerical methods for the analysis of prestressed nets and pin-jointed frame structures," *Computers and Structures*, 33, 791-800.

Love, J. and Milligan, G. (2003). Design methods for basally reinforced pile-supported embankments over soft ground. *Ground Engineering*, March, 39-43.

Mazursky, L. A. (2006). "Three-dimensional analysis of geosynthetic reinforcement used in column-supported embankments," M.S. Thesis. Virginia Polytechnic Institute and State University, Blacksburg, Virginia.

Madhav, M. R. and Poorooshasb, H. B. (1989). "Modified Pasternak model for Reinforced soil," *Mathematical and Computer Modelling*, 12, 1505-1509.

McGuire, M. P. and Filz, G. M. (2008). "Quantitative comparison of theories for geosynthetic reinforcement of column-supported embankments," The First Pan American Geosynthetic Conference & Exhibition (GeoAmericas 2008).

Moraci, N., and Recalcati, P. (2006). "Factors affecting the pullout behavior of extruded geogrids embedded in a compacted granular soil,"*Geotextiles and Geomembranes*, 24, 220-242.

Pinto, M. (2003). "Applications of geosynthetics for soil reinforcement,"*Ground Improvement*, 7, 61-72.

Rogbeck, Y., Gustavsson, S., Sodergren, I., and Lindquist, D. (1998). "Reinforced piled embankments in Sweden – design aspects," *Proceedings, Sixth International Conference on Geosynthetics*, 755-762.

Russell, D. and Pierpoint, N. (1997). "An assessment of design methods for piled embankments," *Ground Engineering*, November, 39-44.

Shukla, S. K. and Chandra S. (1994). **"**A generalized mechanical model for geosynthetic-reinforced foundation soil,"*Geotextiles and Geomembranes*, 13, 1994, 813-825.

Shukla, S. K. and Chandra, S. (1996). **"**A study on a new mechanical model for foundations and it's elastic settlement response," *International Journal for Numerical and Analytical Methods in Geomechanics*, 20, 595-604.

Stewart, M. E. (2005). "Design of bridging layers in geosynthetic-reinforced column-supported embankments," Ph.D. Dissertation. Virginia Polytechnic Institute and State University, Blacksburg, Virginia.

Villard, P. and Giraud, H. (1998). "Three-dimensional modeling of the behavior of geotextile sheets as membranes," *Textile Research Journal*, 68, 797-806.

Villard, P., Kotake, N., and Otani, J. (2002). "Modeling of reinforced soil in finite element analysis," *Geosynthetics: State of the Art, Recent Developments: Proceedings of the Seventh International Conference on Geosynthetics*, 1, September, 39-95.

Yan, L., Yang, J. S., and Han, J. (2006). "Parametric study on geosynthetic-reinforced pile-supported embankments,"*Geotechnical Special* Publication *151, Advances in Earth Structures: Research to Practice - Proceedings of the GeoShanghai Conference*, 2006, 255-261.

