Form-Finding Structures

By

Ariane Ida Fund

Sc.B. Engineering
Brown University, 2006

SUBMITTED TO THE DEPARTMENT OF CIVIL AND ENVIRONMENTAL ENGINEERING IN PARTIAL FULLFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF ENGINEERING IN CIVIL AND ENVIRONMENTAL ENGINEERING
AT THE
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

JUNE 2008

Signature of Author:_____
Department of Civil and Environmental Engineering
May 19, 2008

Certified by:_____
Jerome J. Connor
Professor of Civil and Environmental Engineering
Thesis Supervisor

Accepted by:_____
Daniele Veneziano
Chairman, Departmental Committee for Graduate Students

Form –Finding Structures

By

Ariane Ida Fund

Submitted to the Department of Civil and Environmental Engineering on May 19, 2008 in Partial Fulfillment of the Requirements for the Degree of Master of Engineering in Civil and Environmental Engineering

ABSTRACT

Inherently characterized by the interaction of geometry and forces, the unique nature of long span dome, shell, and membrane structures readily allows collaboration between architects and engineers in the examination of their optimal form. Through the elimination of bending and shear forces in the structure, less material and reinforcement is needed. By minimizing the use of materials, a form that is economical, sustainable and aesthetically attractive emerges. However, this optimization must be done through form-finding methods, whereby the structure itself defines its own shape based on its figure of equilibrium under applied loads. Unlike free forms which are defined mathematically, form-finding shapes rely on the structure and loads themselves for definition. Before the use of computers, these equilibrium shapes could only be found through cumbersome physical models. As technology has advanced, numerical methods have evolved to solve for the optimal shape.

This paper presents a brief history of physical methods formerly used, as well as common applications for these structures. Two numerical methods, the Pucher's equation method and the force-density method (FDM), are then presented. Pucher's equation relies on a prescribed stress resultant throughout the structure, while the force-density method relies on prescribed force-to-length ratios in each bar or cable, leading to a single system of linear equations. Advantages and disadvantages of both methods are discussed, as well as examples illustrating the types of structures that can be formed. These methods are shown to be powerful tools that can be generalized to a number of situations with minimal input required by the designer. The structures are able to define themselves, leading to extremely rational and beautiful forms.

Thesis Supervisor: Jerome J. Connor

Title: Professor of Civil and Environmental Engineering

Acknowledgements

I would like to thank everyone who has helped me in making this thesis and degree possible. To Professor Jerome J. Connor, who has provided his incredible knowledge, advice, and continuous encouragement to me throughout the year. To the MEng class of 2008, for your friendship and hours of work in the MEng room, I never would have made it without you. To Kate, who has always been an amazing friend and the best editor. To Conall, for being so understanding and supportive through everything. And mostly to my parents, who have always believed in me and encouraged me in everything that I do—I owe this all to you.

Table of Contents

Table of Figures

Chapter 1: Introduction

A distinct emergence in the modern system of building design is the lack of interaction between engineer and architect in the earliest stages of conceptualization. Oftentimes the architect will determine the shape and form of a building, while the engineer is purely an aide to make the imagined form work from a practical, structural point of view. One major exception is that of long span dome, shell and membrane forms, which are inherently characterized by the interaction of geometry and forces. Consequently, a structure emerges that is governed by neither aesthetics nor mathematics alone, but is a collaboration of engineering and architecture that together create a rational form that expresses its true function. The minimization of materials, which allows for abundant light and large open spaces, often results in an aesthetically pleasing, economical and sustainable structure.

As David P. Billington states, "The disciplines of structural art are efficiency and economy, and the freedom lies in the potential it offers for the expression of a personal style motivated by a conscious aesthetic search for engineering elegance" (Billington, 1983). If a structure is efficient in resisting its applied forces, then it will be economical in material usage and can achieve elegance in form. In order to achieve this, shape must be derived solely from applied forces and self weight. If the form of a tension or compression surface, which can only carry axial forces, reflects the load path, and thus represents the equilibrium shape, no bending will occur. The alternative to employing these self-equilibrium shapes to define a surface is using free form shapes. Free forms are defined by elements of geometrical shapes, such as lines, spheres, ellipses and hyperbolas, and are in no way functions of the forces acting in them. While this makes the surfaces easy to describe mathematically, the loads are completely independent from the shape determination process, leading to bending forces under both self weight and applied loads. These forces must be resisted by increasing the thickness and reinforcement of the structure, and do not make effective use of material.

Before the use of computers, efficient equilibrium-shaped forms could only be found through measurements on scaled physical models. With the improvement of technology, a number of numerical methods have been developed to combat the deficiencies of experimental models in achieving precision and generalibility.

This thesis examines the use of Pucher's equation in the 1960s to solve for the optimal shape, and more recently the use of the force-density method, which addresses a number of problems previously faced in quickly and easily visualizing the optimal surface. Examples illustrating the scope and limitations of both methods are presented, along with comparisons between the two.

Chapter 2: Historical Context

Evidence of dwellings from ancient prehistoric communities indicates that efficient and optimal structures were formed through intuitive and experimental means well before these very theories were even formulated as the ideal. The earliest evidence of human dwellings, located in southern France in a camp called Terra Amata, can be dated back approximately 380,000 years. These remains indicate that stick-framed dome structures were developed by prehistoric people, similar to those found throughout the world in indigenous communities. The structures discovered at Terra Amata were believed to have been formed by closely placing tree branches in an oval shape on the ground, surrounding a central ridge beam supported by posts. When bent inward toward the ridge beam and attached to each other, the branches formed a series of arches, often tied together with horizontal branches effectively acting as ring beams (Berger, 2007). Rocks and boulders were also placed around the edges to support the frame. This form reflects aspects of the two primary types of prehistoric stick-framed habitations found throughout the world. These are radial structures with arches meeting in the center, or orthogonal structures with sets of arches crossing at right angles, tied with horizontal members. While these structures of arches and hoops were obviously determined based on experience and available materials, it is an extremely efficient system that carries the forces of self weight and wind to the ground in the most direct fashion. It is now known that a funicular shape is the most materially efficient form, able to resist the maximum load per structural weight or area. This funicular form is the shape a structure takes under its own weight and loads, acting like a chain with no stiffness, hinged at every point. When many of these chains are linked together into a two-way net, and reversed to form arches, a very stable grid dome structure emerges, greatly resembling prehistoric structures. These ancient constructions were formed based on practical considerations such as ease of construction, maximization of internal space and empirical structural efficiency through minimizing materials. The result was the development of extremely efficient structures, only limited by the materials from which they were built. Today,

these same considerations continue to dominate in choosing the structural form, and thus many of the same forms found by prehistoric peoples continue to be employed in long span shell structures today.

Early pioneers in form-finding lightweight structure design, which began in the 1950s, relied on physical models and natural forms to mimic desired behavior. Frei Otto used soap bubbles and soap film within edges to represent membranes in equilibrium with uniform stress in all directions and no bending or shear stresses. When these same shapes are used in buildings, they form tensile membranes and prestressed cable nets. These minimal, uniformly loaded membrane surfaces were the start of today's interest in fabric tensile design (Wakefield, 2006). This shape cannot be described mathematically, thus close measurements of the models had to be performed to replicate the shape. As technology advanced, Otto was one of the first to move away from physical models and to use computer methods to investigate and quantify these optimal forms.

Another equilibrium shape is the freely hanging cable, which forms a catenary curve carrying forces only in tension. An arch in perfect compression results when this shape is flipped. The architect Gaudi was one of the first to investigate this property, and made many physical models of hanging cables, which were used to construct vaults in a number of his projects in Barcelona.

Taking this catenary concept further, Heinz Isler was a pioneer in thin-shell concrete construction through his investigations of the catenary behavior of suspended fabric. His method for finding the shape of a doubly curved surface involved hanging wet fabric and letting it freeze, or using hung fabric soaked in a cement mix that was allowed to dry. The sagged form acts like a catenary in tension under its own weight, which could then be inverted into a structural shell model in compression. As a result, he was able to achieve extremely durable, thin and light structures. Furthermore, Isler found that even when using the same amount of material, his doubly curved shells could support up to three times the load as a flat structure before failure (Linkwitz, 1999). By using these methods, Isler was able to design the service station in Deitingen Sud, Switzerland as a concrete shell structure with a thickness of just nine centimeters. The model of frozen fabric can clearly be visualized in this form (Figure 2-1).

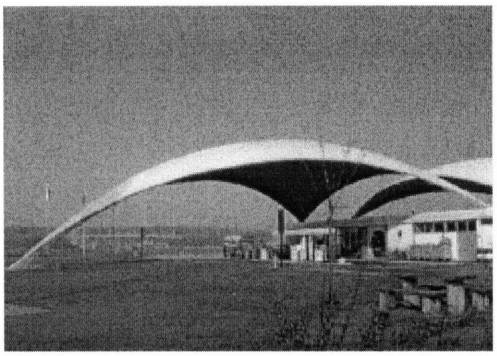

Figure 2-1: Service station, Deitingen Sud, Switzerland (Brew and Lewis, 2007)

Chapter 3: Applications

At present, sports arenas and airport terminals are driving long span roof structural design. Airports have become "inextricably linked to crystal-clear wayfinding, expansive views, and abundant daylight" (Solomon, 2005). Long span roofs that minimize vertical supports allow for both maximum light and circulation through the space. Their shape also mimics the aerospace industry in terms of expressing flexibility and fluidity, and the large scale reflects that of the airplane itself. These roofs can also have a psychological effect by which "the volumes and undulating shapes made possible by their inherent geometries can help lift the spirits of most any weary traveler" (Solomon, 2005). Stadiums roofs are often very similar to airport terminal roofs. Also motivated by the desire to create large unobstructed spaces, many times they too are the key signature feature of this type of structure.

Fabric has a long history of use in building design. Simple tents have been used for centuries, and continue to be employed as temporary accommodations for recreational or refugee purposes. However, modern fabric structures for permanent buildings have also been used for approximately 40 years (Berger, 1999). Their lightness, speed of erection, control of air quality and daylight, acoustic properties and visual appeal make them a desirable feature for large venues and arenas. While mainly seen in canopy roofs, structural fabric is also now being employed as part of the building enclosure itself.

The most basic tensile roof surface requires at least four supports, one of which must be outside of the plane defined by the other three. Using an orthogonal grid, the four-point structure requires cables along the edges, or alternatively, a polar coordinate grid could be used to form a radial tent with catenary edges. Since the material lacks substantial weight, elements must be added to resist upward wind loads as well. The placement of support points, which can be at the high points, low points, and perimeter of the structure, are what determine the ultimate shape. Using form-finding computer programs, discussed later, an equilibrium shape can be determined by prescribing a specified surface stress pattern along with these fixed support points. As Berger states,

modifying the popular principal of Louis Sullivan, "form clearly follows structural function." In addition, "...the structural form defines the sculptural shape of the building on the outside and the form of the space on the inside. There is no longer any distinction between engineering and architecture."

Fabric tensile roofs may be divided into categories based on their supports, namely point-supported structures, A-frame supported structures and arch-supported structures (Berger, 1999). Point-supported structures are formed by ridge cables draped over high mast supports. The roof structure at the Denver International airport, shown in Figure 3-1, makes use of this system. Ridge cables carry the downward loads, while valley cables placed between ridge cables run parallel and form arches to carry the required upward load. The downside is the addition of horizontal forces in the anchors, which are seen in Figure 3-2 to be resisted through diagonal members in the supports.

Figure 3-1: Denver International Airport (Berger, 1999)

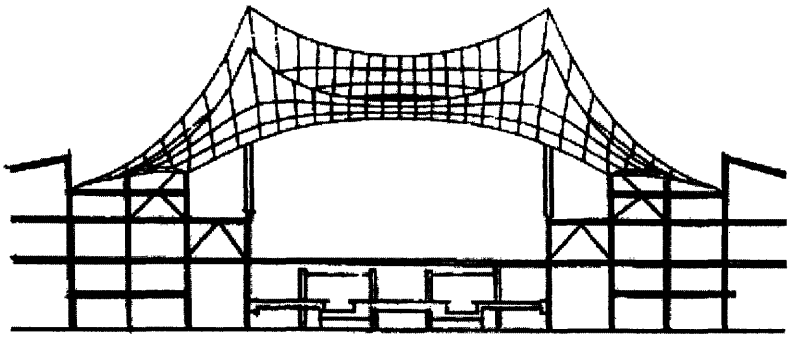

Figure 3-2: Denver Airport Elevation (Berger, 1999)

The Riyadh Stadium, in Saudi Arabia, uses pole supported tent units to form one of the largest span roof structures of the present day. Twenty-four units are arranged in a circle, with each unit having "a vertical main mast, a pair of suspension and stabilizing cables, and an anchorage system of stay cables; the units are tied together and stressed by a central ring cable" (Knudson, 1991), as demonstrated in Figure 3-3 and Figure 3-4.

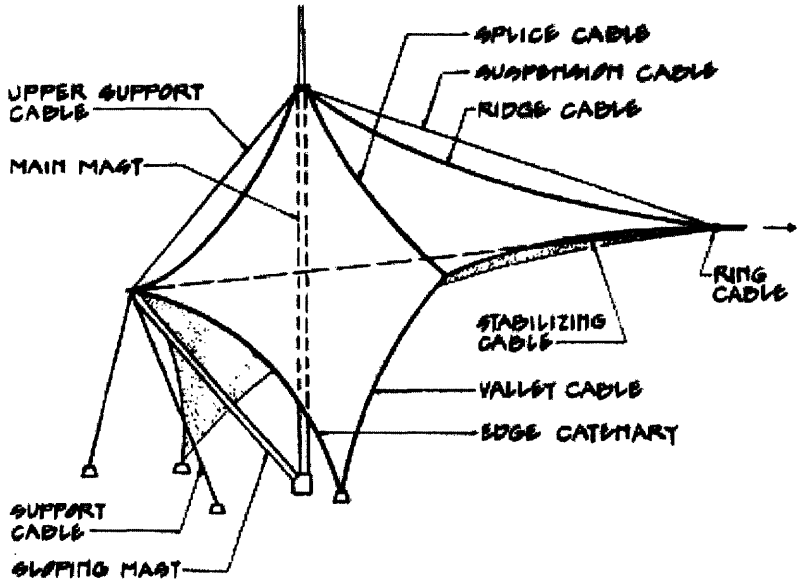

Figure 3-3: One of the 24 unites of the Riyadh Stadium Roof (Berger, 1999)

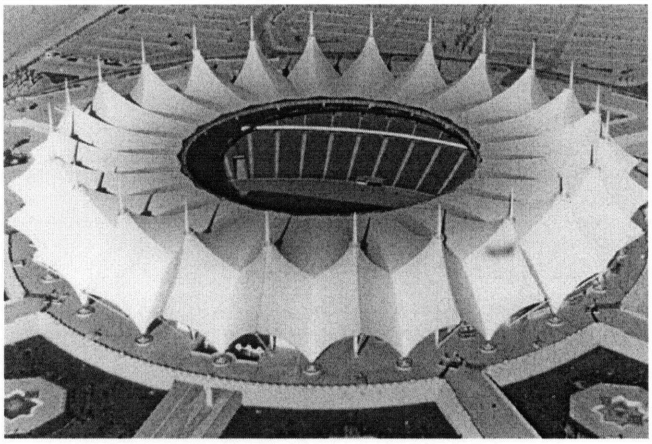

Figure 3-4: Point-Supported Structure: Riyadh Stadium (www.skyscrapercity.com)

Point-supported structures can sometimes result in spatially inefficient forms, however. Either the mast of the peak is forced to be at the center, providing height at the expense of free interior space. Alternatively, masts moved to the exterior would remove the interior obstructions, but would create a tent high at the edges and low at the center. A-frame supports can be used instead. Employed in the roof of the Cynthia Woods Mitchell Center of the Performing Arts, Figure 3-5, three A-frames along with the stage house structure form the support system (Berger, 1999). Here, compression struts connect the support columns and edge cable anchors to the stage house. The supports of the A-frames now form the low points of the surface, while compression struts connecting the support columns and edge cable anchors to the stage house balance the horizontal forces in the membrane.

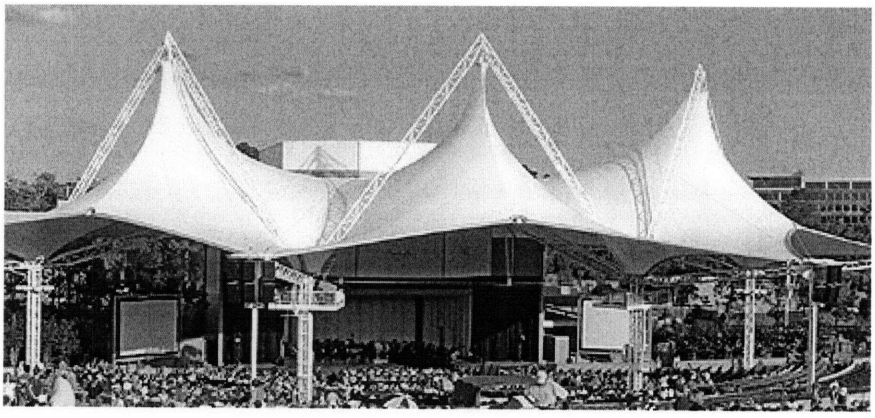

Figure 3-5: A-frame Supported Structure: Mitchell Performing Arts Center (Berger, 1999)

Finally, arch-supported structures are particularly efficient for spans in rectilinear structures. In the McClain Practice facility at the University of Wisconsin, prefabricated steel arches were able to be used to span the full width of the field. Fabricated in the shop, they were bolted together in the field to form half arches, before being lifted and pinned in the center. By doing so, each arch took only one day to erect, illustrating a great benefit of this arch system (Berger, 1999). An additional feature is the roof material itself. The outer edges are covered by a stainless steel roof, while the center is covered by a fabric membrane spanning between the arches. This allows light to concentrate in the center, a desirable effect for a sports arena, as well as a reduction in thermal energy consumption (Berger, 1999). Arches can also be placed on the outside to avoid the visual obstruction in the interior space, as was employed at the Wimbledon Indoor Tennis Facility (Figure 3-6).