Appendix A

Additional WWTS Vs. Strain Plots of Geogrid Tests 11-30

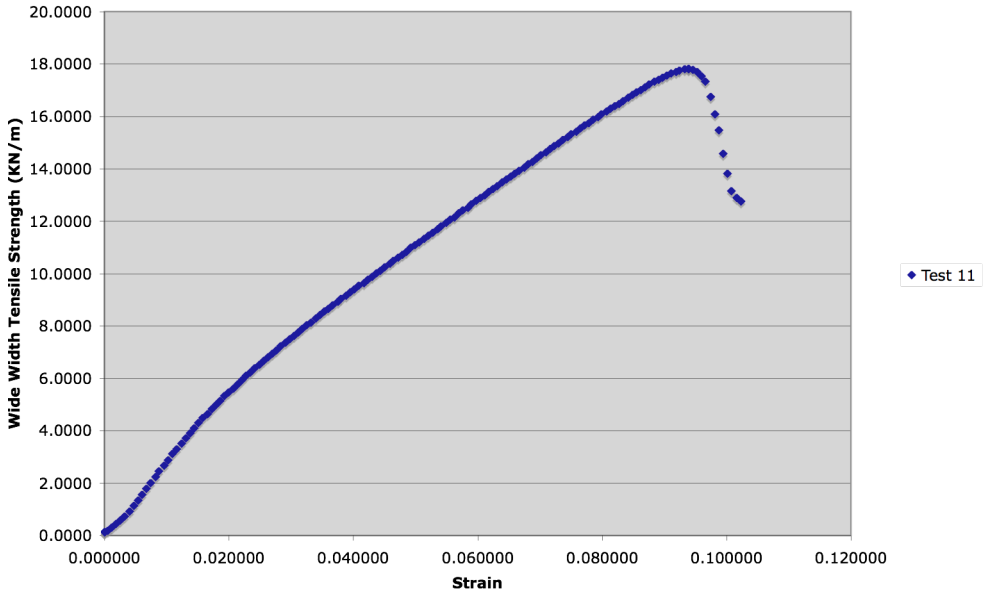

Figure A.1 Fortrac 80 Cross Direction

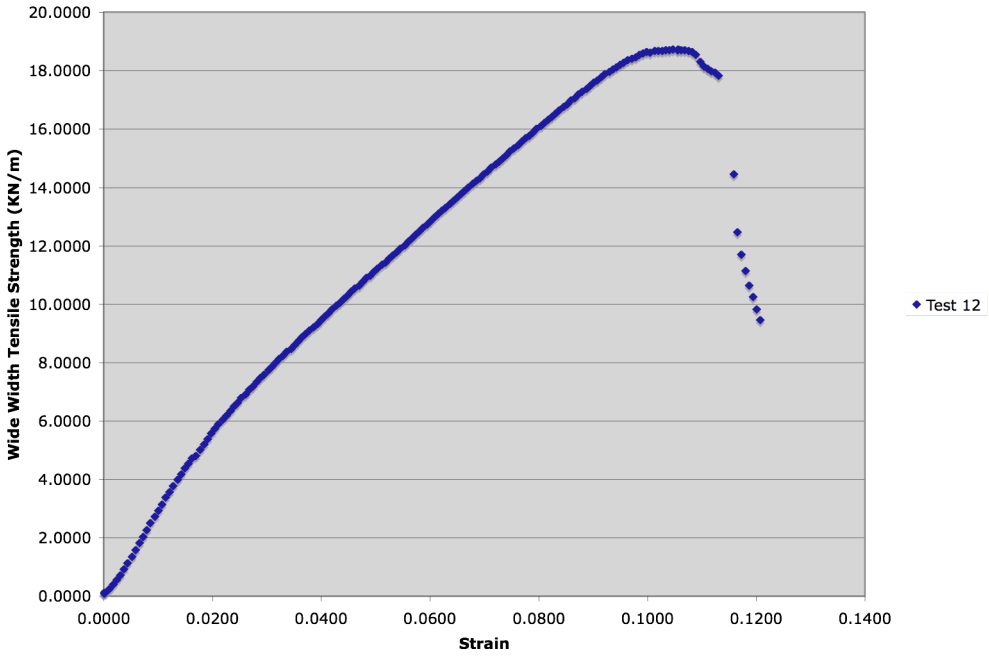

Figure A.2 Fortrac 80 Cross Direction

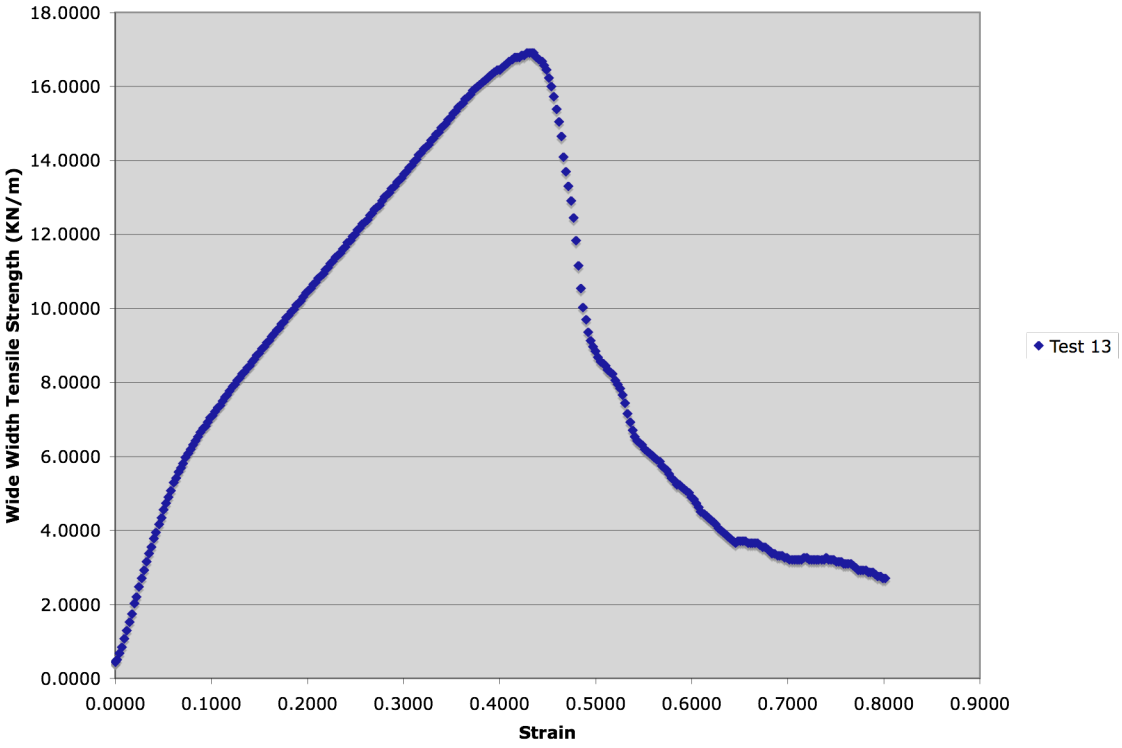

Figure A.3 Fortrac 35 Machine Direction

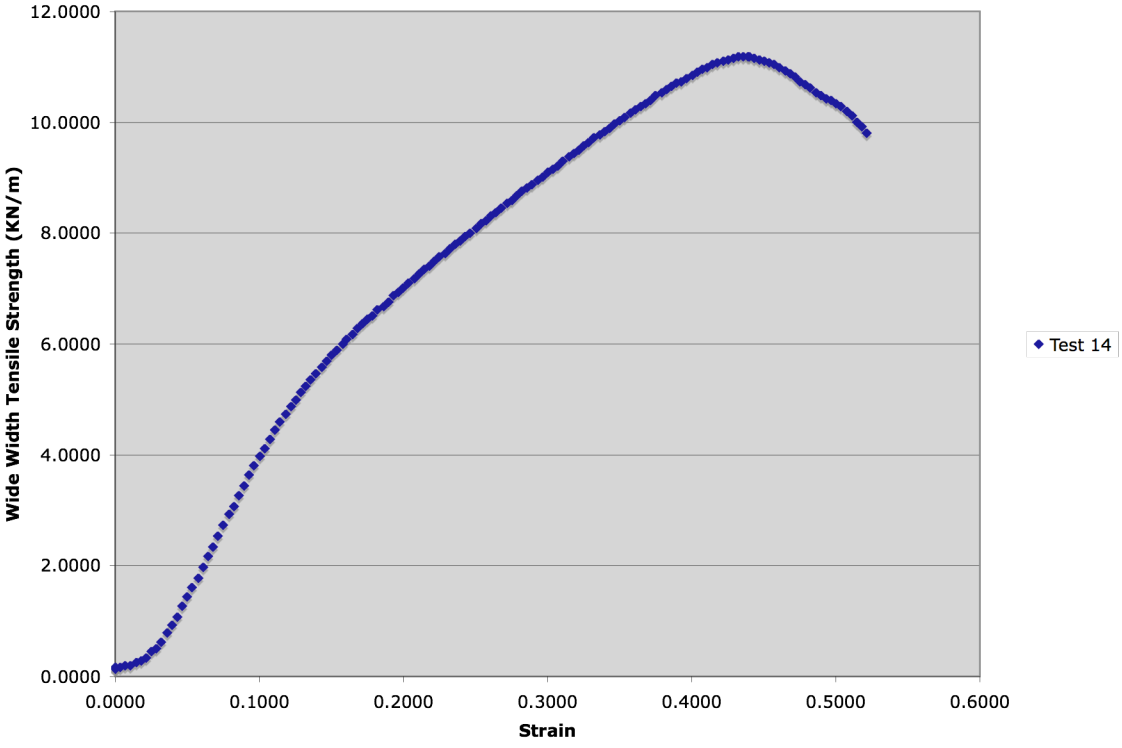

Figure A.4 Fortrac 35 Machine Direction

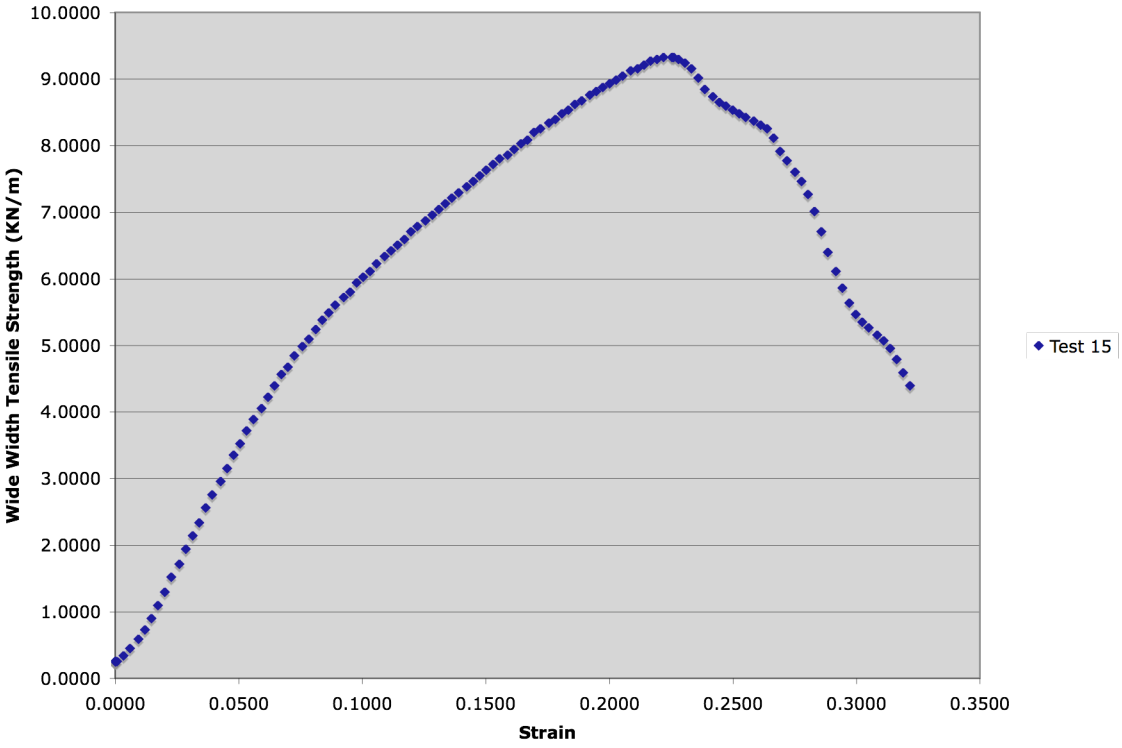

Figure A.5 Fortrac 35 Machine Direction

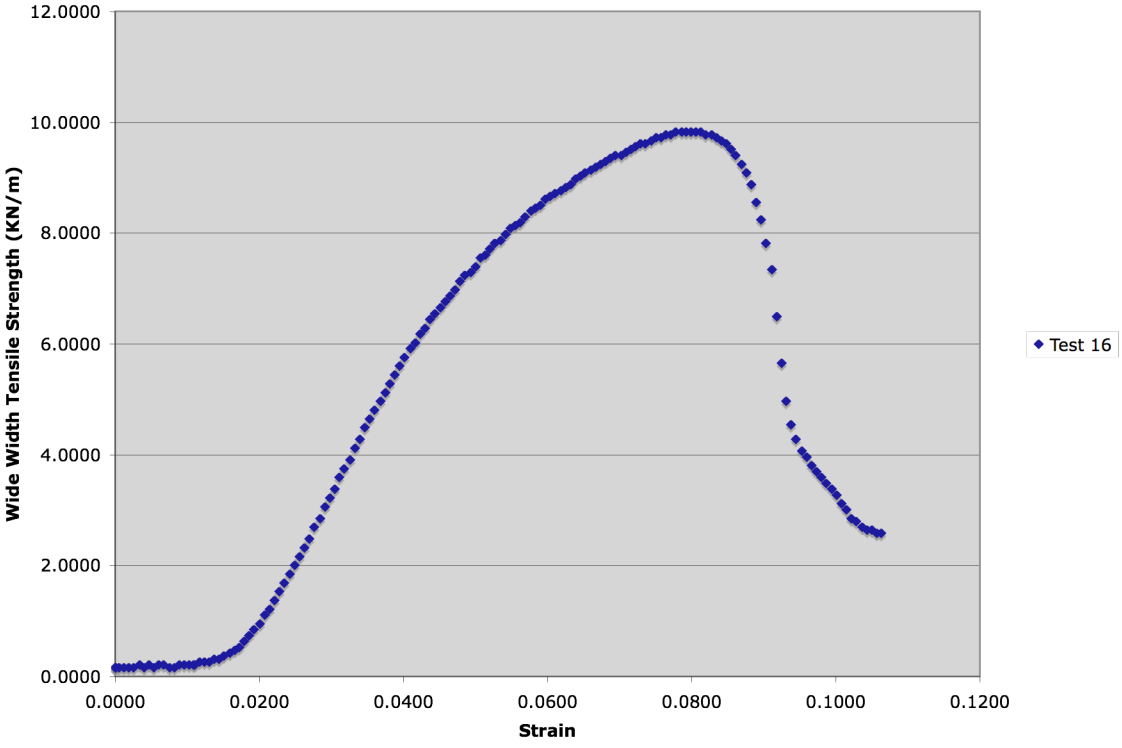

Figure A.6 Fortrac 110 Cross Direction

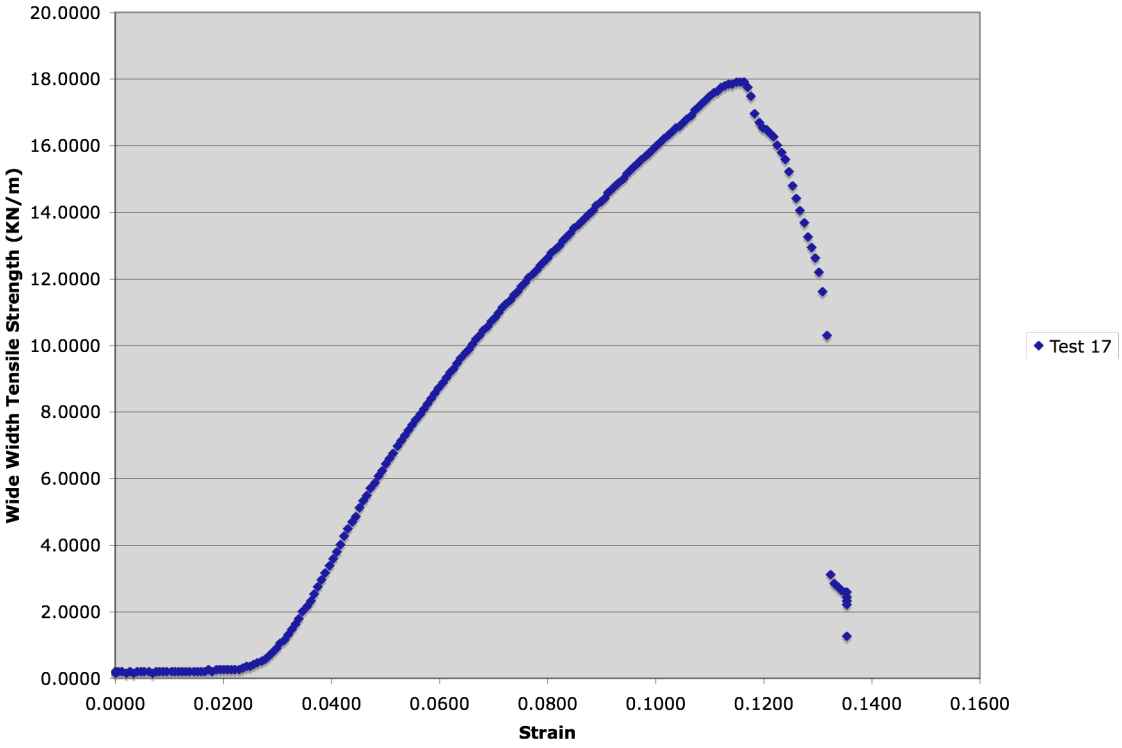

Figure A.7 Fortrac 110 Cross Direction

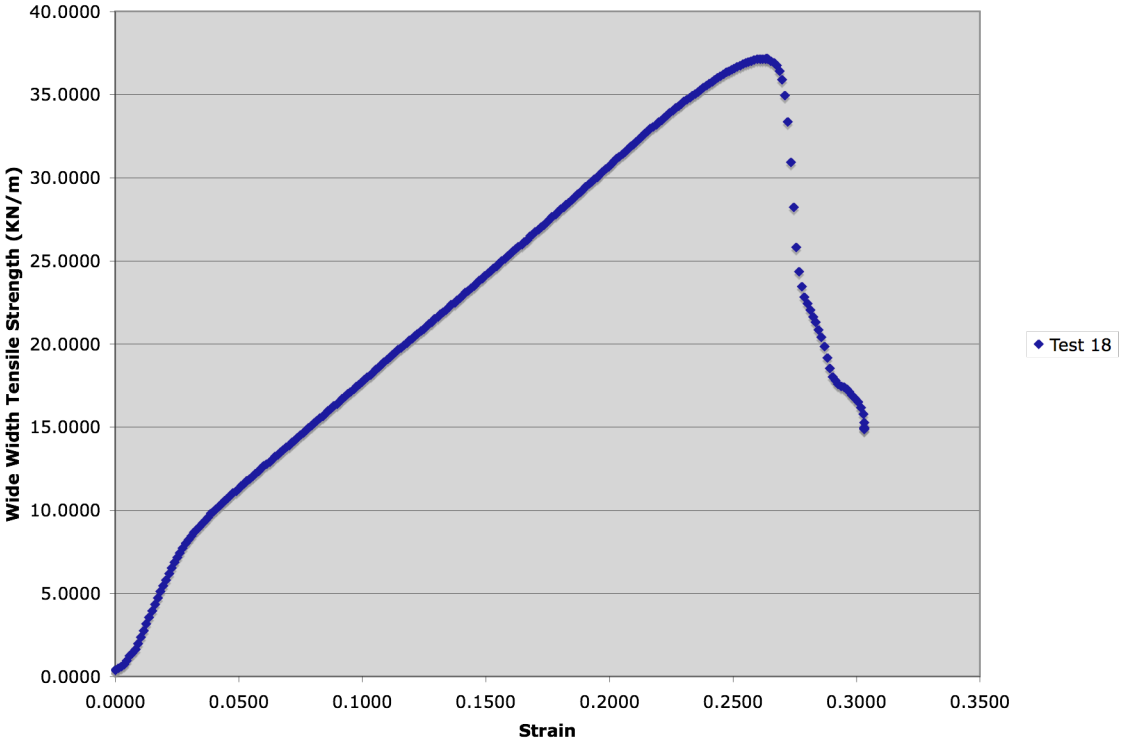

Figure A.8 Fortrac 55 Machine Direction

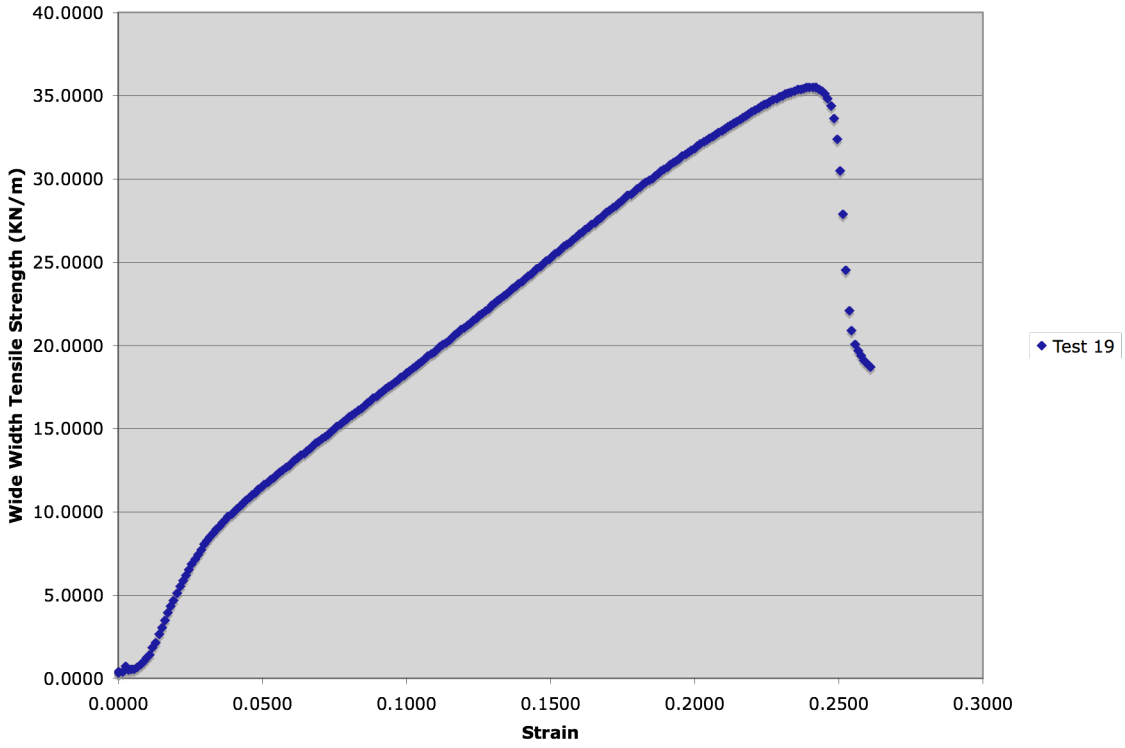

Figure A.9 Fortrac 55 Machine Direction

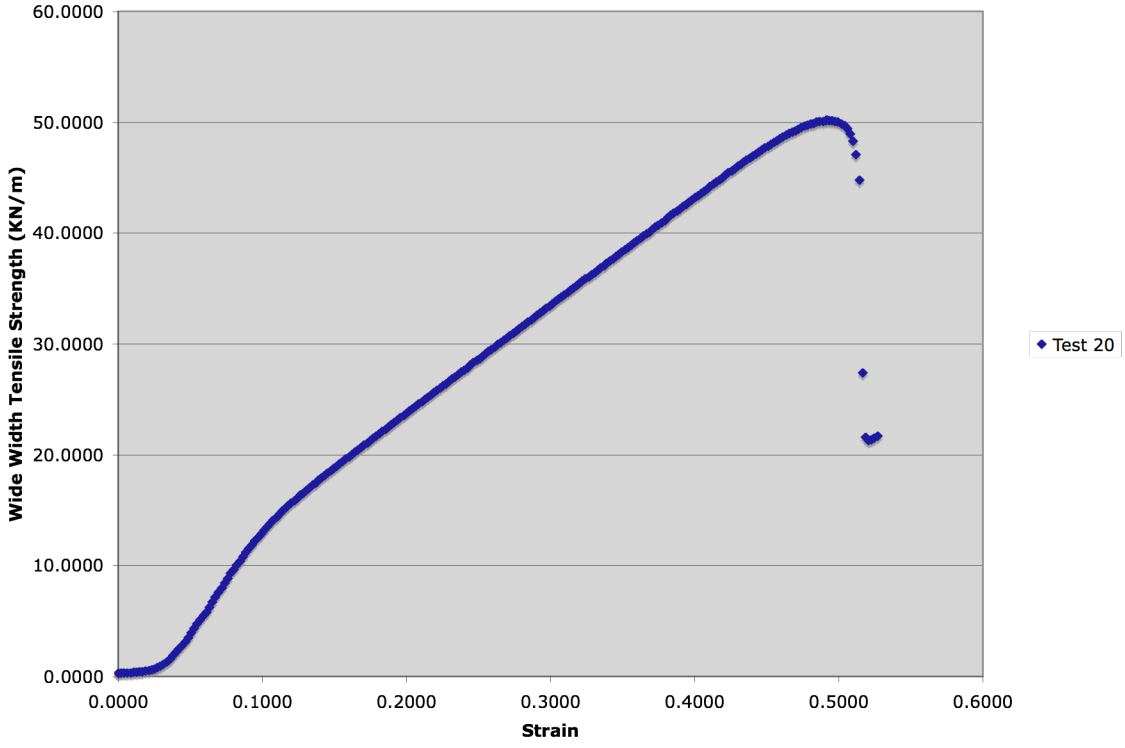

Figure A.10 Fortrac 80 Machine Direction

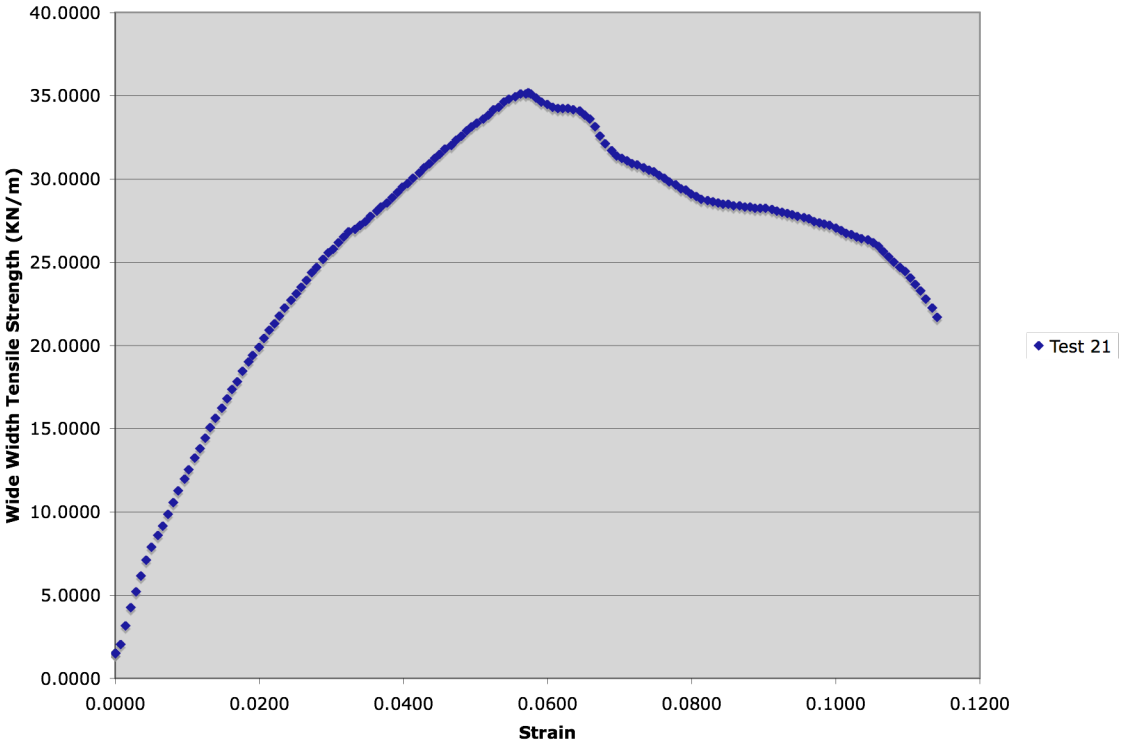

Figure A11 Fornit 30 Machine Direction

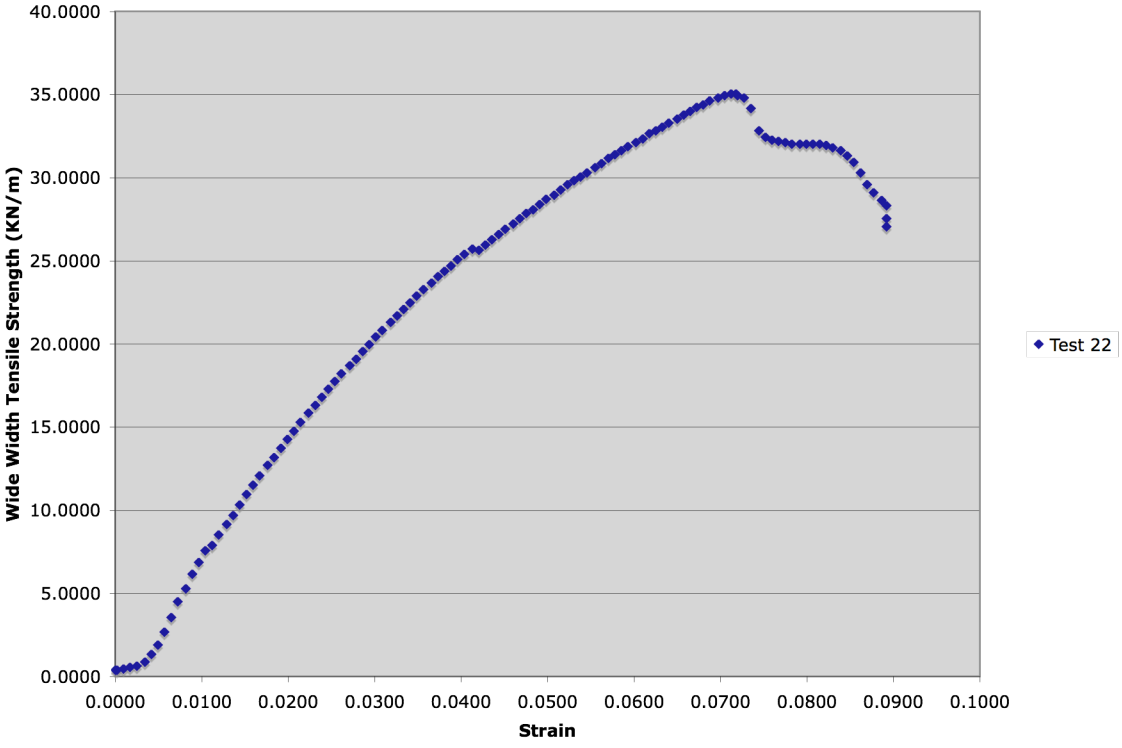

Figure A.12 Fornit 30 Machine Direction

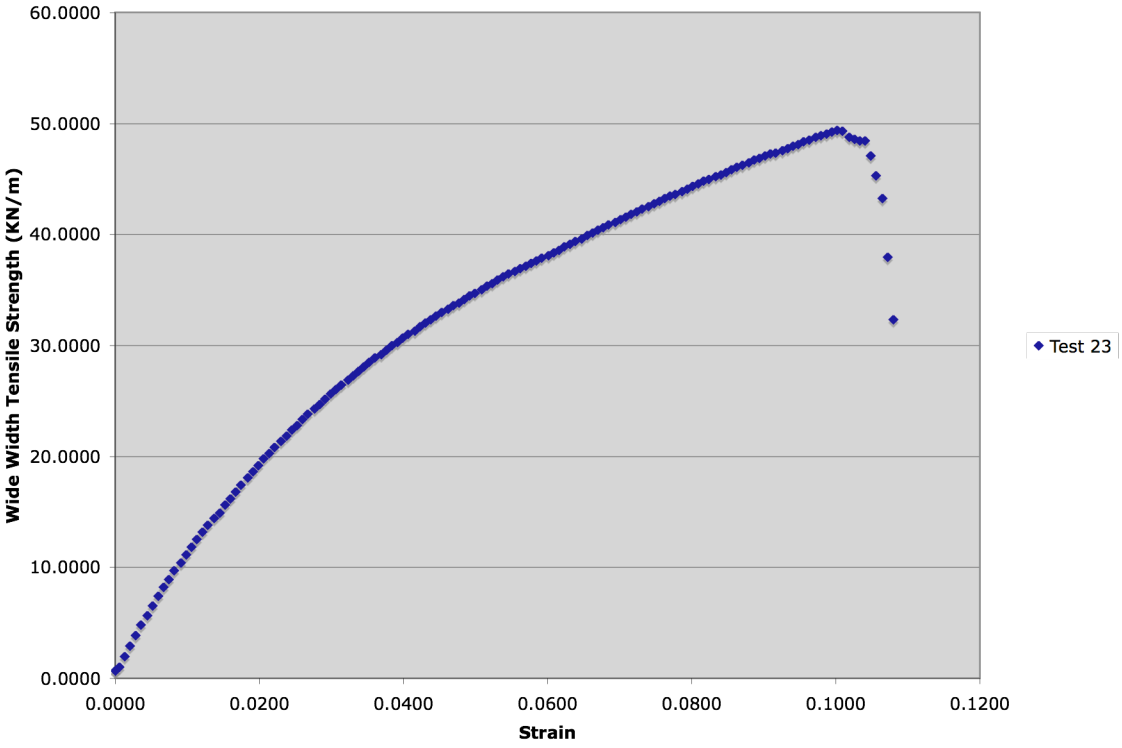

Figure A.13 Fornit 30 Machine Direction

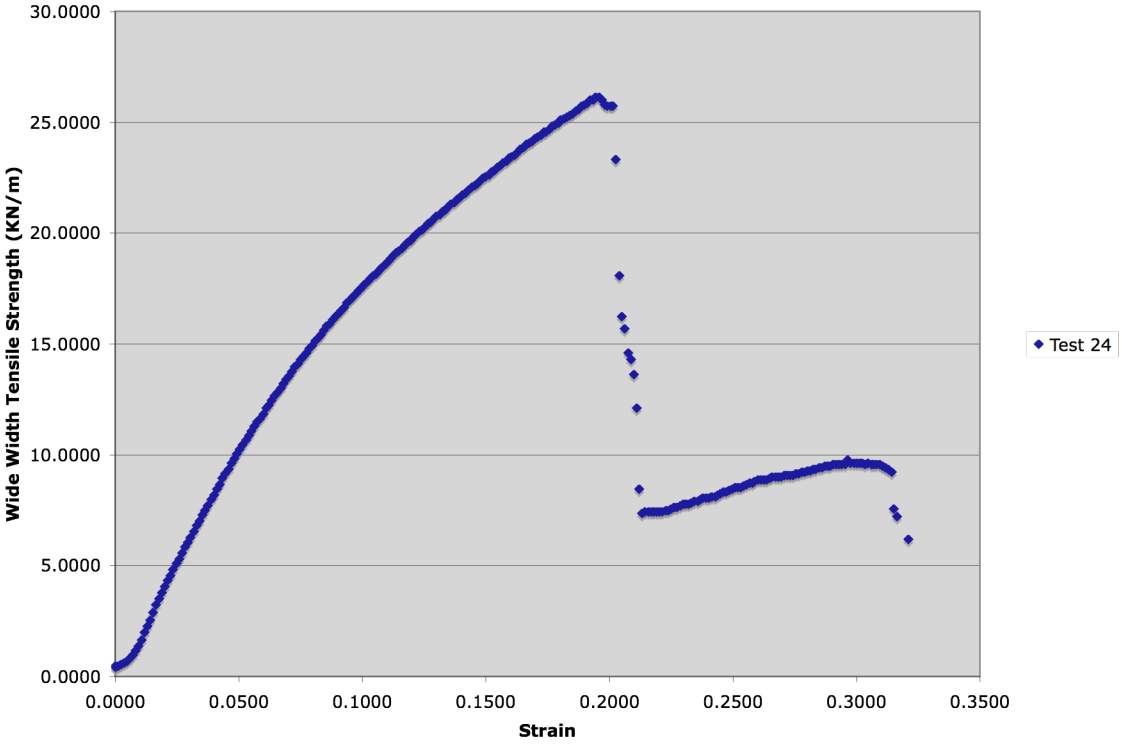

Figure A.14 Fornit 30 Cross Direction

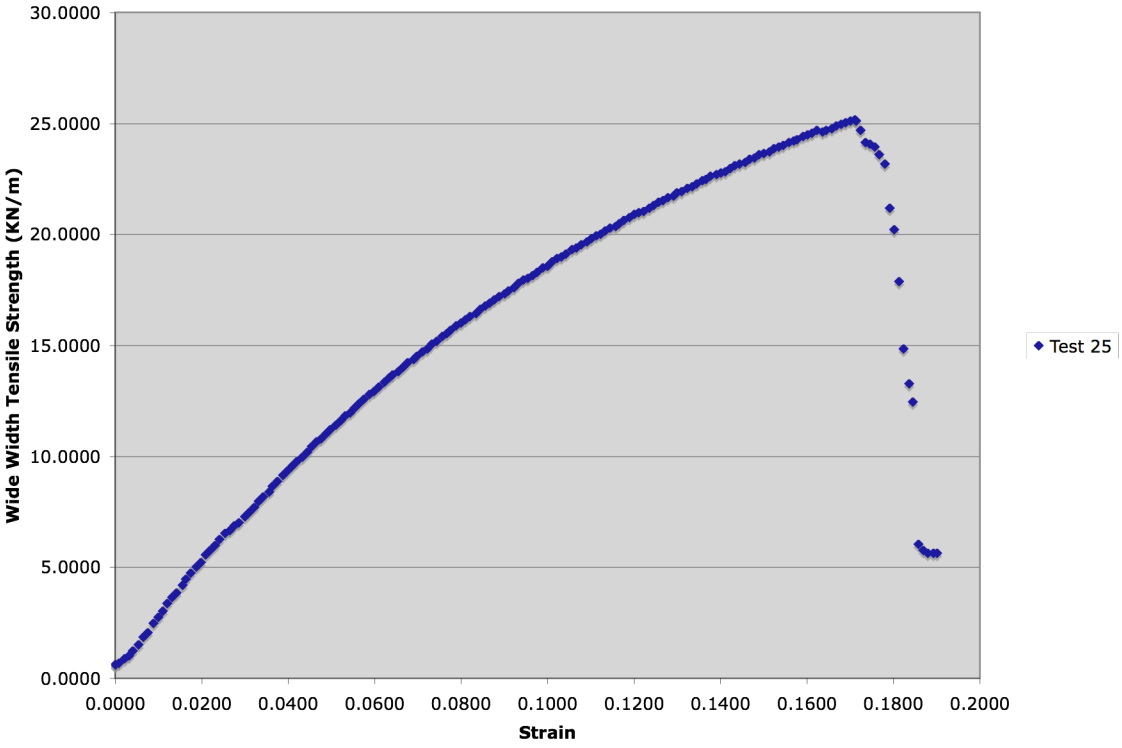

Figure A.15 Fornit 30 Cross Direction

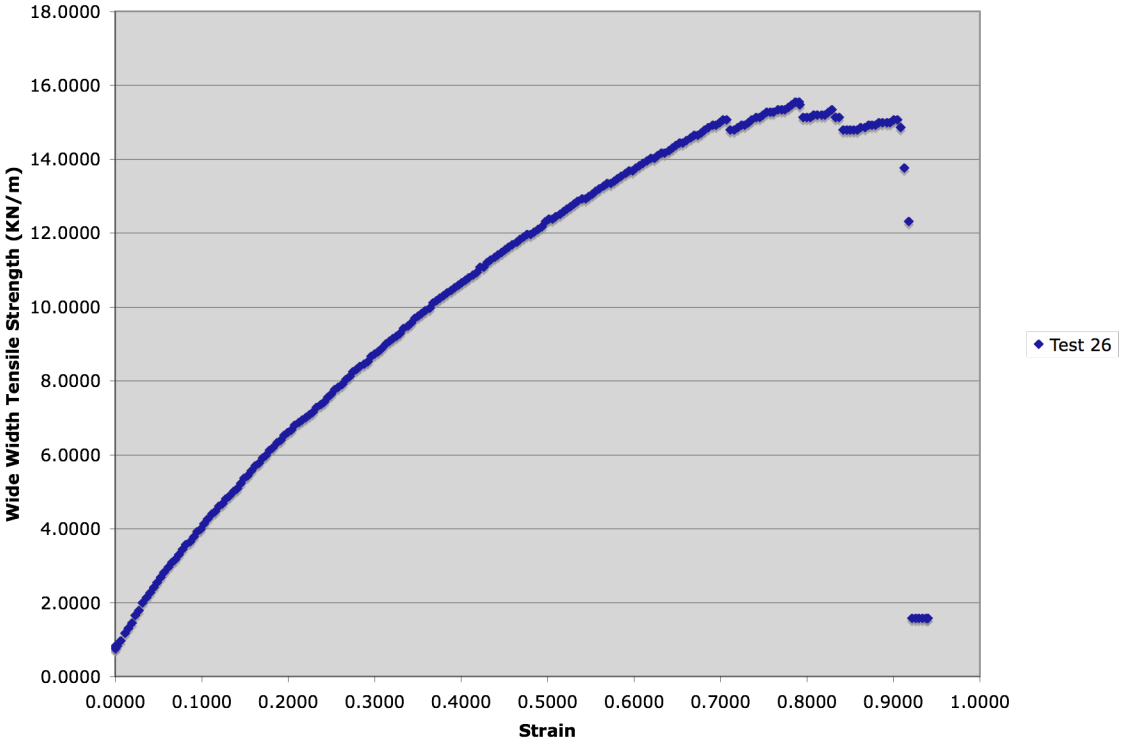

Figure A.16 Fornit 20 Cross Direction

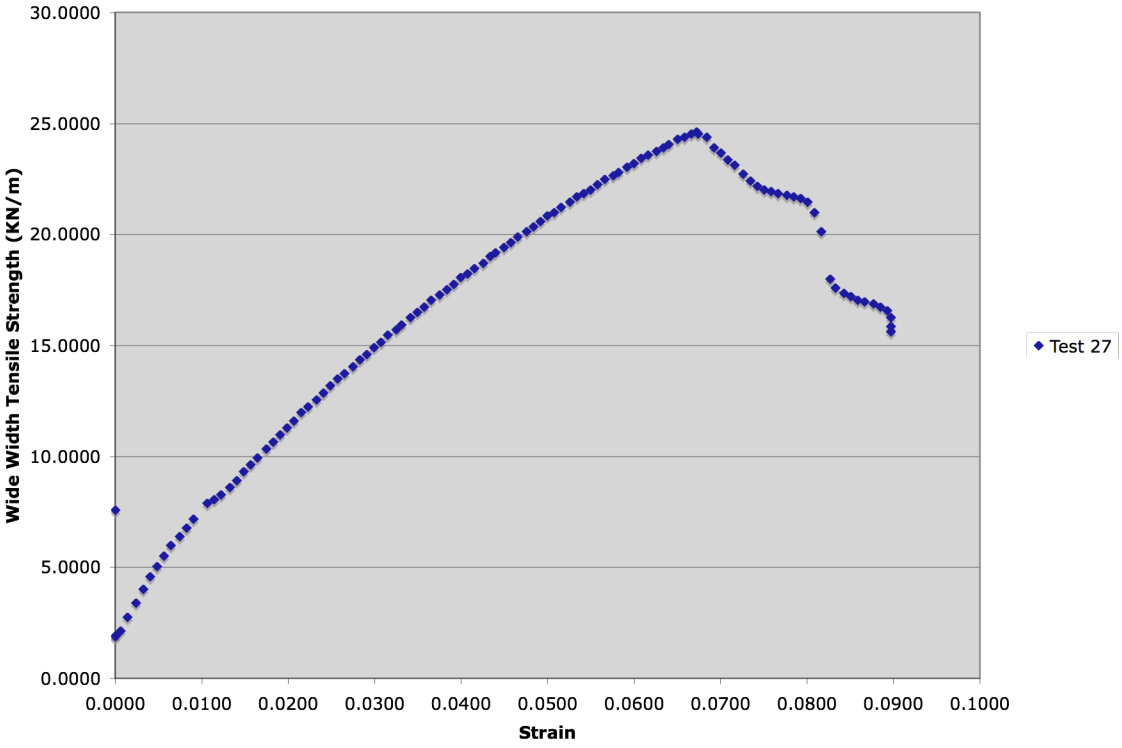

Figure A.17 Fornit 20 Cross Direction

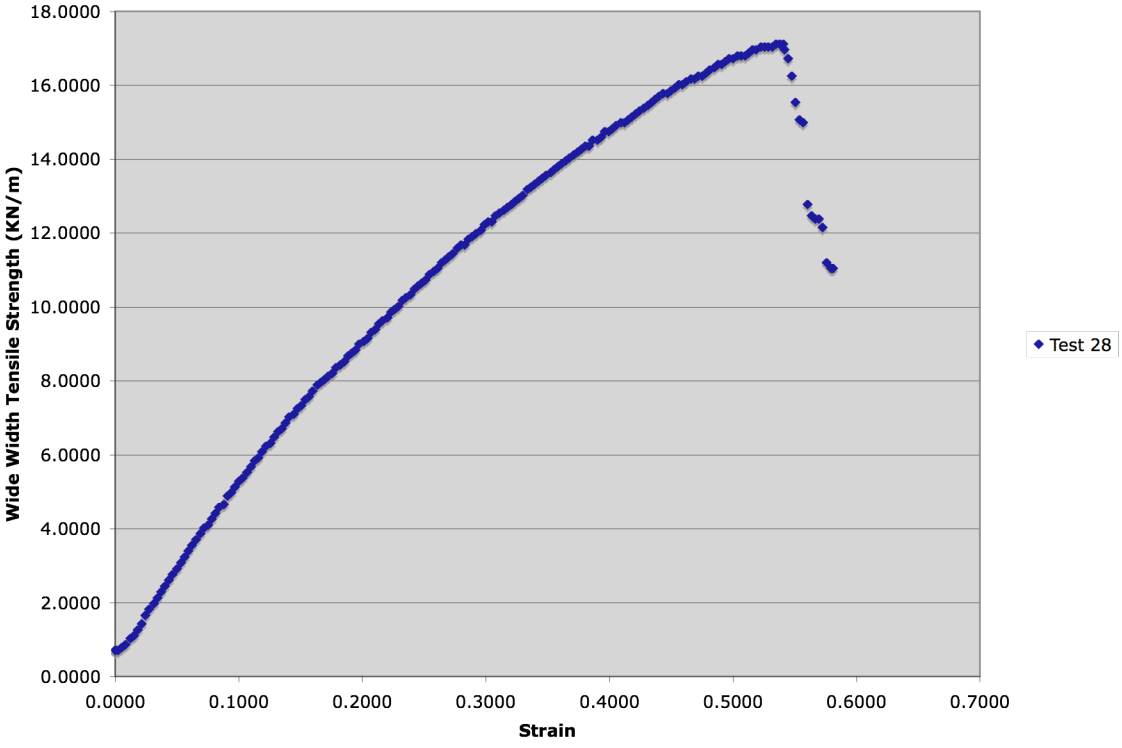

Figure A.18 Fornit 20 Machine Direction

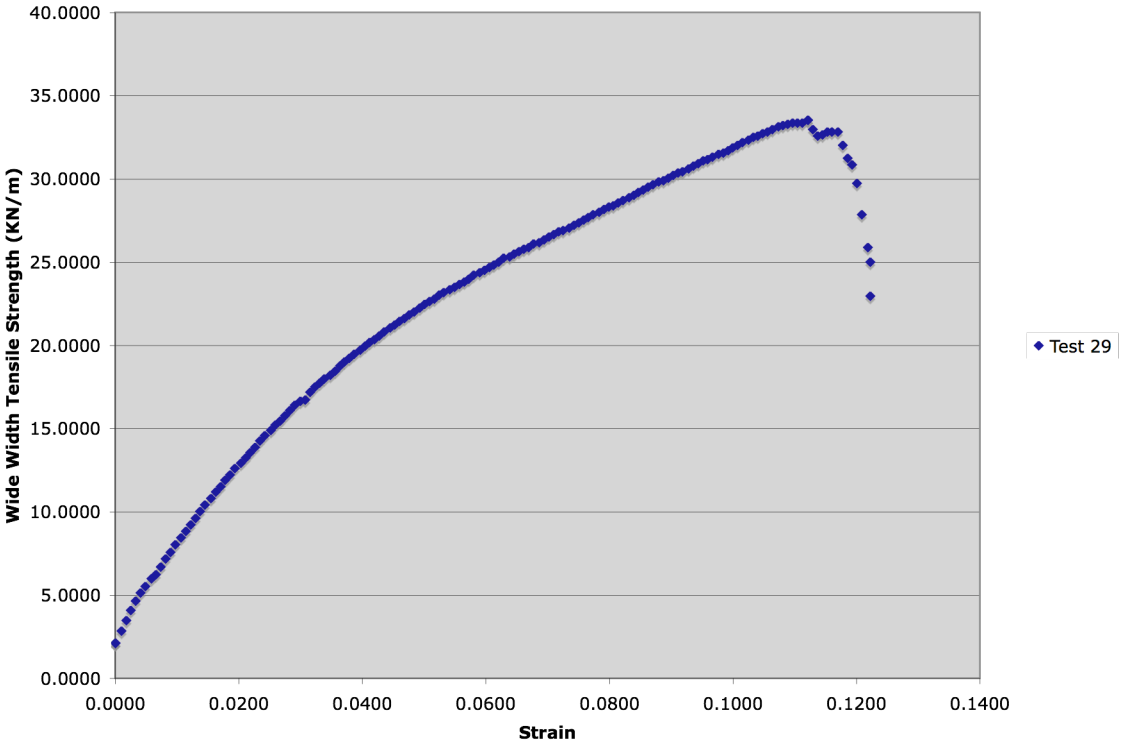

Figure A.19 Fornit 20 Machine Direction

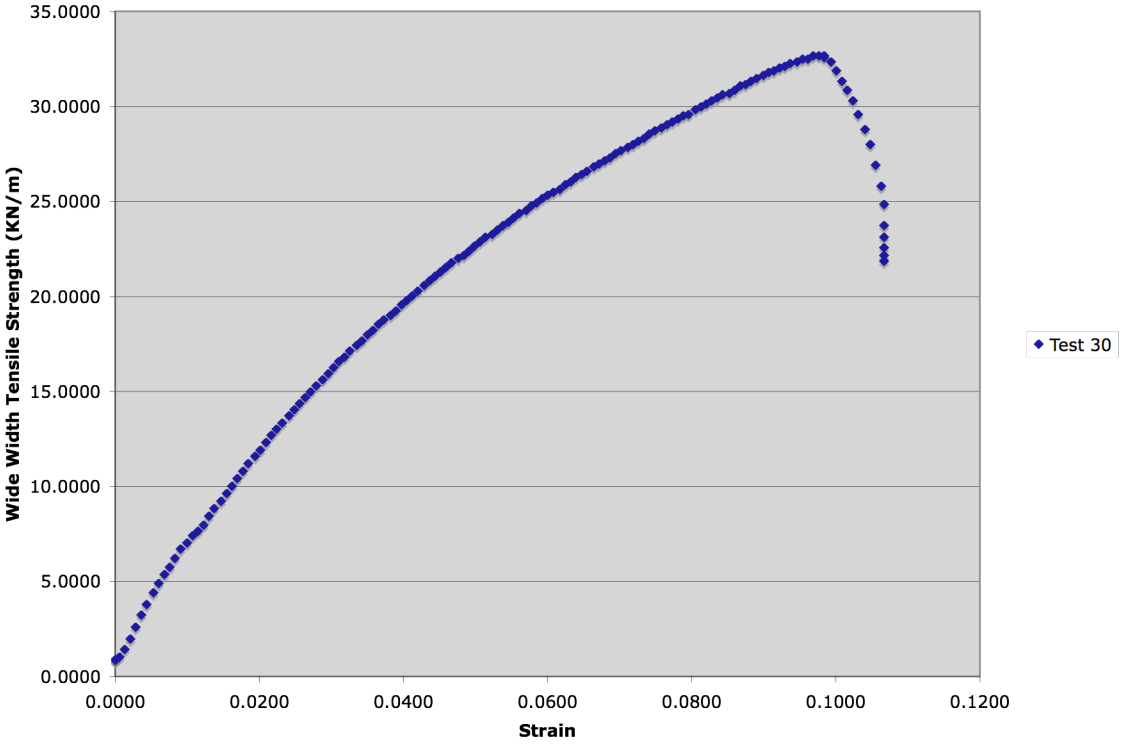

Figure A.20 Fornit 20 Machine Direction

Appendix B

Supporting Mathematica Worksheet for Cable Net Examples

Problem 1: Example 1

```
Clear[XYZ, xyz, DOF, dXYZ, dxyz, Solution, S,
  Sol, Energy, e, fn, z, L, c, d, n, cdim, ndim, CON, R, F, k, EA];

XYZ = {{400, 0, 0}, {0, 400, 0}, {400, 400, 0}, {800, 400, 0}, {400, 800, 0}};
CON = {{2, 3}, {3, 4}, {1, 3}, {3, 5}};
R = {{1, 1, 1}, {1, 1, 1}, {0, 0, 0}, {1, 1, 1}, {1, 1, 1}};
F = {{0, 0, 0}, {0, 0, 0}, {0, 0, -15}, {0, 0, 0}, {0, 0, 0}};
T0 = 200; EMod = 124800; XArea = 0.785; EA = EMod * XArea;
EA
cdim = Dimensions[CON];
ndim = Dimensions[XYZ];
c = cdim[[1]]
d = ndim[[2]]
n = ndim[[1]]

97968.

4

3

5
```

The notation was slightly different when these examples were solved. XYZ is the initial position matrix. CON is the connectivity matrix, and it describes which nodes are connected by cables. R is the restraint matrix and if the node has a 1 in than that node is restrained in the direction that the column represents (x, y, z). F is the force matrix. T0 is the pretension in each cable. EMod is the modulus of elasticity, XArea is the cross-sectional area of a cable, EA is the modulus of elasticity times the cross-sectional area,

```
L = Table[Sqrt[Sum[(XYZ[[CON[[c, i]], m]] - XYZ[[CON[[c, j]], m]])^2, {m, 3}]], {c, c}];

L // MatrixForm
z = 1;
ID = Table[If[R[[i, j]] == 0, z++, 0], {i, n}, {j, d}];
ID // MatrixForm
dXYZ = Table[XYZ[[CON[[c, 1]], D]] - XYZ[[CON[[c, 2]], D]], {c, c}, {D, d}];
dXYZ // MatrixForm
fn = Max[ID]
```

$$\begin{pmatrix} 400 \\ 400 \\ 400 \\ 400 \end{pmatrix}$$

$$\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 2 & 3 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

$$\begin{pmatrix} -400 & 0 & 0 \\ -400 & 0 & 0 \\ 0 & -400 & 0 \\ 0 & -400 & 0 \end{pmatrix}$$

```
3
```

L is the initial length vector, ID numbers each degree of freedom, dXYZ is the initial distance between two nodes on a cable with respect to each axis. fn is the number of free nodes

```
DOF = Table[x[k], {k, fn}];
DOF // MatrixForm
xyz = Table[If[ID[[i, j]] > 0, x[ID[[i, j]]], 0], {i, n}, {j, d}];
xyz // MatrixForm
dxyz = Table[xyz[[CON[[C, 1]], D]] - xyz[[CON[[C, 2]], D]], {C, c}, {D, d}];
dxyz //MatrixForm
```

$$\begin{pmatrix} x[1] \\ x[2] \\ x[3] \end{pmatrix}$$

$$\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ x[1] & x[2] & x[3] \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

$$\begin{pmatrix} -x[1] & -x[2] & -x[3] \\ x[1] & x[2] & x[3] \\ -x[1] & -x[2] & -x[3] \\ x[1] & x[2] & x[3] \end{pmatrix}$$

DOF is the degree of freedom matrix it and it assigns a variable to each degree of freedom based on the ID matrix. xyz i the displacement matrix, and it assigns the variable to the appropriate node (nodal displacement). dxyz is the distance between two nodes on a cable with respect to each axis.

```
e = Table[√(L[[C]]^2 + Sum[(2*dXYZ[[C, d]]*dxyz[[C, d]] + dxyz[[C, d]]^2), {d, 3}]) - L[[C]], {C, c}];
e //MatrixForm
```

$$\begin{pmatrix} -400+\sqrt{160000+800\,x[1]+x[1]^2+x[2]^2+x[3]^2} \\ -400+\sqrt{160000-800\,x[1]+x[1]^2+x[2]^2+x[3]^2} \\ -400+\sqrt{160000+x[1]^2+800\,x[2]+x[2]^2+x[3]^2} \\ -400+\sqrt{160000+x[1]^2-800\,x[2]+x[2]^2+x[3]^2} \end{pmatrix}$$

e is the elongation matrix

$$\text{Energy} = \sum_{i,1}^{c} \frac{EA * e[[i]]^2}{2 * L[[i]]} + \sum_{i,1}^{c} T0 * e[[1]] - \sum_{j,1}^{a} \sum_{k,1}^{d} F[[j,k]] * xyz[[j,k]]$$

$$15\,x[3] + 200\left(-400 + \sqrt{160000 - 800x[1] + x[1]^2 + x[2]^2 + x[3]^2}\right) +$$
$$122.46\left(-400 + \sqrt{160000 - 800x[1] + x[1]^2 + x[2]^2 + x[3]^2}\right)^2 +$$
$$200\left(-400 + \sqrt{160000 + 800x[1] + x[1]^2 + x[2]^2 + x[3]^2}\right) +$$
$$122.46\left(-400 + \sqrt{160000 + 800x[1] + x[1]^2 + x[2]^2 + x[3]^2}\right)^2 +$$
$$200\left(-400 + \sqrt{160000 + x[1]^2 - 800x[2] + x[2]^2 + x[3]^2}\right) +$$
$$122.46\left(-400 + \sqrt{160000 + x[1]^2 - 800x[2] + x[2]^2 + x[3]^2}\right)^2 +$$
$$200\left(-400 + \sqrt{160000 + x[1]^2 + 800x[2] + x[2]^2 + x[3]^2}\right) +$$
$$122.46\left(-400 + \sqrt{160000 + x[1]^2 + 800x[2] + x[2]^2 + x[3]^2}\right)^2$$

Energy sums up the energy of each component of the system (in order: cables, pretension of cables, force)

```
S = NMinimize[Energy, DOF];
S //MatrixForm
```

$$\begin{pmatrix} -54.1668 \\ \{x[1] \to 2.88247 \times 10^{-14},\ x[2] \to 8.0373 \times 10^{-15},\ x[3] \to -6.98051\} \end{pmatrix}$$

Minimize the energy by varying the degree of freedom matrix, DOF

```
Clear[Nxyz]
Nxyz = xyz /. S[[2]]
Clear[NewXYZ]
NXYZ = XYZ + Nxyz;
NXYZ //MatrixForm
MaxX = Max[Table[Nxyz[[i, 1]], {i, n}]]
MaxY = Max[Table[Nxyz[[i, 2]], {i, n}]]
MinZ = Min[Table[NXYZ[[i, 3]], {i, n}]]
```

```
{{0, 0, 0}, {0, 0, 0}, {2.88247 * 10^-14, 8.0373 * 10^-15, -6.98051}, {0, 0, 0}, {0, 0, 0}}
```

$$\begin{pmatrix} 400 & 0 & 0 \\ 0 & 400 & 0 \\ 400. & 400. & -6.98051 \\ 800 & 400 & 0 \\ 400 & 800 & 0 \end{pmatrix}$$

```
2.88247 * 10^-14

8.0373 * 10^-15

-6.98051
```

Nxyz is the new displacements that were found for the system in equilibrium, NXYZ is the final position matrix

```
cables = Table[Line[Table[NXYZ[[CON[[i, j]]]], {j, 2}]], {i, c}];
points = Table[If[NXYZ[[i, 1]] == 0|| NXYZ[[i, 1]] == MaxX, {Hue[0.9, 1, 0.8],
     Cuboid[{NXYZ[[i, 1]] - 1, NXYZ[[i, 2]] - 0.03*MaxY, NXYZ[[i, 3]] - 0.12*MinZ},
      {NXYZ[[i, 1]] + 1, NXYZ[[i, 2]] + 0.03*MaxY, NXYZ[[i, 3]] + 0.12*MinZ}]},
    If[NXYZ[[i, 2]] == 0|| NXYZ[[i, 2]] == MaxY, {Hue[0.9, 1, 0.8],
      Cuboid[{NXYZ[[i, 1]] - 0.03*MaxX, NXYZ[[i, 2]] - 1, NXYZ[[i, 3]] - 0.12*MinZ},
       {NXYZ[[i, 1]] + 0.03*MaxX, NXYZ[[i, 2]] + 1, NXYZ[[i, 3]] + 0.12*MinZ}]},
     {GrayLevel[0], Text[i, {NXYZ[[i, 1]], NXYZ[[i, 2]],
        NXYZ[[i, 3]] - 0.2*MinZ}]}]], {i, n}];
forces = DeleteCases[Table[If[F[[i, 3]] ≠ 0, {Polygon[
      {{NXYZ[[i, 1]] - 0.0125*MaxX, NXYZ[[i, 2]], NXYZ[[i, 3]] + 0.2*MinZ},
       {NXYZ[[i, 1]] + 0.0125*MaxX, NXYZ[[i, 2]], NXYZ[[i, 3]] + 0.2*MinZ},
       {NXYZ[[i, 1]], NXYZ[[i, 2]], NXYZ[[i, 3]] + 0.3*MinZ}}], Line[{NXYZ[[i]],
       {NXYZ[[i, 1]], NXYZ[[i, 2]], NXYZ[[i, 3]] + 0.2*MinZ}}]}, 0], {i, n}], 0]
cablegrid = Flatten[Append[cables, points]]
```

```
{{Polygon[{{400., 400., -8.37661}, {400., 400., -8.37661}, {400., 400., -9.07466}}],
  Line[{{400., 400., -6.98051}, {400., 400., -8.37661}}]}}
```

```
{Line[{{0, 400, 0}, {400., 400., -6.98051}}],
 Line[{{400., 400., -6.98051}, {800, 400, 0}}],
 Line[{{400, 0, 0}, {400., 400., -6.98051}}],
 Line[{{400., 400., -6.98051}, {400, 800, 0}}], Hue[0.9, 1, 0.8],
 Cuboid[{400., -1, 0.837661}, {400., 1, -0.837661}],
 Hue[0.9, 1, 0.8], Cuboid[{-1, 400., 0.837661}, {1, 400., -0.837661}],
 GrayLevel[0], Text[3, {400., 400., -5.58441}], GrayLevel[0],
 Text[4, {800, 400, 1.3961}], GrayLevel[0], Text[5, {400, 800, 1.3961}]}
```

The cables and points tables are used to form a cablegrid table. The cables table prints a line for every cable. The points table prints the node number of an unrestained node and prints a polygon for a restrined node is fixed. The forces table draws a an arrow at every node in which there is a force.

```
Show[{Graphics3D[{Hue[0, 0, 0], cablegrid}],
  Graphics3D[{Hue[0.8, 1, 0.6], EdgeForm[], Thickness[0.004], forces}]},
 Lighting → False, BoxRatios → {1, 1, 0.25}, PlotRange → All, ViewPoint → {1.3, -2.4, 1},
 Boxed → False, Boxed → False, Axes → True, AxesLabel → {"x", "y", "z"}]
```

Show[{Graphics3D plots the displaced shape as shown in Figure 4.1

Problem 2: Example 2

```
Clear[XYZ, xyz, DOF, dXYZ, dxyz, Solution, S,
  Sol, Energy, e, fn, z, L, c, d, n, cdim, ndim, CON, R, F, k, EA];

XYZ = {{4000, 4000, 0}, {4000, 2000, 0}, {8000, 4000, 0},
   {8000, 2000, 0}, {4000, 0, 0}, {0, 2000, 0}, {0, 4000, 0}, {4000, 6000, 0}};
CON = {{1, 7}, {1, 3}, {2, 6}, {2, 4}, {1, 8}, {1, 2}, {2, 5}};
R = {{0, 0, 0}, {0, 0, 0}, {1, 1, 1}, {1, 1, 1}, {1, 1, 1}, {1, 1, 1}, {1, 1, 1}, {1, 1, 1}};
F = {{0, 0, -200}, {0, 0, -200}, {0, 0, 0},
   {0, 0, 0}, {0, 0, 0}, {0, 0, 0}, {0, 0, 0}, {0, 0, 0}};
T0 = 500; EA = 220000;
cdim = Dimensions[CON];
ndim = Dimensions[XYZ];
c = cdim[[1]]
d = ndim[[2]]
n = ndim[[1]]


7

3

8


L = Table[√(Sum[(XYZ[[CON[[C, 1]], D]] - XYZ[[CON[[C, 2]], D]])^2, {D, 2}]), {C, c}];

L // MatrixForm
z = 1;
ID = Table[If[R[[i, j]] == 0, z++, 0], {i, n}, {j, d}];
ID // MatrixForm
dXYZ = Table[XYZ[[CON[[C, 1]], D]] - XYZ[[CON[[C, 2]], D]], {C, c}, {D, d}];
dXYZ // MatrixForm
fn = Max[ID]
```

$$\begin{pmatrix} 4000 \\ 4000 \\ 4000 \\ 4000 \\ 2000 \\ 2000 \\ 2000 \end{pmatrix}$$

$$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

$$\begin{pmatrix} 4000 & 0 & 0 \\ -4000 & 0 & 0 \\ 4000 & 0 & 0 \\ -4000 & 0 & 0 \\ 0 & -2000 & 0 \\ 0 & 2000 & 0 \\ 0 & 2000 & 0 \end{pmatrix}$$

```
6
```

```
DOF = Table[x[k], {k, fn}];
DOF // MatrixForm
xyz = Table[If[ID[[i, j]] > 0, x[ID[[i, j]]], 0], {i, n}, {j, d}];
xyz // MatrixForm
dxyz = Table[xyz[[CON[[C, 1]], D]] - xyz[[CON[[C, 2]], D]], {C, c}, {D, d}];
dxyz // MatrixForm
```

$$\begin{pmatrix} x[1] \\ x[2] \\ x[3] \\ x[4] \\ x[5] \\ x[6] \end{pmatrix}$$

$$\begin{pmatrix} x[1] & x[2] & x[3] \\ x[4] & x[5] & x[6] \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

$$\begin{pmatrix} x[1] & x[2] & x[3] \\ x[1] & x[2] & x[3] \\ x[4] & x[5] & x[6] \\ x[4] & x[5] & x[6] \\ x[1] & x[2] & x[3] \\ x[1] - x[4] & x[2] - x[5] & x[3] - x[6] \\ x[4] & x[5] & x[6] \end{pmatrix}$$

```
e = Table[Sqrt[L[[C]]^2 + Sum[(2*dXYZ[[C, d]]*dxyz[[C, d]] + dxyz[[C, d]]^2), {d, 3}]] - L[[C]], {C, c}];
e // MatrixForm
```

$$\begin{pmatrix} -4000 + \sqrt{16000000 + 8000\,x[1] + x[1]^2 + x[2]^2 + x[3]^2} \\ -4000 + \sqrt{16000000 - 8000\,x[1] + x[1]^2 + x[2]^2 + x[3]^2} \\ -4000 + \sqrt{16000000 + 8000\,x[4] + x[4]^2 + x[5]^2 + x[6]^2} \\ -4000 + \sqrt{16000000 - 8000\,x[4] + x[4]^2 + x[5]^2 + x[6]^2} \\ -2000 + \sqrt{4000000 + x[1]^2 - 4000\,x[2] + x[2]^2 + x[3]^2} \\ -2000 + \sqrt{4000000 + (x[1] - x[4])^2 + 4000\,(x[2] - x[5]) + (x[2] - x[5])^2 + (x[3] - x[6])^2} \\ -2000 + \sqrt{4000000 + x[4]^2 + 4000\,x[5] + x[5]^2 + x[6]^2} \end{pmatrix}$$

$$\text{Energy} = \sum_{i,3}^{c} \frac{EA*e[[i]]^2}{2*L[[i]]} + \sum_{i,3}^{c} T0*e[[1]] - \sum_{j,3}^{n}\sum_{k,3}^{d} F[[j,k]]*xyz[[j,k]]$$

$$200x[3] + 500\left(-4000 + \sqrt{16000000 - 8000x[1] + x[1]^2 + x[2]^2 + x[3]^2}\right) +$$
$$\frac{55}{2}\left(-4000 + \sqrt{16000000 - 8000x[1] + x[1]^2 + x[2]^2 + x[3]^2}\right)^2 +$$
$$500\left(-4000 + \sqrt{16000000 + 8000x[1] + x[1]^2 + x[2]^2 + x[3]^2}\right) +$$
$$\frac{55}{2}\left(-4000 + \sqrt{16000000 + 8000x[1] + x[1]^2 + x[2]^2 + x[3]^2}\right)^2 +$$
$$500\left(-2000 + \sqrt{4000000 + x[1]^2 - 4000x[2] + x[2]^2 + x[3]^2}\right) +$$
$$55\left(-2000 + \sqrt{4000000 + x[1]^2 - 4000x[2] + x[2]^2 + x[3]^2}\right)^2 +$$
$$500\left(-2000 + \sqrt{4000000 + (x[1] - x[4])^2 + 4000(x[2] - x[5]) + (x[2] - x[5])^2 + (x[3] - x[6])^2}\right) +$$
$$55\left(-2000 + \sqrt{4000000 + (x[1] - x[4])^2 + 4000(x[2] - x[5]) + (x[2] - x[5])^2 + (x[3] - x[6])^2}\right)^2 +$$
$$200x[6] + 500\left(-4000 + \sqrt{16000000 - 8000x[4] + x[4]^2 + x[5]^2 + x[6]^2}\right) +$$
$$\frac{55}{2}\left(-4000 + \sqrt{16000000 - 8000x[4] + x[4]^2 + x[5]^2 + x[6]^2}\right)^2 +$$
$$500\left(-4000 + \sqrt{16000000 + 8000x[4] + x[4]^2 + x[5]^2 + x[6]^2}\right) +$$
$$\frac{55}{2}\left(-4000 + \sqrt{16000000 + 8000x[4] + x[4]^2 + x[5]^2 + x[6]^2}\right)^2 +$$
$$500\left(-2000 + \sqrt{4000000 + x[4]^2 + 4000x[5] + x[5]^2 + x[6]^2}\right) +$$
$$55\left(-2000 + \sqrt{4000000 + x[4]^2 + 4000x[5] + x[5]^2 + x[6]^2}\right)^2$$

```
S = NMinimize[Energy, DOF]
S // MatrixForm
```

```
{-49942.9, {x[1] → 3.96119 × 10^-6, x[2] → 3.29977,
  x[3] → -199.75, x[4] → 1.4092 × 10^-6, x[5] → -3.29977, x[6] → -199.75}}
```

```
Clear[Nxyz]
Nxyz = xyz /. S[[2]]
Clear[NewXYZ]
NXYZ = XYZ + Nxyz;
NXYZ // MatrixForm
MaxX = Max[Table[Nxyz[[i, 1]], {i, n}]]
MaxY = Max[Table[Nxyz[[i, 2]], {i, n}]]
MinZ = Min[Table[Nxyz[[i, 3]], {i, n}]]
```

```
{{3.96119 × 10^-6, 3.29977, -199.75}, {1.4092 × 10^-6, -3.29977, -199.75},
 {0, 0, 0}, {0, 0, 0}, {0, 0, 0}, {0, 0, 0}, {0, 0, 0}, {0, 0, 0}}
```

$$\begin{pmatrix} 4000. & 4003.3 & -199.75 \\ 4000. & 1996.7 & -199.75 \\ 8000 & 4000 & 0 \\ 8000 & 2000 & 0 \\ 4000 & 0 & 0 \\ 0 & 2000 & 0 \\ 0 & 4000 & 0 \\ 4000 & 6000 & 0 \end{pmatrix}$$

```
3.96119 × 10^-6

3.29977

-199.75
```

```
cables = Table[Line[Table[NXYZ[[CON[[i, j]]]], {j, 2}]], {i, c}];
points = Table[If[NXYZ[[i, 1]] == 0|| NXYZ[[i, 1]] == MaxX, {Hue[0.6, 1, 0.8],
     Cuboid[{NXYZ[[i, 1]] - 1, NXYZ[[i, 2]] - 0.03*MaxY, NXYZ[[i, 3]] - 0.12*MinZ},
      {NXYZ[[i, 1]] + 1, NXYZ[[i, 2]] + 0.03*MaxY, NXYZ[[i, 3]] + 0.12*MinZ}]},
    If[NXYZ[[i, 2]] == 0|| NXYZ[[i, 2]] == MaxY, {Hue[0.6, 1, 0.8],
      Cuboid[{NXYZ[[i, 1]] - 0.03*MaxX, NXYZ[[i, 2]] - 1, NXYZ[[i, 3]] - 0.12*MinZ},
       {NXYZ[[i, 1]] + 0.03*MaxX, NXYZ[[i, 2]] + 1, NXYZ[[i, 3]] + 0.12*MinZ}]},
     {GrayLevel[0], Text[i, {NXYZ[[i, 1]], NXYZ[[i, 2]],
        NXYZ[[i, 3]] - 0.2*MinZ}]}]], {i, n}];
forces = DeleteCases[Table[If[F[[i, 3]] ≠ 0, {Polygon[
      {{NXYZ[[i, 1]] - 0.0125*MaxX, NXYZ[[i, 2]], NXYZ[[i, 3]] + 0.2*MinZ},
       {NXYZ[[i, 1]] + 0.0125*MaxX, NXYZ[[i, 2]], NXYZ[[i, 3]] + 0.2*MinZ},
       {NXYZ[[i, 1]], NXYZ[[i, 2]], NXYZ[[i, 3]] + 0.3*MinZ}}], Line[{NXYZ[[i]],
       {NXYZ[[i, 1]], NXYZ[[i, 2]], NXYZ[[i, 3]] + 0.2*MinZ}}]}, 0], {i, n}], 0]
cablegrid = Flatten[Append[cables, points]]
```

```
{{Polygon[{{4000., 4003.3, -239.7}, {4000., 4003.3, -239.7}, {4000., 4003.3, -259.675}}],
  Line[{{4000., 4003.3, -199.75}, {4000., 4003.3, -239.7}}]},
 {Polygon[{{4000., 1996.7, -239.7}, {4000., 1996.7, -239.7}, {4000., 1996.7, -259.675}}],
  Line[{{4000., 1996.7, -199.75}, {4000., 1996.7, -239.7}}]}}
```

```
{Line[{{4000., 4003.3, -199.75}, {0, 4000, 0}}],
 Line[{{4000., 4003.3, -199.75}, {8000, 4000, 0}}],
 Line[{{4000., 1996.7, -199.75}, {0, 2000, 0}}],
 Line[{{4000., 1996.7, -199.75}, {8000, 2000, 0}}],
 Line[{{4000., 4003.3, -199.75}, {4000, 6000, 0}}],
 Line[{{4000., 4003.3, -199.75}, {4000., 1996.7, -199.75}}],
 Line[{{4000., 1996.7, -199.75}, {4000, 0, 0}}], GrayLevel[0],
 Text[1, {4000., 4003.3, -159.8}], GrayLevel[0],
 Text[2, {4000., 1996.7, -159.8}], GrayLevel[0],
 Text[3, {8000, 4000, 39.9499}], GrayLevel[0], Text[4, {8000, 2000, 39.9499}],
 Hue[0.6, 1, 0.8], Cuboid[{4000., -1, 23.97}, {4000., 1, -23.97}],
 Hue[0.6, 1, 0.8], Cuboid[{-1, 1999.9, 23.97}, {1, 2000.1, -23.97}],
 Hue[0.6, 1, 0.8], Cuboid[{-1, 3999.9, 23.97}, {1, 4000.1, -23.97}],
 GrayLevel[0], Text[8, {4000, 6000, 39.9499}]}
```

```
Show[{Graphics3D[{Hue[0, 0, 0], cablegrid}],
  Graphics3D[{Hue[0.5, 1, 0.6], EdgeForm[], Thickness[0.004], forces}]},
 Lighting → False, BoxRatios → {1, 1, 0.25}, PlotRange → All, ViewPoint → {1.3, -2.4, 1},
 Boxed → False, Boxed → False, Axes → True, AxesLabel → {"x", "y", "z"}]
```

Problem 3: Example 3

```
Clear[XYZ, xyz, DOF, dXYZ, dxyz, Solution, S,
  Sol, Energy, e, fn, z, L, c, d, n, cdim, ndim, CON, R, F, k, EA];

XYZ = {{400, 0, 0}, {800, 0, 0}, {0, 400, 0},
   {400, 400, 0}, {800, 400, 0}, {1200, 400, 0}, {0, 800, 0}, {400, 800, 0},
   {800, 800, 0}, {1200, 800, 0}, {400, 1200, 0}, {800, 1200, 0}};
CON = {{3, 4}, {4, 5}, {5, 6}, {7, 8}, {8, 9}, {9, 10}, {1, 4},
   {4, 8}, {8, 11}, {2, 5}, {5, 9}, {9, 12}};
R = {{1, 1, 1}, {1, 1, 1}, {1, 1, 1}, {0, 0, 0}, {0, 0, 0}, {1, 1, 1},
   {1, 1, 1}, {0, 0, 0}, {0, 0, 0}, {1, 1, 1}, {1, 1, 1}, {1, 1, 1}};
F = {{0, 0, 0}, {0, 0, 0}, {0, 0, 0}, {0, 0, -15}, {0, 0, -15}, {0, 0, 0},
   {0, 0, 0}, {0, 0, -15}, {0, 0, 0}, {0, 0, 0}, {0, 0, 0}, {0, 0, 0}};
T0 = 200; EMod = 124800; XArea = 0.785; EA = EMod * XArea
EA
cdim = Dimensions[CON];
ndim = Dimensions[XYZ];
c = cdim[[2]]
d = ndim[[2]]
n = ndim[[1]]


97968.

97968.

12

3

12

L = Table[Sqrt[Sum[(XYZ[[CON[[c, 1]], a]] - XYZ[[CON[[c, 2]], a]])^2, {a, 1, 2}]], {c, c}];
L // MatrixForm
z = 1;
ID = Table[If[R[[i, j]] == 0, z++, 0], {i, n}, {j, d}];
ID // MatrixForm
dXYZ = Table[XYZ[[CON[[C, 1]], D]] - XYZ[[CON[[C, 2]], D]], {C, c}, {D, d}];
dXYZ // MatrixForm
fn = Max[ID]
```

$$\begin{pmatrix} 400 \\ 400 \\ 400 \\ 400 \\ 400 \\ 400 \\ 400 \\ 400 \\ 400 \\ 400 \\ 400 \\ 400 \end{pmatrix}$$

$$\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 2 & 3 \\ 4 & 5 & 6 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 7 & 8 & 9 \\ 10 & 11 & 12 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

$$\begin{pmatrix} -400 & 0 & 0 \\ -400 & 0 & 0 \\ -400 & 0 & 0 \\ -400 & 0 & 0 \\ -400 & 0 & 0 \\ -400 & 0 & 0 \\ 0 & -400 & 0 \\ 0 & -400 & 0 \\ 0 & -400 & 0 \\ 0 & -400 & 0 \\ 0 & -400 & 0 \\ 0 & -400 & 0 \end{pmatrix}$$

12

```
DOF = Table[x[k], {k, fn}];
DOF // MatrixForm
xyz = Table[If[ID[[i, j]] > 0, x[ID[[i, j]]], 0], {i, n}, {j, d}];
xyz // MatrixForm
dxyz = Table[xyz[[CON[[C, 1]], D]] - xyz[[CON[[C, 2]], D]], {C, c}, {D, d}];
dxyz // MatrixForm
```

$$\begin{pmatrix} x[1] \\ x[2] \\ x[3] \\ x[4] \\ x[5] \\ x[6] \\ x[7] \\ x[8] \\ x[9] \\ x[10] \\ x[11] \\ x[12] \end{pmatrix}$$

$$\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ x[1] & x[2] & x[3] \\ x[4] & x[5] & x[6] \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ x[7] & x[8] & x[9] \\ x[10] & x[11] & x[12] \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

$$\begin{pmatrix} -x[1] & -x[2] & -x[3] \\ x[1]-x[4] & x[2]-x[5] & x[3]-x[6] \\ x[4] & x[5] & x[6] \\ -x[7] & -x[8] & -x[9] \\ x[7]-x[10] & x[8]-x[11] & x[9]-x[12] \\ x[10] & x[11] & x[12] \\ -x[1] & -x[2] & -x[3] \\ x[1]-x[7] & x[2]-x[8] & x[3]-x[9] \\ x[7] & x[8] & x[9] \\ -x[4] & -x[5] & -x[6] \\ x[4]-x[10] & x[5]-x[11] & x[6]-x[12] \\ x[10] & x[11] & x[12] \end{pmatrix}$$

```
e = Table[Sqrt[L[[C]]^2 + Sum[(2*dXYZ[[C, d]]*dxyz[[C, d]] + dxyz[[C, d]]^2), {d, 3}]] - L[[C]], {C, c}];
e //MatrixForm
```

$$\begin{pmatrix} -400+\sqrt{160000+800x[1]+x[1]^2+x[2]^2+x[3]^2} \\ -400+\sqrt{160000-800(x[1]-x[4])+(x[1]-x[4])^2+(x[2]-x[5])^2+(x[3]-x[6])^2} \\ -400+\sqrt{160000-800x[4]+x[4]^2+x[5]^2+x[6]^2} \\ -400+\sqrt{160000+800x[7]+x[7]^2+x[8]^2+x[9]^2} \\ -400+\sqrt{160000-800(x[7]-x[10])+(x[7]-x[10])^2+(x[8]-x[11])^2+(x[9]-x[12])^2} \\ -400+\sqrt{160000-800x[10]+x[10]^2+x[11]^2+x[12]^2} \\ -400+\sqrt{160000+x[1]^2+800x[2]+x[2]^2+x[3]^2} \\ -400+\sqrt{160000+(x[1]-x[7])^2-800(x[2]-x[8])+(x[2]-x[8])^2+(x[3]-x[9])^2} \\ -400+\sqrt{160000+x[7]^2-800x[8]+x[8]^2+x[9]^2} \\ -400+\sqrt{160000+x[4]^2+800x[5]+x[5]^2+x[6]^2} \\ -400+\sqrt{160000+(x[4]-x[10])^2-800(x[5]-x[11])+(x[5]-x[11])^2+(x[6]-x[12])^2} \\ -400+\sqrt{160000+x[10]^2-800x[11]+x[11]^2+x[12]^2} \end{pmatrix}$$

$$\text{Energy} = \sum_{i=1}^{c} \frac{EA*e[[i]]^2}{2*L[[i]]} + \sum_{i=1}^{c} T0*e[[1]] - \sum_{j=1}^{s}\sum_{k=1}^{d} F[[j,k]]*xyz[[j,k]]$$

$$15\,x[3] + 200\left(-400 + \sqrt{160000 + 800x[1] + x[1]^2 + x[2]^2 + x[3]^2}\right) +$$
$$122.46\left(-400 + \sqrt{160000 + 800x[1] + x[1]^2 + x[2]^2 + x[3]^2}\right)^2 +$$
$$200\left(-400 + \sqrt{160000 + x[1]^2 + 800x[2] + x[2]^2 + x[3]^2}\right) +$$
$$122.46\left(-400 + \sqrt{160000 + x[1]^2 + 800x[2] + x[2]^2 + x[3]^2}\right)^2 +$$
$$200\left(-400 + \sqrt{160000 - 800(x[1] - x[4]) + (x[1] - x[4])^2 + (x[2] - x[5])^2 + (x[3] - x[6])^2}\right) +$$
$$122.46\left(-400 + \sqrt{160000 - 800(x[1] - x[4]) + (x[1] - x[4])^2 + (x[2] - x[5])^2 + (x[3] - x[6])^2}\right)^2 +$$
$$15\,x[6] + 200\left(-400 + \sqrt{160000 - 800x[4] + x[4]^2 + x[5]^2 + x[6]^2}\right) +$$
$$122.46\left(-400 + \sqrt{160000 - 800x[4] + x[4]^2 + x[5]^2 + x[6]^2}\right)^2 +$$
$$200\left(-400 + \sqrt{160000 + x[4]^2 + 800x[5] + x[5]^2 + x[6]^2}\right) +$$
$$122.46\left(-400 + \sqrt{160000 + x[4]^2 + 800x[5] + x[5]^2 + x[6]^2}\right)^2 +$$
$$200\left(-400 + \sqrt{160000 + (x[1] - x[7])^2 - 800(x[2] - x[8]) + (x[2] - x[8])^2 + (x[3] - x[9])^2}\right) +$$
$$122.46\left(-400 + \sqrt{160000 + (x[1] - x[7])^2 - 800(x[2] - x[8]) + (x[2] - x[8])^2 + (x[3] - x[9])^2}\right)^2 +$$
$$15\,x[9] + 200\left(-400 + \sqrt{160000 + 800x[7] + x[7]^2 + x[8]^2 + x[9]^2}\right) +$$
$$122.46\left(-400 + \sqrt{160000 + 800x[7] + x[7]^2 + x[8]^2 + x[9]^2}\right)^2 +$$
$$200\left(-400 + \sqrt{160000 + x[7]^2 - 800x[8] + x[8]^2 + x[9]^2}\right) +$$
$$122.46\left(-400 + \sqrt{160000 + x[7]^2 - 800x[8] + x[8]^2 + x[9]^2}\right)^2 +$$
$$200\left(-400 + \sqrt{160000 + (x[4] - x[10])^2 - 800(x[5] - x[11]) + (x[5] - x[11])^2 + (x[6] - x[12])^2}\right) +$$
$$122.46\left(-400 + \sqrt{160000 + (x[4] - x[10])^2 - 800(x[5] - x[11]) + (x[5] - x[11])^2 + (x[6] - x[12])^2}\right)^2 +$$
$$200\left(-400 + \sqrt{160000 - 800(x[7] - x[10]) + (x[7] - x[10])^2 + (x[8] - x[11])^2 + (x[9] - x[12])^2}\right) +$$
$$122.46\left(-400 + \sqrt{160000 - 800(x[7] - x[10]) + (x[7] - x[10])^2 + (x[8] - x[11])^2 + (x[9] - x[12])^2}\right)^2 +$$
$$200\left(-400 + \sqrt{160000 - 800x[10] + x[10]^2 + x[11]^2 + x[12]^2}\right) +$$
$$122.46\left(-400 + \sqrt{160000 - 800x[10] + x[10]^2 + x[11]^2 + x[12]^2}\right)^2 +$$
$$200\left(-400 + \sqrt{160000 + x[10]^2 - 800x[11] + x[11]^2 + x[12]^2}\right) +$$
$$122.46\left(-400 + \sqrt{160000 + x[10]^2 - 800x[11] + x[11]^2 + x[12]^2}\right)^2$$

```
S = NMinimize[Energy, DOF];
S // MatrixForm
```

```
Clear[Nxyz]
Nxyz = xyz /. S[[2]]
Clear[NewXYZ]
NXYZ = XYZ + Nxyz;
NXYZ //MatrixForm
MaxX = Max[Table[NXYZ[[i, 1]], {i, n}]]
MaxY = Max[Table[NXYZ[[i, 2]], {i, n}]]
MinZ = Min[Table[NXYZ[[i, 3]], {i, n}]]
```

```
{{0, 0, 0}, {0, 0, 0}, {0, 0, 0},
 {-0.0707755, -0.0707756, -12.1719}, {0.0418385, -0.0777647, -11.1827},
 {0, 0, 0}, {0, 0, 0}, {-0.0777647, 0.0418384, -11.1827},
 {-0.0387404, -0.0387404, -5.59219}, {0, 0, 0}, {0, 0, 0}, {0, 0, 0}}
```

$$\begin{pmatrix} 400 & 0 & 0 \\ 800 & 0 & 0 \\ 0 & 400 & 0 \\ 399.929 & 399.929 & -12.1719 \\ 800.042 & 399.922 & -11.1827 \\ 1200 & 400 & 0 \\ 0 & 800 & 0 \\ 399.922 & 800.042 & -11.1827 \\ 799.961 & 799.961 & -5.59219 \\ 1200 & 800 & 0 \\ 400 & 1200 & 0 \\ 800 & 1200 & 0 \end{pmatrix}$$

```
1200
```

```
1200
```

```
-12.1719
```

```
cables = Table[Line[Table[NXYZ[[CON[[i, j]]]], {j, 2}]], {i, c}];
points = Table[If[NXYZ[[i, 1]] == 0 || NXYZ[[i, 1]] == MaxX, {Hue[0.3, 1, 0.8],
     Cuboid[{NXYZ[[i, 1]] - 1, NXYZ[[i, 2]] - 0.03*MaxY, NXYZ[[i, 3]] - 0.12*MinZ},
      {NXYZ[[i, 1]] + 1, NXYZ[[i, 2]] + 0.03*MaxY, NXYZ[[i, 3]] + 0.12*MinZ}]},
    If[NXYZ[[i, 2]] == 0 || NXYZ[[i, 2]] == MaxY, {Hue[0.3, 1, 0.8],
      Cuboid[{NXYZ[[i, 1]] - 0.03*MaxX, NXYZ[[i, 2]] - 1, NXYZ[[i, 3]] - 0.12*MinZ},
       {NXYZ[[i, 1]] + 0.03*MaxX, NXYZ[[i, 2]] + 1, NXYZ[[i, 3]] + 0.12*MinZ}]},
     {GrayLevel[0], Text[i, {NXYZ[[i, 1]], NXYZ[[i, 2]],
        NXYZ[[i, 3]] - 0.2*MinZ}]}], {i, n}];
forces = DeleteCases[Table[If[F[[i, 3]] ≠ 0, {Polygon[
      {{NXYZ[[i, 1]] - 0.0125*MaxX, NXYZ[[i, 2]], NXYZ[[i, 3]] + 0.2*MinZ},
       {NXYZ[[i, 1]] + 0.0125*MaxX, NXYZ[[i, 2]], NXYZ[[i, 3]] + 0.2*MinZ},
       {NXYZ[[i, 1]], NXYZ[[i, 2]], NXYZ[[i, 3]] + 0.3*MinZ}}], Line[{NXYZ[[i]],
       {NXYZ[[i, 1]], NXYZ[[i, 2]], NXYZ[[i, 3]] + 0.2*MinZ}}]}, 0], {i, n}], 0]
cablegrid = Flatten[Append[cables, points]]
```

```
{{Polygon[{{384.929, 399.929, -14.6062},
    {414.929, 399.929, -14.6062}, {399.929, 399.929, -15.8234}}],
  Line[{{399.929, 399.929, -12.1719}, {399.929, 399.929, -14.6062}}]},
 {Polygon[{{785.042, 399.922, -13.6171},
    {815.042, 399.922, -13.6171}, {800.042, 399.922, -14.8343}}],
  Line[{{800.042, 399.922, -11.1827}, {800.042, 399.922, -13.6171}}]},
 {Polygon[{{384.922, 800.042, -13.6171},
    {414.922, 800.042, -13.6171}, {399.922, 800.042, -14.8343}}],
  Line[{{399.922, 800.042, -11.1827}, {399.922, 800.042, -13.6171}}]}}
```

```
{Line[{{0, 400, 0}, {399.929, 399.929, -12.1719}}],
 Line[{{399.929, 399.929, -12.1719}, {800.042, 399.922, -11.1827}}],
 Line[{{800.042, 399.922, -11.1827}, {1200, 400, 0}}],
 Line[{{0, 800, 0}, {399.922, 800.042, -11.1827}}],
 Line[{{399.922, 800.042, -11.1827}, {799.961, 799.961, -5.59219}}],
 Line[{{799.961, 799.961, -5.59219}, {1200, 800, 0}}],
 Line[{{400, 0, 0}, {399.929, 399.929, -12.1719}}],
 Line[{{399.929, 399.929, -12.1719}, {399.922, 800.042, -11.1827}}],
 Line[{{399.922, 800.042, -11.1827}, {400, 1200, 0}}],
 Line[{{800, 0, 0}, {800.042, 399.922, -11.1827}}],
 Line[{{800.042, 399.922, -11.1827}, {799.961, 799.961, -5.59219}}],
 Line[{{799.961, 799.961, -5.59219}, {800, 1200, 0}}],
 Hue[0.3, 1, 0.8], Cuboid[{364., -1, 1.46062}, {436., 1, -1.46062}],
 Hue[0.3, 1, 0.8], Cuboid[{764., -1, 1.46062}, {836., 1, -1.46062}],
 Hue[0.3, 1, 0.8], Cuboid[{-1, 364., 1.46062}, {1, 436., -1.46062}],
 GrayLevel[0], Text[4, {399.929, 399.929, -9.73748}], GrayLevel[0],
 Text[5, {800.042, 399.922, -8.74835}], Hue[0.3, 1, 0.8],
 Cuboid[{1199, 364., 1.46062}, {1201, 436., -1.46062}],
 Hue[0.3, 1, 0.8], Cuboid[{-1, 764., 1.46062}, {1, 836., -1.46062}],
 GrayLevel[0], Text[8, {399.922, 800.042, -8.74835}], GrayLevel[0],
 Text[9, {799.961, 799.961, -3.15782}], Hue[0.3, 1, 0.8],
 Cuboid[{1199, 764., 1.46062}, {1201, 836., -1.46062}],
 Hue[0.3, 1, 0.8], Cuboid[{364., 1199, 1.46062}, {436., 1201, -1.46062}],
 Hue[0.3, 1, 0.8], Cuboid[{764., 1199, 1.46062}, {836., 1201, -1.46062}]}

Show[{Graphics3D[{Hue[0, 0, 0], cablegrid}],
  Graphics3D[{Hue[0.2, 1, 0.6], EdgeForm[], Thickness[0.004], forces}]},
 Lighting → False, BoxRatios → {1, 1, 0.25}, PlotRange → All, ViewPoint → {1.3, -2.4, 1},
 Boxed → False, Boxed → False, Axes → True, AxesLabel → {"x", "y", "z"}]
```

Problem 4: Example 4

```
Clear[XYZ, xyz, DOF, dXYZ, dxyz, Solution, S,
  Sol, Energy, e, fn, z, L, c, d, n, cdim, ndim, CON, R, F, k, EA];

XYZ = {{400, 0, 0}, {800, 0, 0}, {1200, 0, 0}, {0, 400, 0}, {400, 400, 40/3},
   {800, 400, 80/3}, {1200, 400, 40}, {2000, 400, 200/3}, {0, 800, 0}, {400, 800, 80/3},
   {800, 800, 160/3}, {1200, 800, 80}, {2000, 800, 400/3}, {0, 1200, 0},
   {400, 1200, 40}, {800, 1200, 80}, {1200, 1200, 120}, {2000, 1200, 200},
   {0, 1600, 0}, {400, 1600, 160/3}, {800, 1600, 320/3}, {1200, 1600, 160},
   {2000, 1600, 800/3}, {400, 2700, 90}, {800, 2700, 180}, {1200, 2700, 270}};
CON = {{4, 5}, {5, 6}, {6, 7}, {7, 8}, {9, 10}, {10, 11}, {11, 12}, {12, 13},
   {14, 15}, {15, 16}, {16, 17}, {17, 18}, {19, 20}, {20, 21}, {21, 22},
   {22, 23}, {1, 5}, {5, 10}, {10, 15}, {15, 20}, {20, 24}, {2, 6}, {6, 11},
   {11, 16}, {16, 21}, {21, 25}, {3, 7}, {7, 12}, {12, 17}, {17, 22}, {22, 26}};
R = {{1, 1, 1}, {1, 1, 1}, {1, 1, 1}, {1, 1, 1}, {0, 0, 0}, {0, 0, 0}, {0, 0, 0},
   {1, 1, 1}, {1, 1, 1}, {0, 0, 0}, {0, 0, 0}, {0, 0, 0}, {1, 1, 1},
   {1, 1, 1}, {0, 0, 0}, {0, 0, 0}, {0, 0, 0}, {1, 1, 1}, {1, 1, 1}, {0, 0, 0},
   {0, 0, 0}, {0, 0, 0}, {1, 1, 1}, {1, 1, 1}, {1, 1, 1}, {1, 1, 1}};
F = {{0, 0, 0}, {0, 0, 0}, {0, 0, 0}, {0, 0, 0}, {0, 0, -15.7}, {0, 0, -15.7},
   {0, 0, -15.7}, {0, 0, 0}, {0, 0, 0}, {0, 0, -15.7}, {0, 0, -15.7}, {0, 0, -15.7},
   {0, 0, 0}, {0, 0, 0}, {0, 0, -15.7}, {0, 0, -15.7}, {0, 0, 0}, {0, 0, 0}, {0, 0, 0},
   {0, 0, -15.7}, {0, 0, 0}, {0, 0, 0}, {0, 0, 0}, {0, 0, 0}, {0, 0, 0}, {0, 0, 0}};
T0 = 200; EMod = 124800; XArea = 0.785; EA = EMod * XArea;
EA
cdim = Dimensions[CON];
ndim = Dimensions[XYZ];
c = cdim[[2]]
d = ndim[[2]]
n = ndim[[1]]
```

97968.