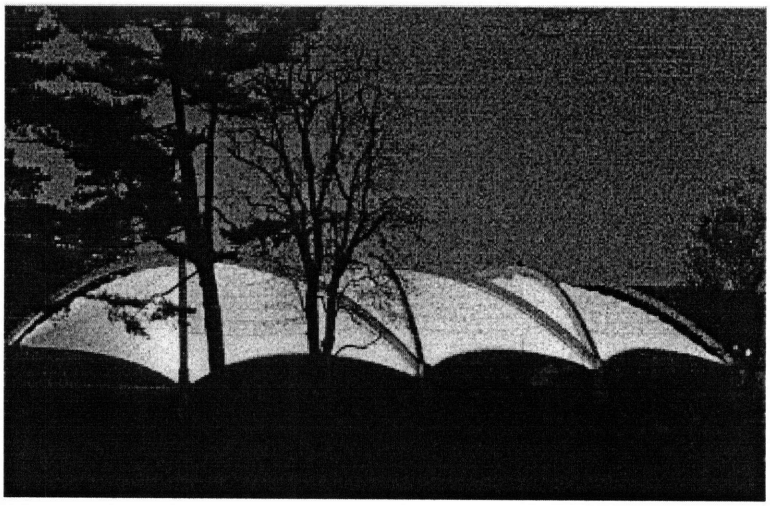

Figure 3-6: Arch-Supported Structure—Wimbledon Indoor Tennis Facility (Berger, 1999)

Chapter 4: Design Methods

Until 30 or 40 years ago, form-finding structures could only be determined through the use of physical models to ascertain their figures of equilibrium under applied loads. While these methods were adequate for simple structures, they were not precise enough for increasingly complex situations. The desire among designers arose to have a method that, unlike modeling, was not problem specific and could be generalized to a wide variety of applications and materials. Rapidly evolving computer methods, such as finite element analysis, have revolutionized the way these structures can be formulated. A few methods that have been developed and employed will be discussed.

4.1: Pucher's Equation Method

4.1.1: Formulation

In the design of membrane shell structures, an initial approximation for the optimal shape can be determined through a specialized form of the membrane force-equilibrium equations. For isotropic materials in compression, the strength is limited by the maximum principle stress. An idealized optimal membrane is assumed to carry axial forces, but can not resist bending or shear forces. An optimal shape, therefore, is one in which the material behaves in the most efficient manner by having the two principal stresses equal at every point. With the assumption that the two principal stresses are equal, the condition of zero shear force is implicitly satisfied.

The force-equilibrium equations can be reduced to one differential equation, called Pucher's equation, which is a function of the membrane stress, $F(x,y)$, and the elevation $z(x,y)$ at each point. Assuming a load per unit projected area in the positive vertical direction, $\overline{P}_z$, Pucher's equation can be expressed as

$$z_{,yy}\,F_{,xx} - 2\,z_{,xy}\,F_{,xy} + z_{,xx}\,F_{,yy} = -\overline{P}_z \qquad (4.1)$$

using subscript notation to represent partial derivatives.

In the design phase to determine the shell shape, F is specified everywhere and the elevation, z, at each point is determined. Using N_{ij} to represent stress resultants, and $\overline{N}_{ij}$ the projected stress resultants in the x-y plane, see Figure 4-1, the following definitions apply

$$\begin{aligned} \overline{N}_{xx} &= F_{,yy} \\ \overline{N}_{yy} &= F_{,xx} \\ \overline{N}_{xy} &= -F_{,xy} \end{aligned} \qquad (4.1)$$

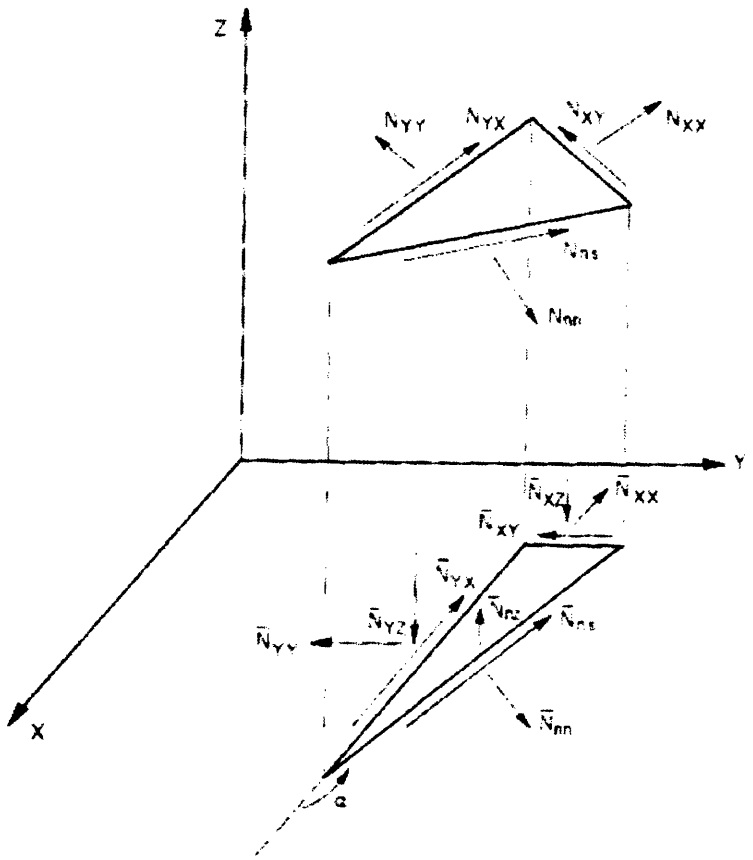

Figure 4-1: Actual and Projected Stress Resultants (Connor, Wolf, Miller, 1965)

As mentioned, to optimize the shape, the stress resultants in the two principal directions are assumed to be equal, while the shear stress is assumed to be zero. If $\overline{N}$ represents the desired stress resultant everywhere, then

$$\begin{aligned} F_{,yy} &= \overline{N}_{xx} = \overline{N} \\ F_{,xx} &= \overline{N}_{yy} = \overline{N} \\ F_{,xy} &= 0 \\ \overline{N} &= \text{specified value} = \text{constant} \end{aligned} \qquad (4.2)$$

Pucher's equation, from equation (4.1), can then be rewritten as

$$z_{,xx} + z_{,yy} = -\frac{\overline{P}_z}{\overline{N}} \qquad (4.4)$$

subjected to prescribed elevations, *z(x,y)*, at specified locations.

Equation 4.4 represents a form of Laplace's equation, which can be solved by using finite element analysis principles. Applying Galerkin's method, the equation is transformed to

$$\iint \left(\frac{\partial z}{\partial x}\frac{\partial \delta z}{\partial x} + \frac{\partial z}{\partial y}\frac{\partial \delta z}{\partial y} \right) dx\, dy = -\int_A \frac{\overline{P}_z}{\overline{N}} \delta z\, dA \qquad (4.5)$$

If triangular elements are used, as shown in Figure 4-2, a relation between Cartesian coordinates, x,y,z, and triangular coordinates, ξ_1, ξ_2, ξ_3, must be established. For an arbitrary point p(x,y) = p(ξ_1, ξ_2) from Figure 4-2 it can be written

$$\begin{aligned} x &= x_3 + (x_1 - x_3)\,\xi_1 + (x_2 - x_3)\,\xi_2 \\ y &= y_3 + (y_1 - y_3)\,\xi_1 + (y_2 - y_3)\,\xi_2 \end{aligned} \qquad (4.6)$$

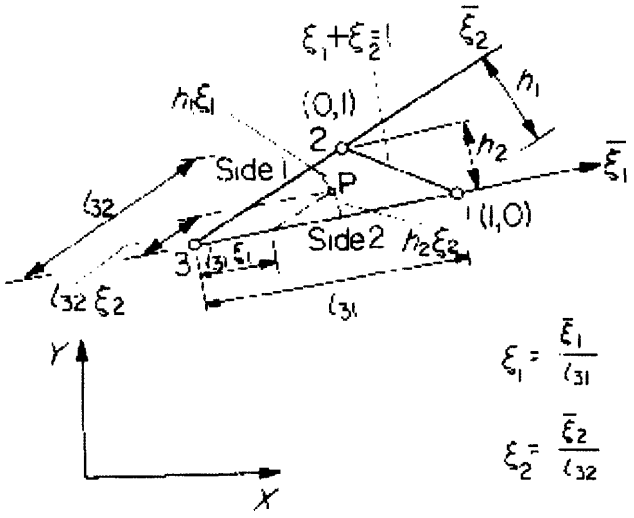

Figure 4-2 : Triangular Coordinate System

It also follows that of the three triangular area coordinates ξ_i, only two are independent such that

$$\xi_1 + \xi_2 + \xi_3 = 1 \qquad (4.7)$$

From equations 6 and 7, the triangular coordinates are written as

$$\begin{aligned}
\xi_1 &= \frac{1}{2A}\left(2A_1^0 + b_1 x + a_1 y\right) \\
\xi_2 &= \frac{1}{2A}\left(2A_2^0 + b_2 x + a_2 y\right) \qquad (4.8) \\
\xi_3 &= \frac{1}{2A}\left(2A_3^0 + b_3 x + a_3 y\right)
\end{aligned}$$

where the parameters are defined as

$$\begin{aligned}
&a_1 = x_3 - x_2 \;\; ; \;\; b_1 = y_2 - y_3 \;\; ; \;\; 2A_1^0 = x_2 y_3 - x_3 y_2 \\
&a_2 = x_1 - x_3 \;\; ; \;\; b_1 = y_3 - y_1 \;\; ; \;\; 2A_2^0 = x_3 y_1 - x_1 y_3 \qquad (4.9) \\
&a_3 = -a_1 - a_2 = x_2 - x_1 \;\; ; \;\; b_1 = -b_1 - b_2 = y_1 - y_2 \;\; ; \;\; 2A_3^0 = A - A_1^0 - A_2^0 \\
&A = \frac{1}{2}\left(b_1 a_2 - b_2 a_1\right) = \text{area of triangle}
\end{aligned}$$

Using these interpolation functions, a function for z can be proposed that varies linearly over the triangular element

$$z = \xi_1 z_1 + \xi_2 z_2 + \xi_3 z_3 = \Phi^T \mathbf{z} \qquad (4.10)$$

where

$$z_i = nodal\ values, i = 1,2,3$$
$$\mathbf{z} = [z_1\ z_2\ z_3]$$
$$\Phi = [\xi_1\ \xi_2\ \xi_3]$$

From equation 4.9, and using the chain rule, it is readily seen that

$$\begin{aligned}
&\frac{\partial z}{\partial x} = \frac{\partial z}{\partial \xi_1}\frac{\partial \xi_1}{dx} + \frac{\partial z}{\partial \xi_2}\frac{\partial \xi_2}{dx} + \frac{\partial z}{\partial \xi_3}\frac{\partial \xi_3}{dx} = \frac{1}{2A}\left(z_1 b_1 + z_2 b_2 + z_3 b_3\right) = \frac{1}{2A}\mathbf{b}^{\mathrm{T}}\mathbf{z} \\
&\frac{\partial \delta z}{\partial x} = \frac{1}{2A}\mathbf{b}^{\mathrm{T}}\delta\mathbf{z} \\
&\frac{\partial z}{\partial y} = \frac{\partial z}{\partial \xi_1}\frac{\partial \xi_1}{dy} + \frac{\partial z}{\partial \xi_2}\frac{\partial \xi_2}{dy} + \frac{\partial z}{\partial \xi_3}\frac{\partial \xi_3}{dy} = \frac{1}{2A}\left(z_1 a_1 + z_2 a_2 + z_3 a_3\right) = \frac{1}{2A}\mathbf{a}^{\mathrm{T}}\mathbf{z} \\
&\frac{\partial \delta z}{\partial y} = \frac{1}{2A}\mathbf{a}^{\mathrm{T}}\delta\mathbf{z}
\end{aligned} \qquad (4.11)$$

where

$$\mathbf{a} = [a_1\ a_2\ a_3]\,;\ \mathbf{b} = [b_1\ b_2\ b_3]$$

In order to solve the right side of equation 4.5, which from equation 4.10 reduces to

$$-\frac{\overline{P}_z}{\overline{\overline{N}}}\delta\mathbf{z}\int_A \Phi^T\, dA \qquad (4.12)$$

the relation can be used that

$$\int_A \xi_i\, dA = \frac{1}{3!}2A = \frac{A}{3} \qquad (4.13)$$

Substituting equations 4.11 and 4.13 into equation 4.5 and simplifying, Pucher's equation reduces to

$$\frac{1}{4A}\left(\mathbf{b}\mathbf{b}^T + \mathbf{a}\mathbf{a}^T\right)\mathbf{z} = -\frac{\overline{P}_z A}{3\overline{N}} \qquad (4.14)$$

Noting that equation 4.14 is of the form KU = R, the stiffness and load matrices for each element can be obtained. From these, the global stiffness and global load matrices can be formulated, and the nodal elevations, z, for the structure can be solved for. It is also important to note that the boundary conditions must be applied. In general, these are fixed nodal elevations at specified locations. To do so, the displacement, stiffness and load vector are modified. The element of the load matrix, R, corresponding to the prescribed condition is modified by subtracting the corresponding K row multiplied by the given elevation. Namely, for a prescribed condition at node i, the R term is now R $-z_{i,\text{prescribed}}$ $K_{(\text{row i})}$ Also, the row and column of the stiffness matrix, K, corresponding to each prescribed elevation is set to zero for the off diagonal elements and one for the diagonal element.

The above methods were used to generate a program in Matlab to solve Pucher's equation. The full program is given in Appendix A.

4.1.2: Design Examples

A preliminary example using the Pucher's equation method was performed to illustrate the results that can be obtained. The desired structure is rectangular in plan, with prescribed elevations along the boundaries following the equation of a parabola with the corners at elevation zero. The surface was discretized into triangular elements, shown in Figure 4-3, resulting in 60 elements and 39 nodes (note: nodes are numbered inside circles). Each node has fixed (x,y) locations, given in Figure 4-3, using a uniform two by two grid. All values of dimensions, loads, and stresses are just presented for the examination of their effects and ratios and are not based on actual measurements. Thus,

the results can be scaled by an appropriate amount to simulate a realistic situation. Of these 39 nodes, the 16 perimeter nodes have prescribed elevations, while the remaining node elevations are to be calculated. Included in Appendix B are the coordinate, interconnectivity and boundary condition matrices that were used as input for the program.

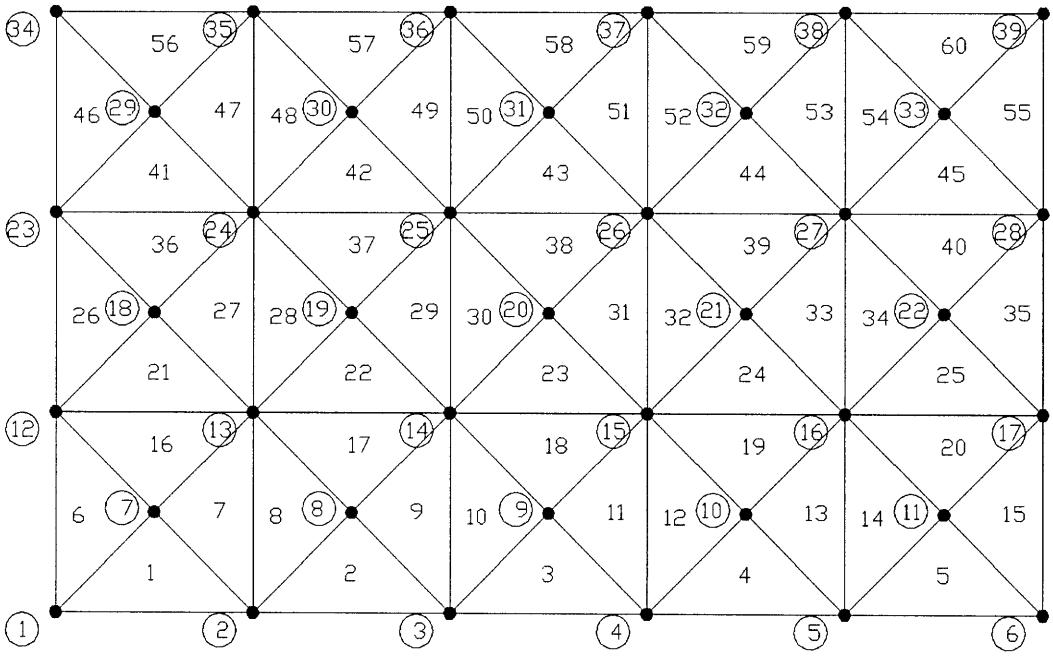

Figure 4-3: Plan of Surface with Element and Node Numbers

The surface was subjected to an arbitrary compression load of $\overline{P}_z$ = -1 and a stress resultant of $\overline{N}$ = 10. Figure 4-4 shows the resultant shape. Since the shape is defined at a set of discrete points, the surface itself is modeled as a series of triangular flat surfaces. In reality a spline function would have to be used to determine the actual curved shape to fit the calculated points, and thus the true membrane. In addition, the mesh that was used could be refined by adding more points and reducing the distances between them, which would create a smoother approximation of the curved surface. For purposes of visualization here, this refinement is unnecessary.

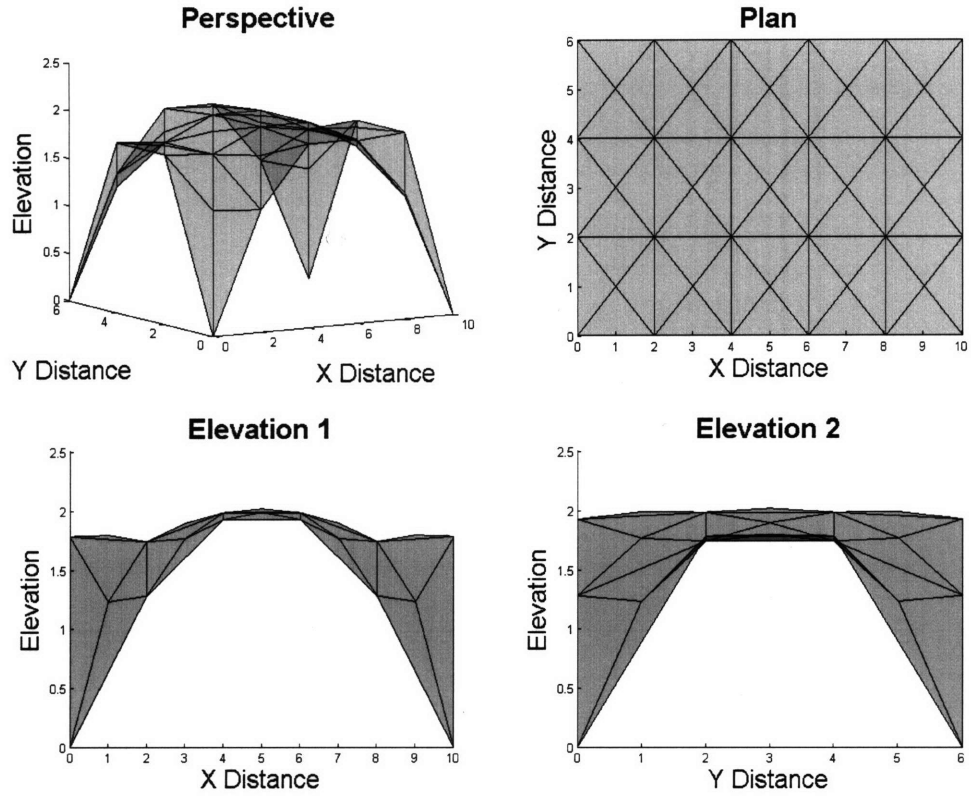

Figure 4-4: Resultant Equilibrium Shape from Pucher's Equation Method

To validate this method, the obtained geometry from Matlab was modeled in the finite element analysis program ADINA. A shell of unit thickness was modeled as a triangularly faceted surface, as previously described. The same design uniform load of -1 in the vertical direction only was applied to all surfaces. An additional outcome using this form of Pucher's equation is that the projected normal boundary force, $\overline{N}_{nn}$ in Figure 4-1, is not zero, but is actually equal to $\overline{N}$. Therefore, a force along the boundaries parallel to the xy plane equal to $\overline{N}$, in this case ten, was applied. If this is not done, the ADINA solution will not be as expected. In addition, constraints were placed along the boundary lines to fix motion in the vertical direction, while allowing the other degrees of freedom to remain unrestrained.

The results to be examined are the principal stresses. As mentioned, the two in plane principal stresses are expected to be equal and have the value of the prescribed stress resultant $\overline{N}$, equaling ten in this situation since there is a unit thickness. Since the

structure is under compression, the values given by ADINA will be negative. In addition, since P1>P2>P3, it is expected that P1 will equal zero everywhere, while P2 = P3 = -10.

The structure was modeled with varying degrees of mesh refinement by altering the number of subdivisions along each line in order to achieve a more accurate solution. Figure 4-6 graphically illustrates the two principal stresses obtained from a very refined mesh. While the expected result is a uniform value across the entire surface, the band plots clearly show variation. However, closer inspection of the legend indicates that the range is quite small, though it is difficult to see just how large this disparity is. To more clearly understand the variance in stresses and how far they are from the expected value, sections were cut through the surface in two directions, shown in Figure 4-6 by lines A and B. The stress values as a function of position along the two lines were exported from ADINA and plotted in Figure 4-6. The sections verify that the stress values, P2 and P3, are actually each quite close to the expected value of -10. In addition, P1 is zero as expected.

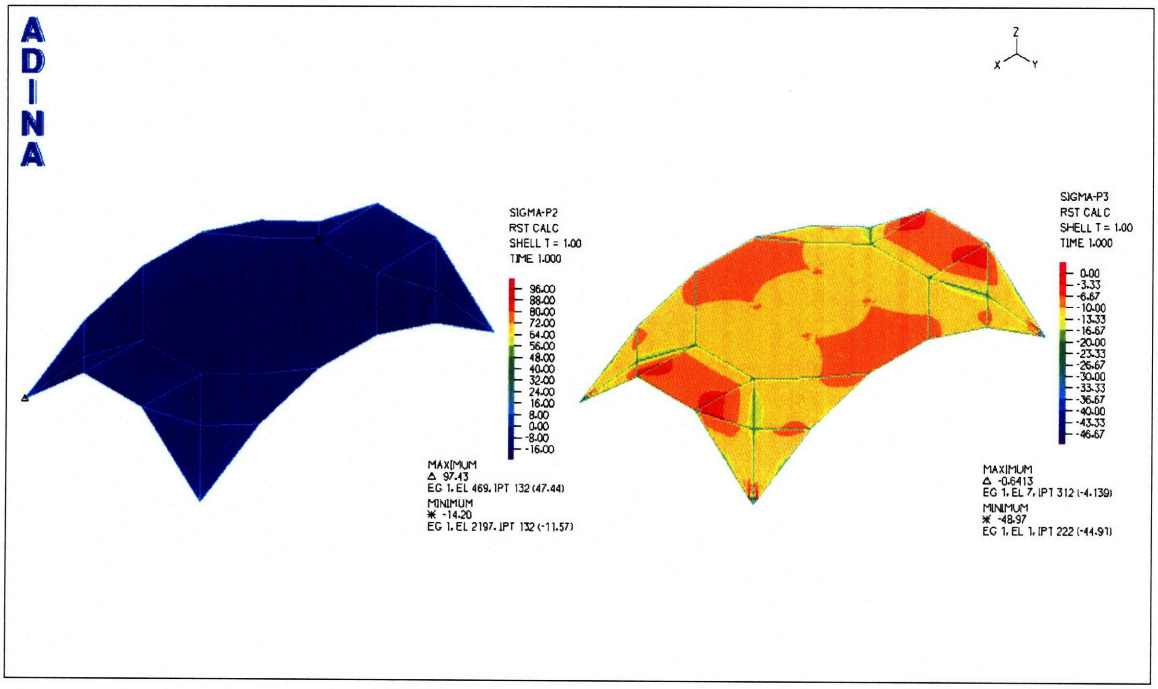

Figure 4-5: Isometric view of the two principal stresses

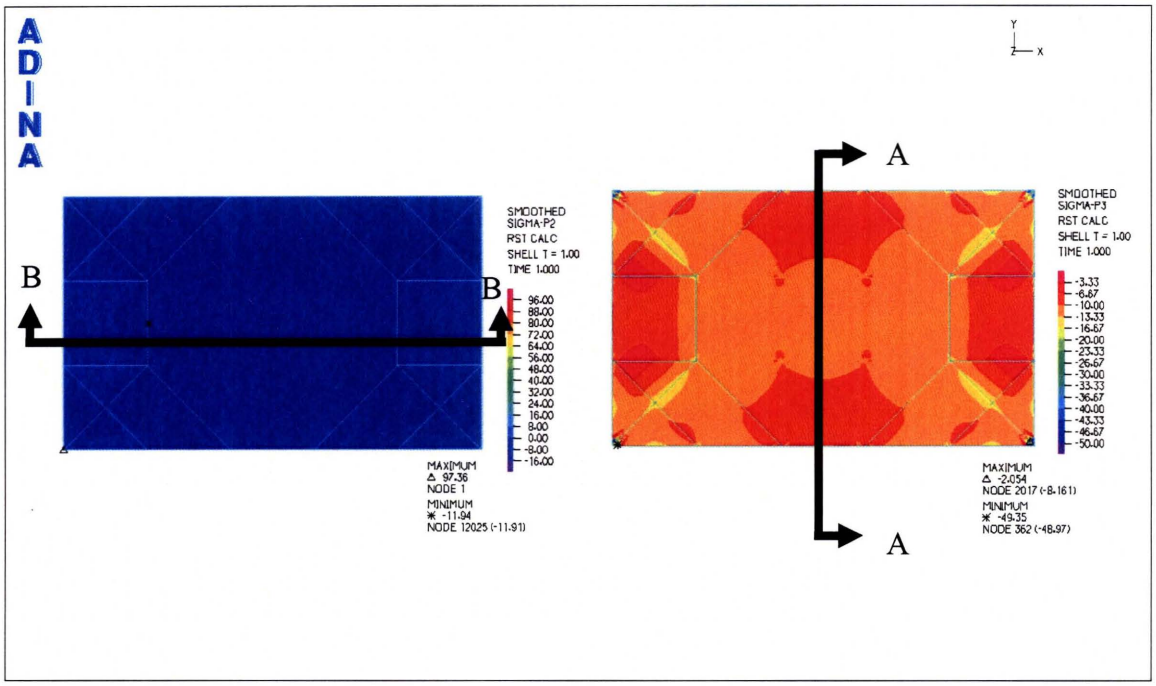

Figure 4-6: Plan view of the two principal stresses

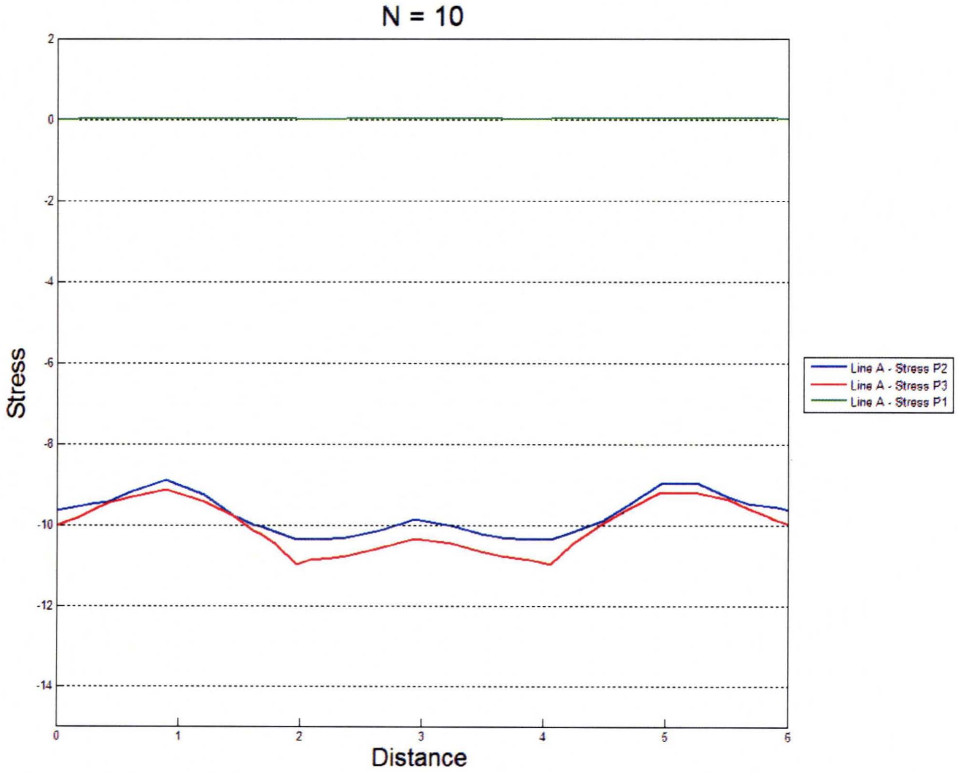

Figure 4-7: Stresses along section line A

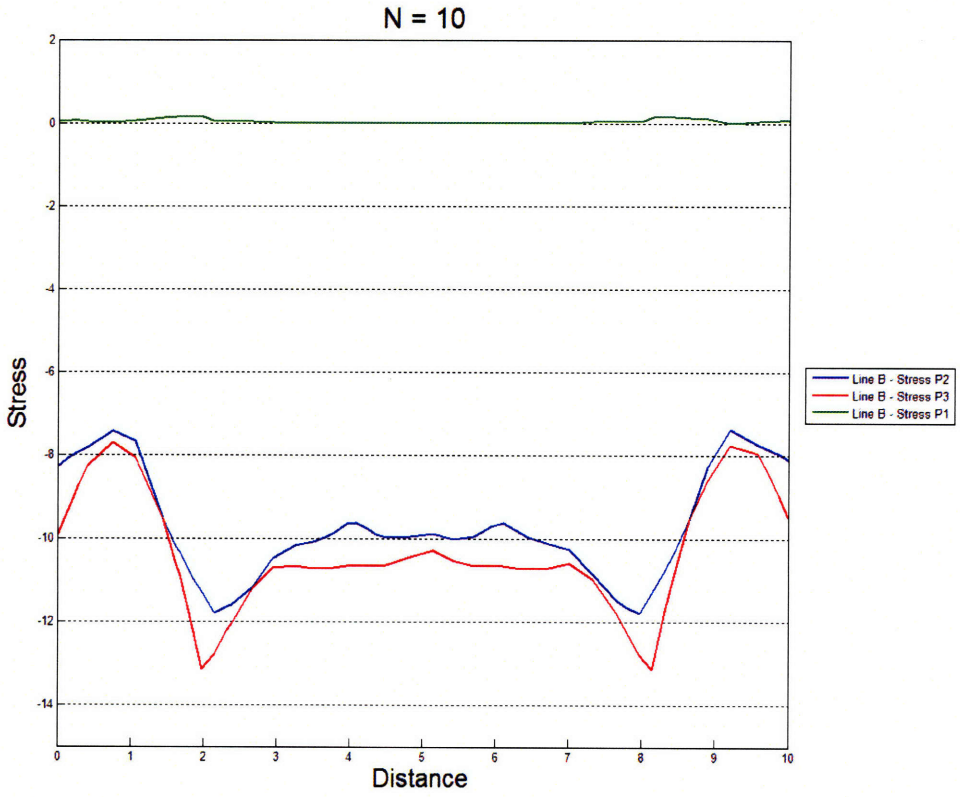

Figure 4-8: Stresses along section line B

Overall, the results clearly yield principal stresses that are approximately -10, as expected. However, it is obvious that the results are not uniform at specific locations, namely where the structure changes curvature, such as at a distance of two and eight along Line B. This is justified, though, since the surface is not modeled as curved, as is the actual case. Instead, the surface is approximated by breaking it into triangular flat panels, leading to slight errors in the results. Aside from this, the values only vary by small amounts, and can be considered uniform for most practical engineering purposes. This validates the initial shell shape that was determined through Pucher's equation.

Effect of Load

The shape of the structure greatly depends upon the applied load. Figure 4-9 shows the structure under three different uniform surface loads, clearly seen in the elevation drawings. For the three cases the stress resultant $\overline{N}$ is equal to one. Again,

these loads do not represent actual values but are used to show the shape under different ratios of load to stress resultant, length, etc. The highest peak is that formed under an applied downward vertical load of two, the middle under a downward load of one, and the shallowest under a downward load of one-half. Clearly, as the load down onto the structure increases the structure must increase in height and have larger vertical components of the member forces to resist these loads.

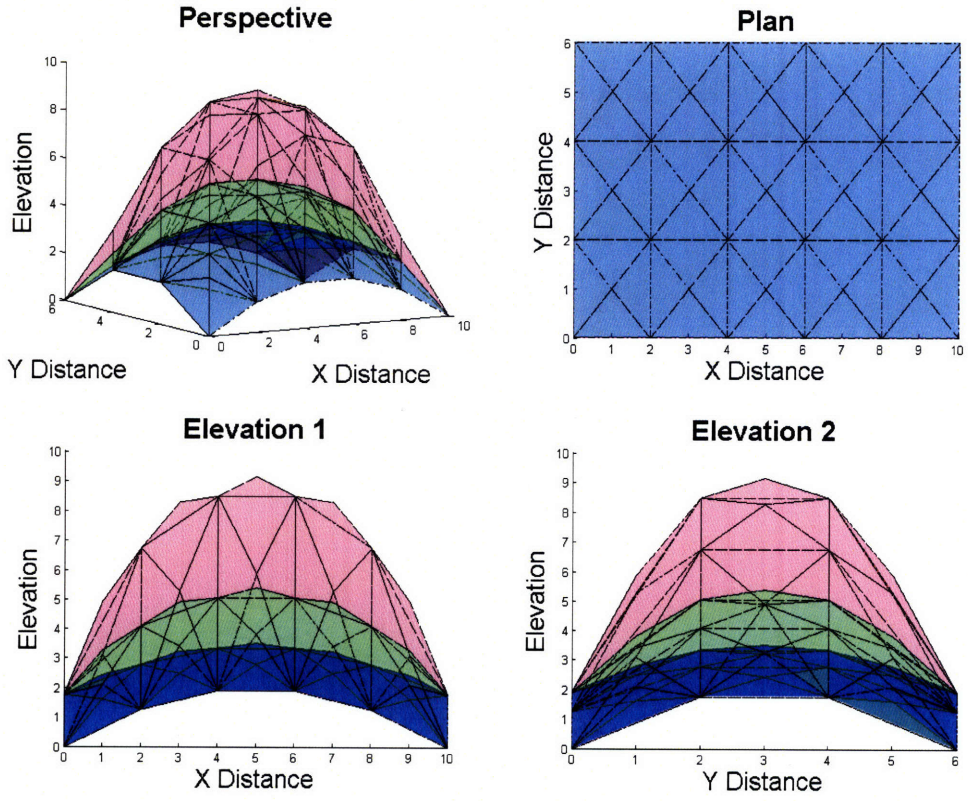

Figure 4-9: Effect of load. P = -1/2 (blue), -1 (green), -2 (red)

Effect of Stress Resultant

Figure 4-10 through Figure 4-12 illustrate different combinations of prescribed forces and stress resultants, and their influence on the shape. The first example presented using this method, shown in Figure 4-4, gave an example with a compressive load. Figure 4-10 shows the opposite condition of an upward tensile load onto the structure,

maintaining the same stress resultant of 10. Clearly the structure now dips in more to offset the upward load. However, if the stress resultant is decreased to one, as is the case in Figure 4-11, it is obvious that this stress resultant is not large enough and the structure has snapped through to compensate for the upward load. If instead a downward load is used, the mirror image of this situation is formed, given in Figure 4-12. This can be directly compared to the case previously shown, with a downward load of one and a stress resultant of ten, shown again in Figure 4-13. Clearly this structure is reduced in height since the desired stress resultant is to be higher. The lower desired stress resultant leads to a higher, more rounded shape.