31

3

26

```
L = Table[Sqrt[Sum[(XYZ[[CON[[C, 1]], q]] - XYZ[[CON[[C, 2]], q]])^2, {q, 3}]], {C, c}];
L // MatrixForm
z = 1;
ID = Table[If[R[[i, j]] == 0, z++, 0], {i, n}, {j, d}];
ID // MatrixForm
dXYZ = Table[XYZ[[CON[[C, 1]], D]] - XYZ[[CON[[C, 2]], D]], {C, c}, {D, d}];
dXYZ // MatrixForm
fn = Max[ID]
```

$$\begin{pmatrix} \frac{40\sqrt{385}}{5} \\ \frac{40\sqrt{385}}{5} \\ \frac{40\sqrt{385}}{5} \\ \frac{80\sqrt{385}}{5} \\ \frac{80\sqrt{385}}{5} \\ \frac{80\sqrt{385}}{5} \\ \frac{80\sqrt{385}}{5} \\ \frac{160\sqrt{385}}{5} \\ 40\sqrt{101} \\ 40\sqrt{101} \\ 40\sqrt{101} \\ 80\sqrt{101} \\ \frac{80\sqrt{385}}{5} \\ \frac{80\sqrt{385}}{5} \\ \frac{80\sqrt{385}}{5} \\ \frac{160\sqrt{385}}{5} \\ \frac{40\sqrt{385}}{5} \\ \frac{40\sqrt{385}}{5} \\ \frac{40\sqrt{385}}{5} \\ \frac{40\sqrt{385}}{5} \\ \frac{330\sqrt{385}}{5} \\ \frac{80\sqrt{385}}{5} \\ \frac{80\sqrt{385}}{5} \\ \frac{80\sqrt{385}}{5} \\ \frac{80\sqrt{385}}{5} \\ \frac{220\sqrt{385}}{5} \\ 40\sqrt{101} \\ 40\sqrt{101} \\ 40\sqrt{101} \\ 40\sqrt{101} \\ 110\sqrt{101} \end{pmatrix}$$

$$\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 10 & 11 & 12 \\ 13 & 14 & 15 \\ 16 & 17 & 18 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 19 & 20 & 21 \\ 22 & 23 & 24 \\ 25 & 26 & 27 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 28 & 29 & 30 \\ 31 & 32 & 33 \\ 34 & 35 & 36 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

$$\begin{pmatrix}
-400 & 0 & -\frac{41}{5} \\
-400 & 0 & -\frac{41}{5} \\
-400 & 0 & -\frac{41}{5} \\
-800 & 0 & -\frac{82}{5} \\
-400 & 0 & -\frac{82}{5} \\
-400 & 0 & -\frac{82}{5} \\
-400 & 0 & -\frac{82}{5} \\
-800 & 0 & -\frac{328}{5} \\
-400 & 0 & -40 \\
-400 & 0 & -40 \\
-400 & 0 & -40 \\
-800 & 0 & -80 \\
-400 & 0 & -\frac{168}{5} \\
-400 & 0 & -\frac{168}{5} \\
-400 & 0 & -\frac{168}{5} \\
-800 & 0 & -\frac{528}{5} \\
0 & -400 & -\frac{41}{5} \\
0 & -400 & -\frac{41}{5} \\
0 & -400 & -\frac{41}{5} \\
0 & -400 & -\frac{82}{5} \\
0 & -1100 & -\frac{328}{5} \\
0 & -400 & -\frac{82}{5} \\
0 & -400 & -\frac{82}{5} \\
0 & -400 & -\frac{82}{5} \\
0 & -400 & -\frac{82}{5} \\
0 & -1100 & -\frac{328}{5} \\
0 & -400 & -40 \\
0 & -400 & -40 \\
0 & -400 & -40 \\
0 & -400 & -40 \\
0 & -1100 & -110
\end{pmatrix}$$

36

```
DOF = Table[x[k], {k, fn}];
DOF // MatrixForm
xyz = Table[If[ID[[i, j]] > 0, x[ID[[i, j]]], 0], {i, n}, {j, d}];
xyz // MatrixForm
dxyz = Table[xyz[[CON[[C, 1]], D]] - xyz[[CON[[C, 2]], D]], {C, c}, {D, d}];
dxyz // MatrixForm
```

$$\begin{pmatrix} x[1] \\ x[2] \\ x[3] \\ x[4] \\ x[5] \\ x[6] \\ x[7] \\ x[8] \\ x[9] \\ x[10] \\ x[11] \\ x[12] \\ x[13] \\ x[14] \\ x[15] \\ x[16] \\ x[17] \\ x[18] \\ x[19] \\ x[20] \\ x[21] \\ x[22] \\ x[23] \\ x[24] \\ x[25] \\ x[26] \\ x[27] \\ x[28] \\ x[29] \\ x[30] \\ x[31] \\ x[32] \\ x[33] \\ x[34] \\ x[35] \\ x[36] \end{pmatrix}$$

$$\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ x[1] & x[2] & x[3] \\ x[4] & x[5] & x[6] \\ x[7] & x[8] & x[9] \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ x[10] & x[11] & x[12] \\ x[13] & x[14] & x[15] \\ x[16] & x[17] & x[18] \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ x[19] & x[20] & x[21] \\ x[22] & x[23] & x[24] \\ x[25] & x[26] & x[27] \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ x[28] & x[29] & x[30] \\ x[31] & x[32] & x[33] \\ x[34] & x[35] & x[36] \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

$$\begin{pmatrix}
-x[1] & -x[2] & -x[3] \\
x[1]-x[4] & x[2]-x[5] & x[3]-x[6] \\
x[4]-x[7] & x[5]-x[8] & x[6]-x[9] \\
x[7] & x[8] & x[9] \\
-x[10] & -x[11] & -x[12] \\
x[10]-x[13] & x[11]-x[14] & x[12]-x[15] \\
x[13]-x[16] & x[14]-x[17] & x[15]-x[18] \\
x[16] & x[17] & x[18] \\
-x[19] & -x[20] & -x[21] \\
x[19]-x[22] & x[20]-x[23] & x[21]-x[24] \\
x[22]-x[25] & x[23]-x[26] & x[24]-x[27] \\
x[25] & x[26] & x[27] \\
-x[28] & -x[29] & -x[30] \\
x[28]-x[31] & x[29]-x[32] & x[30]-x[33] \\
x[31]-x[34] & x[32]-x[35] & x[33]-x[36] \\
x[34] & x[35] & x[36] \\
-x[1] & -x[2] & -x[3] \\
x[1]-x[10] & x[2]-x[11] & x[3]-x[12] \\
x[10]-x[19] & x[11]-x[20] & x[12]-x[21] \\
x[19]-x[28] & x[20]-x[29] & x[21]-x[30] \\
x[28] & x[29] & x[30] \\
-x[4] & -x[5] & -x[6] \\
x[4]-x[13] & x[5]-x[14] & x[6]-x[15] \\
x[13]-x[22] & x[14]-x[23] & x[15]-x[24] \\
x[22]-x[31] & x[23]-x[32] & x[24]-x[33] \\
x[31] & x[32] & x[33] \\
-x[7] & -x[8] & -x[9] \\
x[7]-x[16] & x[8]-x[17] & x[9]-x[18] \\
x[16]-x[25] & x[17]-x[26] & x[18]-x[27] \\
x[25]-x[34] & x[26]-x[35] & x[27]-x[36] \\
x[34] & x[35] & x[36]
\end{pmatrix}$$

```
e = Table[Sqrt[L[[C]]^2 + Sum[(2*dXYZ[[C, d]]*dxyz[[C, d]] + dxyz[[C, d]]^2), {d, 3}]] - L[[C]], {C, c}];
e //MatrixForm
```

$$
\begin{pmatrix}
-\frac{44\sqrt{778}}{5}+\sqrt{\frac{3491888}{5}+800x[1]+x[1]^2+x[2]^2+\frac{88x[15]}{5}+x[3]^2}\\
-\frac{44\sqrt{778}}{5}+\sqrt{\frac{3491888}{5}-800(x[1]-x[4])+(x[1]-x[4])^2+(x[2]-x[5])^2-\frac{88}{5}(x[3]-x[6])+(x[3]-x[6])^2}\\
-\frac{44\sqrt{778}}{5}+\sqrt{\frac{3491888}{5}-800(x[4]-x[7])+(x[4]-x[7])^2+(x[5]-x[8])^2-\frac{88}{5}(x[6]-x[9])+(x[6]-x[9])^2}\\
-\frac{44\sqrt{778}}{5}+\sqrt{\frac{876888}{5}-1600x[7]+x[7]^2+x[8]^2-\frac{368x[19]}{5}+x[9]^2}\\
-\frac{44\sqrt{778}}{5}+\sqrt{\frac{3491888}{5}+800x[10]+x[10]^2+x[11]^2+\frac{368x[38]}{5}+x[12]^2}\\
-\frac{44\sqrt{778}}{5}+\sqrt{\frac{3491888}{5}-800(x[10]-x[13])+(x[10]-x[13])^2+(x[11]-x[14])^2-\frac{368}{5}(x[12]-x[15])+(x[12]-x[15])^2}\\
-\frac{44\sqrt{778}}{5}+\sqrt{\frac{3491888}{5}-800(x[13]-x[16])+(x[13]-x[16])^2+(x[14]-x[17])^2-\frac{368}{5}(x[15]-x[18])+(x[15]-x[18])^2}\\
-\frac{368\sqrt{778}}{5}+\sqrt{\frac{876888}{5}-1600x[16]+x[16]^2+x[17]^2-\frac{528x[38]}{5}+x[18]^2}\\
-40\sqrt{101}+\sqrt{161600+800x[19]+x[19]^4+x[20]^4+80x[21]+x[21]^4}\\
-40\sqrt{101}+\sqrt{161600-800(x[19]-x[22])+(x[19]-x[22])^4+(x[20]-x[23])^4-80(x[21]-x[24])+(x[21]-x[24])^4}\\
-40\sqrt{101}+\sqrt{161600-800(x[22]-x[25])+(x[22]-x[25])^4+(x[23]-x[26])^4-80(x[24]-x[27])+(x[24]-x[27])^4}\\
-80\sqrt{101}+\sqrt{646400-1600x[25]+x[25]^4+x[26]^4-160x[27]+x[27]^4}\\
-\frac{44\sqrt{778}}{5}+\sqrt{\frac{3491888}{5}+800x[28]+x[28]^2+x[29]^2+\frac{528x[38]}{5}+x[30]^2}\\
-\frac{44\sqrt{778}}{5}+\sqrt{\frac{3491888}{5}-800(x[28]-x[31])+(x[28]-x[31])^2+(x[29]-x[32])^2-\frac{528}{5}(x[30]-x[33])+(x[30]-x[33])^2}\\
-\frac{44\sqrt{778}}{5}+\sqrt{\frac{3491888}{5}-800(x[31]-x[34])+(x[31]-x[34])^2+(x[32]-x[35])^2-\frac{528}{5}(x[33]-x[36])+(x[33]-x[36])^2}\\
-\frac{368\sqrt{778}}{5}+\sqrt{\frac{3502488}{5}-1600x[34]+x[34]^2+x[35]^2-\frac{4576x[38]}{5}+x[36]^2}\\
-\frac{44\sqrt{778}}{5}+\sqrt{\frac{3491888}{5}+x[1]^2+800x[2]+x[2]^2+\frac{88x[15]}{5}+x[3]^2}\\
-\frac{44\sqrt{778}}{5}+\sqrt{\frac{3491888}{5}+(x[1]-x[10])^2-800(x[2]-x[11])+(x[2]-x[11])^2-\frac{88}{5}(x[3]-x[12])+(x[3]-x[12])^2}\\
-\frac{44\sqrt{778}}{5}+\sqrt{\frac{3491888}{5}+(x[10]-x[19])^2-800(x[11]-x[20])+(x[11]-x[20])^2-\frac{88}{5}(x[12]-x[21])+(x[12]-x[21])^2}\\
-\frac{44\sqrt{778}}{5}+\sqrt{\frac{3491888}{5}+(x[19]-x[28])^2-800(x[20]-x[29])+(x[20]-x[29])^2-\frac{88}{5}(x[21]-x[30])+(x[21]-x[30])^2}\\
-\frac{328\sqrt{778}}{5}+\sqrt{\frac{3282888}{5}+x[28]^2-2200x[29]+x[29]^2-\frac{2288x[38]}{5}+x[30]^2}\\
-\frac{44\sqrt{778}}{5}+\sqrt{\frac{3491888}{5}+x[4]^2+800x[5]+x[5]^2+\frac{368x[16]}{5}+x[6]^2}\\
-\frac{44\sqrt{778}}{5}+\sqrt{\frac{3491888}{5}+(x[4]-x[13])^2-800(x[5]-x[14])+(x[5]-x[14])^2-\frac{368}{5}(x[6]-x[15])+(x[6]-x[15])^2}\\
-\frac{44\sqrt{778}}{5}+\sqrt{\frac{3491888}{5}+(x[13]-x[22])^2-800(x[14]-x[23])+(x[14]-x[23])^2-\frac{368}{5}(x[15]-x[24])+(x[15]-x[24])^2}\\
-\frac{44\sqrt{778}}{5}+\sqrt{\frac{3491888}{5}+(x[22]-x[31])^2-800(x[23]-x[32])+(x[23]-x[32])^2-\frac{368}{5}(x[24]-x[33])+(x[24]-x[33])^2}\\
-\frac{328\sqrt{778}}{5}+\sqrt{\frac{3285488}{5}+x[31]^2-2200x[32]+x[32]^2-\frac{4488x[55]}{5}+x[33]^2}\\
-40\sqrt{101}+\sqrt{161600+x[7]^4+800x[8]+x[8]^4+80x[9]+x[9]^4}\\
-40\sqrt{101}+\sqrt{161600+(x[7]-x[16])^4-800(x[8]-x[17])+(x[8]-x[17])^4-80(x[9]-x[18])+(x[9]-x[18])^4}\\
-40\sqrt{101}+\sqrt{161600+(x[16]-x[25])^4-800(x[17]-x[26])+(x[17]-x[26])^4-80(x[18]-x[27])+(x[18]-x[27])^4}\\
-40\sqrt{101}+\sqrt{161600+(x[25]-x[34])^4-800(x[26]-x[35])+(x[26]-x[35])^4-80(x[27]-x[36])+(x[27]-x[36])^4}\\
-110\sqrt{101}+\sqrt{1222100+x[34]^4-2200x[35]+x[35]^4-220x[36]+x[36]^4}
\end{pmatrix}
$$

$$
\text{Energy}=\sum_{i,1}^{c}\frac{EA*e[[i]]^2}{2*L[[i]]}+\sum_{l,1}^{c}T0*e[[l]]-\sum_{j,1}^{a}\sum_{k,1}^{d}F[[j,k]]*xyz[[j,k]]
$$

```
S = NMinimize[Energy, DOF];
S // MatrixForm
```

```
Clear[Nxyz]
Nxyz = xyz /. S[[2]]
Clear[NewXYZ]
NXYZ = XYZ + Nxyz;
Nxyz //MatrixForm
NXYZ //MatrixForm
MaxX =Max[Table[NXYZ[[i, 1]], {i, n}]]
MaxY =Max[Table[NXYZ[[i, 2]], {i, n}]]
MaxZ =Max[Table[NXYZ[[i, 3]], {i, n}]]
```

```
{{0, 0, 0}, {0, 0, 0}, {0, 0, 0}, {0, 0, 0}, {0.350249, 0.289041, -19.5864},
 {0.683028, 1.0701, -25.7298}, {0.772551, 1.78178, -23.3843}, {0, 0, 0},
 {0, 0, 0}, {1.18356, 0.564417, -25.9545}, {1.94449, 1.72803, -34.1985},
 {1.91394, 2.49226, -29.615}, {0, 0, 0}, {0, 0, 0}, {1.99754, 0.674753, -25.8858},
 {2.76861, 1.72208, -31.4399}, {1.91513, 1.74624, -21.5087}, {0, 0, 0},
 {0, 0, 0}, {2.42924, 0.619843, -21.5612}, {2.38056, 0.994919, -20.0739},
 {1.74787, 1.12872, -14.4634}, {0, 0, 0}, {0, 0, 0}, {0, 0, 0}, {0, 0, 0}}
```

0	0	0
0	0	0
0	0	0
0	0	0
0.350249	0.289041	-19.5864
0.683028	1.0701	-25.7298
0.772551	1.78178	-23.3843
0	0	0
0	0	0
1.18356	0.564417	-25.9545
1.94449	1.72803	-34.1985
1.91394	2.49226	-29.615
0	0	0
0	0	0
1.99754	0.674753	-25.8858
2.76861	1.72208	-31.4399
1.91513	1.74624	-21.5087
0	0	0
0	0	0
2.42924	0.619843	-21.5612
2.38056	0.994919	-20.0739
1.74787	1.12872	-14.4634
0	0	0
0	0	0
0	0	0
0	0	0

```
{{400       0         0},
 {800       0         0},
 {1200      0         0},
 {0         400       0},
 {400.35    400.289   -6.25311},
 {800.683   401.07    0.936818},
 {1200.77   401.782   16.6157},
 {2000      400       2 Pi/5},
 {0         800       0},
 {401.184   800.564   0.712196},
 {801.944   801.728   19.1348},
 {1201.91   802.492   50.385},
 {2000      800       4 Pi/5},
 {0         1200      0},
 {401.998   1200.67   14.1142},
 {802.769   1201.72   48.5601},
 {1201.92   1201.75   98.4913},
 {2000      1200      200},
 {0         1600      0},
 {402.429   1600.62   31.7721},
 {802.381   1600.99   86.5928},
 {1201.75   1601.13   145.537},
 {2000      1600      6 Pi/5},
 {400       2700      90},
 {800       2700      180},
 {1200      2700      270}}

2000

2700

270

cables = Table[Line[Table[NXYZ[[CON[[i, j]]]], {j, 2}]], {i, c}];
points = Table[If[NXYZ[[i, 1]] == 0 || NXYZ[[i, 1]] == MaxX, {Hue[0.1, 1, 0.8],
     Cuboid[{NXYZ[[i, 1]] - 1, NXYZ[[i, 2]] - 0.03*MaxY, NXYZ[[i, 3]] - 0.06*MaxZ},
      {NXYZ[[i, 1]] + 1, NXYZ[[i, 2]] + 0.03*MaxY, NXYZ[[i, 3]] + 0.06*MaxZ}]},
    If[NXYZ[[i, 2]] == 0 || NXYZ[[i, 2]] == MaxY, {Hue[0.1, 1, 0.8],
      Cuboid[{NXYZ[[i, 1]] - 0.03*MaxX, NXYZ[[i, 2]] - 1, NXYZ[[i, 3]] - 0.06*MaxZ},
       {NXYZ[[i, 1]] + 0.03*MaxX, NXYZ[[i, 2]] + 1, NXYZ[[i, 3]] + 0.06*MaxZ}]},
     {GrayLevel[0], Text[i, {NXYZ[[i, 1]], NXYZ[[i, 2]],
        NXYZ[[i, 3]] + 0.1*MaxZ}]}]], {i, n}];
forces = DeleteCases[Table[If[F[[i, 3]] ≠ 0, {Polygon[
      {{NXYZ[[i, 1]] - 0.0125*MaxX, NXYZ[[i, 2]], NXYZ[[i, 3]] - 0.1*MaxZ},
       {NXYZ[[i, 1]] + 0.0125*MaxX, NXYZ[[i, 2]], NXYZ[[i, 3]] - 0.1*MaxZ},
       {NXYZ[[i, 1]], NXYZ[[i, 2]], NXYZ[[i, 3]] - 0.15*MaxZ}}], Line[{NXYZ[[i]],
       {NXYZ[[i, 1]], NXYZ[[i, 2]], NXYZ[[i, 3]] - 0.1*MaxZ}}]}, 0], {i, n}], 0]
cablegrid = Flatten[Append[cables, points]]
```

Appendix C

Standard Mathematica Algorithm

```
Off[General::"spell1"];
Clear[XYZ, xyz, DOF, dxyz, S, Energy, e, tn, z, L, tc, d, n, CON, R, P, k, EA, ID];
FindMax = Function[{M, tn, L}, MaxX = 0; X = 0; MaxY = 0; Y = 0; MaxZ = 0; Z = 0;
   For[i = 1, i ≤ tn, i++, If[Abs[M[[i, 1]]] > Abs[MaxX], MaxX = M[[i, 1]]; X = i, MaxX; X];
    If[Abs[M[[i, 2]]] > Abs[MaxY], MaxY = M[[i, 2]]; Y = i, MaxY; Y];
    If[Abs[M[[i, 3]]] > Abs[MaxZ], MaxZ = M[[i, 3]]; Z = i, MaxZ; 1]];
   Print[TableForm[{{"Max X is ", MaxX * L, " at ", X},
      {"Max Y is ", MaxY * L, " at ", Y}, {"Max Z is ", MaxZ * L, " at ", Z}}]]];
Add = Function[x, a = 0; For[k = 1, k < x, k++, a = a + k]; a];
$TextStyle = {FontFamily → "Times", FontSize → 10};
<< Graphics`MultipleListPlot`
<< Graphics`Legend`
```

The spelling error has been turned off, and some of the variables have been cleared in the kernal. Find Max is a function that finds the maximum displacements that occur along each axis in dimensional form. Add is a function that is used to convert the nodal positions of the one-eighth portion of the unit cell to the full unit cell. $TextStyle specifies the text to be used on all output, and the <<Graphics loads the mathematica data necessary to create multiple list plots and legends.

```
B = 60; L = 300; Qp = -14.6; Qs = -3.06; Kp = 29.2; Ks = 0.16; EApW = 7300;
n = 50; H = L/(2n); b = B/L; h = H/L; EA = EApW * H; qp = Qp * L^2/EA; qs = Qs * L^2/EA;
pp = qp * h^2; ps = qs * h^2; kp = Kp * L^2/EA; ks = Ks * L^2/EA; tp = kp * h^2; ts = ks * h^2;
h
b
qp
qs
kp
ks
EA
pp
ps
tp
ts

1/100

1/5

-60.

-12.5753

36000.

197.26

21900

-0.006

-0.00125753

3.6

0.019726
```

B is half of the width of the pile cap (cm), L is the cell unit length (cm), Qp is the pressure acting on the geogrid above the pile (N/cm^2), Qs is the pressure acting on the geogrid above the soil (N/cm^2), Kp is the soil stiffness above the pile (N/cm^2/cm), Ks is the soil stiffness above the soil (N/cm^2/cm), EApW is the Elastic Modulus multiplied by the geogrid rib cross-sectional area divided by the number of ribs in a cm (N/cm), n is the number of geogrid ribs (cables) in half of the cell unit length. H is geogrid rib length (cable length) (cm), EA (N), qp, qs, kp, ks, b and h are dimensionless. pp is the dimensionless point load acting above the pile , ps is the dimensionless point load acting above the soft soil, tp is the dimensionless stiffness of the soil springs on top of the pile , and ts is the dimensionless stiffness of the soft soil springs

```
xyz = Partition[Flatten[Table[{i*h, j*h, 0}, {j, 0, n}, {i, j, n}]], 3];
```

xyz is the initial position matrix of the cable net nodes in the one-eighth unit cell. The coordinates are ordered in the same way as they are listed in the title.

```
a1 = 1; a2 = 2; b1 = 0; b2 = 1; b3 = n + 2;
Con = Partition[
   Flatten[Join[Table[{If[j == a2, a2++; a1 += 1, a1], a1 += 1}, {j, 1, n}, {i, j, n}],
     Table[{If[i == b2, b2++; b1 += 2, b1 += 1], b3++}, {i, 1, n}, {j, i, n}]]], 2];
Transpose[xyz] // MatrixForm
Transpose[Con] // MatrixForm
```

Con is the connectivity matrix. It is a list of the nodes that are connected by ribs (cables). It is a joined table, the first list is of the nodes that make up cables parallel to the x-axis and the second list is of the nodes that make up cables parallel to the y-axis.

```
tn = Dimensions[xyz][[1]]
tc = Dimensions[Con][[1]]
```

tn is the number of nodes in a cable net for one-eighth of the unit cell. tc is the number of cables in one-eighth of the unit cell.

```
c1 = n; c2 = n;
CoC = Join[Table[If[i ≤ n, 1/2, 1], {i, tc/2}],
   Table[If[i == c1, c2 -= 1; c1 += c2; 1/2, 1], {i, tc/2}]];
```

CoC is the cable coefficient vector. This vector considers the effect of perimeter cables, and reduces the energy a perimeter cable absorbs by one half. Perimeter cables are accounted for twice when the reduced unit cell is transformed to the full unit cell. The list must be joined so that the values correspond to the order of the Con matrix.

```
a = 0;
uvw = Table[{If[xyz[[i, 1]] == 1/2 || xyz[[i, 1]] == 0, 0, x[a += 1]], If[xyz[[i, 2]] == 0 ||
      xyz[[i, 2]] == 1/2, 0, If[xyz[[i, 1]] == xyz[[i, 2]], x[a], x[a += 1]]],
    x[a += 1]}, {i, tn}];
```

uvw is the displacement matrix. All degrees of freedom that will be restrained are given a value of zero the rest are given a variable x[a] where a is an increasing integer. The diagonal nodes that make up the eighth slice of the grid are restrained so that the x and y displacements are the same.

```
nDOF = a
DOF = Table[x[i], {i, nDOF}];

3825
```

nDOF is the number of degrees of freedom in the eighth. DOF is a list of all the variables, x[a], that represent degrees of freedom. This list will be used in the NMinimize function.