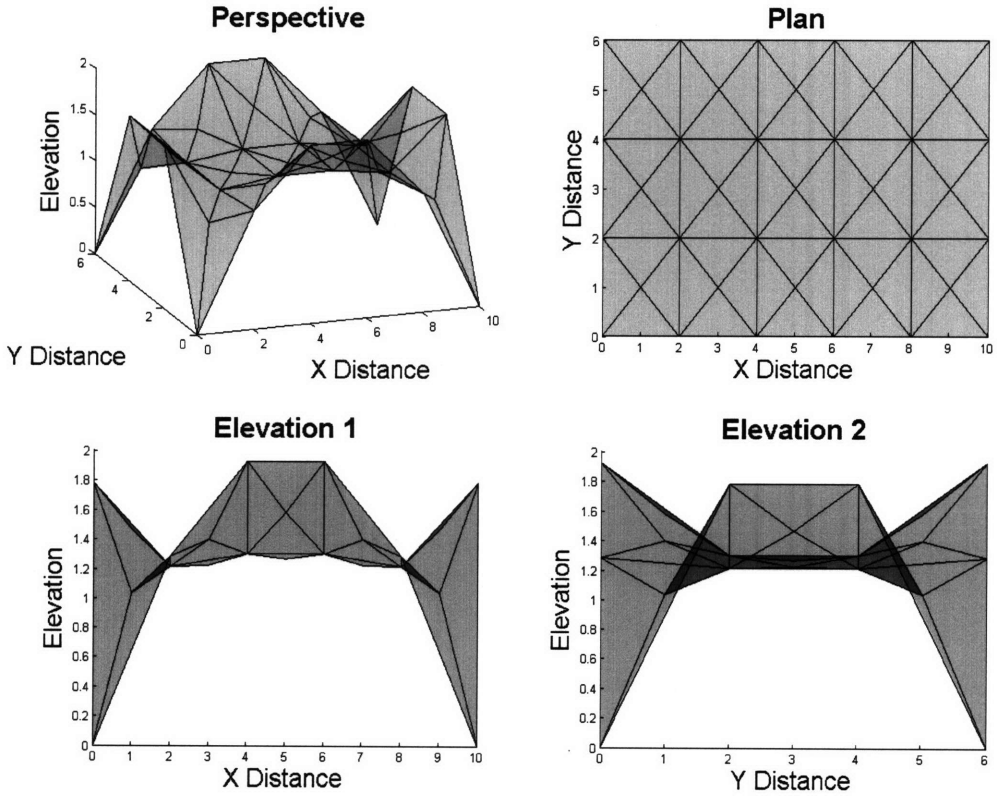

Figure 4-10: Effect of stress resultant. P = 1, N = 10. Ratio = 0.1

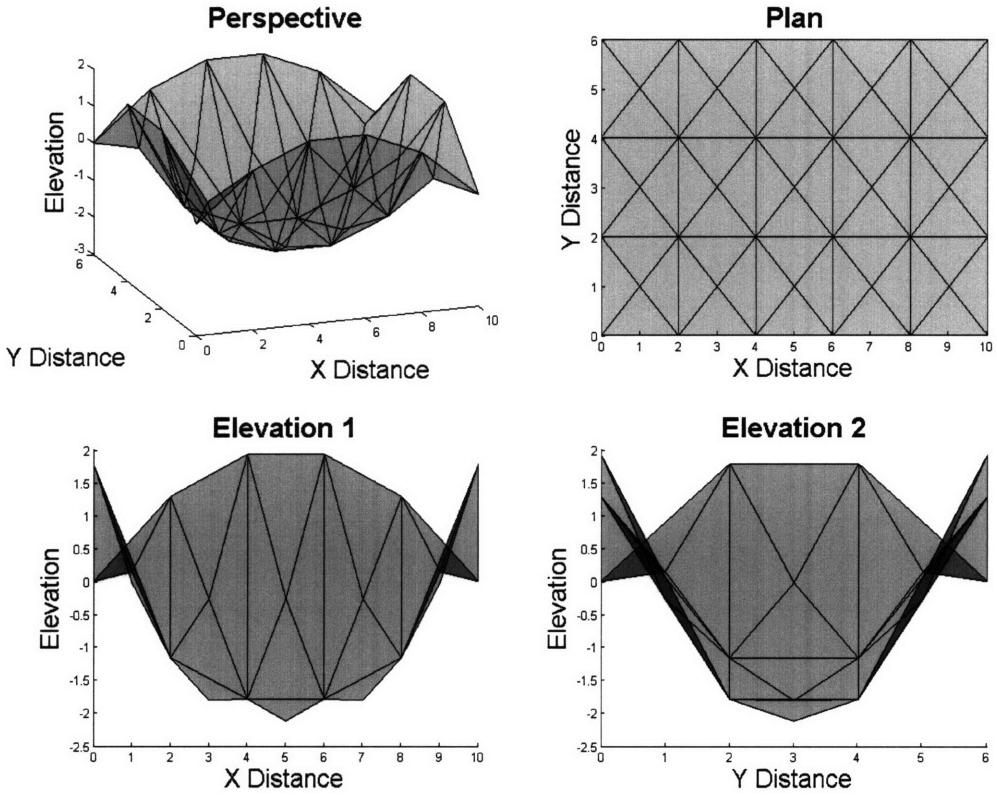

Figure 4-11: Effect of stress resultant. P = 1, N = 1. Ratio = 1

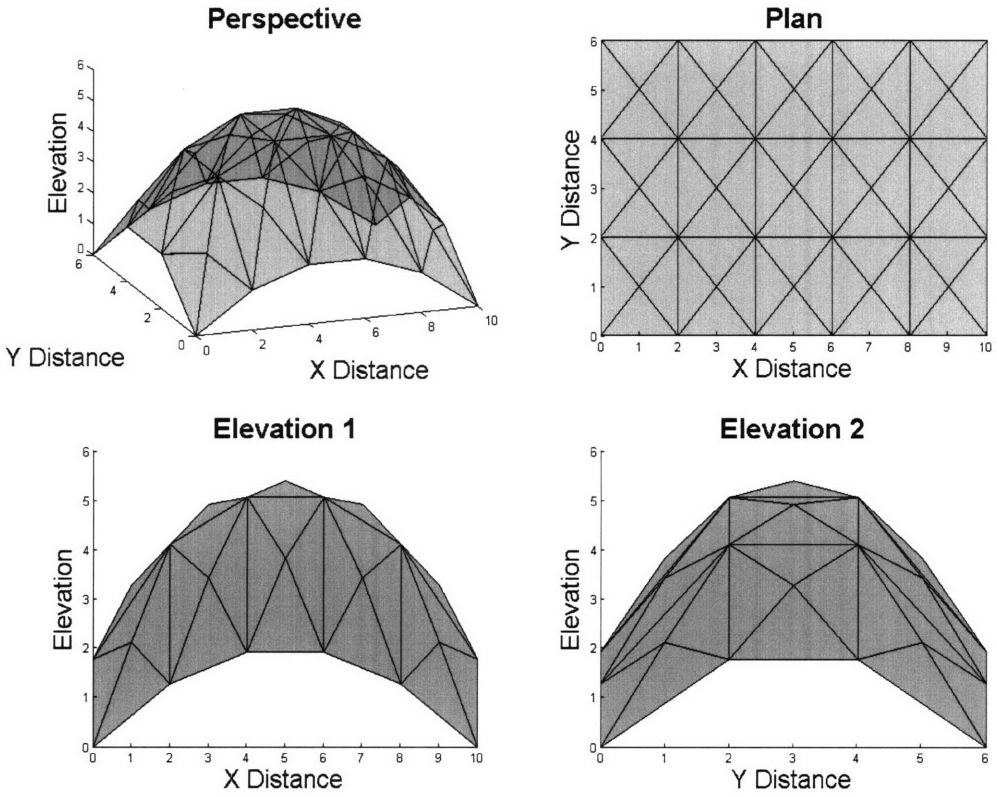

Figure 4-12: Effect of stress resultant. P = -1, N = 1. Ratio = -1

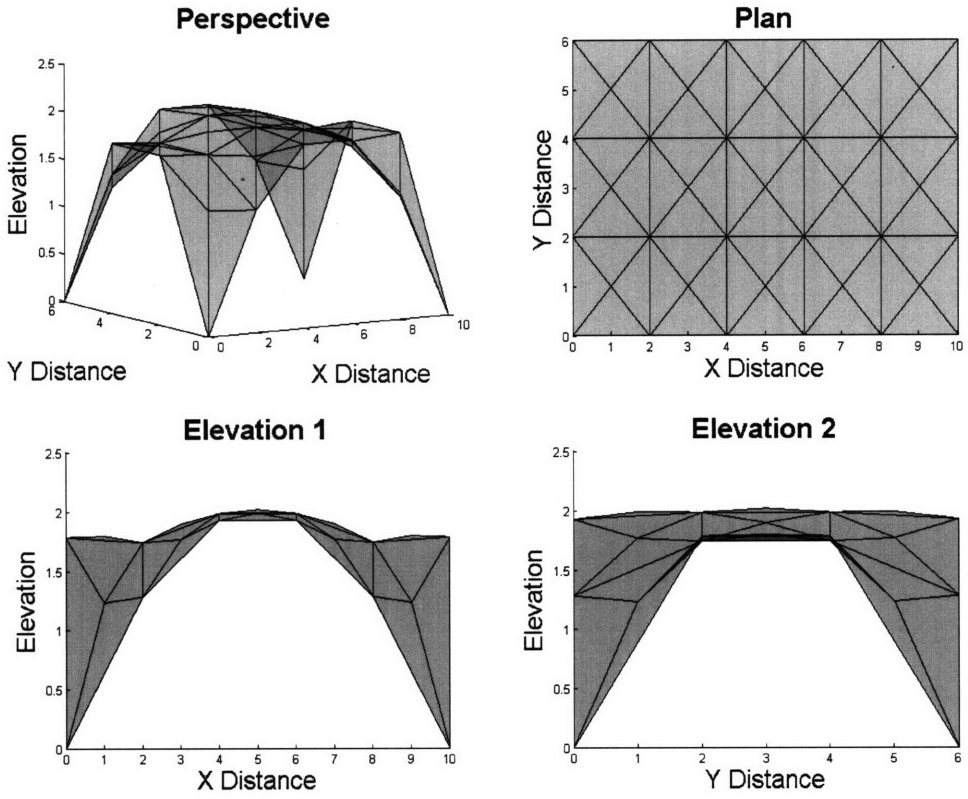

Figure 4-13: Effect of stress resultant. P = -1, N = 10. Ratio = -0.1

4.1.3: Advantages

Pucher's equation lends itself to simple and quick visualization of a curved surface under specified conditions. For most situations in which this would be applied, such as roof structures, the applied load is uniform and vertical as required. In addition, there is very minimal input needed. The only requirement by the user is the discretization of the surface, numbering of the elements and nodes, and formulation of the interconnectivity matrix indicating which nodes correspond to which elements. The accuracy of the surface is dependent on the refinement of the grid, thus it is up to the user to determine how many elements are desired and how much input they would like to use.

4.1.4: Limitations

As mentioned, the normal boundary force, $\overline{N}_{nn,}$ is not zero but is actually equal to $\overline{N}$. Thus, in practical applications this boundary force would have to be applied to mimic the desired behavior. In addition, the form of Pucher's equation described here only considers a uniform load that is purely in the vertical direction. However, there are many situations that require variable loading, or loads that are in different directions, which cannot be accounted for with this method. Furthermore, the prescribed stress resultant, $\overline{N}$, must also be uniform everywhere, a condition that is not always desired. Finally, the x and y coordinates of each node must be known. This implies that something about the final shape must also be known prior to design. While this is not an issue for simple structures, for complex shapes knowing where the non-fixed nodes should be placed in the x and y directions is harder to ascertain if certain conditions, such as equidistant final node location, are desired. Aside from vertical adjustment to satisfy equilibrium, this method does nothing to allow for the modification of the nodal locations to meet other constraints such as desired bar length or actual forces in the members.

4.2: Force-Density Method

4.2.1: Background

The force density method (FDM), first developed by Klaus Linkwitz and H.-J. Sheck in 1971, was motivated by the desire to analytically solve for the shape of a doubly curved surface simply and with minimal input. This method is based on the force-to-length ratio, called the force-density, of each branch of the net structure. The branches are physically either the cables or rods that make up the structure, and are schematically drawn as the lines that connect node points. The result is a simple, single system of linear equations, with the only requirement that the shape must be in equilibrium, implying that the sum of all forces at each node is zero.

The development of this method was based on the physical experiments on a free hanging membrane performed by Heinz Isler, as previously described. This idealized membrane is assumed to be inelastic and transmit tensile forces, but has no resistance to bending or shear stresses. While it has a fixed area, the membrane has no fixed shape, and its geometry is allowed to change based on the applied forces. Again using Mohr's circle, the two principal stresses must be equal to eliminate shear forces. As a result, by requiring the principal stresses at every point equal in all directions and using membrane theory, the shape that a free hanging membrane would take can be found. While the principal stresses must be uniform in all directions for each point, this value does not have to remain constant over the surface.

To achieve this condition of uniform stress at each point, the FDM prescribes a value of the force-density for each branch of the structure. With no other constraints, the equilibrium structure having those prescribed force-densities can be determined.

4.2.2: Formulation

The formulation of the FDM described below is based off of the methods initially presented by Schek. To start, an arrangement of a pin-jointed network of branches, numbered 1 to m, and nodes, numbered 1 to n, is formed in the 2D plane, called topology mapping. This topology graph that is formed only indicates the connectivity between nodes, and thus the same graph can be used for many equilibrium shapes depending on the applied loads, as explained by Hernandez-Montes, Jurado-Pina, and Bayo (2006). These authors have also examined different methods for topological mapping based on the desired network and final configuration. However, this is beyond the scope of the present discussion, and in fact has little impact on the initial visualization. It is therefore assumed that the structure is mapped somewhat arbitrarily, but includes enough branches to adequately define the surface. It is also helpful for later equation formulation that the nodes are numbered such that the free nodes are first, numbered 1 through nfree, while the fixed nodes are at the end of the numbering sequence, nfree+1 to n, totaling nfixed.

Each branch, j, is connected to two nodes, numbered i(j) and k(j). Requiring i < k, meaning node number i is the smaller of the two connecting nodes, the branch-node matrix, C_s is an m x n matrix defined by

$$C_s(j,r) = \begin{cases} +1 & \text{for i(j)} = \text{r} \\ -1 & \text{for k(j)} = \text{r} \\ 0 & \text{otherwise} \end{cases} \qquad (4.15)$$

This overall branch-node connectivity matrix is really a combination of the branch-node matrix for the free nodes, C, which are the columns 1 through nfree, and for the fixed nodes C_f, columns nfree+1 through n, in the C_s matrix.

The nodal coordinates of all points are grouped in the size nfree vectors ***x, y, z***, for the free nodes, and the size nfixed vectors x_f, y_f, z_f, for the fixed nodes, containing the x,y, and z coordinates of each node. The size m vectors ***l*** and ***s*** contain the length l_j of each branch j, and the branch forces s_j, respectively. The nodal loads are formed by the ***p*** matrix of size nfree x 3, containing the vectors ***px, py, pz***, representing the respective component of the applied load at each node.

The coordinate differences, represented by ***u, v, w*** of the nodes that are connected by branches can be obtained through the branch-node matrix as

$$\begin{aligned} \boldsymbol{u} &= \boldsymbol{C}\boldsymbol{x} + \boldsymbol{C}_f \boldsymbol{x}_f \\ \boldsymbol{v} &= \boldsymbol{C}\boldsymbol{y} + \boldsymbol{C}_f \boldsymbol{y}_f \\ \boldsymbol{w} &= \boldsymbol{C}\boldsymbol{z} + \boldsymbol{C}_f \boldsymbol{z}_f \end{aligned} \qquad (4.16)$$

With **U, V, W,** and **L** representing the diagonal matrixes of ***u, v, w,*** and ***l,*** respectively, imposing the constraint of equilibrium, in other words summing the forces of each node to be zero, results in

$$\begin{aligned} \boldsymbol{C}^T\boldsymbol{U}\boldsymbol{L}^{-1}\boldsymbol{s} &= \boldsymbol{p}_x \\ \boldsymbol{C}^T\boldsymbol{V}\boldsymbol{L}^{-1}\boldsymbol{s} &= \boldsymbol{p}_y \\ \boldsymbol{C}^T\boldsymbol{W}\boldsymbol{L}^{-1}\boldsymbol{s} &= \boldsymbol{p}_z \end{aligned} \qquad (4.17)$$

Defining **q** as the m sized vector containing the force-density ratios $q_j = s_j/l_j$ of each branch, for the equations in 4.17 to be linear, it is assumed, that

$$\boldsymbol{q} = \boldsymbol{L}^{-1}\boldsymbol{s} \qquad (4.18)$$

With **Q** defined as the diagonal matrix of $\boldsymbol{q}$, the following identities can be written

$$\begin{aligned} \boldsymbol{U}\boldsymbol{q} &= \boldsymbol{Q}\boldsymbol{u} \\ \boldsymbol{V}\boldsymbol{q} &= \boldsymbol{Q}\boldsymbol{v} \\ \boldsymbol{W}\boldsymbol{q} &= \boldsymbol{Q}\boldsymbol{w} \end{aligned} \qquad (4.19)$$

Combining equations 4.17, 4.18, and 4.19 with equation 4.16, the equilibrium equations can be rewritten as

$$\begin{aligned} \boldsymbol{C}^T\boldsymbol{Q}\boldsymbol{C}\boldsymbol{x} + \boldsymbol{C}^T\boldsymbol{Q}\boldsymbol{C}_f\boldsymbol{x}_f &= \boldsymbol{p}_x \\ \boldsymbol{C}^T\boldsymbol{Q}\boldsymbol{C}\boldsymbol{y} + \boldsymbol{C}^T\boldsymbol{Q}\boldsymbol{C}_f\boldsymbol{y}_f &= \boldsymbol{p}_y \\ \boldsymbol{C}^T\boldsymbol{Q}\boldsymbol{C}\boldsymbol{z} + \boldsymbol{C}^T\boldsymbol{Q}\boldsymbol{C}_f\boldsymbol{z}_f &= \boldsymbol{p}_z \end{aligned} \qquad (4.20)$$

Defining $\boldsymbol{D} = \boldsymbol{C}^T\boldsymbol{Q}\boldsymbol{C}$ and $\boldsymbol{D}_f = \boldsymbol{C}^T\boldsymbol{Q}\boldsymbol{C}_f$, for simplicity, the final equilibrium shape for a given $\boldsymbol{q}$ matrix is

$$\begin{aligned} \boldsymbol{x} &= \boldsymbol{D}^{-1}\left(\boldsymbol{p}_x - \boldsymbol{D}_f\boldsymbol{x}_f\right) \\ \boldsymbol{y} &= \boldsymbol{D}^{-1}\left(\boldsymbol{p}_y - \boldsymbol{D}_f\boldsymbol{y}_f\right) \\ \boldsymbol{z} &= \boldsymbol{D}^{-1}\left(\boldsymbol{p}_z - \boldsymbol{D}_f\boldsymbol{z}_f\right) \end{aligned} \qquad (4.21)$$

$\boldsymbol{L}$ can then be found from the nodal coordinates given by equation 4.21

$$l_j = \sqrt{\left(u_j\right)^2 + \left(v_j\right)^2 + \left(w_j\right)^2} \qquad (4.22)$$

thus giving the forces in each branch as

$$\boldsymbol{s} = \boldsymbol{L}\,\boldsymbol{q} \qquad (4.23)$$

The above equations, given the interconnection, load, fixed nodes and force-densities, are enough to solve for the complete set of free node coordinates and branch forces. These equations were used to generate a program in Matlab, given in full in Appendix A. It is also clear that the shape changes if the force-densities, q, change. While it initially may seem that prescribing the force-densities prior to shape determination is difficult, it has been shown through many investigations that trivial force densities are adequate to generate figures of equilibrium. In most cases a value of one assigned for each branch inside the network and a value inversely proportional to the branch length in the boundary is sufficient (Linkwitz, 1999).

4.2.3: Modified Force-Density Method

While the previous method described the formation of an equilibrium state without consideration of materials or other geometrical requirements, in practical situations these factors are often significant. Additional conditions are, in general, nonlinear, thus the modified force density method will also be nonlinear. However, if the constraints can be formed as linear functions with respect to the force densities and coordinates, the above formulation can be used as the starting shape for iterations on the non-linear behavior. Without going into great detail here, some of the constraints that can be prescribed are described by Schek in "The Force Density Method for Form Finding and Computation of General Networks." These include prescribed node distances in specified branches to account for desired bar or cable lengths, prescribed forces in specified branches and prescribed unstrained lengths of the branches. Various factors of the project and desired structure dictate which conditions to use and where these conditions are applicable. As previously mentioned, these nonlinear additional

conditions will not be fulfilled by the initial linear equilibrium shape. Therefore, iterations must be performed until the shape is satisfactory.

4.2.4: Advantages

There are incredible advantages to using the force-density method to form the shape of a doubly-curved surface. For initial visualization, the only parameters to be defined are the force densities of all branches in the net, the coordinates of any fixed nodes and the connectivity of the structure, which simply specifies the nodes that each branch is desired to span between. The result is a system of linear equations which can be solved for, without iterations, in one step. Further, the equations for this initial shape do not contain anything relating to the materials used in the structure, which would only be considered if other geometrical constraints on the structure are required. This completely separates the first step, which involves pure shape determination based on the force-densities coming from prestress in the structure, from the second step, which involves material considerations and the behavior under applied loads. Furthermore, aside from the fixed nodes, this method eliminates the need to specify any initial coordinates. In addition, there is no limitation on the direction of applied loading it can handle, since the system of linear equations involves equilibrium in the directions of x,y, and z.

4.2.5: Limitations

For most structures to which this method would be employed, the nature of the applied loading is a uniform vertical surface load. However, the formulation of the force-density method requires the load to be specified as a point load at the nodal locations. Since the shape of the structure is obviously unknown, and the nodes are free to move in the x, y, and z directions, it is impossible to calculate the tributary area per node ahead of shape determination. Therefore, if an accurate result is required there will have to be some initial estimation of the amount of load taken by each node. When the initial shape is determined, the actual tributary area, and thus tributary load to each node, can be

found. The nodal loads can then be adjusted as such, and a more accurate shape determined. This can be iterated on until the user determines that the loads are acceptable representations of the true surface area load. This is primarily an issue for complex shapes, where it is difficult to initially estimate the distribution of load.

4.2.6: Design Examples

The same structure that was previously designed with Pucher's equation method in section 4.1.2: is designed using the force-density method. The same topology graph as before is to be used. However, in the FDM the branches, as opposed to the triangular elements, must be numbered. In addition, for ease programming the nodes must be renumbered so that the free nodes in the interior are labeled first and the fixed, perimeter nodes are last. As shown in Figure 4-14, there are 39 nodes and 98 branches. The same fixed nodes were used as in section 4.1.2. A downward vertical load of $p_z = 1$ was applied to each node. Note that a positive value of load indicates down for this method, whereas the opposite is true in Pucher's equation. The force-density was set to one for all of the interior branches, and 100 for the perimeter. This helps to ensure that the edges will be straight. Included in Appendix B are the input matrices of connectivity, fixed nodes, force-density and load. Figure 4-15 shows the resultant shape.

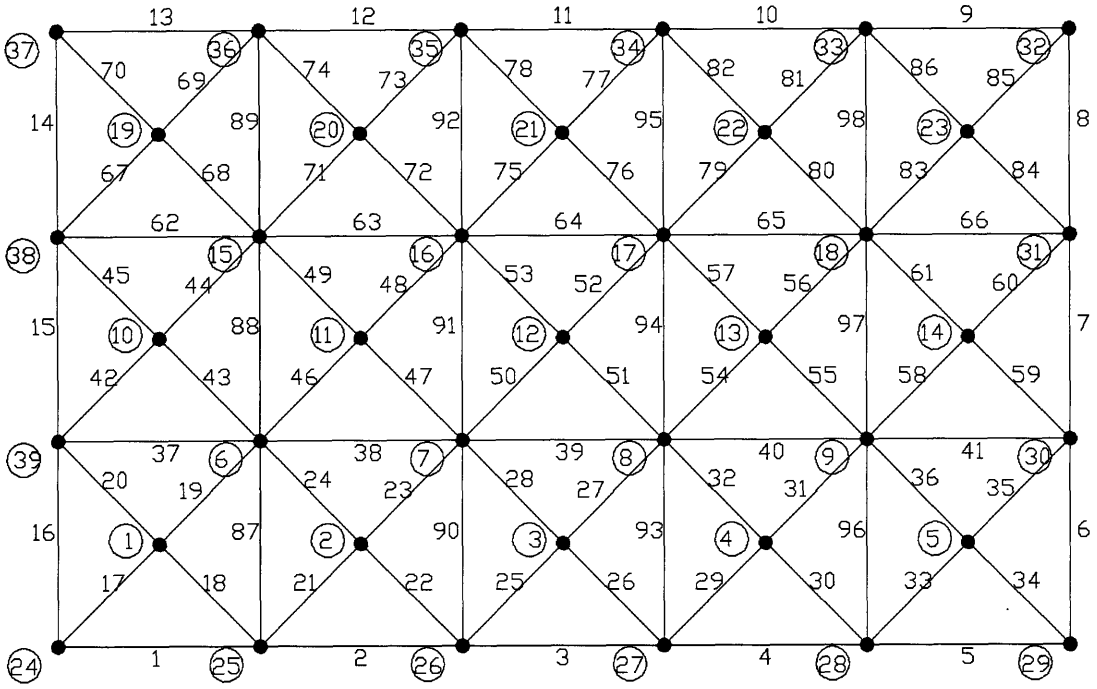

Figure 4-14: Plan of Surface with Element and Node Numbers (Topology Graph)

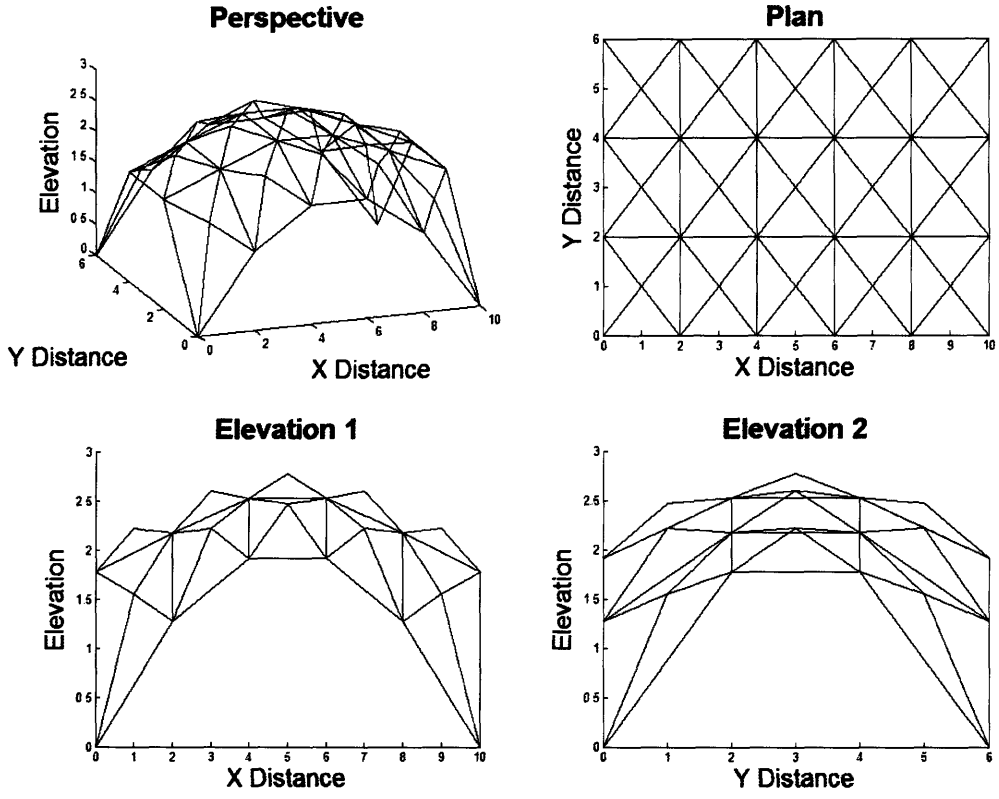

Figure 4-15: Resultant equilibrium shape

Effect of Load

To examine the influence of the load on the shape the same input as previously described was used, with the ***p*** matrix modified. As stated, Figure 4-15 gives the shape under compressive loading with nodal loads equaling one in the vertical direction (down). Figure 4-16 gives the reverse shape, that of the structure under tensile loading of negative one (up). Finally, Figure 4-17 shows the pure equilibrium shape under no applied loads.

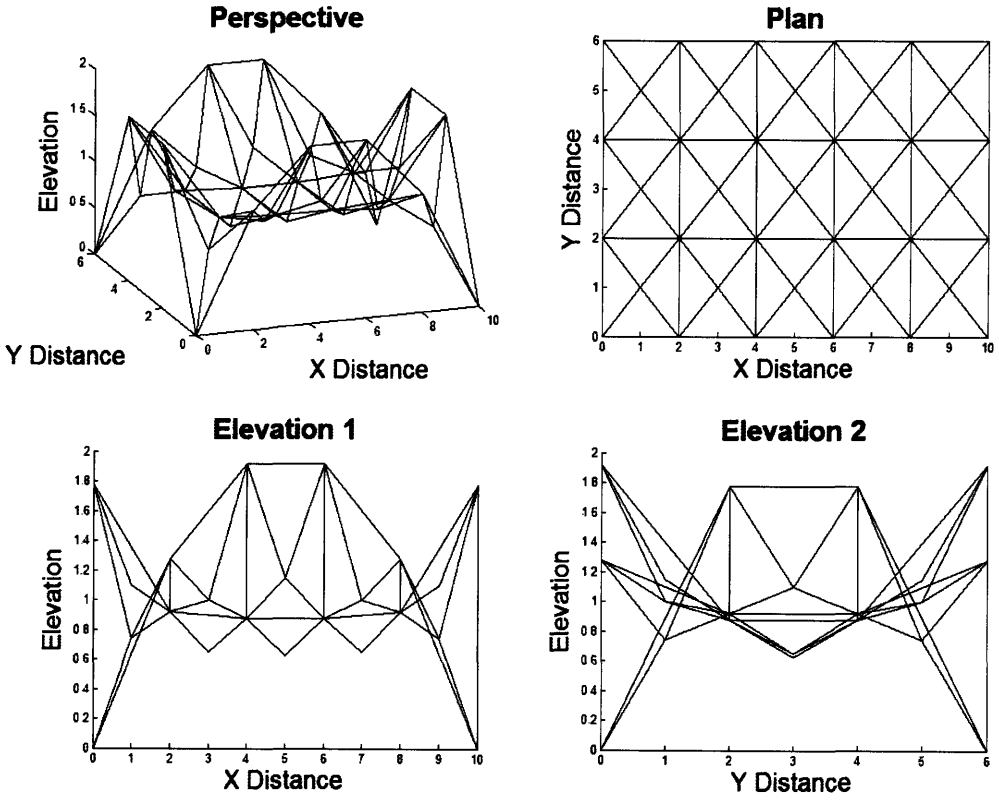

Figure 4-16: Effect of Load. Pz = -1 (up)

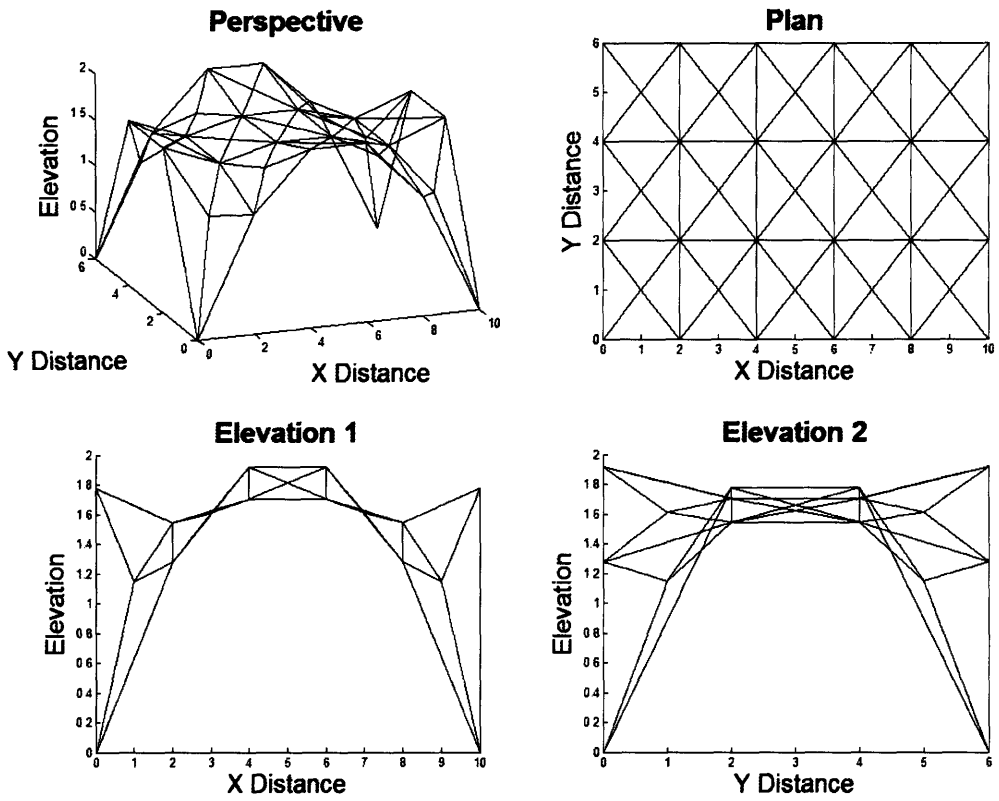

Figure 4-17: Effect of Load. Pz = 0

Effect of Force-Densities

Since the FDM is based around prescribed force-densities, it is important to understand what effect the modification of these values has on the structure so that desired behavior can be obtained. To illustrate, the structure used previously is modified. While the nodes and branches remain the same, the fixed points are changed so that only the corners have prescribed conditions. They are set so that two opposite corners have the same prescribed height, while the other two are on the ground with a height of zero. This action can be likened to propping up two corners of a tent. It is important to note that for the program to run, the nodes must be renumbered so that the four corner nodes are labeled last. This also means that the connectivity matrix must be reformulated. The

designers must therefore decide ahead of time which nodes they may want fixed so that the numbering is done in a way that requires the least amount of changes.

The ratio of the perimeter to the interior force-densities is examined in Figure 4-18, Figure 4-19, and Figure 4-20. As previously mentioned, a force-density of 100 for the perimeter branches yields fairly straight borders. As the desired force-density is decreased in the perimeter branches, the branches themselves increase in length to achieve this result, and the perimeter curves. This is why, for the structure in Figure 4-15, the force-density was set to 100 for the perimeter and 1 for the interior to obtain the desired result of straight borders in plan.