Rectangle Piles

```
Co = Table[If[xyz[[i, 1]] == xyz[[i, 2]] == 0 || xyz[[i, 1]] == xyz[[i, 2]] == 1/2,
    1/8, If[xyz[[i, 1]] == 1/2 && xyz[[i, 2]] == 0, 1/4,
     If[xyz[[i, 1]] == 1/2 || xyz[[i, 2]] == 0 || xyz[[i, 1]] == xyz[[i, 2]], 1/2, 1]]], {i, tn}];
p = Table[If[xyz[[i, 1]] ≤ b, pp * Co[[i]], ps * Co[[i]]], {i, tn}];
t = Table[If[xyz[[i, 1]] ≤ b, tp * Co[[i]], ts * Co[[i]]], {i, tn}];
```

Co is a list of load coefficients that account for the tributary areas that result from the geometry of a one eighth slice. The nodes on the exterior of the slice are one-half of the full load, The two corners that are at 45 degrees (the center node and corner node of the full grid) are one-eighth of the full load, and the corner that is 90 degrees (on the full grid it is the center edge node) are one-fourth of the full load. p and t are the list of point loads and soil stiffnesses acting on each node of the unit cell. The table multiplies the load coefficients by the pile or soil values depending on whether the node is over a pile or not.

```
e = Table[If[i ≤ tc / 2, c1 = Con[[i, 1]]; c2 = Con[[i, 2]];
    √((h + uvw[[c2, 1]] - uvw[[c1, 1]])^2 + (uvw[[c2, 2]] - uvw[[c1, 2]])^2 +
        (uvw[[c2, 3]] - uvw[[c1, 3]])^2) - h, c1 = Con[[i, 1]]; c2 = Con[[i, 2]];
    √((uvw[[c2, 1]] - uvw[[c1, 1]])^2 + (h + uvw[[c2, 2]] - uvw[[c1, 2]])^2 +
        (uvw[[c2, 3]] - uvw[[c1, 3]])^2) - h], {i, 1, tc}];
```

e is the elongation list. It uses the Con matrix and the displacement matrix to find the total elongation of a cable due to displacement.

$$\text{Energy} = \sum_{i,1}^{tc} \frac{CoC[[i]] * e[[i]]^2}{2 * h} + \sum_{j,1}^{t*} \frac{1}{2} * t[[j]] * (uvw[[j, 3]])^2 - \sum_{k,1}^{t*} p[[k]] * uvw[[k, 3]];$$

Energy is the sum of the total energy for the system of cables, soil stiffnesses, and point loads.

```
S = Timing[NMinimize[Energy, DOF]];
S[[1]]
```

S is the solution of the NMinimize function for the Energy equation and the list of degrees of freedom (x[a]). The timing function is used to time how long it took the function to find the solution.

```
uvw /. S[[2, 2]] >> Desktop/SchoolStuff/Research/Mathematica/ParametricStudyData/uvw
e /. S[[2, 2]] >> Desktop/SchoolStuff/Research/Mathematica/ParametricStudyData/e
```

The solution for each DOF is then placed in the displacement and elongation matrices and stored in seperate files on the hard drive.

```
Nuvw = << Desktop/SchoolStuff/Research/Mathematica/ParametricStudyData/uvw;
Ne = << Desktop/SchoolStuff/Research/Mathematica/ParametricStudyData/e;
Nxyz = xyz + Nuvw;
```

The solution is retrieved from the uvw and e files and called Nuvw and Ne respectively. This was done so that it was not necessary to run the NMinimize function everytime a notebook is opened. Nxyz is the new position matrix of the displaced grid. It is calculated by adding the initial position matrix and the displacement matrix.

```
ϵ = Table[Nε[[i]] / h, {i, tc}];
T = Table[EA * ϵ[[i]], {i, tc}];
Max[ϵ]
Max[T]
Min[Nuvw]
FindMax[Nuvw, tn, L]
Ds = Nuvw[[1, 3]] - Min[Nuvw]
Ds * L
pos = Table[If[xyz[[i, 1]] ≤ b, 0, 1], {i, tn}];
LDR = Total[Table[t[[i]] * Nuvw[[i, 3]] * pos[[i]], {i, tn}]] /
  Total[Table[p[[j]] * pos[[j]], {j, tn}]]
```

ϵ is a list of the strain values for the cables. T is a list of tension values for the cables. The maximum strain, tension, and vertical displacement is found. The FindMax function is used to find the maximum displacements in dimensional form. The differential settlement is found in non-dimensional and dimensional form by subtracting the smallest vertical displacement from the largest vertical displacement. pos is a list of all the nodes in the reduced unit cell, a 0 means the node is initially above the pile and a 1 means the node is initially above the soft soil. LDR is the ratio of the load the soft soil springs support to the load on the soft soil. It is found by finding the total of the soil stiffness times the vertical displacement of each node supported by the soft soil and dividing it by the total of the point loads acting on each node supported by the soft soil.

```
cn = n + 1; cx = 2 n (2 n + 1); cy = cx; fc = cx + cy; fn = (2 n + 1)^2
Fullxyz =
  Table[If[ j ≤ i && i ≤ cn && j ≤ cn, n1 = i + (j - 1) cn - Add[j]; {Nxyz[[n1, 1]], Nxyz[[n1, 2]],
     Nxyz[[n1, 3]]}, If[i > cn && 2cn - i ≥ j && j ≤ cn, n2 = cn - i + (j) cn - Add[j];
     {1 - Nxyz[[n2, 1]], Nxyz[[n2, 2]], Nxyz[[n2, 3]]}, If[i ≤ cn && j > i && j ≤ cn,
      n3 = j + (i - 1) cn - Add[i]; {Nxyz[[n3, 2]], Nxyz[[n3, 1]], Nxyz[[n3, 3]]},
      If[i > cn && j > 2cn - i && j ≤ cn, n4 = j + (2cn - i - 1) cn - Add[2cn - i];
       {1 - Nxyz[[n4, 2]], Nxyz[[n4, 1]], Nxyz[[n4, 3]]},
       If[i ≤ cn && cn - i ≥ j - cn && j > cn, n5 = cn - j + (i) cn - Add[i]; {Nxyz[[n5, 2]],
         1 - Nxyz[[n5, 1]], Nxyz[[n5, 3]]}, If[i ≤ cn && cn - i < j - cn && j > cn,
         n6 = i + (2cn - j - 1) cn - Add[2cn - j]; {Nxyz[[n6, 1]], 1 - Nxyz[[n6, 2]],
          Nxyz[[n6, 3]]}, If[i > cn && i ≤ j && j > cn, n7 = cn - i + (2cn - j) cn - Add[2cn - j];
          {1 - Nxyz[[n7, 1]], 1 - Nxyz[[n7, 2]], Nxyz[[n7, 3]]}, If[i > cn && i > j && j > cn,
           n8 = cn - j + (2cn - i) cn - Add[2cn - i]; {1 - Nxyz[[n8, 2]], 1 - Nxyz[[n8, 1]],
            Nxyz[[n8, 3]]}, {0, 0, 0}]]]]]]]], {j, 2n + 1}, {i, 2n + 1}];
```

cn is the number of nodes on the edge of the one eighth slice. cx is the total number of cables in the full grid that are parallel to the x-axis. cy is the total number of cables in the full grid that are parallel to the y-axis. fn is the total number of nodes in the full grid. Add is a function that takes into account the "triangle effect" of the one eighth slice. The function adds all integers starting at one all the way up to the value x. Fullxyz takes the final position matrix of the one eighth slice and creates a matrix of the final positions for a full grid. Each of the eight sections of the full grid has a seperate if statement. i is the column number of the grid and j is the row number. So that i will start at one and increase until i equals the number of columns in a row, and this will occur for every row, j, starting at one. i and j is essentially the position in the grid and the criteria of a if statement will be met depending on which of the eight sections of the gird the position is in. n1, n2, n3... are the node numbers of the eighth slice that correspond to the position.

```
a21 = 1; b21 = 1; a22 = 1;
FullCon = Join[Table[{If[i == (2n) a22 + 1, a22 += 1; a21 += 1, a21], a21 += 1}, {i, 1, cx}],
   Table[{b21++, b21 + 2n}, {i, 1, cy}]];
```

FullCon is the the full connectivity matrix. The matrix is made up of the two nodes that make up a cable. It is used to draw the cables in the Graphics 3D Plot. The matrix is joined because the first table is cables parallel to the x-axis and the second table makes up cables parallel to the y-axis.

```
Fullxyz = Partition[Flatten[Fullxyz], 3];
cablegrid = Table[Line[Table[Fullxyz[[FullCon[[i, j]]]], {j, 2}]], {i, fc}];
Show[Graphics3D[cablegrid, Axes → True, AxesLabel → {"x", "y", "z"},
   BoxRatios → {1, 1, 0.5}, ViewPoint → {1.3, -2.4, 1}, Boxed → False]];
```

Cablegrid is a list of all the lines (cables) in the grid so the Show[Graphics3D function draws the lines that represent each cable. Axes-True draws the axes and includes values. BoxRatios-is the ratio for each dimension of the box. ViewPoint {} is the angle to show the box. Boxed-False tells the kernal not to draw the box. AxesLabel is the labels to print on each axes.

$$tcx = \frac{tc}{2}$$

```
Fulle = Flatten[Table[If[ j ≤ i && i ≤ n && j ≤ n, c1 = i + (j - 1) n - Add[j]; №[[c1]],
     If[i > n && 2n + 1 - i ≥ j && j ≤ n, c2 = n - i + 1 + (j) n - Add[j]; №[[c2]],
      If[i ≤ n && j > i && j ≤ n + 1, c3 = tcx - 1 + j + (i - 1) n - Add[i]; №[[c3]],
       If[i > n && j > 2n + 1 - i && j ≤ n + 1, c4 = tcx - 1 + j + (2n - i) n - Add[2n - i + 1];
        №[[c4]], If[i ≤ n && i > 2n + 1 - j && j > n + 1, c5 = i + (2n - j + 1) n - Add[2n - j + 2];
         №[[c5]], If[i ≤ n && i ≤ 2n + 1 - j && j > n + 1, c6 = tcx + n - j + 1 + (i) n - Add[i];
          №[[c6]], If[i > n && i < j && j > n + 1, c7 = 3n - i + 1 + (2n - j) n - Add[2n - j + 2];
           №[[c7]], If[i > n && i ≥ j && j > n + 1, c8 = tcx + 2n - j + 1 + (2n - i) n -
              Add[2n - i + 1]; №[[c8]], 0]]]]]]], {j, 2n + 1}, {i, 2n}]];

Dimensions[Fulle][[1]]
Dimensions[Fullxyz][[1]]
```

Fulle is a list of the elongations of each cable in the unit cell. The order of the list corresponds to the FullCon list. Each if statement represents an eighth of the unit cell like the Fullxyz matrix.

```
xt0 = -h/2; xt20 = xt0; xt50 = xt0; xe0 = xt0; xe25 = xt0; xe50 = xt0;
TwoDXSection0 = Table[{Fullxyz[[i, 1]], Fullxyz[[i, 3]]}, {i, 1 + 2n}];
TwoDXSection25 =
  Table[{Fullxyz[[i, 1]], Fullxyz[[i, 3]]}, {i, n^2 + 0.5n + 1, n^2 + 2.5n + 1}];
TwoDXSection50 = Table[{Fullxyz[[i, 1]], Fullxyz[[i, 3]]},
   {i, 2*n^2 + n + 1, 2*n^2 + 3n + 1}];
MultipleListPlot[TwoDXSection0, TwoDXSection25, TwoDXSection50,
 PlotJoined → True, PlotLegend → {"y = 0", "y = 0.25", "y = 0.5"},
 LegendPosition → {0.95, -0.4}, LegendSize → {0.5, 0.9},
 PlotLabel → StyleForm["w Versus x", FontWeight → Bold, FontSize → 12],
 Frame → True, FrameLabel → {"x", "w"}, RotateLabel → False, SymbolShape → None]
Fulle = Table[Fulle[[i]]/h, {i, cx}];
FullT = Table[EA*Fulle[[i]], {i, cx}];
TPlot0 = Table[xt0 += h; {xt0, FullT[[i]]}, {i, 2n}];
TPlot20 = Table[xt20 += h; {xt20, FullT[[i]]}, {i, 0.8n^2 + 1, 0.8n^2 + 2n}];
TPlot50 = Table[xt50 += h; {xt50, FullT[[i]]}, {i, 2*n^2 + 1, 2*n^2 + 2*n}];
MultipleListPlot[TPlot0, TPlot20, TPlot50,
 PlotJoined → True, PlotLegend → {Y = "0", "Y = 0.25", "Y = 0.5"},
 LegendPosition → {0.95, -0.4}, LegendSize → {0.5, 0.9},
 PlotLabel → StyleForm["Tension", FontWeight → Bold, FontSize → 12],
 Frame → True, FrameLabel → {"X", "T"}, RotateLabel → False, SymbolShape → None]
ePlot0 = Table[xe0 += h; {xe0, Fulle[[i]]}, {i, 2n}];
ePlot25 = Table[xe25 += h; {xe25, Fulle[[i]]}, {i, n^2 + 1, n^2 + 2n}];
ePlot50 = Table[xe50 += h; {xe50, Fulle[[i]]}, {i, 2*n^2 + 1, 2*n^2 + 2*n}];
MultipleListPlot[ePlot0, ePlot25, ePlot50,
 PlotJoined → True, PlotLegend → {"y = 0", "y = 0.25", "y = 0.5"},
 LegendPosition → {0.95, -0.4}, LegendSize → {0.5, 0.9},
 PlotLabel → StyleForm["ϵx Versus x", FontWeight → Bold, FontSize → 12],
 Frame → True, FrameLabel → {"x", "ϵ"}, RotateLabel → False, SymbolShape → None]
```

xt0, xt20, ...etc are the initial variables used to plot the position of each cable along the x-axis. TwoDXSection0 forms a table of the x and z coordinates along the edge, TwoDXSection25 forms a table of the x and z coordinates at the quarter of the unit cell, TwoDXSection50 forms a table of the x and z coordinates along the centerline of the unit cell. The values of all three of these tables are plotted on a multiple list plot. Specifications for the legend are made using the functions PlotLegend, LegendPosition, and LegendSize, PlotJoined→True plots lines for all the data, PlotLabel→StyleForm specifies the title of the plot, that the title should be bold, and the size of the font. Frame→True plots a frame around the whole plot. FrameLabel→{"x","w"} specifies the labels that should be printed on each axis of the plot. x is the bottom axis and w is the vertical axis. RotateLabel→False prints the vertical axis label horizontal. SymbolShape→None eliminates all symbols that would be plotted for each point. Fulle is a table of the strain of the cables parallel to the x-axis. Tension is a table of the tension values of the cables parallel to the x-axis. TPlot0...etc and ePlot0...etc. serve the same purpose as the TwoDXSection tables except for the Tension and Strain values. Different ranges of values were used when creating these tables because the organization of the Fullxyz table was different than the FullCon table.

```
a31 = -(n + 1); b31 = n + 2; xt45 = -h; xe45 = xt45;
DiagXSection = Table[If[i ≤ n + 1, a31 += b31; b31 -= 1; {Nxyz[[a31, 1]], Nxyz[[a31, 3]]},
    b31 += 1; a31 -= b31; {1 - Nxyz[[a31, 1]], Nxyz[[a31, 3]]}], {i, 2n + 1}];
ListPlot[DiagXSection, PlotJoined → True, PlotLabel →
  StyleForm["w Versus x Along Diagonal", FontWeight → Bold, FontSize → 12],
 Frame → True, FrameLabel → {"x", "w"}, RotateLabel → False]
DiagTPlot = Table[xt45 += h;
   {xt45, If[i == 1, FullT[[2n*(i - 1) + i]], If[i == 2n + 1, FullT[[2n*(i - 1) + i - 1]],
      1/2 (FullT[[2n*(i - 1) + i - 1]] + FullT[[2n*(i - 1) + i]])]]}, {i, 2n + 1}];
ListPlot[DiagTPlot, PlotJoined → True, PlotLabel →
  StyleForm["T Versus x Along Diagonal", FontWeight → Bold, FontSize → 12],
 Frame → True, FrameLabel → {"x", "T"}, RotateLabel → False]
DiagePlot = Table[xe45 += h;
   {xe45, If[i == 1, Fulle[[2n*(i - 1) + i]], If[i == 2n + 1, Fulle[[2n*(i - 1) + i - 1]],
      1/2 (Fulle[[2n*(i - 1) + i - 1]] + Fulle[[2n*(i - 1) + i]])]]}, {i, 2n + 1}];
ListPlot[DiagePlot, PlotJoined → True, PlotLabel →
  StyleForm["ε Versus x Along Diagonal", FontWeight → Bold, FontSize → 12],
 Frame → True, FrameLabel → {"x", "ε"}, RotateLabel → False]
```

The diagonal tension and strain plots take the average of the two cable values that meet at each point along the diagonal. This seemed most accurate because it was uncertain which cable should be compared to the cables along the diagonal of the 45 degree orientation.

```
Fullw = Partition[Table[Fullxyz[[i, 3]], {i, fn}], 2n + 1];
wMin = Min[Fullw]
wMax = Max[Fullw]
ListPlot3D[Fullw, Mesh → False,
 MeshRange → {{0, 1}, {0, 1}}, AxesLabel → {"x", "y", "z    "}]
ListContourPlot[Fullw, MeshRange → {{0, 1}, {0, 1}},
 FrameLabel → {"x", "y"}, RotateLabel → False]
```

Fullw is the vertical displacements of the cable net. Both of these plots don't take into account the small x and y displacements that occur.

```
Fullex = Partition[Fulle, 2n];
ListPlot3D[Fullex, Mesh → False,
 MeshRange → {{0, 1}, {0, 1}}, AxesLabel → {"x", "y", "ε    "}]
ListContourPlot[Fullex, MeshRange → {{0, 1}, {0, 1}},
  FrameLabel → {"x", "y"}, RotateLabel → False];
```

Fullex is a list of all the strains of the cables parallel to the x-axis

```
Total[Flatten[Fullw]]
Total[Flatten[Fullex]]
```

Appendix D

45-Degree Orientation Algorithm

```
Off[General::"spell1"];
Clear[XYZ, xyz, DOF, dxyz, S, Energy, e, tn, z, L, tc, d, n, CON, R, P, k, EA, ID];
FindMax = Function[{M, tn, L}, MaxX = 0; X = 0; MaxY = 0; Y = 0; MaxZ = 0; Z = 0;
   For[i = 1, i ≤ tn, i++, If[Abs[M[[i, 1]]] > Abs[MaxX], MaxX = M[[i, 1]]; X = i, MaxX; X];
    If[Abs[M[[i, 2]]] > Abs[MaxY], MaxY = M[[i, 2]]; Y = i, MaxY; Y];
    If[Abs[M[[i, 3]]] > Abs[MaxZ], MaxZ = M[[i, 3]]; Z = i, MaxZ; 1]];
   Print[TableForm[{{"Max X is ", MaxX*L, " at ", X},
      {"Max Y is ", MaxY*L, " at ", Y}, {"Max Z is ", MaxZ*L, " at ", Z}}]]];
Add = Function[x, a = 0; For[k = 1, k < x, k++, a = a + k]; a];
$TextStyle = {FontFamily → "Times", FontSize → 10};
<< Graphics`MultipleListPlot`
<< Graphics`Legend`
```

The spelling error has been turned off, and some of the variables have been cleared in the kernal. Find Max is a function that finds the maximum displacements that occur along each axis in dimensional form. Add is a function that is used to convert the nodal positions of the one-eighth portion of the unit cell to the full unit cell. $TextStyle specifies the text to be used on all output, and the <<Graphics loads the mathematica data neccessary to create multiple list plots and legends.

```
B = 60; L = 300; Qp = -14.6; Qs = -3.06; Kp = 29.2; Ks = 0.16; EApW = 7300; n = 7;
H = L√2/(4n); b = B/L; h = H/L; h45 = 1/(2n); EA = EApW*H; qp = Qp*L²/EA; qs = Qs*L²/EA;
pp = qp*h²; ps = qs*h²; kp = Kp*L²/EA; ks = Ks*L²/EA; tp = kp*h²; ts = ks*h²;
h
h45
b
qp
qs
kp
ks
EA
pp
ps
tp
ts
```

$$\frac{1}{14\sqrt{2}}$$

$$\frac{1}{14}$$

$$\frac{1}{5}$$

```
-11.8794

-2.48979

7127.64

39.0555
```

$$\frac{547500\sqrt{2}}{7}$$

```
-0.00606092

-0.0012703

3.63655

0.0199263
```

B is half of the width of the pile cap (cm), L is the cell unit length (cm), Qp is the pressure acting on the geogrid above the pile (N/cm^2), Qs is the pressure acting on the geogrid above the soil (N/cm^2), Kp is the soil stiffness above the pile (N/cm^2/cm), Ks is the soil stiffness above the soil (N/cm^2/cm), EApW is the Elastic Modulus multiplied by the geogrid rib cross-sectional area divided by the number of ribs in a cm (N/cm), n is the number of geogrid ribs (cables) in half of the cell unit length. H is geogrid rib length (cable length) (cm), EA (N), qp, qs, kp, ks, b and h are dimensionless. pp is the dimensionless point load acting above the pile , ps is the dimensionless point load acting above the soft soil, tp is the dimensionless stiffness of the soil spring on top of the pile , and ts is the dimensionless stiffness of the soft soil spring.

$$xyz = \text{Partition}\left[\text{Flatten}\left[\text{Table}\left[\left\{i * h45,\ j * \frac{h45}{2},\ 0\right\},\ \{j,\ 0,\ 2n\},\ \left\{i,\ \frac{j}{2},\ n\right\}\right]\right],\ 3\right];$$

xyz is the initial position matrix of the cable net nodes that make up one-eighth of an unit cell. The coordinates are ordered in the same way as they are listed in the title.

```
a1 = 0; b1 = n + 1;
Con = Partition[Flatten[Table[If[OddQ[j] == True && i == 1, {a1 += 2, b1 += 1},
      If[OddQ[i] == True, {a1 += 1, b1 += 1}, {b1, a1 + 1}]], {j, 0, 2n}, {i, 2n - j}]], 2];
Transpose[xyz] // MatrixForm
Transpose[Con] // MatrixForm
```

Con is the connectivity matrix. It is a list of the nodes that are connected by ribs (cables). It is an joined table, the first list is of the nodes that make up the cables parallel to the x-axis and the second list is of the nodes that make up the cables parallel to the y-axis.

```
tn = Dimensions[xyz][[1]]
tc = Dimensions[Con][[1]]
```

tn is the number of nodes in a cable net for one-eighth of the unit cell. tc is the number of cables in one-eighth of the unit cell.

$$CoC = \text{Flatten}\left[\text{Table}\left[\text{If}\left[i == j,\ \frac{1}{2},\ 1\right],\ \{j,\ 2n\},\ \{i,\ j,\ 2n\}\right]\right];$$

CoC is the cable coefficient vector. This vector considers the effect of perimeter cables, and reduces the energy a perimeter cable absorbs by one half. Perimeter cables are accounted for twice when the reduced unit cell is transformed to the full unit cell.

$$a = 0;\ uvw = \text{Table}\left[\left\{\text{If}\left[xyz[[i,\ 1]] == \frac{1}{2} \,||\, xyz[[i,\ 1]] == 0,\ 0,\ x[a += 1]\right],\ \text{If}\left[xyz[[i,\ 2]] == 0 \,||\, xyz[[i,\ 2]] == \frac{1}{2},\ 0,\ \text{If}[xyz[[i,\ 1]] == xyz[[i,\ 2]],\ x[a],\ x[a += 1]]\right],\ x[a += 1]\right\},\ \{i,\ tn\}\right];$$

uvw is the displacement matrix. All degrees of freedom that will be restrained are given a value of zero, the rest are given a variable x[a] where a is an increasing integer. The diagonal nodes along the edge of the grid that make up the reduced one-eighth unit cell are restrained so that the x and y displacements are equal.

```
nDOF = a
DOF = Table[x[i], {i, nDOF}];
```

nDOF is the number of degrees of freedom in one-eighth of the unit cell. DOF is a list of all the variables, x[a], that represent degrees of freedom. This list will be used in the NMinimize function.

Rectangle Piles

$$\text{Co} = \text{Table}\left[\text{If}\left[\text{xyz}[[i,1]] == \text{xyz}[[i,2]] == 0 \,||\, \text{xyz}[[i,1]] == \text{xyz}[[i,2]] == \frac{1}{2},\ \frac{1}{8},\ \text{If}\left[\text{xyz}[[i,1]] == \frac{1}{2} \,\&\&\, \text{xyz}[[i,2]] == 0,\ \frac{1}{4},\ \text{If}\left[\text{xyz}[[i,1]] == \frac{1}{2} \,||\, \text{xyz}[[i,2]] == 0 \,||\, \text{xyz}[[i,1]] == \text{xyz}[[i,2]],\ \frac{1}{2},\ 1\right]\right], \{i, tn\}\right];$$

```
p = Table[If[xyz[[i, 1]] ≤ b, pp * Co[[i]], ps * Co[[i]]], {i, tn}];
t = Table[If[xyz[[i, 1]] ≤ b, tp * Co[[i]], ts * Co[[i]]], {i, tn}];
```

Co is a list of load coefficients that account for the tributary areas that result from the geometry of one-eighth of the unit cell. The nodes on the perimeter of the cable net that makes up one-eighth of the unit cell will be one-half of the full load. The two corners with 45-degree angles (the center node and corner node of the full unit cell) will be one-eighth of the full load, and the corner with a 90 degree angle (on the full grid it is the center edge node) will be one-fourth of the full load. p and k are the list of point loads and soil stiffness that multiplies the load coefficients by the pile or soil values depending on whether the node is over a pile or not.

$$\text{e} = \text{Table}\left[c1 = \text{Con}[[i,1]];\ c2 = \text{Con}[[i,2]];\ \text{If}\left[c1 < c2,\ \sqrt{\left(\left(\frac{h}{\sqrt{2}} + \text{uvw}[[c2,1]] - \text{uvw}[[c1,1]]\right)^2 + \left(\frac{h}{\sqrt{2}} + \text{uvw}[[c2,2]] - \text{uvw}[[c1,2]]\right)^2 + (\text{uvw}[[c2,3]] - \text{uvw}[[c1,3]])^2\right)} - h,\ \sqrt{\left(\left(\frac{h}{\sqrt{2}} + \text{uvw}[[c2,1]] - \text{uvw}[[c1,1]]\right)^2 + \left(\frac{h}{\sqrt{2}} - \text{uvw}[[c2,2]] + \text{uvw}[[c1,2]]\right)^2 + (\text{uvw}[[c2,3]] - \text{uvw}[[c1,3]])^2\right)} - h\right], \{i, 1, tc\}\right];$$

```
e = Table[c1 = Con[[i, 1]]; c2 = Con[[i, 2]];
   √((xyz[[c2, 1]] - xyz[[c1, 1]] + uvw[[c2, 1]] - uvw[[c1, 1]])^2 +
      (xyz[[c2, 2]] - xyz[[c1, 2]] + uvw[[c2, 2]] - uvw[[c1, 2]])^2 +
      (uvw[[c2, 3]] - uvw[[c1, 3]])^2) - h, {i, 1, tc}];
```

e is the elongation list. It uses the Con matrix and the displacement matrix to find the total elongation of a cable due to displacement.

$$\text{Energy} = \sum_{i,1}^{tc} \frac{\text{CoC}[[i]] * \text{e}[[i]]^2}{2 * h} + \sum_{j,1}^{tp} \frac{1}{2} * \text{t}[[j]] * (\text{uvw}[[j,3]])^2 - \sum_{k,1}^{tp} \text{p}[[k]] * \text{uvw}[[k,3]];$$

Energy is the sum of the total energy for the system of cables, soil stiffnesses, and point loads.

```
S = Timing[NMinimize[Energy, DOF]];
S[[1]]
```

S is the solution of the NMinimize function for the Energy equation and the list of degrees of freedom (x[a]). The timing function is used to time how long it took the function to find the solution.

```
uvw /. S[[2, 2]] >> Desktop/SchoolStuff/Research/Mathematica/ParametricStudyData/uvw45
e /. S[[2, 2]] >> Desktop/SchoolStuff/Research/Mathematica/ParametricStudyData/e45
```

The solution for each DOF is then placed in the displacement and elongation matrices and stored in seperate files on the hard drive.

```
Nuvw = << Desktop/SchoolStuff/Research/Mathematica/ParametricStudyData/uvw45;
Ne = << Desktop/SchoolStuff/Research/Mathematica/ParametricStudyData/e45;
Nxyz = xyz + Nuvw;
```

The solution is retrieved fom the uvw and e files and called Nuvw and Ne respectively. This was done so that it was not necessary to run the NMinimize function everytime a notebook is opened. Nxyz is the new position matrix of the displaced grid. It is calculated by adding the initial position matrix and the displacement matrix.

```
ϵ = Table[Ne[[i]]/h, {i, tc}];
T = Table[EA * ϵ[[i]], {i, tc}];
Max[ϵ]
Max[T]
Min[Nuvw]
Min[Nuvw] * L
FindMax[Nuvw, tn, L]
Ds = Nuvw[[1, 3]] - Min[Nuvw]
Ds * L
pos = Table[If[xyz[[i, 1]] ≤ b, 0, 1], {i, tn}];
LDR = Total[Table[t[[i]] * Nuvw[[i, 3]] * pos[[i]], {i, tn}]] /
  Total[Table[p[[j]] * pos[[j]], {j, tn}]]
```

ϵ is a list of the strain values for the cables. T is a list of tension values for the cables. The maximum strain, tension, and vertical displacement are found. The FindMax function is used to find the maximum displacements in dimensional form. The differential settlement is found in non-dimensional and dimensional form by subtracting the smallest vertical displacement from the largest vertical displacement. pos is a list of all the nodes in the reduced unit cell, a 0 means the node is initially above the pile and 1 means the node is initially above the soft soil. LDR is the ratio of the load the soft soil springs support to the load on the soft soil. It is found by finding the total of the soil stiffness times the vertical displacement of each node supported by the soft soil and dividing it by the total of the point loads acting on each node supported by the soft soil.

```
cn = n + 1; fn = (2n + 1)^2 + (2n)^2
Fullxyz = Sort[
   Partition[Flatten[Table[If[xyz[[i, 1]] == xyz[[i, 2]] == 1/2, {xyz[[i, 1]], xyz[[i, 2]],
        Nxyz[[i, 1]], Nxyz[[i, 2]], Nxyz[[i, 3]]}, If[xyz[[i, 1]] == xyz[[i, 2]],
       {{xyz[[i, 1]], xyz[[i, 2]], Nxyz[[i, 1]], Nxyz[[i, 2]], Nxyz[[i, 3]]},
        {1 - xyz[[i, 1]], xyz[[i, 2]], 1 - Nxyz[[i, 1]], Nxyz[[i, 2]], Nxyz[[i, 3]]},
        {xyz[[i, 1]], 1 - xyz[[i, 2]], Nxyz[[i, 1]], 1 - Nxyz[[i, 2]], Nxyz[[i, 3]]},
        {1 - xyz[[i, 1]], 1 - xyz[[i, 2]], 1 - Nxyz[[i, 1]],
          1 - Nxyz[[i, 2]], Nxyz[[i, 3]]}}, If[xyz[[i, 1]] == 1/2,
        {{xyz[[i, 1]], xyz[[i, 2]], Nxyz[[i, 1]], Nxyz[[i, 2]], Nxyz[[i, 3]]},
         {1 - xyz[[i, 2]], xyz[[i, 1]], 1 - Nxyz[[i, 2]], Nxyz[[i, 1]], Nxyz[[i, 3]]},
         {xyz[[i, 2]], xyz[[i, 1]], Nxyz[[i, 2]], Nxyz[[i, 1]], Nxyz[[i, 3]]},
         {xyz[[i, 1]], 1 - xyz[[i, 2]], Nxyz[[i, 1]], 1 - Nxyz[[i, 2]], Nxyz[[i, 3]]}},
        {{xyz[[i, 1]], xyz[[i, 2]], Nxyz[[i, 1]], Nxyz[[i, 2]], Nxyz[[i, 3]]},
         {1 - xyz[[i, 1]], xyz[[i, 2]], 1 - Nxyz[[i, 1]], Nxyz[[i, 2]], Nxyz[[i, 3]]},
         {xyz[[i, 1]], 1 - xyz[[i, 2]], Nxyz[[i, 1]], 1 - Nxyz[[i, 2]], Nxyz[[i, 3]]},
         {1 - xyz[[i, 1]], 1 - xyz[[i, 2]], 1 - Nxyz[[i, 1]], 1 - Nxyz[[i, 2]], Nxyz[[i, 3]]},
         {xyz[[i, 2]], xyz[[i, 1]], Nxyz[[i, 2]], Nxyz[[i, 1]], Nxyz[[i, 3]]},
         {1 - xyz[[i, 2]], xyz[[i, 1]], 1 - Nxyz[[i, 2]], Nxyz[[i, 1]], Nxyz[[i, 3]]},
         {xyz[[i, 2]], 1 - xyz[[i, 1]], Nxyz[[i, 2]], 1 - Nxyz[[i, 1]], Nxyz[[i, 3]]},
         {1 - xyz[[i, 2]], 1 - xyz[[i, 1]], 1 - Nxyz[[i, 2]],
          1 - Nxyz[[i, 1]], Nxyz[[i, 3]]}}]], {i, tn}]], 5]];
Dimensions[
 Fullxyz]
```

cn is the number of nodes on the edge of the one eighth slice. fn is the total number of nodes in the full unit cell. Fullxyz takes the final position matrix of the cable net that makes up one-eighth of the unit cell and creates a matrix of the final positions for a full unit cell. The first if statement meets the criteria of the center node in the full unit cell (the upper corner of the reduced one-eighth unit cell), and it adds one row in the fullxyz matrix for this node. The second if statement meets the criteria of a node along the 45-degree diagonal in the full unit cell (diagonal edge of one-eighth reduced unit cell), and it adds four rows in the fullxyz matrix for this node. The third if statement meets the criteria of a node along the centerline of the full unit cell (the vertical edge of one-eighth of the unit cell), and it adds four rows in the fullxyz matrix for this node. The rest of the nodes are either central nodes or edge nodes on the full unit cell (central nodes or horizontal edge nodes on one-eighth of the unit cell), and it adds eight rows in the fullxyz matrix for this node. There are five columns added for each row, the first two are the initial position along the x and y axes, and the final three are the final coordinates of the node. The first two columns are used to sort the rows so that the matrix is in proper order.

```
a21 = 0; b21 = 2n; fc = (4n)^2
FullCon = Partition[Flatten[Table[If[OddQ[j] == True, If[i == 1,
       {a21 += 1, b21 += 2}, If[OddQ[i] == True, {a21 += 1, b21 += 1}, {b21, a21 + 1}]],
      If[OddQ[i] == True, If[i == 1, {b21 += 1, a21 += 2}, {b21 += 1, a21 += 1}],
       {a21, b21 + 1}]], {j, 4n}, {i, 4n}]], 2];
```

FullCon is the the full connectivity matrix. The matrix is made up of the two nodes that make up a cable. It is used to draw the cables in the Graphics 3D Plot. The matrix is joined because the first table makes up the cables that are parallel to the x-axis and the second table makes up the cables parallel to the y-axis.

```
cablegrid = Flatten[
   Table[Line[Table[{Fullxyz[[FullCon[[i, j]], 3]], Fullxyz[[FullCon[[i, j]], 4]],
       Fullxyz[[FullCon[[i, j]], 5]]}, {j, 2}]], {i, fc}]];
Show[Graphics3D[cablegrid, Axes → True, AxesLabel → {"x", "y", "z"},
   BoxRatios → {1, 1, 0.5}, ViewPoint → {1.3, -2.4, 1}, Boxed → False]];
```