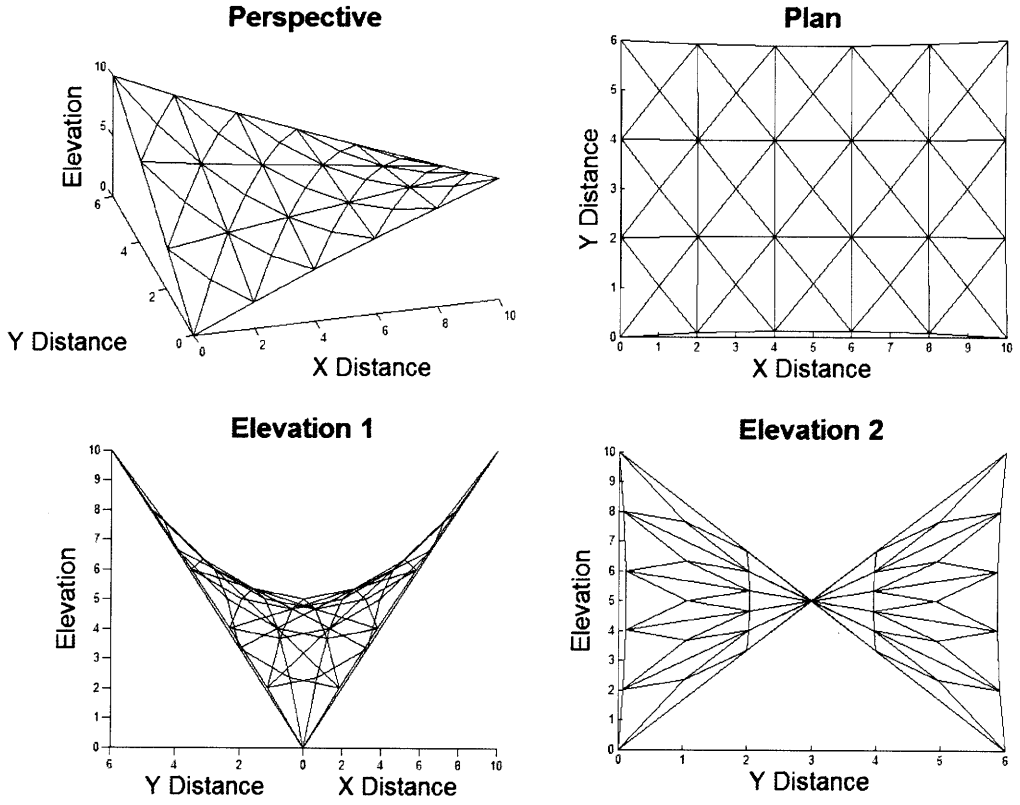

Figure 4-18: Effect of force-density. Perimeter q = 100, interior q =1

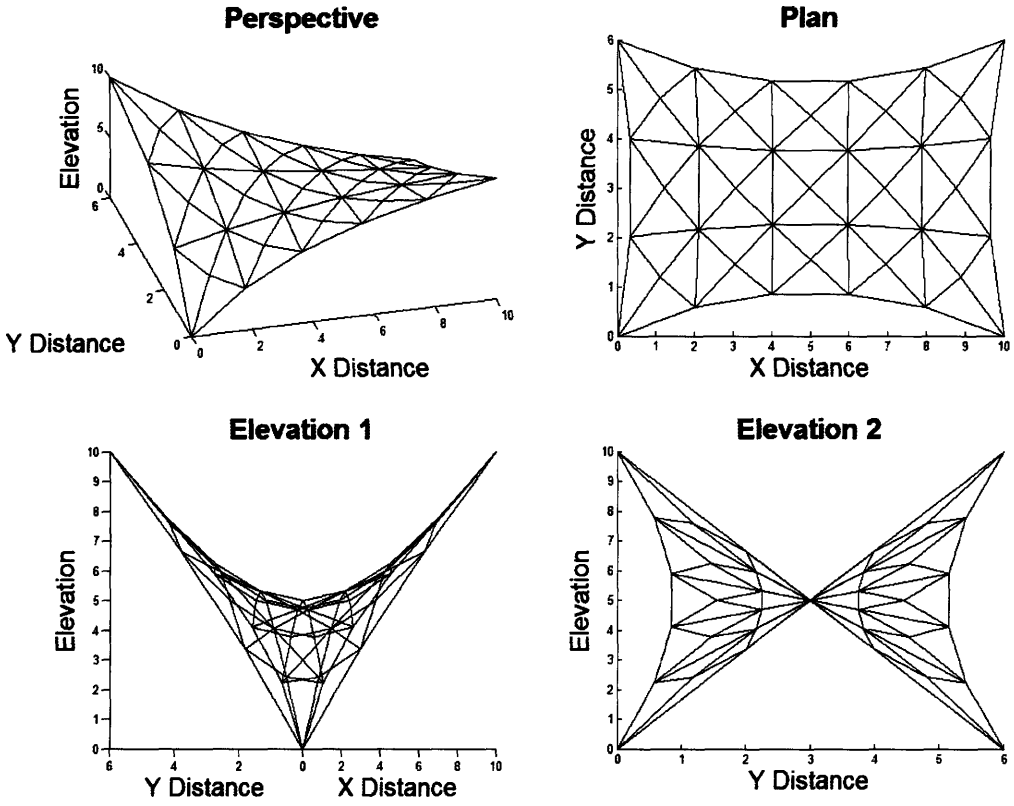

Figure 4-19: Effect of force-density. Perimeter q = 10, interior q =1

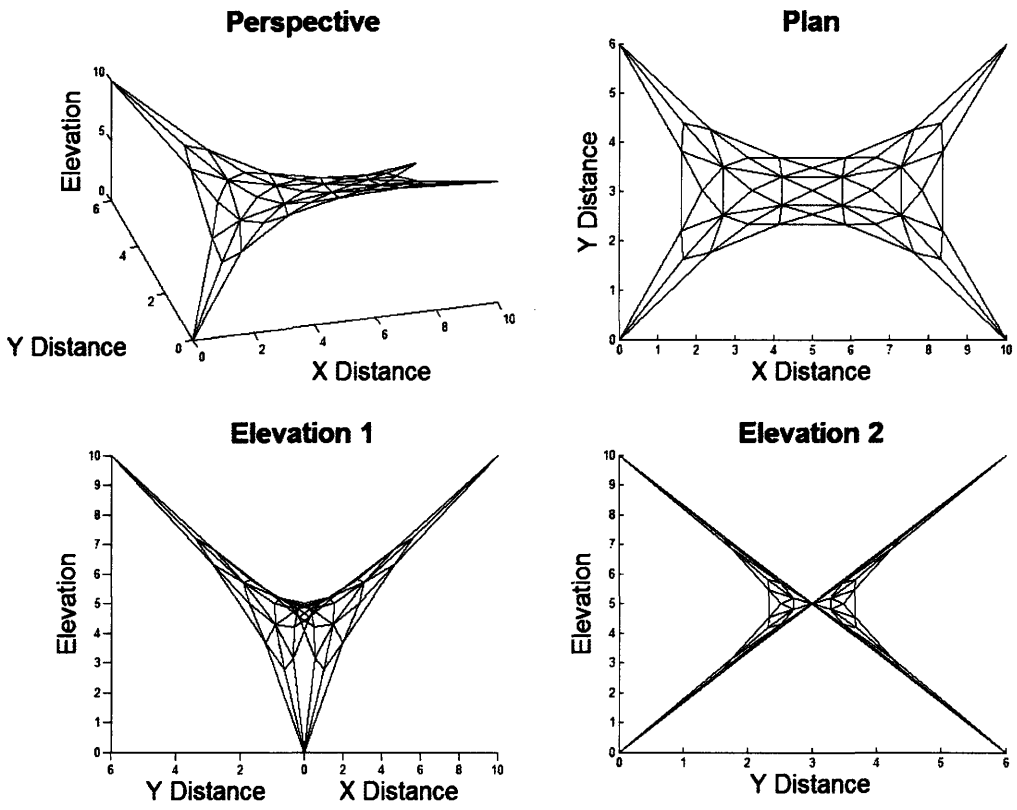

Figure 4-20: Effect of force-density. Perimeter q = 1, interior q =1

Effect of Fixed Nodes

The same structure has already been looked at with two different fixed node conditions. Specifically, in Figure 4-15 the borders were given the shape of parabolic openings, while in Figure 4-18 the corners were given set heights for openings. Both of these involved fixed nodes at the edge of the structure. Another desired condition might be to employ prescribed nodes in the center, representing the desired height the structure is to reach. This is analogous to having a post in the center with a tent draped over it. With a height of five at the central node and a height of three for the four surrounding nodes, Figure 4-21 shows the equilibrium shape from this situation.

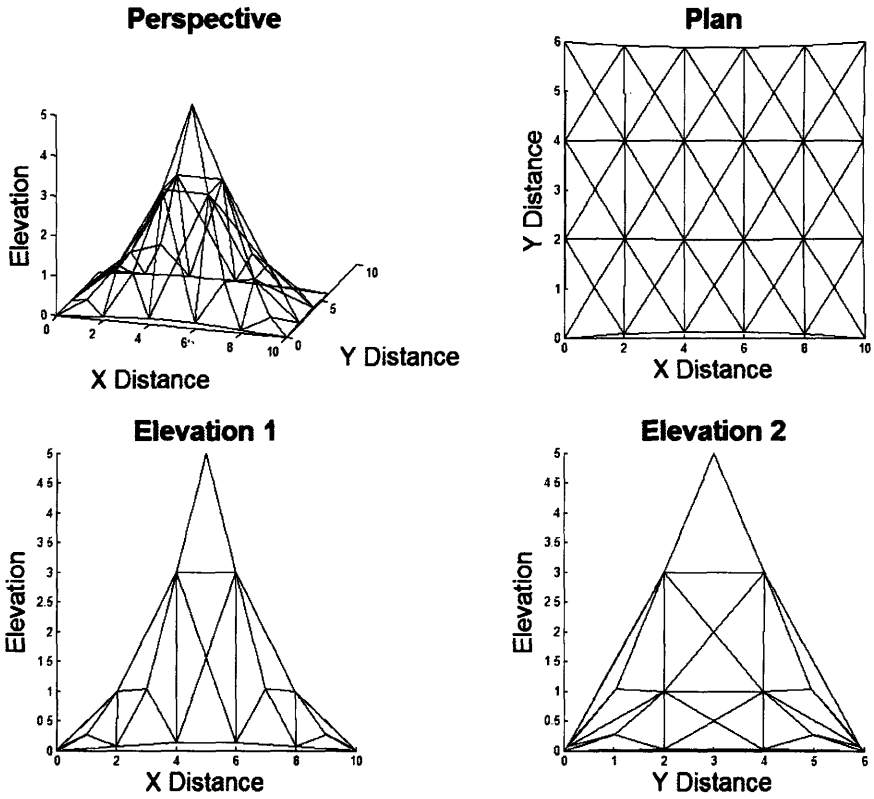

Figure 4-21: Effect of fixed nodes. Prescribed center height

Chapter 5: Comparison

The Pucher's equation method and the force-density method both have their advantages and disadvantages, as previously described. All of these considerations must be taken into account when considering which method is appropriate for the specific situation. However, it is also important to examine whether the two methods are actually analogous and will yield the same results.

To compare the two methods, the structure previously used with prescribed edge coordinates forming parabolic openings is examined. For ease of initial comparison, the shape is formed using both methods with an applied load of zero, representing the pure equilibrium shape of the structure itself. This is done for preliminary comparison since it is more difficult to compare the situations with applied loading due to the different nature of specified loads. As already stated, Pucher's equation requires a uniform area load, while the FDM requires this load to be discretized into concentrated loads at the nodal locations. Furthermore, the relationship between prescribed stress resultant and prescribed force-densities must be established to relate the two methods. It is possible to estimate the loading, stress resultant and force-densities to obtain equivalent conditions between the two methods, which is illustrated later.

Figure 5-1 shows the resultant shapes from the two methods with the applied load of zero. Pucher's equation method is shown with blue surfaces and lines, while the FDM is shown with red lines. Clearly the two structures almost exactly overlap and match each other. The nodal locations in the plan view show that they are in the same x and y positions. The elevation views show that the structures follow almost the same shape. However, there is a slight discrepancy between the two methods in the doubly curved central portion, where the elevations are slightly off.

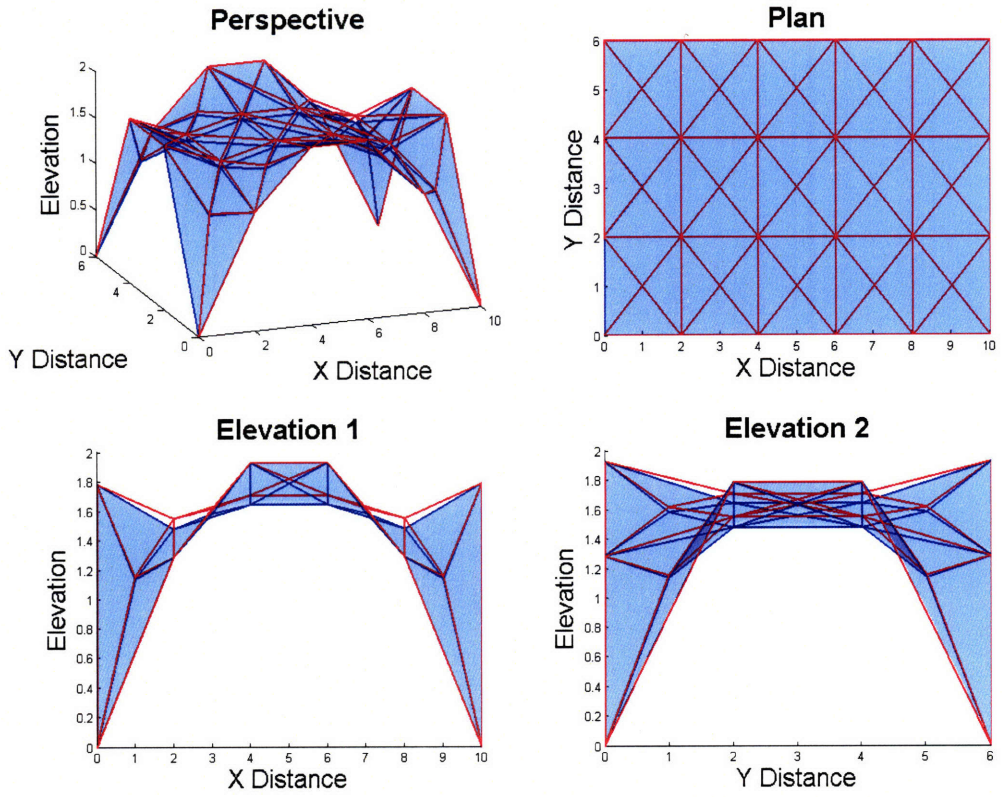

Figure 5-1: Pucher's equation method (blue) and FDM (red) comparison. Prescribed border nodes.

This discrepancy between the shapes obtained from the two methods is even more pronounced in the case where only the corner nodes are prescribed and the borders are free to move. The designed shapes from the two methods are shown in Figure 5-2, again with Pucher's method in blue surfaces with blue lines and the FDM method in red lines. The two structures follow the same overall shape, and they both reach the same elevation in the center. However, the nature of the curved interior is different between the two. Since the nodes are free to move in any direction in the FDM, as compared to the fixed x and y locations of the nodes in the Pucher's equation method, the FDM yields a smoother and more rounded shape. Since the nodes can adjust, the branches can increase or decrease in length, as required, to form a smooth surface. This is also seen in the plan view, where the borders are no longer projected as straight lines but are now required to be curved inward. In contrast, the nodes in Pucher's method are fixed in x and y, thus the borders must be fixed as straight. In addition, to accommodate the x and y nodal coordinates the surface is not smooth, and has visible bulges and portions that stick out.

In this situation is it more beneficial to use the FDM to visualize the surface since it results in a smooth shape that is more realistic.

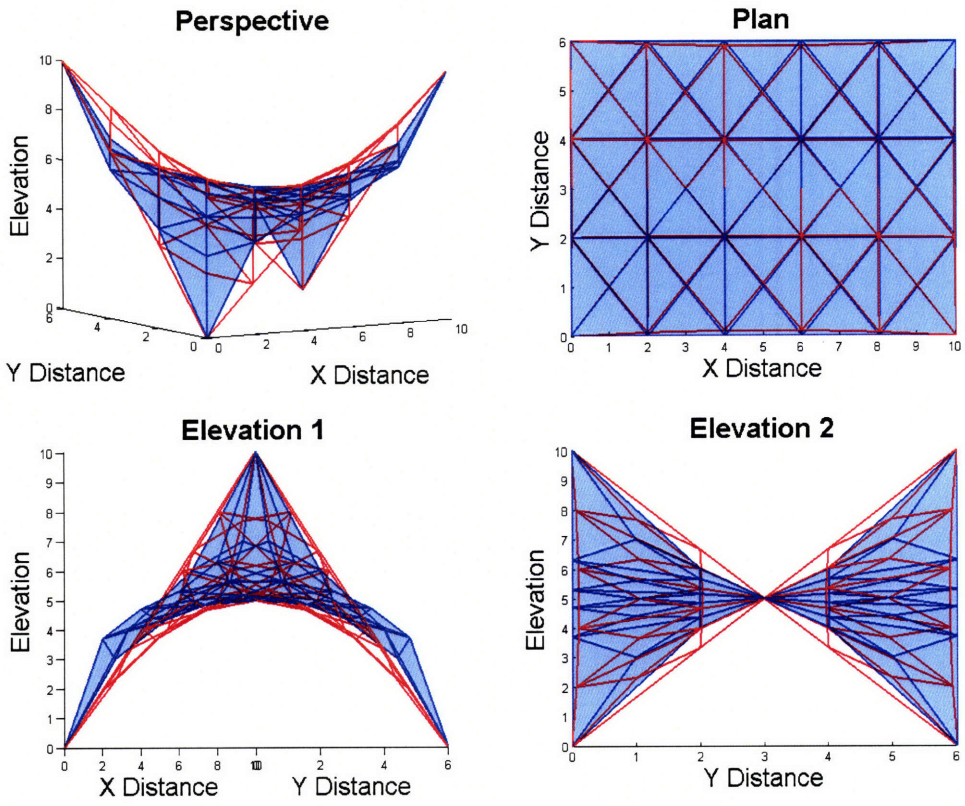

Figure 5-2: Pucher's equation method (blue) and FDM (red) comparison. Prescribed corner nodes.

As briefly mentioned, it is more difficult to compare the two methods with the situation of structures under applied loads since the input of the two methods is different. However, with some thought and initial trial and error, the structure was formed having approximately the same conditions in each method. Using the situation of prescribed perimeter nodes, the FDM structure was modeled with force densities of one for the interior branches and 100 for the exterior branches, ensuring straight borders as previously illustrated. In addition a load of one was applied to each node. Pucher's equation method was formed with a downward area load of one and a stress resultant of four. This yielded approximately the same form, shown in Figure 5-3. However, it is clear the elevations deviate slightly from each other. Here the Pucher's method gives a smoother surface. The lengths of the interior branches of the FDM structure are too long,

and must be adjusted to give a smoother structure. This is beyond the scope of design here. However, it is clear that both methods yield approximately the same shape, which would be sufficient for initial visualization and design.