Cablegrid is a list of all the lines (cables) in the grid so the *Show[Graphics3D* function draws the lines that represent each cable. Axes-True draws the axes and includes values. BoxRatios-is the ratio for each dimension of the box. ViewPoint {} is the angle to show the box. Boxed-False tells the kernal not to draw the box. AxesLabel is the labels to print on each axes.

```
Fulle = Flatten[Table[If[ j ≤ i && i ≤ 2n && j ≤ 2n, c1 = i + (j - 1) 2n - Add[j];
     №[[c1]], If[i > 2n && 4n - i + 1 ≥ j && j ≤ 2n, c2 = 2n - i + 1 + (j) 2n - Add[j];
      №[[c2]], If[i ≤ 2n && j > i && j ≤ 2n, c3 = j + (i - 1) 2n - Add[i]; №[[c3]],
       If[i > 2n && j > 4n - i && j ≤ 2n, c4 = j + (4n - i) 2n - Add[4n - i + 1]; №[[c4]],
        If[i ≤ 2n && i > 4n - j && j > 2n, c5 = i + (4n - j) 2n - Add[4n - j + 1]; №[[c5]],
         If[i ≤ 2n && i ≤ 4n - j && j > 2n, c6 = 2n - j + 1 + (i) 2n - Add[i]; №[[c6]],
          If[i > n && i < j && j > 2n, c7 = 4n - i + 1 + (4n - j) 2n - Add[4n - j + 1]; №[[c7]],
           If[i > n && i ≥ j && j > 2n, c8 = 4n - j + 1 + (4n - i) 2n - Add[4n - i + 1];
            №[[c8]], 0]]]]]]], {j, 4n}, {i, 4n}]];
Dimensions[Fulle][[1]]
Dimensions[Fullxyz][[1]]
```

Fulle is a list of the elongation of each cable in the unit cell. The order of the list corresponds to the FullCon list. Each if statement represents an eighth of the unit cell like the Fullxyz matrix.

```
xt0 = -h45/4 ; xe0 = xt0; xt20 = xt0; xe30 = xt0; xt50 = xt0; xe50 = xt0;
TwoDXSection0 = Table[{Fullxyz[[i, 4]], Fullxyz[[i, 5]]}, {i, 1 + 2 n}];
TwoDXSection30 =
  Table[{Fullxyz[[i, 4]], Fullxyz[[i, 5]]}, {i, 2.4*n^2 + 0.6*n + 1, 2.4*n^2 + 2.6*n + 1}];
TwoDXSection50 = Table[{Fullxyz[[i, 4]], Fullxyz[[i, 5]]},
   {i, 4*n^2 + n + 1, 4*n^2 + 3*n + 1}];
MultipleListPlot[TwoDXSection0, TwoDXSection30, TwoDXSection50,
 PlotJoined → True, PlotLegend → {"y = 0", "y = 0.30", "y = 0.5"},
 LegendPosition → {0.95, -0.4}, LegendSize → {0.5, 0.9},
 PlotLabel → StyleForm["w Versus x", FontWeight → Bold, FontSize → 12],
 Frame → True, FrameLabel → {"x", "w"}, RotateLabel → False, SymbolShape → None]
Fulle = Table[(Fulle[[i]]/h), {i, fc}];
FullT = Table[EA * Fulle[[i]], {i, fc}];
TPlot0 = Table[xt0 += h/Sqrt[2] : {xt0, FullT[[i]]}, {i, 1, 4 n}];
TPlot20 = Table[xt20 += h/Sqrt[2] : {xt20, FullT[[i]]}, {i, 3.2*n^2 + 1, 3.2*n^2 + 4 n}];
TPlot50 = Table[xt50 += h/Sqrt[2] : {xt50, FullT[[i]]}, {i, 8*n^2 + 1, 8*n^2 + 4 n}];
MultipleListPlot[TPlot0, TPlot20, TPlot50,
 PlotJoined → True, PlotLegend → {"y = 0", "y = 0.2", "y = 0.5"},
 LegendPosition → {0.95, -0.4}, LegendSize → {0.5, 0.9},
 PlotLabel → StyleForm["Tension", FontWeight → Bold, FontSize → 12],
 Frame → True, FrameLabel → {"x", "T"}, RotateLabel → False, SymbolShape → None]
ePlot0 = Table[xe0 += h/Sqrt[2] : {xe0, Fulle[[i]]}, {i, 1, 4 n}];
ePlot30 = Table[xe30 += h/Sqrt[2] : {xe30, Fulle[[i]]}, {i, 4.8*n^2 + 1, 4.8*n^2 + 4 n}];
ePlot50 = Table[xe50 += h/Sqrt[2] : {xe50, Fulle[[i]]}, {i, 8*n^2 + 1, 8*n^2 + 4 n}];
MultipleListPlot[ePlot0, ePlot30, ePlot50,
 PlotJoined → True, PlotLegend → {"y = 0", "y = 0.30", "y = 0.5"},
 LegendPosition → {0.95, -0.4}, LegendSize → {0.5, 0.9},
 PlotLabel → StyleForm["e Versus x", FontWeight → Bold, FontSize → 12],
 Frame → True, FrameLabel → {"x", "e"}, RotateLabel → False, SymbolShape → None]
```

xt0, xt20, ...etc are the initial variables used to plot the position of each cable along the x-axis. TwoDXSection0 forms a table of the x and z coordinates along the edge, TwoDXSection25 forms a table of the x and z coordinates at the quarter of the unit cell, TwoDXSection50 forms a table of the x and z coordinates along the centerline of the unit cell. The values of all three of these tables are plotted on a multiple list plot. Specifications for the legend are made using the functions PlotLegend, LegendPosition, and LegendSize, PlotJoined→True plots lines for all the data, PlotLabel→ StyleForm specifies the title of the plot, that the title should be bold, and the size of the font. Frame→True plots a frame around the whole plot. FrameLabel→{"x","w"} specifies the labels that should be printed on each axis of the plot. x is the bottom axis and w is the vertical axis. RotateLabel→False prints the vertical axis label horizontal. SymbolShape→None eliminates all symbols that would be plotted for each point. Fulle is a table of the strain of the cables parallel to the x-axis. Tension is a table of the tension values of the cables parallel to the x-axis. TPlot0...etc and ePlot0...etc. serve the same purpose as the TwoDXSection tables except for the tension andsStrain values. Different ranges of values were used when creating these tables because the organization of the Fullxyz table was different than the FullCon table.Note: The graph is zig zagged because one cable is 45 degrees in one direction and the next cable is 45 degrees in the opposite direction so the cable in the difection of the load will have the larger strain and tension

```
a31 = 1; b31 = n + 1; xt45 = -h45/4; xe45 = xt45;
DiagXSection = DeleteCases[Table[If[Fullxyz[[i, 1]] == Fullxyz[[i, 2]],
    {Fullxyz[[i, 4]], Fullxyz[[i, 5]]}, {}], {i, fn}], {}];
ListPlot[DiagXSection, PlotJoined → True, PlotLabel →
  StyleForm["w Versus x Along Diagonal", FontWeight → Bold, FontSize → 12],
 Frame → True, FrameLabel → {"x", "w"}, RotateLabel → False]
DiagTPlot = Table[xt45 += h/Sqrt[2]; {xt45, FullT[[4n*(i - 1) + i]]}, {i, 4n}];
ListPlot[DiagTPlot, PlotJoined → True,
 PlotLabel → StyleForm["T Versus x Along Diagonal", FontWeight → Bold, FontSize → 12],
 Frame → True, FrameLabel → {"x", "T"}, RotateLabel → False]
DiagePlot = Table[xe45 += h/Sqrt[2]; {xe45, Fulle[[4n*(i - 1) + i]]}, {i, 4n}];
ListPlot[DiagePlot, PlotJoined → True,
 PlotLabel → StyleForm["e Versus x Along Diagonal", FontWeight → Bold, FontSize → 12],
 Frame → True, FrameLabel → {"x", "e"}, RotateLabel → False]
```

DiagXSection is a list of the vertical displacements of the nodes along the 45-degree diagonal. DiagTPlot and DiagePlot are the tension forces and strains of the cables along the 45-degree diagonal. These values are plotted.

```
w1 = 0;
Fullw = Table[If[EvenQ[j] == True && i == n,
    Fullxyz[[w1, 5]], w1 += 1; Fullxyz[[w1, 5]]], {j, 4n}, {i, 0, 2n}];
Total[Flatten[Table[Fullxyz[[i, 5]], {i, fn}]]]
ListPlot3D[Fullw, Mesh → False,
 MeshRange → {{0, 1}, {0, 1}}, AxesLabel → {"x", "y", "z    "}]
ListContourPlot[Fullw, MeshRange → {{0, 1}, {0, 1}},
 FrameLabel → {"x", "y"}, RotateLabel → False]
wMin = Min[Fullw]
wMax = Max[Fullw]
```

Fullw is the vertical displacements of the cable net. Both of these plots don't take into account the small x and y displacements that occur.

```
Fullex = Partition[Fulle/h, 2n];
ListPlot3D[Fullex, Mesh → False,
 MeshRange → {{0, 1}, {0, 1}}, AxesLabel → {"x", "y", "e    "}]
ListContourPlot[Fullex, MeshRange → {{0, 1}, {0, 1}},
 FrameLabel → {"x", "y"}, RotateLabel → False]
Total[Flatten[Fullw]]
Total[Flatten[Fullex]]
```

Fullex is a list of all the strains of the cables parallel to the x-axis

Full w is the vertical displacements of the grid.Both of these plots don't take into account the small x and y displacements that occur.It is assumed that there was no x and y displacements.

The 3-D strain plots did not work out because of the orientation of the cables, perhaps if the strains of two perpendicular cables that meet at one node were averaged or the resultant was found than a better strain plot would have been produced.

Appendix E

Anisotropic Geogrid Algorithm

```
Off[General::"spell1"];
Clear[XYZ, xyz, DOF, dxyz, S, Energy, e, tn, z, L, tc, d, n, CON, R, P, k, EA, ID];
FindMax = Function[{M, tn, L}, MaxX = 0; X = 0; MaxY = 0; Y = 0; MaxZ = 0; Z = 0;
   For[i = 1, i ≤ tn, i++, If[Abs[M[[i, 1]]] > Abs[MaxX], MaxX = M[[i, 1]]; X = i, MaxX; X];
    If[Abs[M[[i, 2]]] > Abs[MaxY], MaxY = M[[i, 2]]; Y = i, MaxY; Y];
    If[Abs[M[[i, 3]]] > Abs[MaxZ], MaxZ = M[[i, 3]]; Z = i, MaxZ; 1]];
   Print[TableForm[{{"Max X is ", MaxX*L, " at ", X},
      {"Max Y is ", MaxY*L, " at ", Y}, {"Max Z is ", MaxZ*L, " at ", Z}}]]];
$TextStyle = {FontFamily → "Times", FontSize → 10};
<< Graphics`MultipleListPlot`
<< Graphics`Legend`
```

The spelling error has been turned off, and some of the variables have been cleared in the kernal. Find Max is a function that finds the maximum displacements that occur along each axis in dimensional form. $TextStyle specifies the text to be used on all output, and the <<Graphics loads the mathematica data necessary to create multiple list plots and legends.

```
B = 60; L = 300; Qp = -14.6; Qs = -3.06; Kp = 29.2; Ks = 0.16; EA = 21900;
nx = 25; ny = 50; Hx = L/(2 nx); Hy = L/(2 ny); b = B/L; hx = Hx/L; hy = Hy/L; qp = Qp*L^2/EA;
pp = qp*hx*hy; qs = Qs*L^2/EA; ps = qs*hx*hy; kp = Kp*L^2/EA; tp = kp*hx*hy;
ks = Ks*L^2/EA; ts = ks*hx*hy; EApWx = EA/Hy; EApWy = EA/Hx;
hx
hy
b
qp
pp
qs
ps
kp
tp
ks
ts
EApWx
EApWy
```

$\frac{1}{50}$

$\frac{1}{100}$

$\frac{1}{5}$

```
-60.

-0.012

-12.5753

-0.00251507

36000.

7.2
```

```
197.26

0.0394521

7300

3650
```

B is half of the width of the pile cap (cm), L is the cell unit length (cm), Qp is the pressure acting on the geogrid above the pile (N/cm^2), Qs is the pressure acting on the geogrid above the soil (N/cm^2), Kp is the soil stiffness above the pile (N/cm^2/cm), Ks is the soil stiffness above the soil (N/cm^2/cm), EA (N) is the Elastic Modulus multiplied by the geogrid rib cross-sectional area , nx is the number of geogrid ribs (cables) along the x-axis in half of the cell unit length, ny is the number of geogrid ribs (cables) along the y-axis in half of the cell unit length. Hx is the geogrid rib length along the x-axis (cable length) (cm), Hy is the geogrid rib length along the y-axis (cable length) (cm), qp, qs, kp, ks, b and h are dimensionless. pp is the dimensionless point load acting above the pile , ps is the dimensionless point load acting above the soft soil, tp is the dimensionless stiffness of the soil spring on top of the pile , and ts is the dimensionless stiffness of the soft soil spring.

```
xyz = Partition[Flatten[Table[{i*hx, j*hy, 0}, {j, 0, ny}, {i, 0, nx}]], 3];
```

xyz is the initial position matrix of the cable net nodes that make up one-fourth of an unit cell. The coordinates are ordered in the same way as they are listed in the title.

```
a1 = 1; a2 = 2; b1 = 1; b2 = nx + 2;
Con = Partition[
   Flatten[Join[Table[{If[j == a2, a2++; a1 += 1, a1], a1 += 1}, {j, 1, ny + 1}, {i, 1, nx}],
     Table[{b1++, b2++}, {j, 1, nx + 1}, {i, 1, ny}]]], 2];
Transpose[xyz] // MatrixForm
Transpose[Con] // MatrixForm
```

Con is the connectivity matrix. It is a list of the nodes that are connected by ribs (cables). It is a joined table, the first list is of the nodes that make up the cables parallel to the x-axis and the second list is of the nodes that make up the cables parallel to the y-axis.

```
tn = Dimensions[xyz][[1]]
tc = Dimensions[Con][[1]]
```

tn is the number of nodes in a cable net for one-fourth of the unit cell. tc is the number of cables in one-fourth of the unit cell.

```
tcx = nx*(ny + 1); tcy = ny*(nx + 1); c1 = 1; c2 = nx + 1;
CoC = Flatten[Join[Table[If[i ≤ nx || i > nx^2, 1/2, 1], {i, tcx}],
    Table[If[i == c1, c1 += nx + 1; 1/2, If[i == c2, c2 += nx + 1; 1/2, 1]], {i, tcy}]]];
tc
tcx
tcy
```

CoC is the cable coefficient vector. This vector considers the effect of perimeter cables, and reduces the energy a perimeter cable absorbs by one half. Perimeter cables are accounted for twice when the reduced unit cell is transformed to the full unit cell. The list must be joined so that the values correspond to the order of the Con matrix.

```
a = 0;
uvw = Table[{If[xyz[[i, 1]] == 1/2 || xyz[[i, 1]] == 0, 0, x[a += 1]],
    If[xyz[[i, 2]] == 0 || xyz[[i, 2]] == 1/2, 0, x[a += 1]],
    x[a += 1]}, {i, tn}];
```

uvw is the displacement matrix. All degrees of freedom that will be restrained are given a value of zero, the rest are given a variable x[a] where a is an increasing integer.

```
nDOF = a
DOF = Table[x[i], {i, nDOF}];
```

nDOF is the number of degrees of freedom in one-fourth of the unit cell. DOF is a list of all the variables, x[a], that represent degrees of freedom. This list will be used in the NMinimize function.

Rectangle Piles

```
Co = Table[If[xyz[[i, 1]] == 0 || xyz[[i, 1]] == 1/2, If[xyz[[i, 2]] == 0 || xyz[[i, 2]] == 1/2,
      1/4, 1/2], If[xyz[[i, 2]] == 0 || xyz[[i, 2]] == 1/2, 1/2, 1]], {i, tn}];
p = Table[If[xyz[[i, 1]] ≤ b && xyz[[i, 2]] ≤ b, pp*Co[[i]], ps*Co[[i]]], {i, tn}];
t = Table[If[xyz[[i, 1]] ≤ b && xyz[[i, 2]] ≤ b, tp*Co[[i]], ts*Co[[i]]], {i, tn}];
```

Co is a list of load coefficients that account for the tributary areas that result from the geometry of one-fourth of the unit cell. The nodes on the perimeter of the cable net that makes up one-fourth of the unit cell will be one half of the full load, the four corners (on the full grid they are the center edge nodes) will be one-fourth of the full load. p and k are the list of point loads and soil stiffness that multiplies the load coefficients by the pile or soil values depending on whether the node is over a pile or not.

```
e = Table[If[i ≤ tcx, c1 = Con[[i, 1]]; c2 = Con[[i, 2]];
    √((hx + uvw[[c1, 1]] - uvw[[c2, 1]])^2 + (uvw[[c1, 2]] - uvw[[c2, 2]])^2 +
      (uvw[[c1, 3]] - uvw[[c2, 3]])^2) - hx, c1 = Con[[i, 1]]; c2 = Con[[i, 2]];
    √((uvw[[c1, 1]] - uvw[[c2, 1]])^2 + (hy + uvw[[c1, 2]] - uvw[[c2, 2]])^2 +
      (uvw[[c1, 3]] - uvw[[c2, 3]])^2) - hy], {i, 1, tc}];
```

e is the elongation list. It uses the Con matrix and the displacement matrix to find the total elongation of a cable due to displacement.

$$\text{Energy} = \sum_{i,1}^{tcx} \frac{CoC[[i]]*e[[i]]^2}{2*hx} + \sum_{i,tcx+1}^{tc} \frac{CoC[[i]]*e[[i]]^2}{2*hy} + \sum_{j,1}^{tt} \frac{1}{2}*t[[j]]*uvw[[j,3]]^2 - \sum_{k,1}^{tt} p[[k]]*uvw[[k,3]];$$

Energy is the sum of the total energy for the system of cables parallel to the x-axis, cables parallel to y-axis, soil stiffnesses and point loads.

```
S = Timing[NMinimize[Energy, DOF]];
S[[1]]
```

S is the solution of the NMinimize function for the Energy equation and the list of degrees of freedom (x[a]). The timing function is used to time how long it took the function to find the solution.

```
uvw /. S[[2, 2]] >> Desktop/SchoolStuff/Research/Mathematica/ParametricStudyData/uvwAn
e /. S[[2, 2]] >> Desktop/SchoolStuff/Research/Mathematica/ParametricStudyData/eAn
```

The solution for each DOF is then placed in the displacement and elongation matrices and stored in seperate files on the hard drive.

```
Nuvw = << Desktop/SchoolStuff/Research/Mathematica/ParametricStudyData/uvwAn;
Ne = << Desktop/SchoolStuff/Research/Mathematica/ParametricStudyData/eAn;
Nxyz = xyz + Nuvw;
```

The solution is retrieved from the uvw and e files and called Nuvw and Ne respectively. This was done so that it was not necessary to run the NMinimize function everytime a notebook is opened. Nxyz is the new position matrix of the displaced grid. It is a simple addition of the initial position matrix and the displacement matrix.

```
ϵ = Table[If[i ≤ tcx, №[[i]]/hx, №[[i]]/hy], {i, tc}];
T = Table[EA * ϵ[[i]], {i, tc}];
Max[ϵ]
Max[T]
Min[Nuvw]
Min[Nuvw] * L
FindMax[Nuvw, tn, L]
Ds = Nuvw[[1, 3]] - Min[Nuvw]
Ds * L
pos = Table[If[xyz[[i, 1]] ≤ b, 0, 1], {i, tn}];
LDR = Total[Table[t[[i]] * Nuvw[[i, 3]] * pos[[i]], {i, tn}]] /
  Total[Table[p[[j]] * pos[[j]], {j, tn}]]
```

ϵ is a list of the strain values for the cables. T is a list of tension values for the cables. The maximum strain, tension, and vertical displacement are found. The FindMax function is used to find the maximum displacements in dimensional form. The differential settlement is found in non-dimensional and dimensional form by subtracting the smallest vertical displacement from the largest vertical displacement. pos is a list of all the nodes in the reduced unit cell, a 0 means the node is initially above the pile and 1 means the node is initially above the soft soil. LDR is the ratio of the load the soft soil springs support to the load on the soft soil. It is found by finding the total of the soil stiffness times the vertical displacement of each node supported by the soft soil and dividing it by the total of the point loads acting on each node supported by the soft soil.

```
cnx = nx + 1; cny = ny + 1; cx = 2nx (2ny + 1);
cy = 2ny (2nx + 1); fc = cx + cy; fn = (2nx + 1) * (2ny + 1);
Fullxyz = Table[If[ i ≤ cnx && j ≤ cny, n1 = i + (j - 1) cnx;
    {Nxyz[[n1, 1]], Nxyz[[n1, 2]], Nxyz[[n1, 3]]}, If[i > cnx && j ≤ cny,
     n2 = cnx - i + (j) cnx; {1 - Nxyz[[n2, 1]], Nxyz[[n2, 2]], Nxyz[[n2, 3]]},
     If[i ≤ cnx && j > cny, n3 = i + (2cny - j - 1) cnx; {Nxyz[[n3, 1]], 1 - Nxyz[[n3, 2]],
       Nxyz[[n3, 3]]}, If[i > cnx && j > cny, n4 = cnx - i + (2cny - j) cnx; {1 - Nxyz[[n4, 1]],
         1 - Nxyz[[n4, 2]], Nxyz[[n4, 3]]}, {0, 0, 0}]]]], {j, 2ny + 1}, {i, 2nx + 1}];
```

cn is the number of nodes on the edge of the cable net that makes up one-fourth of the unit cell. cx is the total number of cables in the full grid that are parallel to the x-axis. cy is the total number of cables in the full grid that are parallel to the y-axis. fn is the total number of nodes in the full grid. Fullxyz takes the final position matrix of the one-fourth of the unit cell and creates a matrix of the final positions for a full unit cell. Each of the four sections of the full unit cell has a seperate if statement. i is the column number of the grid and j is the row number. So that i will start at one and increase until i equals the number of columns in a row, and this will occur for every row, j, starting at one. i and j is essentially the position of the node in the full unit cell and the criteria of an if statement will be met depending on which of the four sections the node is in. n1, n2, n3... are the node numbers of the cable net that makes up one-fourth of the unit cell that correspond to the nodal position on the full unit cell.

```
a21 = 1; b21 = 1; a22 = 1;
FullCon = Join[Table[{If[i == (2nx) a22++, a22 += 1; a21 += 1, a21], a21 += 1}, {i, 1, cx}],
   Table[{b21++, b21 + 2nx}, {i, 1, cy}]];
```

FullCon is the the full connectivity matrix. The matrix is made up of the two nodes that make up a cable. It is used to draw the cables in the Graphics 3D Plot. The matrix is joined because the first table makes up the cables that are parallel to the x-axis and the second table makes up the cables parallel to the y-axis.

```
Fullxyz = Partition[Flatten[Fullxyz], 3];
cablegrid = Table[Line[Table[Fullxyz[[FullCon[[i, j]]]], {j, 2}]], {i, fc}];
Show[Graphics3D[cablegrid, Axes → True, AxesLabel → {"x", "y", "z"},
   BoxRatios → {1, 1, 0.5}, ViewPoint → {1.3, -2.4, 1}, Boxed → False]];
```

Cablegrid is a list of all the lines (cables) in the grid so the Show[Graphics3D function draws the lines that represent each cable. Axes-True draws the axes and includes values. BoxRatios-is the ratio for each dimension of the box. ViewPoint {} is the angle to show the box. Boxed-False tells the kernal not to draw the box. AxesLabel is the labels to print on each axes.

```
Fullex = Flatten[Table[If[i ≤ nx && j ≤ ny + 1, c1 = i + (j - 1) nx;
     №[[c1]], If[i > nx  && j ≤ ny + 1, c2 = 2nx - i + 1 + (j - 1) nx; №[[c2]],
      If[i ≤ nx && j > ny + 1, c3 = i + (2ny - j + 1) nx; №[[c3]], If[i > nx && j > ny + 1,
        c4 = 2nx - i + 1 + (2ny - j + 1) nx; №[[c4]], 0]]]], {j, 2ny + 1}, {i, 2nx}]];
Fulley = Table[If[i ≤ nx + 1 && j ≤ ny, c1 = tcx + i + (j - 1) (nx + 1); №[[c1]],
    If[i > nx + 1  && j ≤ ny, c2 = tcx + 2nx - i + 2 + (j - 1) (nx + 1); №[[c2]],
     If[i ≤ nx + 1 && j > ny, c3 = tcx + i + (2ny - j) (nx + 1); №[[c3]], If[i > nx + 1 && j > ny,
       c4 = tcx + 2nx - i + 2 + (2ny - j) (nx + 1); №[[c4]], 0]]]], {j, 2ny}, {i, 2nx + 1}];
```

Fullex is a list of the elongation of each cable parallel to the x-axis in the unit cell. Fulley is a list of the elongation of each cable parallel to the y-axis in the unit cell. The order of the list corresponds to the FullCon list. Each if statement represents a section of one-fourth of the unit cell like the Fullxyz matrix.

```
xt0 = -hx/2; xt20 = xt0; xt50 = xt0; xe0 = xt0; xe24 = xt0; xe50 = xt0;
yt0 = -hy/2; yt20 = yt0; yt50 = yt0; ye0 = yt0; ye24 = yt0;
ye50 = yt0; y0 = 1; y24 = 0.48 ny + 1; y50 = ny + 1;
TwoDXSectionx0 = Table[{Fullxyz[[i, 1]], Fullxyz[[i, 3]]}, {i, 1, 2 nx}];
TwoDXSectionx24 = Table[{Fullxyz[[i, 1]], Fullxyz[[i, 3]]},
   {i, nx * ny + 0.48 ny + 1, 0.98 nx * ny + 0.48 ny + 2nx + 1}];
TwoDXSectionx50 = Table[{Fullxyz[[i, 1]], Fullxyz[[i, 3]]},
   {i, 2 * nx * ny + ny + 1, 2 * nx * ny + ny + 2 nx + 1}];
MultipleListPlot[TwoDXSectionx0, TwoDXSectionx24, TwoDXSectionx50,
 PlotJoined → True, PlotLegend → {"y = 0", "y = 0.24", "y = 0.5"},
 LegendPosition → {0.95, -0.4}, LegendSize → {0.5, 0.9},
 PlotLabel → StyleForm["w Versus x", FontWeight → Bold, FontSize → 12],
 Frame → True, FrameLabel → {"x", "w"}, RotateLabel → False, SymbolShape → None]
Fullex = Table[Fullex[[i]]/hx, {i, cx}];
FullTx = Table[EA * Fullex[[i]], {i, cx}];
TPlotx0 = Table[xt0 += hx; {xt0, FullTx[[i]]}, {i, 2nx}];
TPlotx20 = Table[xt20 += hx; {xt20, FullTx[[i]]}, {i, 0.8 * nx * ny + 1, 0.8 * nx * ny + 2nx}];
TPlotx50 = Table[xt50 += hx; {xt50, FullTx[[i]]}, {i, 2 * nx * ny + 1, 2 * nx * ny + 2 * nx}];
MultipleListPlot[TPlotx0, TPlotx20, TPlotx50,
 PlotJoined → True, PlotLegend → {"y = 0", "y = 0.20", "y = 0.5"},
 LegendPosition → {0.95, -0.4}, LegendSize → {0.5, 0.9},
 PlotLabel → StyleForm["Tension Parallel to x-Axis", FontWeight → Bold, FontSize → 12],
 Frame → True, FrameLabel → {"x", "T"}, RotateLabel → False, SymbolShape → None]
ePlotx0 = Table[xe0 += hx; {xe0, Fullex[[i]]}, {i, 2nx}];
ePlotx24 = Table[xe24 += hx; {xe24, Fullex[[i]]}, {i, 0.96 nx * ny + 1, 0.96 nx * ny + 2nx}];
ePlotx50 = Table[xe50 += hx; {xe50, Fullex[[i]]}, {i, 2 * nx * ny + 1, 2 * nx * ny + 2 * nx}];
MultipleListPlot[ePlotx0, ePlotx24, ePlotx50,
 PlotJoined → True, PlotLegend → {"y = 0", "y = 0.24", "y = 0.5"},
 LegendPosition → {0.95, -0.4}, LegendSize → {0.5, 0.9},
 PlotLabel → StyleForm["ϵx Versus x", FontWeight → Bold, FontSize → 12],
 Frame → True, FrameLabel → {"x", "ϵ"}, RotateLabel → False, SymbolShape → None]
```

xt0, xt20, ...etc are the initial variables used to plot the position of each cable along the x-axis. TwoDXSectionx0 forms a table of the x and z coordinates along the edge, TwoDXSectionx25 forms a table of the x and z coordinates at the quarter of the unit cell, TwoDXSectionx50 forms a table of the x and z coordinates along the centerline of the unit cell. The values of all three of these tables are plotted on a multiple list plot. Specifications for the legend are made using the functions PlotLegend, LegendPosition, and LegendSize, PlotJoined→True plots lines for all the data, PlotLabel→ StyleForm specifies the title of the plot, that the title should be bold, and the size of the font. Frame→True plots a frame around the whole plot. FrameLabel→{"x","w"} specifies the labels that should be printed on each axis of the plot. x is the bottom axis and w is the vertical axis. RotateLabel→False prints the vertical axis label horizontal. SymbolShape→None eliminates all symbols that would be plotted for each point. Fullex is a table of the strain of the cables parallel to the x-axis. Tension is a table of the tension values of the cables parallel to the x-axis. TPlotx0...etc and ePlotx0...etc. serve the same purpose as the TwoDXSectionx tables except for the tension and Strain values. Different ranges of values were used when creating these tables because the organization of the Fullxyz table was different than the FullCon table.

```
TwoDXSectiony0 = Table[If[i ≠ 1, y0 = y0 + 2 πx + 1; {Fullxyz[[y0, 2]], Fullxyz[[y0, 3]]},
    {Fullxyz[[y0, 2]], Fullxyz[[y0, 3]]}], {i, 2 πy + 1}];
TwoDXSectiony24 = Table[If[i ≠ 1, y24 = y24 + 2 πx + 1; {Fullxyz[[y24, 2]],
     Fullxyz[[y24, 3]]}, {Fullxyz[[y24, 2]], Fullxyz[[y24, 3]]}], {i, 2 πy + 1}];
TwoDXSectiony50 = Table[If[i ≠ 1, y50 = y50 + 2 πx + 1; {Fullxyz[[y50, 2]],
     Fullxyz[[y50, 3]]}, {Fullxyz[[y50, 2]], Fullxyz[[y50, 3]]}], {i, 2 πy + 1}];
MultipleListPlot[TwoDXSectiony0, TwoDXSectiony24, TwoDXSectiony50,
 PlotJoined → True, PlotLegend → {"x = 0", "x = 0.24", "x = 0.5"},
 LegendPosition → {0.95, -0.4}, LegendSize → {0.5, 0.9},
 PlotLabel → StyleForm["w Versus y", FontWeight → Bold, FontSize → 12],
 Frame → True, FrameLabel → {"y", "w"}, RotateLabel → False, SymbolShape → None]
Fulley = Fulley / hy;
FullTy = EA * Fulley;
TPloty0 = Table[yt0 += hy; {yt0, FullTy[[i, 1]]}, {i, 2 πy}];
TPloty20 = Table[yt20 += hy; {yt20, FullTy[[i, 0.4 πx + 1]]}, {i, 2 πy}];
TPloty50 = Table[yt50 += hy; {yt50, FullTy[[i, πx + 1]]}, {i, 2 πy}];
MultipleListPlot[TPloty0, TPloty20, TPloty50,
 PlotJoined → True, PlotLegend → {"y = 0", "y = 0.2", "y = 0.5"},
 LegendPosition → {0.95, -0.4}, LegendSize → {0.5, 0.9},
 PlotLabel → StyleForm["Tension Parallel to y-Axis", FontWeight → Bold, FontSize → 12],
 Frame → True, FrameLabel → {"x", "T"}, RotateLabel → False, SymbolShape → None]
ePloty0 = Table[ye0 += hy; {ye0, Fulley[[i, 1]]}, {i, 2 πy}];
ePloty24 = Table[ye24 += hy; {ye24, Fulley[[i, 0.48 πx + 1]]}, {i, 2 πy}];
ePloty50 = Table[ye50 += hy; {ye50, Fulley[[i, πx + 1]]}, {i, 2 πy}];
MultipleListPlot[ePloty0, ePloty24, ePloty50,
 PlotJoined → True, PlotLegend → {"y = 0", "y = 0.24", "y = 0.5"},
 LegendPosition → {0.95, -0.4}, LegendSize → {0.5, 0.9},
 PlotLabel → StyleForm["ε_y Versus x", FontWeight → Bold, FontSize → 12],
 Frame → True, FrameLabel → {"x", "ε"}, RotateLabel → False, SymbolShape → None]
```

yt0, yt20, ...etc are the initial variables used to plot the position of each cable along the y-axis. TwoDXSectiony0 forms a table of the y and z coordinates along the edge, TwoDXSectiony25 forms a table of the y and z coordinates at the quarter of the unit cell, TwoDXSectiony50 forms a table of the y and z coordinates along the centerline of the unit cell. The values of all three of these tables are plotted on a multiple list plot. Specifications for the legend are made using the functions PlotLegend, LegendPosition, and LegendSize, PlotJoined→True plots lines for all the data, PlotLabel→ StyleForm specifies the title of the plot, that the title should be bold, and the size of the font. Frame→True plots a frame around the whole plot. FrameLabel→{"y","w"} specifies the labels that should be printed on each axis of the plot. y is the bottom axis and w is the vertical axis. RotateLabel→False prints the vertical axis label horizontal. SymbolShape→None eliminates all symbols that would be plotted for each point. Fulley is a table of the strain of the cables parallel to the y-axis. Tension is a table of the tension values of the cables parallel to the y-axis. TPloty0...etc and ePloty0...etc serve the same purpose as the TwoDXSectiony tables except for the tension and strain values. Different ranges of values were used when creating these tables because the organization of the Fullxyz table was different than the FullCon table. Also the tension and strain values of the cables parallel to the y-axis were plotted along the x-axis because that was the way it was organized in the Con matrix.

```
Fullw = Partition[Table[Fullxyz[[i, 3]], {i, fn}], 2 πx + 1];
wMin = Min[Fullw]
wMax = Max[Fullw]
ListPlot3D[Fullw, Mesh → False,
 MeshRange → {{0, 1}, {0, 1}}, AxesLabel → {"x", "y", "w    "}]
ListContourPlot[Fullw, MeshRange → {{0, 1}, {0, 1}},
 FrameLabel → {"x", "y"}, RotateLabel → False]
```

Fullw is the vertical displacements of the cable net. Both of these plots don't take into account the small x and y displacements that occur.

```
ListPlot3D[Partition[Fullex, 2πx], ViewPoint→{1.3, -2.4, 2},
 Mesh→False, MeshRange→{{0, 1}, {0, 1}}, AxesLabel→{"x", "y", "ε    "}]
ListPlot3D[Fulley, ViewPoint→{1.3, -2.4, 2}, Mesh→False,
 MeshRange→{{0, 1}, {0, 1}}, AxesLabel→{"x", "y", "ε    "}]
ListContourPlot[Partition[Fullex, 2πx], MeshRange→{{0, 1}, {0, 1}},
 FrameLabel→{"x", "y"}, RotateLabel→False]
ListContourPlot[Fulley, MeshRange→{{0, 1}, {0, 1}},
 FrameLabel→{"x", "y"}, RotateLabel→False]
Total[Flatten[Fullw]]
Total[Fullex]
Total[Flatten[Fulley]]
```

Fullex is a list of all the strains of the cables parallel to the x-axis. Fulley is a list of all the strains of the cables parallel t the y-axis. 3-D plots of the strains were created for the cables parallel to the x-axis and y-axis.