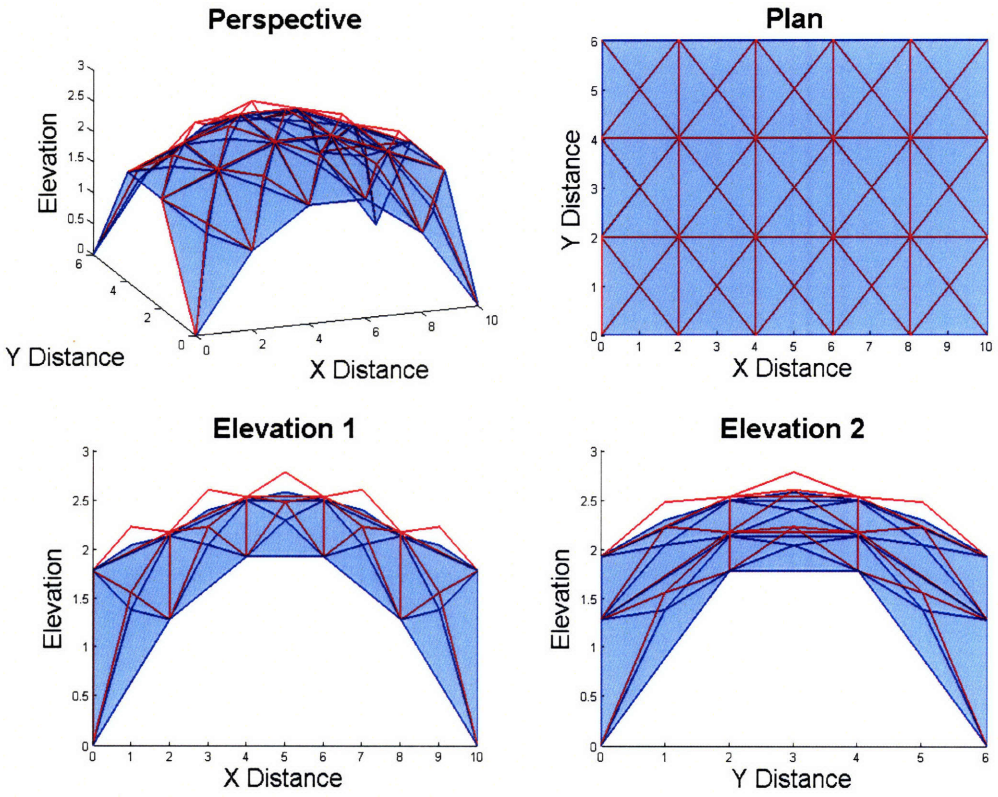

Figure 5-3: Pucher's equation method (blue) and FDM (red) comparison with applied load

To further illustrate the capabilities of these two methods, a design example with a different shape was performed. This form is based on the Kresge auditorium at MIT, shown in Figure 5-4, which is a shell structure supported at three points. The shape of the building in plan was scaled from drawings, as well as the height of the parabolic openings along the borders. These are the fixed nodes to be used in both methods.

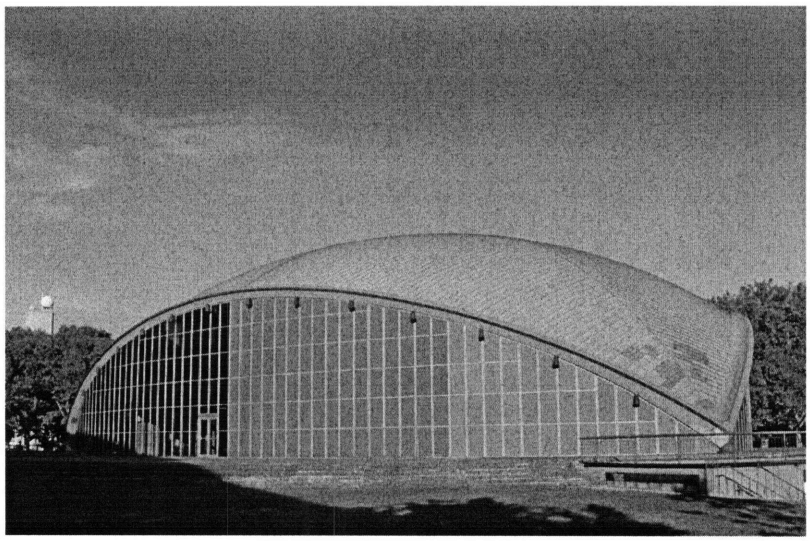

Figure 5-4: Kresge Auditorium (www.wikipedia.com)

While the plan dimensions and fixed elevations were based on Kresge, the prescribed load and other parameters were chosen arbitrarily to generate a shape just for purposes of visualization.

Figure 5-5 shows an example of this structure formed with the FDM with a load of $p_z = 1$ at all nodes and a force-density of one for the interior branches and 100 for the perimeter branches. The structure would have to be iterated on to obtain uniform branch lengths since the initial view, as shown in plan, yields some unsymmetrical conditions which are not ideal. Also, the mesh used to generate the shape could be refined to obtain further accuracy. However, more refinement means more time to label the structure and to generate the connectivity matrix, so an appropriate balance between time and accuracy must be decided upon.

Similarly, Figure 5-6 shows the preliminary design of a structure with these boundary conditions using Pucher's equation method. In this case, the load was a uniform area load of one downward, while the prescribed stress resultant was 100.

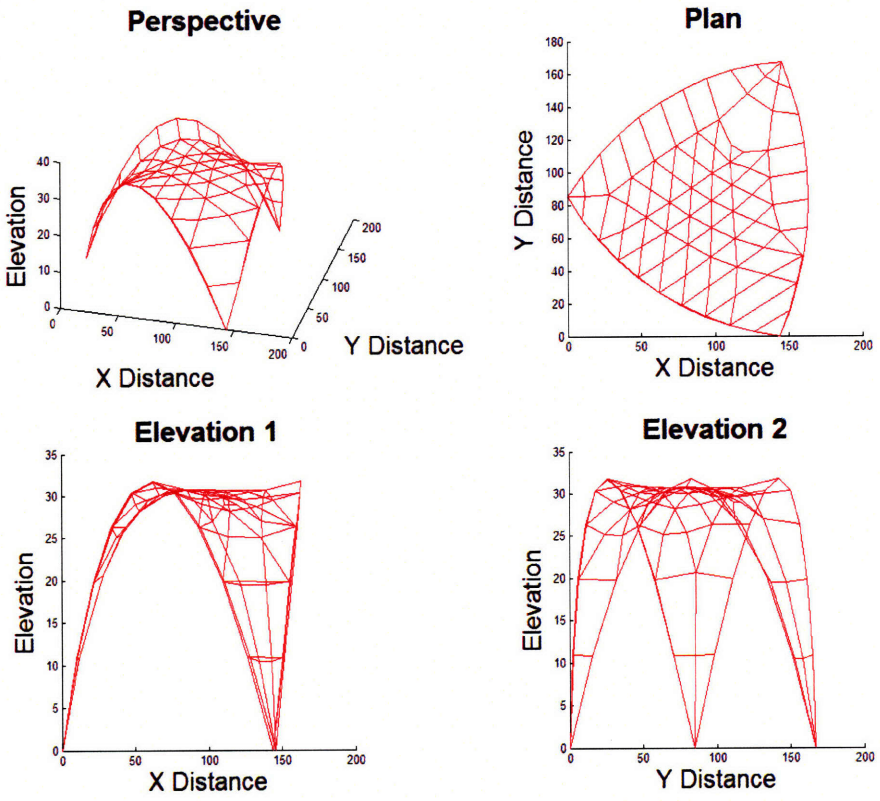

Figure 5-5: FDM. P = 1 everywhere, border q = 100, interior q = 1.

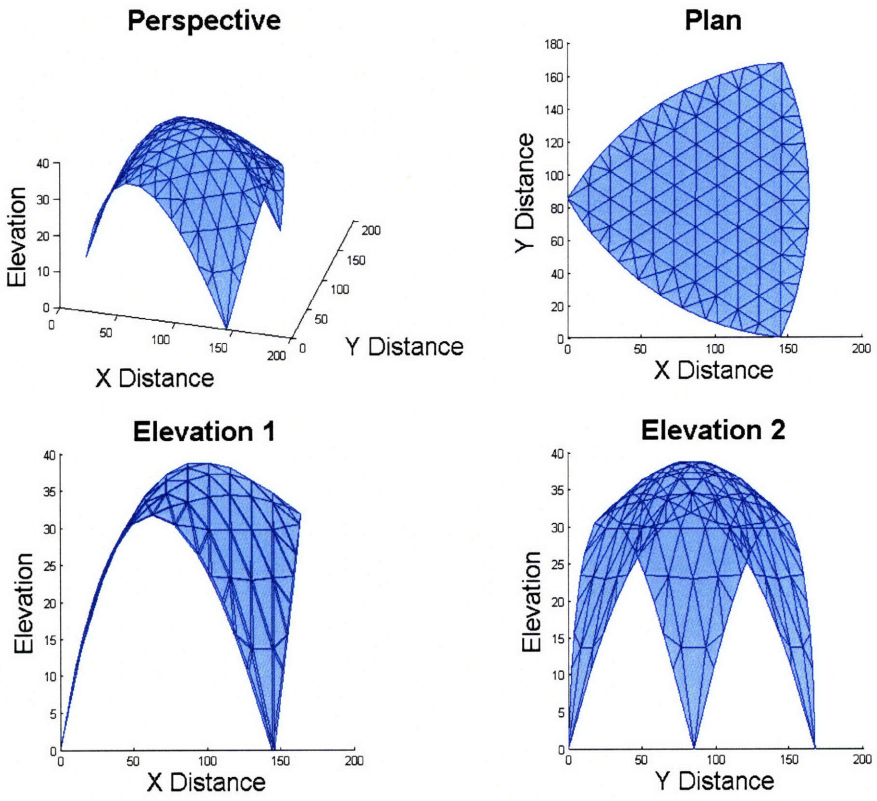

Figure 5-6: Pucher's equation method. Pz = 1, N = 100

To compare this structure between the two methods, the conditions for Pucher's equation were modified to generate a shape comparable to that formed by the FDM. Using a surface area load of one-half, while keeping the desired stress resultant as 100, yields the structure in Figure 5-7. The structure generated with the FDM is overlaid with this shape in Figure 5-8, where the structures are almost exactly the same and it is difficult to see the differences between the two. However, different topology graphs were used so that the outline of branches from Pucher's equation, shown in blue, can be distinguished from the FDM structure, depicted with red lines.

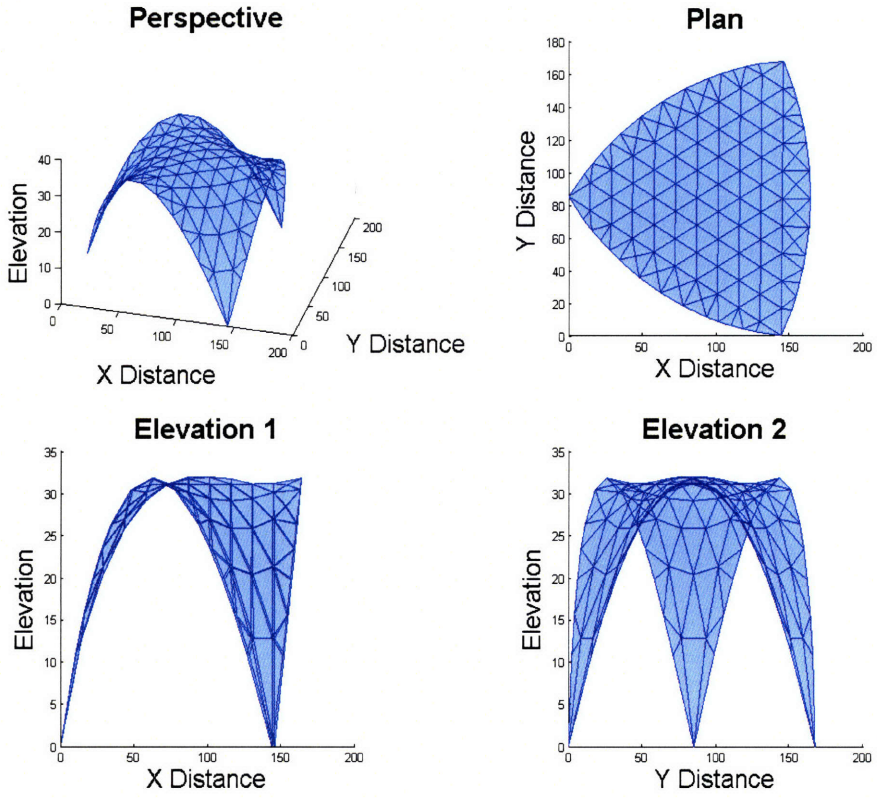

Figure 5-7: Pucher's equation method. Pz = 1/2, N = 100

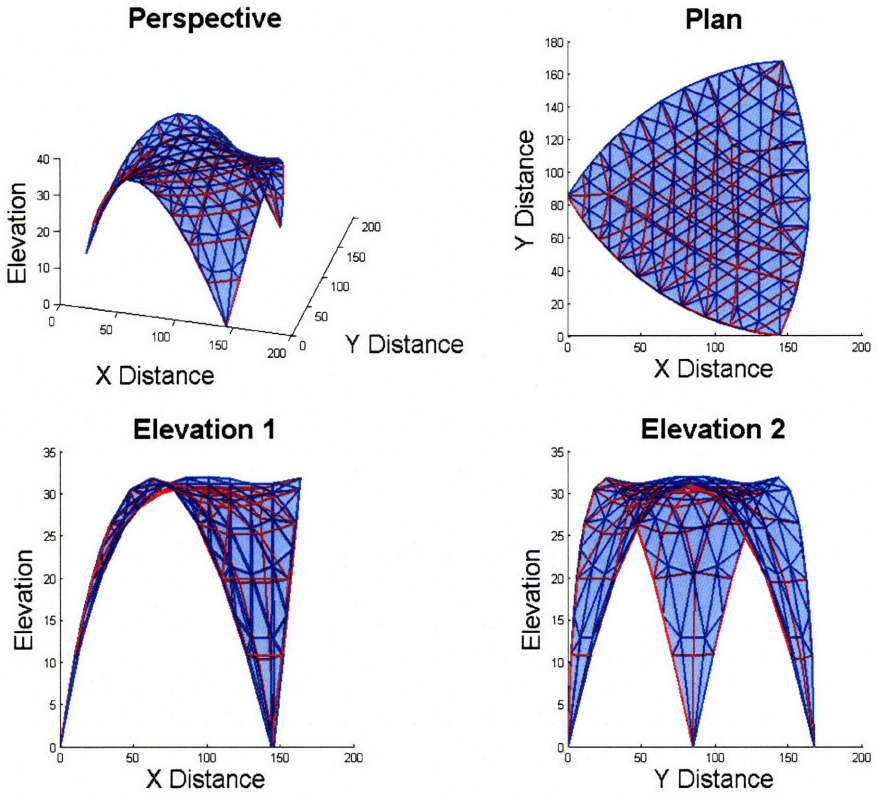

Figure 5-8: Comparison of Pucher's equation method (blue) and the FDM (red)

Chapter 6: Conclusions

The previous section has shown that both the force-density method and Pucher's equation method yield comparable results in the formation of curved membrane or shell surfaces. Both of these methods are powerful tools that enable a designer to quickly visualize a surface with very minimal input. For this initial visualization, both methods are based on statically determinate structures, requiring nothing to be specified regarding the material. This allows great generalization and applicability to a wide range of situations. In addition, it is very easy to modify the prescribed conditions if there is a change in design or if the resultant shape is not as desired.

The implications of these methods are enormous. By forming an equilibrium shape based on the unique design conditions, a structure emerges that is optimal for its own specific nature. This allows efficiency, minimization of material and lightweight rational structures. The thinness offers a unique aesthetic, while the structure itself is economic as well as sustainable.

The force-density method actually has the ability to take these ideas further. With a modified force-density method, the entire structure can be designed. This includes the shape, the branches and the actual membrane surface. Once the overall shape is known, the rods or cables can be adjusted to obtain prescribed lengths to make the structure buildable, and the actual forces within them can be specified so that specific sizes can be designed for. In addition, the surface can be discretized so that the appropriate membrane or shell can be formed.

These methods offer a vast improvement from physical models, which were the only option before the general use of computers. Whereas earlier methods were cumbersome, imprecise and time consuming, analytical methods offer ease, accuracy and flexibility. The result is structures that are able to rationally define their own shape. Rather than the designer giving the structure its form, the structure actually gives its own form to the designer.

References

Berger, Horst. "Form and Function of Tensile Structures for Permanent Buildings." Engineering Structures 21 (1999): 669-679.

Berger, Horst. "Structural Form in Architecture." Structure Magazine (2007): 37-41.

Billington, David. The Tower and the Bridge. New York: Basic Books, 1983.

Brew, J S., and W J. Lewis. "Free Hanging Membrane Model for Shell Structures." International Journal for Numerical Methods in Engineering 71 (2007): 1513-1533.

Connor, J. J., and C. A. Brebbia. Finite Element Techniques for Fluid Flow. Boston: Newnes-Butterworths, 1976.

Connor, J., J. Wolf, and R. Miller. Automatic Solution of Pucher's Equation. MIT & Agency for International Development U.S. Department of State. 1965.

Dickson, Michael. "Frei Otto's Life's Work: an Inspiration for All." The Structural Engineer (2005): 23-29.

Gonchar, Joann, and Peter Reina. "Stadium Roofs Offer Much More Than Shelter." Architectural Record 194 (2006): 257.

Grundig, Lothar, Erik Moncrieff, Peter Singer, and Dieter Strobel. "A History of the Principal Developments and Applications of the Force Density Method in Germany 1970-1999." Fourth International Colliquium on Computation of Shell & Spatial Structures (2000).

Hart, Sara. "Buckminster Fuller's Dreams of Spanning Great Distances are Being Realized in Big Projects; Long-Spans Amplify the Collaborative Relationship Between Architects and Engineers." Architectural Record 190 (2002): 267.

Hernandez-Montes, E, R. Jurado-Pina, and E. Bayo. "Topological Mapping for Tension Structures." Journal of Structural Engineering (2006).

Kawaguchi, M. "Design Problems of Long Span Spatial Structures." Engineering Structures 13 (1991): 144-163.

Knudson, William C. "Recent Advances in the Field of Long Span Tension Structures." Engineering Structures 13 (1991): 164-177.

Linkwitz, Klaus. "About Formfinding of Double-Curved Structures." Engineering Structures 21 (1999): 709-718.

Macdonald, Angus J. "Long Span Structures." Architecture Week (2003): b1.2.