Appendix F

Two Layers of Geogrid Algorithm

```
Off[General::"spell1"];
FindMax = Function[{M, tn, L}, MaxX = 0; X = 0; MaxY = 0; Y = 0; MaxZ = 0; Z = 0;
   For[i = 1, i ≤ tn, i++, If[Abs[M[[i, 1]]] > Abs[MaxX], MaxX = M[[i, 1]]; X = i, MaxX; X];
    If[Abs[M[[i, 2]]] > Abs[MaxY], MaxY = M[[i, 2]]; Y = i, MaxY; Y];
    If[Abs[M[[i, 3]]] > Abs[MaxZ], MaxZ = M[[i, 3]]; Z = i, MaxZ; 1]];
   Print[TableForm[{{"Max X is ", MaxX*L, " at ", X},
      {"Max Y is ", MaxY*L, " at ", Y}, {"Max Z is ", MaxZ*L, " at ", Z}}]]];
LimitCheck = Function[{M, n, xy1, xy2, z1, z2}, chx = 0; chy = 0; chz = 0;
   For[i = 1, i ≤ n, i++, {If[M[[i, 1]] ≤ xy1 || M[[i, 1]] ≥ xy2, Print["x=", i], 0],
     If[M[[i, 2]] ≤ xy1 || M[[i, 2]] ≥ xy2, Print["y=", i], 0],
     If[M[[i, 3]] ≤ z1 || M[[i, 3]] ≥ z2, Print["z=", i], 0]}]];
Add = Function[x, a = 0; For[k = 1, k < x, k++, a = a + k]; a];
$TextStyle = {FontFamily → "Times", FontSize → 10};
<< Graphics`MultipleListPlot`
<< Graphics`Legend`
```

The spelling error has been turned off, and some of the variables have been cleared in the kernal. Find Max is a function that finds the maximum displacements that occur along each axis in dimensional form. LimitCheck is a function that checks to make sure that the displacements are whithin the limits set in the DOF matrix. Add is a function that is used to convert the nodal positions of the one-eighth portion of the unit cell to the full unit cell. $TextStyle specifies the text to be used on all output, and the <<Graphics loads the mathematica data neccessary to create multiple list plots and legends.

```
B = 60; L = 300; Qp = -14.6; Qs = -3.06; Kp = 29.2; Ks = 0.16; EApW = 7300; Dp = 15;
n = 50; d = Dp / L; H = L/(2n); b = B/L; h = H/L; EA = EApW*H; qp = Qp*L^2/EA; qs = Qs*L^2/EA;
pp = qp*h^2; ps = qs*h^2; kp = Kp*L^2/EA; ks = Ks*L^2/EA; tp = kp*h^2; ts = ks*h^2;
h
b
d
EA
qp
qs
kp
ks
pp
ps
tp
ts

 1
100

1
5

 1
20

21900

-60.

-12.5753

36000.

197.26
```

```
-0.006

-0.00125753

3.6

0.019726
```

B is half of the width of the pile cap (cm), L is the cell unit length (cm), Qp is the pressure acting on the geogrid above the pile (N/cm^2), Qs is the pressure acting on the geogrid above the soil (N/cm^2), Kp is the soil stiffness above the pile (N/cm^2/cm), Ks is the soil stiffness above the soil (N/cm^2/cm), EApW is the Elastic Modulus multiplied by the geogrid rib cross-sectional area divided by the number of ribs in a cm (N/cm), n is the number of geogrid ribs (cables) in half of the cell unit length. H is geogrid rib length (cable length) (cm), EA (N), qp, qs, kp, ks, b and h are dimensionless. pp is the dimensionless point load acting above the pile , ps is the dimensionless point load acting above the soft soil, tp is the dimensionless stiffness of the soil spring on top of the pile , and ts is the dimensionless stiffness of the soft soil spring.

```
xyz = Partition[Flatten[Table[{j*h, i*h}, {i, 0, n}, {j, i, n}]], 2];
```

xyz is the initial position matrix of the cable net nodes that make up one-eighth of an unit cell. The coordinates are ordered in the same way as they are listed in the title.

```
a1 = 1; a2 = 2; b1 = 0; b2 = 1; b3 = n + 2;
Con = Partition[
   Flatten[Join[Table[{If[i == a2, a2++; a1 += 1, a1], a1 += 1}, {i, 1, n}, {j, i, n}],
     Table[{If[i == b2, b2++; b1 += 2, b1 += 1], b3++}, {i, 1, n}, {j, i, n}]]], 2];
Transpose[xyz] // MatrixForm
Transpose[Con] // MatrixForm
```

Con is the connectivity matrix. It is a list of the nodes that are connected by ribs (cables). It is an joined table, the first list is of the nodes that make up the cables parallel to the x-axis and the second list is of the nodes that make up the cables parallel to the y-axis.

```
tn = Dimensions[xyz][[1]]
tc = Dimensions[Con][[1]]
```

tn is the number of nodes in an eighth of the unit cell.tc is the number of cables in an eighth of the unit cell.

```
c1 = n; c2 = n;
CoC = Join[Table[If[i ≤ n, 1/2, 1], {i, tc/2}],
   Table[If[i == c1, c2 -= 1; c1 += c2; 1/2, 1], {i, tc/2}]];
```

CoC is the cable coefficient vector. This vector considers the effect of perimeter cables, and reduces the energy a perimeter cable absorbs by one half. Perimeter cables are accounted for twice when the reduced unit cell is transformed to the full unit cell.

```
a = 0;
ID1 = Table[{If[xyz[[i, 1]] == 1/2 || xyz[[i, 1]] == 0, 0, a += 1],
    If[xyz[[i, 2]] == 0 || xyz[[i, 2]] == 1/2, 0, If[xyz[[i, 1]] == xyz[[i, 2]], a, a += 1]],
    a += 1}, {i, tn}];
a
ID2 = Table[{If[xyz[[i, 1]] == 1/2 || xyz[[i, 1]] == 0, 0, a += 1],
    If[xyz[[i, 2]] == 0 || xyz[[i, 2]] == 1/2, 0, If[xyz[[i, 1]] == xyz[[i, 2]], a, a += 1]],
    a += 1}, {i, tn}];
uvw1 = Table[If[ID1[[i, j]] == 0, 0, x[ID1[[i, j]]]], {i, tn}, {j, 3}];
uvw2 = Table[If[ID2[[i, j]] == 0, 0, x[ID2[[i, j]]]], {i, tn}, {j, 3}];
```

ID1 and ID2 are vectors that determine the nodal displacement numbers (DOF #). All the degrees of freedom that are restrained are given a value of zero, and the rest are assigned a number. The diagonal nodes that make up the one-eighth reduced unit cell are restrained so that the x and y displacements are the same. uvw is the nodal displacement matrix. If the degree of freedom is free then a variable is assigned to it, x[ID]. There are ID1 and ID2 and uvw1 and uvw2, one for each cable net

```
nDOF = a
DOF1 = Table[If[ID1[[i, 1]] == 0 && ID1[[i, 2]] == 0, {{uvw1[[i, 3]], -d / 10, -2 d, d / 2}},
    If[ID1[[i, 1]] == 0, {{uvw1[[i, 2]], 0, -0.5, 0.5}, {uvw1[[i, 3]], -d / 10, -2 d, d / 2}},
     If[ID1[[i, 2]] == 0, {{uvw1[[i, 1]], 0, -0.1, 0.1}, {uvw1[[i, 3]], -d / 10, -2 d, d / 2}},
      If[ID1[[i, 1]] == ID1[[i, 2]], {{uvw1[[i, 1]], 0, -0.5, 0.5},
        {uvw1[[i, 3]], -d / 10, -2 d, d / 2}}, {{uvw1[[i, 1]], 0, -0.5, 0.5},
        {uvw1[[i, 2]], 0, -0.5, 0.5}, {uvw1[[i, 3]], -d / 10, -2 d, d / 2}}]]], {i, tn}];
DOF2 = Table[If[ID2[[i, 1]] == 0 && ID2[[i, 2]] == 0, {{uvw2[[i, 3]], 0, -3 d, d / 10}},
    If[ID2[[i, 1]] == 0, {{uvw2[[i, 2]], 0, -0.5, 0.5}, {uvw2[[i, 3]], 0, -3 d, d / 2}},
     If[ID2[[i, 2]] == 0, {{uvw2[[i, 1]], 0, -0.5, 0.5}, {uvw2[[i, 3]], 0, -3 d, d / 2}},
      If[ID2[[i, 1]] == ID2[[i, 2]], {{uvw2[[i, 1]], 0, -0.5, 0.5},
        {uvw2[[i, 3]], 0, -3 d, d / 2}}, {{uvw2[[i, 1]], 0, -0.5, 0.5},
        {uvw2[[i, 2]], 0, -0.5, 0.5}, {uvw2[[i, 3]], 0, -3 d, d / 2}}]]], {i, tn}];
DOF = Partition[Flatten[Join[DOF1, DOF2]], 4]
Dimensions[DOF][[1]]
```

nDOF is the number of degrees of freedom in both of the cablenets one-eighth of the unit cell. DOF is a list of all the variables, x[a], that represent degrees of freedom, and the restrictions that are applied to it. This list will be used in the FindMinimum function. DOF1 are the degrees of freedom of the top gecgrid, and DOF2 are the degrees of freedom of the bottom gecgrid.

Rectangle Piles

```
Co = Table[If[xyz[[i, 1]] == xyz[[i, 2]] == 0 || xyz[[i, 1]] == xyz[[i, 2]] == 1/2,
    1/8, If[xyz[[i, 1]] == 1/2 && xyz[[i, 2]] == 0, 1/4,
    If[xyz[[i, 1]] == 1/2 || xyz[[i, 2]] == 0 || xyz[[i, 1]] == xyz[[i, 2]], 1/2, 1]]], {i, tn}]
p = Table[If[xyz[[i, 1]] ≤ b, pp * Co[[i]], ps * Co[[i]]], {i, tn}]
t = Table[If[xyz[[i, 1]] ≤ b, tp * Co[[i]], ts * Co[[i]]], {i, tn}]
tg = tp * Co
```

Co is a list of load coefficients that account for the tributary areas that result from the geometry of one-eighth of the unit cell. The nodes on the perimeter of the cable net that makes up one-eighth of the unit cell will be one-half of the full load. The two corners with 45-degree angles (the center node and corner node of the full unit cell) will be one-eighth of the full load, and the corner with a 90 degree angle (on the full grid it is the center edge node) will be one-fourth of the full load. p and t are the list of point loads and soil stiffness that multiplies the load coefficients by the pile or soil values depending on whether the node is over a pile or not.

```
e1 = Table[If[i ≤ tc / 2, c1 = Con[[i, 1]]; c2 = Con[[i, 2]];
    √((h + uvw1[[c2, 1]] - uvw1[[c1, 1]])^2 + (uvw1[[c2, 2]] - uvw1[[c1, 2]])^2 +
        (uvw1[[c2, 3]] - uvw1[[c1, 3]])^2) - h, c1 = Con[[i, 1]]; c2 = Con[[i, 2]];
    √((uvw1[[c2, 1]] - uvw1[[c1, 1]])^2 + (h + uvw1[[c2, 2]] - uvw1[[c1, 2]])^2 +
        (uvw1[[c2, 3]] - uvw1[[c1, 3]])^2) - h], {i, 1, tc}];
e2 = Table[If[i ≤ tc / 2, c1 = Con[[i, 1]]; c2 = Con[[i, 2]];
    √((h + uvw2[[c2, 1]] - uvw2[[c1, 1]])^2 + (uvw2[[c2, 2]] - uvw2[[c1, 2]])^2 +
        (uvw2[[c2, 3]] - uvw2[[c1, 3]])^2) - h, c1 = Con[[i, 1]]; c2 = Con[[i, 2]];
    √((uvw2[[c2, 1]] - uvw2[[c1, 1]])^2 + (h + uvw2[[c2, 2]] - uvw2[[c1, 2]])^2 +
        (uvw2[[c2, 3]] - uvw2[[c1, 3]])^2) - h], {i, 1, tc}];
```

e is the elongation list. It uses the Con matrix and the displacement matrix to find the total elongation of a cable due to displacement. e1 is the elongation equation of the top geogrid and e2 is the elongation equation of the bottom geogrid.

$$\text{Energy} = \frac{1}{2h} * \sum_{i,1}^{tc} \text{CoC}[[i]] * e1[[i]]^2 + \frac{1}{2h} * \sum_{j,1}^{tc} \text{CoC}[[j]] * e2[[j]]^2 - \sum_{k,1}^{tn} p[[k]] * uvw1[[k, 3]] + \frac{1}{2} * \sum_{m,1}^{tn} tg[[m]] * (uvw1[[m, 3]] - uvw2[[m, 3]])^2 + \frac{1}{2} * \sum_{n,1}^{tn} t[[n]] * (uvw2[[n, 3]])^2;$$

Energy is the sum of the total energy for the system of cables, soil stiffnesses, and point loads.

```
S = Timing[FindMinimum[Energy, DOF, MaxIterations→ 100000, AccuracyGoal → 10]];
S[[1]]
```

S is the solution of the FindMinimum function for the Energy equation and the list of degrees of freedom (x[a]). The timing function is used to time how long it took the function to find the solution.

```
uvw1 /. S[[2, 2]] >>
 Desktop/SchoolStuff/Research/Mathematica/ParametricStudyData/uvw2G1.3
e1 /. S[[2, 2]] >> Desktop/SchoolStuff/Research/Mathematica/ParametricStudyData/e2G1.3
Nuvw1 = << Desktop/SchoolStuff/Research/Mathematica/ParametricStudyData/uvw2G1.3;
Ne1 = << Desktop/SchoolStuff/Research/Mathematica/ParametricStudyData/e2G1.3;
Nxyz1 = Table[
   {xyz[[i, 1]] + Nuvw1[[i, 1]], xyz[[i, 2]] + Nuvw1[[i, 2]], Nuvw1[[i, 3]]}, {i, tn}];
uvw2 /. S[[2, 2]] >>
 Desktop/SchoolStuff/Research/Mathematica/ParametricStudyData/uvw2G2.3
e2 /. S[[2, 2]] >> Desktop/SchoolStuff/Research/Mathematica/ParametricStudyData/e2G2.3;
Nuvw2 = << Desktop/SchoolStuff/Research/Mathematica/ParametricStudyData/uvw2G2.3;
Ne2 = << Desktop/SchoolStuff/Research/Mathematica/ParametricStudyData/e2G2.3;
Nxyz2 = Table[
   {xyz[[i, 1]] + Nuvw2[[i, 1]], xyz[[i, 2]] + Nuvw2[[i, 2]], -d + Nuvw2[[i, 3]]}, {i, tn}];
```

The solution for each DOF is then placed in the displacement and elongation matrices and stored in seperate files on the hard drive. The solution is retrieved fom the uvw and e files and called Nuvw and Ne respectively. This was done so that it was not necessary to run the findMinimum function everytime a notebook is opened. Nxyz is the new position matrix of the displaced grid. It is calculated by adding the initial position matrix and the displacement matrix. Nxyz1 is the final nodal positions of the upper cablenet and Nxyz2 is the final nodal positions of the lower cablenet.

```
e1 = Table[Ne1[[i]] / h, {i, tc}];
T1 = Table[EA * e1[[i]], {i, tc}];
Max[e1]
Max[T1]
Min[Nuvw1]
Min[Nuvw1] * L
Max[Table[Nuvw1[[i, 3]], {i, tn}]]
Ds1 = Nuvw1[[1, 3]] - Min[Nuvw1]
Ds1 * L
e2 = Table[Ne2[[i]] / h, {i, tc}];
T2 = Table[EA * e2[[i]], {i, tc}];
Max[e2]
Max[T2]
Min[Nuvw2]
Min[Nuvw2] * L
Max[Table[Nuvw2[[i, 3]], {i, tn}]]
Ds2 = Nuvw2[[1, 3]] - Min[Nuvw2]
Ds2 * L
pos = Table[If[xyz[[i, 1]] ≤ b, 0, 1], {i, tn}];
LDR = Total[Table[t[[i]] * Nuvw2[[i, 3]] * pos[[i]], {i, tn}]] /
  Total[Table[p[[j]] * pos[[j]], {j, tn}]]
```

ε is a list of the strain values for the cables. T is a list of tension values for the cables. The maximum strain, tension, and vertical displacement are found. The FindMax function is used to find the maximum displacements in dimensional form. The differential settlement is found in non-dimensional and dimensional form by subtracting the smallest vertical displacement from the largest vertical displacement. pos is a list of all the nodes in the reduced unit cell, a 0 means the node is initially above the pile and 1 means the node is initially above the soft soil. LDR is the ratio of the load the soft soil springs support to the load on the soft soil. It is found by finding the total of the soil stiffness times the vertical displacement of each node supported by the soft soil and dividing it by the total of the point loads acting on each node supported by the soft soil. 1 is the upper geogrid and 2 is the bottom geogrid.

```
cn = n + 1; cx = 2 n (2 n + 1); cy = cx; fc = cx + cy; fn = (2 n + 1)^2
Fullxyz1 = Table[If[ j ≤ i && i ≤ cn && j ≤ cn,
    n1 = i + (j - 1) cn - Add[j]; {Nxyz1[[n1, 1]], Nxyz1[[n1, 2]], Nxyz1[[n1, 3]]},
    If[i > cn && 2cn - i ≥ j && j ≤ cn, n2 = cn - i + (j) cn - Add[j];
     {1 - Nxyz1[[n2, 1]], Nxyz1[[n2, 2]], Nxyz1[[n2, 3]]}, If[i ≤ cn && j > i && j ≤ cn,
      n3 = j + (i - 1) cn - Add[i]; {Nxyz1[[n3, 2]], Nxyz1[[n3, 1]], Nxyz1[[n3, 3]]},
      If[i > cn && j > 2cn - i && j ≤ cn, n4 = j + (2cn - i - 1) cn - Add[2cn - i];
       {1 - Nxyz1[[n4, 2]], Nxyz1[[n4, 1]], Nxyz1[[n4, 3]]},
       If[i ≤ cn && cn - i ≥ j - cn && j > cn, n5 = cn - j + (i) cn - Add[i];
        {Nxyz1[[n5, 2]], 1 - Nxyz1[[n5, 1]], Nxyz1[[n5, 3]]},
        If[i ≤ cn && cn - i < j - cn && j > cn, n6 = i + (2 cn - j - 1) cn - Add[2 cn - j];
         {Nxyz1[[n6, 1]], 1 - Nxyz1[[n6, 2]], Nxyz1[[n6, 3]]},
         If[i > cn && i ≤ j && j > cn, n7 = cn - i + (2 cn - j) cn - Add[2 cn - j];
          {1 - Nxyz1[[n7, 1]], 1 - Nxyz1[[n7, 2]], Nxyz1[[n7, 3]]}, If[i > cn && i > j &&
            j > cn, n8 = cn - j + (2cn - i) cn - Add[2cn - i]; {1 - Nxyz1[[n8, 2]], 1 -
             Nxyz1[[n8, 1]], Nxyz1[[n8, 3]]}, {0, 0, 0}]]]]]]]], {j, 2 n + 1}, {i, 2 n + 1}];
cn = n + 1; cx = 2 n (2 n + 1); cy = cx; fc = cx + cy; fn = (2 n + 1)^2
Fullxyz2 = Table[If[ j ≤ i && i ≤ cn && j ≤ cn,
    n1 = i + (j - 1) cn - Add[j]; {Nxyz2[[n1, 1]], Nxyz2[[n1, 2]], Nxyz2[[n1, 3]]},
    If[i > cn && 2cn - i ≥ j && j ≤ cn, n2 = cn - i + (j) cn - Add[j];
     {1 - Nxyz2[[n2, 1]], Nxyz2[[n2, 2]], Nxyz2[[n2, 3]]}, If[i ≤ cn && j > i && j ≤ cn,
      n3 = j + (i - 1) cn - Add[i]; {Nxyz2[[n3, 2]], Nxyz2[[n3, 1]], Nxyz2[[n3, 3]]},
      If[i > cn && j > 2cn - i && j ≤ cn, n4 = j + (2cn - i - 1) cn - Add[2cn - i];
       {1 - Nxyz2[[n4, 2]], Nxyz2[[n4, 1]], Nxyz2[[n4, 3]]},
       If[i ≤ cn && cn - i ≥ j - cn && j > cn, n5 = cn - j + (i) cn - Add[i];
        {Nxyz2[[n5, 2]], 1 - Nxyz2[[n5, 1]], Nxyz2[[n5, 3]]},
        If[i ≤ cn && cn - i < j - cn && j > cn, n6 = i + (2 cn - j - 1) cn - Add[2 cn - j];
         {Nxyz2[[n6, 1]], 1 - Nxyz2[[n6, 2]], Nxyz2[[n6, 3]]},
         If[i > cn && i ≤ j && j > cn, n7 = cn - i + (2 cn - j) cn - Add[2 cn - j];
          {1 - Nxyz2[[n7, 1]], 1 - Nxyz2[[n7, 2]], Nxyz2[[n7, 3]]}, If[i > cn && i > j &&
            j > cn, n8 = cn - j + (2cn - i) cn - Add[2cn - i]; {1 - Nxyz2[[n8, 2]], 1 -
             Nxyz2[[n8, 1]], Nxyz2[[n8, 3]]}, {0, 0, 0}]]]]]]]], {j, 2 n + 1}, {i, 2 n + 1}];
```

cn is the number of nodes on the edge of the one eighth slice. cx is the total number of cables in the full grid that are parallel to the x-axis. cy is the total number of cables in the full grid that are parallel to the y-axis. fn is the total number of nodes in the full grid. Add is a function that takes into account the "triangle effect" of the one eighth slice. The function adds all integers starting at one all the way up to the value x. Fullxyz takes the final position matrix of the one eighth slice and creates a matrix of the final positions for a full grid. Each of the eight sections of the full grid has a seperate if statement. i is the column number of the grid and j is the row number. So that i will start at one and increase until i equals the number of columns in a row, and this will occur for every row, j, starting at one. i and j is essentially the position in the grid and the criteria of a if statement will be met depending on which of the eight sections of the gird the position is in. n1, n2, n3... are the node numbers of the eighth slice that correspond to the position. 1 is the upper geogrid and 2 is the lower geogrid.

```
a21 = 1; b21 = 1; a22 = 1;
FullCon = Partition[
   Flatten[Join[Table[{If[i == (2n)a22 + 1, a22 += 1; a21 += 1, a21], a21 += 1}, {i, 1, cx}],
     Table[{b21++, b21 + 2n}, {i, 1, cy}]]], 2];
```

FullCon is the the full connectivity matrix. The matrix is made up of the two nodes that make up a cable. It is used to draw the cables in the Graphics3D Plot. The matrix is joined because the first table is cables parallel to the x-axis and the second table makes up cables parallel to the y-axis.

```
Fullxyz1 = Partition[Flatten[Fullxyz1], 3];
cablegrid1 = Table[Line[Table[Fullxyz1[[FullCon[[i, j]]]], {j, 2}]], {i, fc}];
Show[Graphics3D[cablegrid1, Axes → True,
   BoxRatios → {1, 1, 0.5}, ViewPoint → {1.3, -2.4, 2}, Boxed → False]];
Fullxyz2 = Partition[Flatten[Fullxyz2], 3];
cablegrid2 = Table[Line[Table[Fullxyz2[[FullCon[[i, j]]]], {j, 2}]], {i, fc}];
Show[Graphics3D[cablegrid2, Axes → True,
   BoxRatios → {1, 1, 0.5}, ViewPoint → {1.3, -2.4, 2}, Boxed → False]];
cablegrids = Append[cablegrid1, cablegrid2];
Show[Graphics3D[cablegrids, Axes → True,
   BoxRatios → {1, 1, 0.5}, ViewPoint → {1.3, -2.4, 2}, Boxed → False]];
```

Cablegrid1 is a list of all the lines (cables) in the upper gecgrid, cablegrid2 is a list of all the lines (cables) in the lower gecgrid, and cablegrids is both gecgrids so the *Show*[Graphics3D function draws the lines that represent each cable. Axes-True draws the axes and includes values. BoxRatios-is the ratio for each dimension of the box. ViewPoint {} is the angle to show the box. Boxed-False tells the kernal not to draw the box. AxesLabel is the labels to print on each axes. The thir

$$tcx = \frac{tc}{2}$$

```
Fulle1 = Flatten[Table[If[j ≤ i && i ≤ n && j ≤ n, c1 = i + (j - 1) n - Add[j]; №1[[c1]],
     If[i > n && 2n + 1 - i ≥ j && j ≤ n, c2 = n - i + 1 + (j) n - Add[j]; №1[[c2]],
      If[i ≤ n && j > i && j ≤ n + 1, c3 = tcx - 1 + j + (i - 1) n - Add[i]; №1[[c3]],
       If[i > n && j > 2n + 1 - i && j ≤ n + 1, c4 = tcx - 1 + j + (2n - i) n - Add[2n - i + 1];
        №1[[c4]], If[i ≤ n && i > 2n + 1 - j && j > n + 1, c5 = i + (2n - j + 1) n - Add[2n - j + 2];
         №1[[c5]], If[i ≤ n && i ≤ 2n + 1 - j && j > n + 1, c6 = tcx + n - j + 1 + (i) n - Add[i];
          №1[[c6]], If[i > n && i < j && j > n + 1, c7 = 3n - i + 1 + (2n - j) n - Add[2n - j + 2];
           №1[[c7]], If[i > n && i ≥ j && j > n + 1, c8 = tcx + 2n - j + 1 + (2n - i) n -
              Add[2n - i + 1]; №1[[c8]], 0]]]]]]]], {j, 2n + 1}, {i, 2n}]];
Fulle2 = Flatten[Table[If[j ≤ i && i ≤ n && j ≤ n, c1 = i + (j - 1) n - Add[j]; №2[[c1]],
     If[i > n && 2n + 1 - i ≥ j && j ≤ n, c2 = n - i + 1 + (j) n - Add[j]; №2[[c2]],
      If[i ≤ n && j > i && j ≤ n + 1, c3 = tcx - 1 + j + (i - 1) n - Add[i]; №2[[c3]],
       If[i > n && j > 2n + 1 - i && j ≤ n + 1, c4 = tcx - 1 + j + (2n - i) n - Add[2n - i + 1];
        №2[[c4]], If[i ≤ n && i > 2n + 1 - j && j > n + 1, c5 = i + (2n - j + 1) n - Add[2n - j + 2];
         №2[[c5]], If[i ≤ n && i ≤ 2n + 1 - j && j > n + 1, c6 = tcx + n - j + 1 + (i) n - Add[i];
          №2[[c6]], If[i > n && i < j && j > n + 1, c7 = 3n - i + 1 + (2n - j) n - Add[2n - j + 2];
           №2[[c7]], If[i > n && i ≥ j && j > n + 1, c8 = tcx + 2n - j + 1 + (2n - i) n -
              Add[2n - i + 1]; №2[[c8]], 0]]]]]]]], {j, 2n + 1}, {i, 2n}]];

Dimensions[Fulle1][[1]]
Dimensions[Fullxyz1][[1]]
```