Pedreschi, Remo. "Form, Force and Structure: a Brief History." Architectural Design 78 (2008): 12-19.

Sanchez, Javier, Miguel A. Serna, and Paz Morer. "A Multi-Step Force-Density Method and Surface-Fitting Approach for the Preliminary Shape Design of Tensile Structures." Engineering Structures 29 (2007): 1966-1976.

Schek, H.-J. "The Force Density Method for Form Finding and Computation of General Networks." Computer Methods in Applied Mechanics and Engineering 3 (1974): 115-134.

Solomon, Nancy B. "Flights of Fancy in Long-Span Design." Architectural Record 193 (2005): 181.

Wakefield, David. "Tensile Structure Design an Engineer's Perspective." Architectural Design 76 (2006): 92-95.

Zhang, J Y., and M Ohsaki. "Form-Finding of Tensegrity Structures Subjected to Geometrical Constraints." International Journal of Space Structures 21 (2006): 183-196.

Zhang, J. Y., and M. Ohsaki. "Adaptive Force Density Method for Form-Finding Problem of Tensegrity Structures." International Journal of Solids and Structures 43 (2006): 5658-5673.

www.wikipedia.com

www.skyscrapercity.com

Appendix A: Matlab Codes

A.1: Pucher's Equation Code

```
function Pucher(COORD,ICONN,Pz,N,BC)

    %Use 3-node triangular elements
    %Define NN: Number of Nodes; NE: Number of Elements;
    %COORD[NN,3]:Coordinate Matrix with columns 1)node number, 2)x
coordinate, 3)y coordinate;
    %ICONN[NE,4]:Interconnectivity Matrix for the elements with columns
1)element number, 2)node 1 number, 3)node 2 number, 4)node 3 number;
    %nodes entered counterclockwise
    %COORD and ICONN entries can be in any order
    %Ar: Element Area
    %KE[3,3]: Local Element Stiffness/Influence Coefficient Matrix;
    %K[NN,NN]: Global Stiffness Matrix;
    %BC[N,z]: Boundary Condition Matrix for prescribed conditions with
columns 1)node number, 2)prescribed condition;
    %Pz : Prescribed surface loading per unit projected area in +z
direction; N: desired stress;
    %PE[3,1]: Local Element Load Matrix;
    %P[NN,NN]: Global Load Matrix

    %Define arrays of x and y coordinates from input COORD matrix
    X = COORD(:,2); % x coord is second column of coord matrix
    Y = COORD(:,3); % y coord is third column of coord matrix
    [NN,coord] = size(COORD);
    [NE,iconn] = size(ICONN);
    K = zeros(NN,NN); %set global stiffness to zero initially. Size is
Number Nodes x Number Nodes
    P = zeros(NN,1); %sets load matrix to zero initially. size is
Number Nodes x 1

    %Assemble local stiffness matrix
    for E=1:NE %loop over the elements
        a1 = X(ICONN(E,4))-X(ICONN(E,3)); %x3-x2
        a2 = X(ICONN(E,2))-X(ICONN(E,4)); %x1-x3
        a3 = -a1-a2; %x2-x1
        b1 = Y(ICONN(E,3))-Y(ICONN(E,4)); %y2-y3
        b2 = Y(ICONN(E,4))-Y(ICONN(E,2)); %y3-y1
        b3 = -b1-b2; %y1-y2
        %A1 = 1/2*(X(ICONN(E,3))*Y(ICONN(E,4))-
X(ICONN(E,4))*Y(ICONN(E,3)));
        %A2 = 1/2*(X(ICONN(E,4))*Y(ICONN(E,2))-
X(ICONN(E,2))*Y(ICONN(E,4)));
        Ar = 1/2*(b1*a2-b2*a1);
        %A3 = A-A1-A2;
        B = [b1; b2; b3];
        A = [a1; a2; a3];
        KE = zeros(3,3); %set element stiffness matrix to zero
initially
        KE = 1/(4*Ar)*(B*transpose(B)+A*transpose(A));
        %Assemble local element stiffness matrix into global stiffness
matrix
        %loop over nodes in element
```

```
        for i=2:4
            for j=2:4
                K(ICONN(E,i),ICONN(E,j)) =  K(ICONN(E,i),ICONN(E,j)) +
KE(i-1,j-1);
            end
        end

        %Assemble element Loading Matrix, PE, and put into global P
Matrix
        PE = zeros (3,1); %set element load matrix to zero initially
        for i=2:4 %loop over nodes in element
        PE(i-1,1) = -1/3*Pz/N*Ar;
        P(ICONN(E,i),1) = P(ICONN(E,i),1) + PE(i-1,1);
        end
    end
    %K %Prints Global K Matrix
    %P %Prints Global P Matrix

    %Boundary Conditions
    [nrow,ncol] = size(BC); % number of boundary conditions
    for i=1:nrow %loop through prescribed conditions
        P = P - BC(i,2)*(K(:,BC(i,1))); %multiply prescribed condition
by corresponding column of K matrix and subtr from P
        K(:,BC(i,1)) = 0; K(BC(i,1),:) = 0; %set corresponding row and
column of K matrix to zero
        K(BC(i,1),BC(i,1)) = 1; %set corresponding diagonal of K matrix
to 1
    end
    for i = 1:nrow
        P(BC(i,1),1) = BC(i,2);
    end

    %P %Prints Global K Matrix
    %K %Prints Global P Matrix

    Z = inv(K)*P; %Solves Elevation Matrix

    %Nodal Geometry
    Geometry_NodeNumber_X_Y_Z = [COORD(:,1) X Y Z]


    %%%%%%PLOTTING
    f = ICONN; f(:,1) = [];  %f is ICONN matrix with just the node
numbers and not the first col of element number
    vertices1 = [X Y Z]; vertices = sortrows(vertices1,3); %x,y,z
coordinates sorted in ascending z order
    new = sortrows(Geometry_NodeNumber_X_Y_Z,4); % geometry sorted in
ascending z order
    n = 1:1:NN;
    sorted = [n' new]; %matrix with first column the new node number,
second column the old node number, 3-5:x,y,z coordinates

    %rename nodes in f matrix to reflected sorted node number for
plotting
    for i = 1:NN
```

```
        for k = 1:NN
            if sorted(i,2) == k
                for a = 1:NE
                    for j=1:3
                        if f(a,j)== k
                            faces(a,j)=i;
                        end
                    end
                end
            else continue
            end
        end
    end

    [s,c] = size(Z);
    %r = (0:1/3:(s-1)/3)/s;
    r = (1:1:s)/s;
    %g = (0:3:(s-1)*3)/s;
    g = (0:1/3:(s-1)/3)/s;

    b = (s:-1:1)/s;
    color = [r' g' b'];

    %subplot(2,1,1)
    subplot(2,2,1)

patch('Faces',faces,'Vertices',vertices,'FaceVertexCData',color,'FaceCo
lor','black','LineStyle','-.','LineWidth',1,'FaceAlpha',.2)
    xlabel('X Distance','FontSize',20)
    ylabel('Y Distance','FontSize',20)
    zlabel('Elevation','FontSize',20)
    view(15,40)
    title('Perspective','FontWeight','bold','FontSize',23)


    subplot(2,2,3)

patch('Faces',faces,'Vertices',vertices,'FaceVertexCData',color,'FaceCo
lor','black','LineStyle','-.','LineWidth',1,'FaceAlpha',.2)
    zlabel('Elevation','FontSize',20)
    xlabel('X Distance','FontSize',20)
    ylabel('Y Distance','FontSize',20)
    view(0,0)
    title('Elevation 1','FontWeight','bold','FontSize',23)

    subplot(2,2,4)

patch('Faces',faces,'Vertices',vertices,'FaceVertexCData',color,'FaceCo
lor','black','LineStyle','-.','LineWidth',1,'FaceAlpha',.2)
    zlabel('Elevation','FontSize',20)
    xlabel('X Distance','FontSize',20)
    ylabel('Y Distance','FontSize',20)
    view(90,0)
    title('Elevation 2','FontWeight','bold','FontSize',23)
```

```
    subplot(2,2,2)

patch('Faces',faces,'Vertices',vertices,'FaceVertexCData',color,'FaceCo
lor','black','LineStyle','-.','LineWidth',1,'FaceAlpha',.2)
    zlabel('Elevation','FontSize',20)
    xlabel('X Distance','FontSize',20)
    ylabel('Y Distance','FontSize',20)
    view(0,90)
    title('Plan','FontWeight','bold','FontSize',23)
```

A.2: Force-Density Method Code

```
function fd(n,CONN,nfixed,q,p)

% m = number branches
% n = number nodes
%number nodes so free nodes numbered first (1,2,..), fixed nodes
numbered after
% CONN = m x 3,col1=branch number col2 = node 1 number, col3 = node 2
number.
% presorted so node number 1 < node number 2
%Cs[m,n] = branch-node matrix
%C[m,nfree] = free nodes branch-node matrix . Cf(m,fn) = fixed nodes
%branch-node matrix
%nfixed[fn,4] = fixed nodes col1=node col2=x col3=y col4=z
%q[m,1] = force-densities
%p[n,4] = loads, col1=node number(free nodes) col2=px col3=py col4=pz


[m,three] = size(CONN); %determine number of branches

Cs = zeros(m,n); %initialize C to zero
for j = 1:m %loop over branches
    for r = 1:n %loop over nodes
        if CONN(j,2) == r
            Cs(j,r) = 1; %C(branch, node) = 1 if first node
        end
        if CONN(j,3) == r
            Cs(j,r) = -1; %C(brance, node) = -1 if second node
        end
    end
end

[fn,four] = size(nfixed); %fn = number fixed nodes. determine fn

%seperate Cs matrix into fixed (Cf) and free (C) portions corresponding
to
%the node condition

for i = 1: fn
    Cf(:,i) = Cs(:,nfixed(i,1));
end

C = Cs;
for i = fn:-1:1
    C(:,nfixed(i,1))=[];
end

%Cf
%C

Q = diag(q);
```

```
D = transpose(C)*Q*C;
Df = transpose(C)*Q*Cf;

%Solve for free node coordinates
x = inv(D)*(p(:,2)-Df*nfixed(:,2));
y = inv(D)*(p(:,3)-Df*nfixed(:,3));
z = inv(D)*(p(:,4)-Df*nfixed(:,4));

for i =1:n-fn
    Xtot(i,1) = x(i,1);
    Ytot(i,1) = y(i,1);
    Ztot(i,1) = z(i,1);
end

for j = (n-fn)+1:n
    Xtot(j,1) = nfixed(j-(n-fn),2);
    Ytot(j,1) = nfixed(j-(n-fn),3);
    Ztot(j,1) = nfixed(j-(n-fn),4);
end

%Nodal Coordinates
Geometry = [Xtot Ytot Ztot]

%Solve for branch lengths
for i = 1:m
lx(i,1) = abs(Xtot(CONN(i,2),1)-Xtot(CONN(i,3),1));
ly(i,1) = abs(Ytot(CONN(i,2),1)-Ytot(CONN(i,3),1));
lz(i,1) = abs(Ztot(CONN(i,2),1)-Ztot(CONN(i,3),1));
l(i,1) = sqrt((lx(i,1))^2+(ly(i,1))^2+(lz(i,1))^2);
end
l
L = diag(l);

%branch fores
s = L*q

%%%%%PLOTTING
subplot(2,2,1)
for i = 1:m
    line([Xtot(CONN(i,2),1) Xtot(CONN(i,3),1)],[Ytot(CONN(i,2),1)
Ytot(CONN(i,3),1)],[Ztot(CONN(i,2),1)
Ztot(CONN(i,3),1)],'Color','black')
    xlabel('X Distance','FontSize',20)
    ylabel('Y Distance','FontSize',20)
    zlabel('Elevation','FontSize',20)
    view(-20,25)
    title('Perspective','FontWeight','bold','FontSize',23)
end

subplot(2,2,2)
for i = 1:m
    line([Xtot(CONN(i,2),1) Xtot(CONN(i,3),1)],[Ytot(CONN(i,2),1)
Ytot(CONN(i,3),1)],[Ztot(CONN(i,2),1)
Ztot(CONN(i,3),1)],'Color','black')
    xlabel('X Distance','FontSize',20)
```

```
    ylabel('Y Distance','FontSize',20)
    zlabel('Elevation','FontSize',20)
    view(0,90)
    title('Plan','FontWeight','bold','FontSize',23)
end

subplot(2,2,3)
for i = 1:m
    line([Xtot(CONN(i,2),1) Xtot(CONN(i,3),1)],[Ytot(CONN(i,2),1)
Ytot(CONN(i,3),1)],[Ztot(CONN(i,2),1)
Ztot(CONN(i,3),1)],'Color','black')
    xlabel('X Distance','FontSize',20)
    ylabel('Y Distance','FontSize',20)
    zlabel('Elevation','FontSize',20)
    view(0,0)
    title('Elevation 1','FontWeight','bold','FontSize',23)
end

subplot(2,2,4)
for i = 1:m
    line([Xtot(CONN(i,2),1) Xtot(CONN(i,3),1)],[Ytot(CONN(i,2),1)
Ytot(CONN(i,3),1)],[Ztot(CONN(i,2),1)
Ztot(CONN(i,3),1)],'Color','black')
    xlabel('X Distance','FontSize',20)
    ylabel('Y Distance','FontSize',20)
    zlabel('Elevation','FontSize',20)
    view(90,0)
    title('Elevation 2','FontWeight','bold','FontSize',23)
end
```

Appendix B: Sample Program Input

B.1: Pucher's Equation Input

See Figure 4-3 for number reference

COORD		
Node #	X position	Y position
1	0	0
2	2	0
3	4	0
4	6	0
5	8	0
6	10	0
7	1	1
8	3	1
9	5	1
10	7	1
11	9	1
12	0	2
13	2	2
14	4	2
15	6	2
16	8	2
17	10	2
18	1	3
19	3	3
20	5	3
21	7	3
22	9	3
23	0	4
24	2	4
25	4	4
26	6	4
27	8	4
28	10	4
29	1	5
30	3	5
31	5	5
32	7	5
33	9	5
34	0	6
35	2	6
36	4	6
37	6	6
38	8	6
39	10	6

BC	
Node#	Z position
1	0
2	1.28
3	1.92
4	1.92
5	1.28
6	0
12	1.777778
17	1.777778
23	1.777778
28	1.777778
34	0
35	1.28
36	1.92
37	1.92
38	1.28
39	0

ICONN			
Element #	Node1	Node2	Node3
1	1	2	7
2	2	3	8
3	3	4	9
4	4	5	10
5	5	6	11
6	1	7	12
7	2	13	7
8	2	8	13
9	3	14	8
10	3	9	14
11	4	15	9
12	4	10	15
13	5	16	10
14	5	11	16
15	6	17	11
16	7	13	12
17	8	14	13
18	9	15	14
19	10	16	15
20	11	17	16
21	12	13	18
22	13	14	19
23	14	15	20
24	15	16	21
25	16	17	22
26	12	18	23
27	13	24	18
28	13	19	24
29	14	25	19
30	14	20	25
31	15	26	20
32	15	21	26
33	16	27	21
34	16	22	27
35	17	28	22
36	18	24	23
37	19	25	24
38	20	26	25
39	21	27	26
40	22	28	27
41	23	24	29
42	24	25	30
43	25	26	31
44	26	27	32
45	27	28	33
46	23	29	34
47	24	35	29
48	24	30	35
49	25	36	30
50	25	31	36
51	26	37	31
52	26	32	37
53	27	38	32
54	27	33	38
55	28	39	33
56	29	35	34
57	30	36	35
58	31	37	36
59	32	38	37
60	33	39	38

B.2 Force-Density Method Input

See Figure 4-14 for number reference

nfixed			
Node	x Position	y Position	z Position
24	0	0	0
25	2	0	1.28
26	4	0	1.92
27	6	0	1.92
28	8	0	1.28
29	10	0	0
30	10	2	1.777778
31	10	4	1.777778
32	10	6	0
33	8	6	1.28
34	6	6	1.92
35	4	6	1.92
36	2	6	1.28
37	0	6	0
38	0	4	1.777778
39	0	2	1.777778

n = 39

p			
node	px	py	pz
1	0	0	1
2	0	0	1
3	0	0	1
4	0	0	1
5	0	0	1
6	0	0	1
7	0	0	1
8	0	0	1
9	0	0	1
10	0	0	1
11	0	0	1
12	0	0	1
13	0	0	1
14	0	0	1
15	0	0	1
16	0	0	1
17	0	0	1
18	0	0	1
19	0	0	1
20	0	0	1
21	0	0	1
22	0	0	1
23	0	0	1

CONN		
Branch	Node i	Node j
1	24	25
2	25	26
3	26	27
4	27	28
5	28	29
6	29	30
7	30	31
8	31	32
9	32	33
10	33	34
11	34	35
12	35	36
13	36	37
14	37	38
15	38	39
16	24	39
17	1	24
18	1	25
19	1	6
20	1	39
21	2	25
22	2	26
23	2	7
24	2	6
25	3	26
26	3	27
27	3	8
28	3	7
29	4	27
30	4	28
31	4	9
32	4	8
33	5	28
34	5	29
35	5	30
36	5	9
37	6	39
38	6	7
39	7	8
40	8	9
41	9	30
42	10	39
43	6	10
44	10	15
45	10	38
46	6	11
47	7	11
48	11	16
49	11	15

(continued)		
Branch	Node i	Node j
50	7	12
51	8	12
52	12	17
53	12	16
54	8	13
55	9	13
56	13	18
57	13	17
58	9	14
59	14	30
60	14	31
61	14	18
62	15	38
63	15	16
64	16	17
65	17	18
66	18	31
67	19	38
68	15	19
69	19	36
70	19	37
71	15	20
72	16	20
73	20	35
74	20	36
75	16	21
76	17	21
77	21	34
78	21	35
79	17	22
80	18	22
81	22	33
82	22	34
83	18	23
84	23	31
85	23	32
86	23	33
87	6	25
88	6	15
89	15	36
90	7	26
91	7	16
92	16	35
93	8	27
94	8	17
95	17	34
96	9	28
97	9	18
98	18	33

q
100
100
100
100
100
100
100
100
100
100
100
100
100
100
100
100
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1

(continued)
q
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1