Fulle1 is a list of the elongation of each cable in the upper gecgrid of the unit cell, and Fulle1 is a list of the elongation of each cable in the lower gecgrid of the unit cell. The order of the list corresponds to the FullCon list. Each if statement represents an eighth of the unit cell like the Fullxyz matrix.

```
xt01 = -h/2 ; xt201 = xt01; xt501 = xt01; xe01 = xt01; xe251 = xt01; xe501 = xt01;
xt02 = -h/2 ; xt202 = xt02; xt502 = xt02; xe02 = xt02; xe252 = xt02; xe502 = xt02;
TwoDXSection01 = Table[{Fullxyz1[[i, 1]], Fullxyz1[[i, 3]]}, {i, 1, 2n}];
TwoDXSection251 =
  Table[{Fullxyz1[[i, 1]], Fullxyz1[[i, 3]]}, {i, n^2 + 0.5n + 1, n^2 + 2.5n + 1}];
TwoDXSection501 = Table[{Fullxyz1[[i, 1]], Fullxyz1[[i, 3]]},
   {i, 2*n^2 + n + 1, 2*n^2 + 3n + 1}];
TwoDXSection02 = Table[{Fullxyz2[[i, 1]], Fullxyz2[[i, 3]]}, {i, 1, 2n}];
TwoDXSection252 =
  Table[{Fullxyz2[[i, 1]], Fullxyz2[[i, 3]]}, {i, n^2 + 0.5n + 1, n^2 + 2.5n + 1}];
TwoDXSection502 = Table[{Fullxyz2[[i, 1]], Fullxyz2[[i, 3]]},
   {i, 2*n^2 + n + 1, 2*n^2 + 3n + 1}];
MultipleListPlot[TwoDXSection01, TwoDXSection02, PlotJoined → True,
 PlotLegend → {"Top", "Bottom"}, LegendPosition → {0.95, -0.4}, LegendSize → {0.5, 0.9},
 PlotLabel → StyleForm["w Versus x at Edge", FontWeight → Bold, FontSize → 12],
 Frame → True, FrameLabel → {"x", "w"}, RotateLabel → False, SymbolShape → None]
MultipleListPlot[TwoDXSection251, TwoDXSection252, PlotJoined → True,
 PlotLegend → {"Top", "Bottom"}, LegendPosition → {0.95, -0.4}, LegendSize → {0.5, 0.9},
 PlotLabel → StyleForm["w Versus x at Quarter", FontWeight → Bold, FontSize → 12],
 Frame → True, FrameLabel → {"x", "w"}, RotateLabel → False, SymbolShape → None]
MultipleListPlot[TwoDXSection501, TwoDXSection502, PlotJoined → True,
 PlotLegend → {"Top", "Bottom"}, LegendPosition → {0.95, -0.4}, LegendSize → {0.5, 0.9},
 PlotLabel → StyleForm["w Versus x at Center", FontWeight → Bold, FontSize → 12],
 Frame → True, FrameLabel → {"x", "w"}, RotateLabel → False, SymbolShape → None]
Fulle1 = Table[Fulle1[[i]]/h, {i, cx}];
FullT1 = Table[EA*Fulle1[[i]], {i, cx}];
TPlot01 = Table[xt01 += h; {xt01, FullT1[[i]]}, {i, 2n}];
TPlot201 = Table[xt201 += h; {xt201, FullT1[[i]]}, {i, 0.8n^2 + 1, 0.8n^2 + 2n}];
TPlot501 = Table[xt501 += h; {xt501, FullT1[[i]]}, {i, 2*n^2 + 1, 2*n^2 + 2*n}];
Fulle2 = Table[Fulle2[[i]]/h, {i, cx}];
FullT2 = Table[EA*Fulle2[[i]], {i, cx}];
TPlot02 = Table[xt02 += h; {xt02, FullT2[[i]]}, {i, 2n}];
TPlot202 = Table[xt202 += h; {xt202, FullT2[[i]]}, {i, 0.8n^2 + 1, 0.8n^2 + 2n}];
TPlot502 = Table[xt502 += h; {xt502, FullT2[[i]]}, {i, 2*n^2 + 1, 2*n^2 + 2*n}];
ePlot01 = Table[xe01 += h; {xe01, Fulle1[[i]]}, {i, 2n}];
ePlot251 = Table[xe251 += h; {xe251, Fulle1[[i]]}, {i, n^2 + 1, n^2 + 2n}];
ePlot501 = Table[xe501 += h; {xe501, Fulle1[[i]]}, {i, 2*n^2 + 1, 2*n^2 + 2*n}];
ePlot02 = Table[xe02 += h; {xe02, Fulle2[[i]]}, {i, 2n}];
ePlot252 = Table[xe252 += h; {xe252, Fulle2[[i]]}, {i, n^2 + 1, n^2 + 2n}];
ePlot502 = Table[xe502 += h; {xe502, Fulle2[[i]]}, {i, 2*n^2 + 1, 2*n^2 + 2*n}];
MultipleListPlot[ePlot01, ePlot02, PlotJoined → True, PlotLegend → {"Top", "Bottom"},
 LegendPosition → {0.95, -0.4}, LegendSize → {0.5, 0.9}, Axes → False,
 PlotLabel → StyleForm["ϵx versus x at Edge", FontWeight → Bold, FontSize → 12],
 Frame → True, FrameLabel → {"x", "ϵ"}, RotateLabel → False, SymbolShape → None]
MultipleListPlot[ePlot251, ePlot252, PlotJoined → True, PlotLegend → {"Top", "Bottom"},
 Axes → False, LegendPosition → {0.95, -0.4}, LegendSize → {0.5, 0.9},
 PlotLabel → StyleForm["ϵx versus x at Quarter", FontWeight → Bold, FontSize → 12],
 Frame → True, FrameLabel → {"x", "ϵ"}, RotateLabel → False, SymbolShape → None]
MultipleListPlot[ePlot501, ePlot502, PlotJoined → True, PlotLegend → {"Top", "Bottom"},
 Axes → False, LegendPosition → {0.95, -0.4}, LegendSize → {0.5, 0.9},
 PlotLabel → StyleForm["ϵx versus x at Center", FontWeight → Bold, FontSize → 12],
 Frame → True, FrameLabel → {"x", "ϵ"}, RotateLabel → False, SymbolShape → None]
```

xt01, xt201, ...etc are the initial variables used to plot the position of each cable along the x-axis for the upper cablenet. TwoDXSection01 forms a table of the x and z coordinates along the edge of the upper cablenet, TwoDXSection251 forms a table of the x and z coordinates at the quarter of the unit cell of the upper cablenet, TwoDXSection501 forms a table of the x and z coordinates along the centerline of the unit cell upper cablenet. The values of all three of these tables are plotted on a multiple list plot. Specifications for the legend are made using the functions PlotLegend, LegendPosition, and LegendSize, PlotJoined→True plots lines for all the data, PlotLabel→StyleForm specifies the title of the plot, that the title should be bold, and the size of the font. Ftame→True plots a frame around the whole plot. FtameLabel→{"x","w"} specifies the labels that should be printed on each axis of the plot. x is the bottom axis and w is the vertical axis. RotateLabel→False prints the vertical axis label horizontal. SymbolShape→None eliminates all symbols that would be plotted for each point. Fulle1 is a table of the strain of the cables parallel to the x-axis for the upper cablenet. FullT1 is a table of the tension values of the cables parallel to the x-axis for the upper cablenet. TPlot0...etc and ePlot0...etc. serve the same purpose as the TwoDXSection tables except for the tension and Strain values. Different ranges of values were used when creating these tables because the organization of the Fullxyz table was different than the FullCon table. These same plots were created for the lower cablenet with similar variables but instead of the variables ending in 1 the vaiables end in 2.

```
Fullw1 = Partition[Table[Fullxyz1[[i, 3]], {i, fn}], 2 n + 1];
WMin = Min[Fullw1]
WMax = Max[Fullw1]
Fullex1 = Partition[Fulle1/h, 2 n];
ListPlot3D[Fullw1, Mesh → False,
 MeshRange → {{0, 1}, {0, 1}}, AxesLabel → {"x", "y", "w    "}]
ListContourPlot[Fullw1, MeshRange → {{0, 1}, {0, 1}},
 FrameLabel → {"x", "y"}, RotateLabel → False]
ListPlot3D[Fullex1, Mesh → False, MeshRange → {{0, 1}, {0, 1}},
 AxesLabel → {"x", "y", "ε    "}]
ListContourPlot[Fullex1, MeshRange → {{0, 1}, {0, 1}},
 FrameLabel → {"x", "y"}, RotateLabel → False]

Fullw2 = Partition[Table[Fullxyz2[[i, 3]], {i, fn}], 2 n + 1];
WMin = Min[Fullw2]
WMax = Max[Fullw2]
Fullex2 = Partition[Fulle2/h, 2 n];
ListPlot3D[Fullw2, Mesh → False,
 MeshRange → {{0, 1}, {0, 1}}, AxesLabel → {"x", "y", "w    "}]
ListContourPlot[Fullw2, MeshRange → {{0, 1}, {0, 1}},
 FrameLabel → {"x", "y"}, RotateLabel → False]
ListPlot3D[Fullex2, Mesh → False, MeshRange → {{0, 1}, {0, 1}},
 AxesLabel → {"x", "y", "ε    "}]
ListContourPlot[Fullex2, MeshRange → {{0, 1}, {0, 1}},
 FrameLabel → {"x", "y"}, RotateLabel → False]
```

Fullw1 is the vertical displacements of the upper cable net, and Fullw2 is the vertical displacements of the lower cable net. Both of these plots don't take into account the small x and y displacements that occur.

Fullex1 is a list of all the strains of the cables parallel to the x-axis on the upper cablenet, and Fullex2 is a list of all the strains of the cables parallel to the x-axis on the lower cablenet.

```
Total[Flatten[Fullw1]]
Total[Flatten[Fullw2]]
```

Appendix G

Rotational Stiffness Algorithm

```
Off[General::"spell1"];
Clear[XYZ, xyz, DOF, dxyz, S, Energy, e, tn, z, L, tc, d, n, CON, R, P, k, EA, ID];
FindMax = Function[{M, tn, L}, MaxX = 0; X = 0; MaxY = 0; Y = 0; MaxZ = 0; Z = 0;
   For[i = 1, i ≤ tn, i++, If[Abs[M[[i, 1]]] > Abs[MaxX], MaxX = M[[i, 1]]; X = i, MaxX; X];
    If[Abs[M[[i, 2]]] > Abs[MaxY], MaxY = M[[i, 2]]; Y = i, MaxY; Y];
    If[Abs[M[[i, 3]]] > Abs[MaxZ], MaxZ = M[[i, 3]]; Z = i, MaxZ; 1]];
   Print[TableForm[{{"Max X is ", MaxX * L, " at ", X},
      {"Max Y is ", MaxY * L, " at ", Y}, {"Max Z is ", MaxZ * L, " at ", Z}}]]];
Add = Function[x, a = 0; For[k = 1, k < x, k++, a = a + k]; a];
$TextStyle = {FontFamily → "Times", FontSize → 10};
<< Graphics`MultipleListPlot`
<< Graphics`Legend`
```

The spelling error has been turned off, and some of the variables have been cleared in the kernal. Find Max is a function that finds the maximum displacements that occur along each axis in dimensional form. Add is a function that is used to convert the nodal positions of the one-eighth portion of the unit cell to the full unit cell. $TextStyle specifies the text to be used on all output, and the <<Graphics loads the mathematica data neccessary to create multiple list plots and legends.

```
B = 60; L = 300; Qp = -14.6; Qs = -3.06; Kp = 29.2; Ks = 0.16; EApW = 7300;
Cr = 0.02; n = 50; H = L/(2 n); b = B/L; h = H/L; EA = EApW * H; qp = Qp * L^2/EA; qs = Qs * L^2/EA;
pp = qp * h^2; ps = qs * h^2; kp = Kp * L^2/EA; ks = Ks * L^2/EA; tp = kp * h^2; ts = ks * h^2;
h
b
qp
qs
kp
ks
EA
pp
ps
tp
ts
```

$\frac{1}{100}$

$\frac{1}{5}$

-60.

-12.5753

36000.

197.26

21900

-0.006

-0.00125753

3.6

0.019726

B is half of the width of the pile cap (cm), L is the cell unit length (cm), Qp is the pressure acting on the geogrid above the pile (N/cm^2), Qs is the pressure acting on the geogrid above the soil (N/cm^2), Kp is the soil stiffness above the pile (N/cm^2/cm), Ks is the soil stiffness above the soil (N/cm^2/cm), EApW is the Elastic Modulus multiplied by the geogrid rib cross-sectional area divided by the number of ribs in a cm (N/cm), n is the number of geogrid ribs (cables) in half of the cell unit length. H is geogrid rib length (cable length) (cm), EA (N), qp, qs, kp, ks, b and h are dimensionless. pp is the dimensionless point load acting above the pile , ps is the dimensionless point load acting above the soft soil, tp is the dimensionless stiffness of the soil spring on top of the pile , and ts is the dimensionless stiffness of the soft soil spring.

```
xyz = Partition[Flatten[Table[{i*h, j*h, 0}, {j, 0, n}, {i, j, n}]], 3];
```

xyz is the initial position matrix of the cable net nodes that make up one-eighth of an unit cell. The coordinates are ordered in the same way as they are listed in the title.

```
a1 = 1; a2 = 2; b1 = 0; b2 = 1; b3 = n + 2;
Con = Partition[
   Flatten[Join[Table[{If[j == a2, a2++; a1 += 1, a1], a1 += 1}, {j, 1, n}, {i, j, n}],
     Table[{If[i == b2, b2++; b1 += 2, b1 += 1], b3++}, {i, 1, n}, {j, i, n}]]], 2];
Transpose[xyz] // MatrixForm
Transpose[Con] // MatrixForm
```

Con is the connectivity matrix. It is a list of the nodes that are connected by ribs (cables). It is a joined table, the first list is of the nodes that make up the cables parallel to the x-axis and the second list is of the nodes that make up the cables parallel to the y-axis.

```
tn = Dimensions[xyz][[1]]
tc = Dimensions[Con][[1]]
```

```
66
```

```
110
```

tn is the number of nodes in a cable net for one-eighth of the unit cell. tc is the number of cables in one-eighth of the unit cell.

```
c1 = n; c2 = n;
CoC = Join[Table[If[i ≤ n, 1/2, 1], {i, tc/2}],
   Table[If[i == c1, c2 -= 1; c1 += c2; 1/2, 1], {i, tc/2}]];
```

CoC is the cable coefficient vector. This vector considers the effect of perimeter cables, and reduces the energy a perimeter cable absorbs by one half. Perimeter cables are accounted for twice when the reduced unit cell is transformed to the full unit cell. The list must be joined so that the values correspond to the order of the Con matrix.

```
a = 0;
uvw = Table[{If[xyz[[i, 1]] == 1/2 || xyz[[i, 1]] == 0, 0, x[a += 1]], If[xyz[[i, 2]] == 0 ||
      xyz[[i, 2]] == 1/2, 0, If[xyz[[i, 1]] == xyz[[i, 2]], x[a], x[a += 1]]],
    x[a += 1]}, {i, tn}];
```

uvw is the displacement matrix. All degrees of freedom that will be restrained are given a value of zero, the rest are given a variable x[a] where a is an increasing integer. The diagonal nodes along the edge of the grid that make up the reduced one-eighth unit cell are restrained so that the x and y displacements are equal.

```
nDOF = a
DOF = Table[x[i], {i, nDOF}];
```

nDOF is the number of degrees of freedom in one-eighth of the unit cell. DOF is a list of all the variables, x[a], that represent degrees of freedom. This list will be used in the NMinimize function.

Rectangle Piles

```
Co = Table[If[xyz[[i, 1]] == xyz[[i, 2]] == 0 || xyz[[i, 1]] == xyz[[i, 2]] == 1/2,
    1/8, If[xyz[[i, 1]] == 1/2 && xyz[[i, 2]] == 0, 1/4,
     If[xyz[[i, 1]] == 1/2 || xyz[[i, 2]] == 0 || xyz[[i, 1]] == xyz[[i, 2]], 1/2, 1]]], {i, tn}];
p = Table[If[xyz[[i, 1]] ≤ b, pp * Co[[i]], ps * Co[[i]]], {i, tn}];
t = Table[If[xyz[[i, 1]] ≤ b, tp * Co[[i]], ts * Co[[i]]], {i, tn}];
c = Cr * Co;
c[[1]] = 2 * c[[1]];
c[[tn]] = 2 * c[[tn]];
```

Co is a list of load coefficients that account for the tributary areas that result from the geometry of one-eighth of the unit cell. The nodes on the perimeter of the cable net that makes up one-eighth of the unit cell will be one-half of the full load. The two corners with 45-degree angles (the center node and corner node of the full unit cell) will be one-eighth of the full load, and the corner with a 90 degree angle (on the full grid it is the center edge node) will be one-fourth of the full load. p and k are the list of point loads and soil stiffness that multiplies the load coefficients by the pile or soil values depending on whether the node is over a pile or not. c is the list of soil stiffnesses of the soil in between the two geogrids. The c is doubled at the two 45-degree angled corners because these two nodes only have one cable connected them.

```
e = Table[If[i ≤ tc / 2, c1 = Con[[i, 1]]; c2 = Con[[i, 2]];
    √((h + uvw[[c2, 1]] - uvw[[c1, 1]])^2 + (uvw[[c2, 2]] - uvw[[c1, 2]])^2 +
        (uvw[[c2, 3]] - uvw[[c1, 3]])^2) - h, c1 = Con[[i, 1]]; c2 = Con[[i, 2]];
    √((uvw[[c2, 1]] - uvw[[c1, 1]])^2 + (h + uvw[[c2, 2]] - uvw[[c1, 2]])^2 +
        (uvw[[c2, 3]] - uvw[[c1, 3]])^2) - h], {i, 1, tc}];
```

e is the elongation list. It uses the Con matrix and the displacement matrix to find the total elongation of a cable due to displacement.

```
b30 = n + 1; b31 = n + 1; b32 = 0; b33 = n + 1;
ϕ8 = Table[If[i == 1, {2 * ArcTan[(uvw[[i, 3]] - uvw[[i + 1, 3]])/h], 0},
    If[i == tn, b32 += 2; {0, 2 * ArcTan[(uvw[[i, 3]] - uvw[[b32, 3]])/h]}, If[i ≤ n, b33++;
     {ArcTan[(uvw[[i, 3]] - uvw[[i - 1, 3]])/h] + ArcTan[(uvw[[i, 3]] - uvw[[i + 1, 3]])/h],
      2 * ArcTan[(uvw[[i, 3]] - uvw[[b33, 3]])/h]}, If[i == n + 1, b33++;
      {2 * ArcTan[(uvw[[i, 3]] - uvw[[i - 1, 3]])/h], 2 * ArcTan[(uvw[[i, 3]] - uvw[[b33, 3]])/h]},
      If[i == b31 + 1, b30 -= 1; b31 += b30; b32 += 2;
       {ArcTan[(uvw[[i, 3]] - uvw[[i + 1, 3]])/h] + ArcTan[(uvw[[i, 3]] - uvw[[b32, 3]])/h],
        ArcTan[(uvw[[i, 3]] - uvw[[i + 1, 3]])/h] + ArcTan[(uvw[[i, 3]] - uvw[[b32, 3]])/h]},
       If[i == b31, b32++; b33++; {2 * ArcTan[(uvw[[i, 3]] - uvw[[i - 1, 3]])/h],
         ArcTan[(uvw[[i, 3]] - uvw[[b32, 3]])/h] + ArcTan[(uvw[[i, 3]] - uvw[[b33, 3]])/h]},
        b32++; b33++; {ArcTan[(uvw[[i, 3]] - uvw[[i - 1, 3]])/h] +
          ArcTan[(uvw[[i, 3]] - uvw[[i + 1, 3]])/h], ArcTan[(uvw[[i, 3]] - uvw[[b32, 3]])/h] +
          ArcTan[(uvw[[i, 3]] - uvw[[b33, 3]])/h]}]]]]]], {i, tn}];
```

ϕ8 is the matrix of the two angles that are formed by the 4 cables at each node in the x-z and y-z planes.

$$\text{Energy} = \sum_{i=1}^{tc} \frac{CoC[[i]] * e[[i]]^2}{2*h} + \sum_{j=1}^{tn} \frac{1}{2} * t[[j]] * (uvw[[j, 3]])^2 + \sum_{k=1}^{tn} \sum_{i=1}^{2} \frac{1}{2} * c[[k]] * (\phi 8[[k, l]])^2 - \sum_{m=1}^{tn} p[[m]] * uvw[[m, 3]];$$

Energy is the sum of the total energy for the system of cables, soil springs, ,rotational springs and point loads.

```
S = Timing[NMinimize[Energy, DOF]];
S[[1]]
```

S is the solution of the NMinimize function for the Energy equation and the list of degrees of freedom (x[a]). The timing function is used to time how long it took the function to find the solution.

```
uvw /. S[[2, 2]] >> Desktop/SchoolStuff/Research/Mathematica/ParametricStudyData/uvwR
e /. S[[2, 2]] >> Desktop/SchoolStuff/Research/Mathematica/ParametricStudyData/eR
```

The solution for each DOF is then placed in the displacement and elongation matrices and stored in seperate files on the hard drive.

```
Nuvw = << Desktop/SchoolStuff/Research/Mathematica/ParametricStudyData/uvwR;
Ne = << Desktop/SchoolStuff/Research/Mathematica/ParametricStudyData/eR;
Nxyz = xyz + Nuvw;
```

The solution is retrieved fom the uvw and e files and called Nuvw and Ne respectively. This was done so that it was not necessary to run the NMinimize function everytime a notebook is opened. Nxyz is the new position matrix of the displaced grid. It is a simple addition of the initial position matrix and the displacement matrix.

```
ϕ8 /. S[[2, 2]]


ϵ = Table[-Nε[[i]] / h, {i, tc}];
T = Table[EA * ϵ[[i]], {i, tc}];
Max[ϵ]
Max[T]
Min[Nuvw]
FindMax[Nuvw, tn, L]
Ds = Nuvw[[1, 3]] - Min[Nuvw]
Ds * L
pos = Table[If[xyz[[i, 1]] ≤ b, 0, 1], {i, tn}];
LDR = Total[Table[t[[i]] * Nuvw[[i, 3]] * pos[[i]], {i, tn}]] /
  Total[Table[p[[j]] * pos[[j]], {j, tn}]]
```

ϵ is a list of the strain values for the cables. T is a list of tension values for the cables. The maximum strain, tension, and vertical displacement are found. The FindMax function is used to find the maximum displacements in dimensional form. The differential settlement is found in non-dimensional and dimensional form by subtracting the smallest vertical displacement from the largest vertical displacement. pos is a list of all the nodes in the reduced unit cell, a 0 means the node is initially above the pile and 1 means the node is initially above the soft soil. LDR is the ratio of the load the soft soil springs support to the load on the soft soil. It is found by finding the total of the soil stiffness times the vertical displacement of each node supported by the soft soil and dividing it by the total of the point loads acting on each node supported by the soft soil

```
cn = n + 1; cx = 2 n (2 n + 1); cy = cx; fc = cx + cy; fn = (2 n + 1)^2
Fullxyz =
  Table[If[ j ≤ i && i ≤ cn && j ≤ cn, n1 = i + (j - 1) cn - Add[j]; {Nxyz[[n1, 1]], Nxyz[[n1, 2]],
     Nxyz[[n1, 3]]}, If[i > cn && 2cn - i ≥ j && j ≤ cn, n2 = cn - i + (j) cn - Add[j];
     {1 - Nxyz[[n2, 1]], Nxyz[[n2, 2]], Nxyz[[n2, 3]]}, If[i ≤ cn && j > i && j ≤ cn,
      n3 = j + (i - 1) cn - Add[i]; {Nxyz[[n3, 2]], Nxyz[[n3, 1]], Nxyz[[n3, 3]]},
      If[i > cn && j > 2cn - i && j ≤ cn, n4 = j + (2cn - i - 1) cn - Add[2cn - i];
       {1 - Nxyz[[n4, 2]], Nxyz[[n4, 1]], Nxyz[[n4, 3]]},
       If[i ≤ cn && cn - i ≥ j - cn && j > cn, n5 = cn - j + (i) cn - Add[i]; {Nxyz[[n5, 2]],
         1 - Nxyz[[n5, 1]], Nxyz[[n5, 3]]}, If[i ≤ cn && cn - i < j - cn && j > cn,
         n6 = i + (2cn - j - 1) cn - Add[2cn - j]; {Nxyz[[n6, 1]], 1 - Nxyz[[n6, 2]],
          Nxyz[[n6, 3]]}, If[i > cn && i ≤ j && j > cn, n7 = cn - i + (2cn - j) cn - Add[2cn - j];
          {1 - Nxyz[[n7, 1]], 1 - Nxyz[[n7, 2]], Nxyz[[n7, 3]]}, If[i > cn && i > j && j > cn,
           n8 = cn - j + (2cn - i) cn - Add[2cn - i]; {1 - Nxyz[[n8, 2]], 1 - Nxyz[[n8, 1]],
            Nxyz[[n8, 3]]}, {0, 0, 0}]]]]]]], {j, 2n + 1}, {i, 2n + 1}];

441
```

cn is the number of nodes on the edge of the one eighth slice. cx is the total number of cables in the full grid that are parallel to the x-axis. cy is the total number of cables in the full grid that are parallel to the y-axis. fn is the total number of nodes in the full grid. Add is a function that takes into account the "triangle effect" of the one eighth slice. The function adds all integers starting at one all the way up to the value x. Fullxyz takes the final position matrix of the one eighth slice and creates a matrix of the final positions for a full grid. Each of the eight sections of the full grid has a seperate if statement. i is the column number of the grid and j is the row number. So that i will start at one and increase until i equals the number of columns in a row, and this will occur for every row, j, starting at one. i and j is essentially the position in the grid and the criteria of a if statement will be met for depending on which of the eight sections of the gird the position is in. n1, n2, n3... are the node numbers of the eighth slice that correspond to the position.

```
a21 = 1; b21 = 1; a22 = 1;
FullCon = Join[Table[{If[i == (2n) a22 + 1, a22 += 1; a21 += 1, a21], a21 += 1}, {i, 1, cx}],
   Table[{b21++, b21 + 2n}, {i, 1, cy}]];
```

FullCon is the the full connectivity matrix. The matrix is made up of the two nodes that make up a cable. It is used to draw the cables in the Graphics 3D Plot. The matrix is joined because the first table is cables that are parallel to the x-axis and the second table makes up the cables parallel to the y-axis.

```
Fullxyz = Partition[Flatten[Fullxyz], 3];
cablegrid = Table[Line[Table[Fullxyz[[FullCon[[i, j]]]], {j, 2}]], {i, fc}];
Show[Graphics3D[cablegrid, Axes → True, AxesLabel → {"x", "y", "z"},
   BoxRatios → {1, 1, 0.5}, ViewPoint → {1.3, -2.4, 1}, Boxed → False]];
```

Cablegrid is a list of all the lines (cables) in the grid so the Show[Graphics3D function draws the lines that represent each cable. Axes-True draws the axes and includes values. BoxRatios-is the ratio for each dimension of the box. ViewPoint-{} is the angle to show the box. Boxed-False tells the kernal not to draw the box. AxesLabel is the labels to print on each axes.

```
tcx = tc/2
Fulle = Flatten[Table[If[ j ≤ i && i ≤ n && j ≤ n, c1 = i + (j - 1) n - Add[j]; №[[c1]],
     If[i > n && 2n + 1 - i ≥ j && j ≤ n, c2 = n - i + 1 + (j) n - Add[j]; №[[c2]],
      If[i ≤ n && j > i && j ≤ n + 1, c3 = tcx - 1 + j + (i - 1) n - Add[i]; №[[c3]],
       If[i > n && j > 2n + 1 - i && j ≤ n + 1, c4 = tcx - 1 + j + (2n - i) n - Add[2n - i + 1];
        №[[c4]], If[i ≤ n && i > 2n + 1 - j && j > n + 1, c5 = i + (2n - j + 1) n - Add[2n - j + 2];
         №[[c5]], If[i ≤ n && i ≤ 2n + 1 - j && j > n + 1, c6 = tcx + n - j + 1 + (i) n - Add[i];
          №[[c6]], If[i > n && i < j && j > n + 1, c7 = 3n - i + 1 + (2n - j) n - Add[2n - j + 2];
           №[[c7]], If[i > n && i ≥ j && j > n + 1, c8 = tcx + 2n - j + 1 + (2n - i) n -
              Add[2n - i + 1]; №[[c8]], 0]]]]]]], {j, 2n + 1}, {i, 2n}]];


xt0 = -h/2; xt20 = xt0; xt50 = xt0; xϵ0 = xt0; xϵ25 = xt0; xϵ50 = xt0;
TwoDXSection0 = Table[{Fullxyz[[i, 1]], Fullxyz[[i, 3]]}, {i, 1 + 2n}];
TwoDXSection25 =
  Table[{Fullxyz[[i, 1]], Fullxyz[[i, 3]]}, {i, n² + 0.5n + 1, n² + 2.5n + 1}];
TwoDXSection50 = Table[{Fullxyz[[i, 1]], Fullxyz[[i, 3]]},
   {i, 2*n² + n + 1, 2*n² + 3n + 1}];
MultipleListPlot[TwoDXSection0, TwoDXSection25, TwoDXSection50,
 PlotJoined → True, PlotLegend → {"y = 0", "y = 0.25", "y = 0.5"},
 LegendPosition → {0.95, -0.4}, LegendSize → {0.5, 0.9},
 PlotLabel → StyleForm["w Versus x", FontWeight → Bold, FontSize → 12],
 Frame → True, FrameLabel → {"x", "w"}, RotateLabel → False, SymbolShape → None]
Fullϵ = Table[(Fulle[[i]]/h), {i, cx}];
FullT = Table[EA * Fullϵ[[i]], {i, cx}];
TPlot0 = Table[xt0 += h; {xt0, FullT[[i]]}, {i, 2n}];
TPlot20 = Table[xt20 += h; {xt20, FullT[[i]]}, {i, 0.8n² + 1, 0.8n² + 2n}];
TPlot50 = Table[xt50 += h; {xt50, FullT[[i]]}, {i, 2*n² + 1, 2*n² + 2*n}];
MultipleListPlot[TPlot0, TPlot20, TPlot50,
 PlotJoined → True, PlotLegend → {Y = "0", "Y = 0.25", "Y = 0.5"},
 LegendPosition → {0.95, -0.4}, LegendSize → {0.5, 0.9},
 PlotLabel → StyleForm["Tension", FontWeight → Bold, FontSize → 12],
 Frame → True, FrameLabel → {"X", "T"}, RotateLabel → False, SymbolShape → None]
ϵPlot0 = Table[xϵ0 += h; {xϵ0, Fullϵ[[i]]}, {i, 2n}];
ϵPlot25 = Table[xϵ25 += h; {xϵ25, Fullϵ[[i]]}, {i, n² + 1, n² + 2n}];
ϵPlot50 = Table[xϵ50 += h; {xϵ50, Fullϵ[[i]]}, {i, 2*n² + 1, 2*n² + 2*n}];
MultipleListPlot[ϵPlot0, ϵPlot25, ϵPlot50,
 PlotJoined → True, PlotLegend → {"y = 0", "y = 0.25", "y = 0.5"},
 LegendPosition → {0.95, -0.4}, LegendSize → {0.5, 0.9},
 PlotLabel → StyleForm["ϵx Versus x", FontWeight → Bold, FontSize → 12],
 Frame → True, FrameLabel → {"x", "ϵ"}, RotateLabel → False, SymbolShape → None]
```

xt0, xt20, ...etc are the initial variables used to plot the position of each cable along the x-axis. TwoDXSection0 forms a table of the x and z coordinates along the edge, TwoDXSection25 forms a table of the x and z coordinates at the quarter of the unit cell, TwoDXSection50 forms a table of the x and z coordinates along the centerline of the unit cell. The values of all three of these tables are plotted on a multiple list plot. Specifications for the legend are made using the functions PlotLegend, LegendPosition, and LegendSize. PlotJoined→True plots lines for all the data, PlotLabel→ StyleForm specifies the title of the plot, that the title should be bold, and the size of the font. Frame→True plots a frame around the whole plot. FrameLabel→{"x","w"} specifies the labels that should be printed on each axis of the plot. x is the bottom axis and w is the vertical axis. RotateLabel→False prints the vertical axis label horizontal. SymbolShape→None eliminates all symbols that would be plotted for each point. Fulle is a table of the strain of the cables parallel to the x-axis. Tension is a table of the tension values of the cables parallel to the x-axis. TPlot0...etc and ϵPlot0...etc. serve the same purpose as the TwoDXSection tables except for the Tension and Strain values. Different ranges of values were used when creating these tables because the organization of the Fullxyz table was different than the FullCon table.

```
Fullw = Partition[Table[Fullxyz[[i, 3]], {i, fn}], 2n + 1];
wMin = Min[Fullw]
wMax = Max[Fullw]
ListPlot3D[Fullw, Mesh → False,
 MeshRange → {{0, 1}, {0, 1}}, AxesLabel → {"x", "y", "z    "}]
ListContourPlot[Fullw, MeshRange → {{0, 1}, {0, 1}},
 FrameLabel → {"x", "y"}, RotateLabel → False]
```

Fullw is the vertical displacements of the cable net. Both of these plots don't take into account the small x and y displac ments that occur.

```
Fullex = Partition[Fulle, 2n];
ListPlot3D[Fullex, Mesh → False,
 MeshRange → {{0, 1}, {0, 1}}, AxesLabel → {"x", "y", "ϵ    "}]
ListContourPlot[Fullex, MeshRange → {{0, 1}, {0, 1}},
  FrameLabel → {"x", "y"}, RotateLabel → False];
```

Fullex is a list of all the strains of the cables parallel to the x-axis

```
Total[Flatten[Fullw]]
Total[Flatten[Fullex]]
```

Vita

Kyle Halvordson was born the eldest of four boys on March 16, 1984 to Peter and Patricia Halvordson of Mystic, Connecticut. Kyle spent his entire childhood and adolescence in Mystic and graduated from Fitch High School in May of 2002. He attended the University of Connecticut to pursue engineering, and graduated in May of 2006 with a Bachelor of Science in Civil Engineering. Later that year he enrolled at Virginia Polytechnic Institute and State University to further his studies in structural engineering. Kyle earned his Master of Science degree in Civil Engineering in December of 2007.