Towards Synthesis of Tensegrity Structures of Desired Shape and Micro-pipette Aspiration of Red Blood Cells

A Project Report

Submitted in Partial fulfillment of the

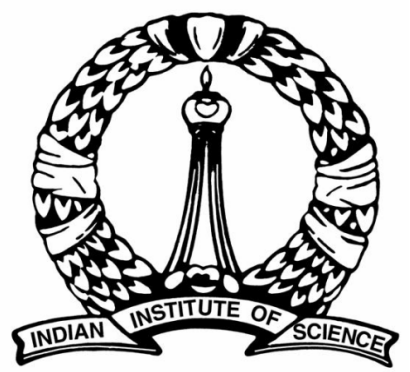
Requirements for the Degree of

Master of Engineering

in the Faculty of Engineering

By

Navaneeth Krishna R. P.



DEPARTMENT OF MECHANICAL ENGINEERING

INDIAN INSTITUTE OF SCIENCE

BANGALORE-560012

INDIA

June 2013

Abstract

In this project, we formulate an optimization problem to synthesize a tensegrity structure of desired shape and conduct preliminary experiments on micro-pipette aspiration of a red blood cell (RBC). In synthesis of tensegrity structures, we use the force density method. The reason for using the force density, defined as the force per unit length in the desired configuration, is twofold: (i) to directly impose constraints of positive and negative force densities in tension and compression elements in class 1 tensegrity where no two compression elements have a common vertex, by designating all members in the desired shape a priori as either a bar in compression or a cable in tension. We solve static equilibrium equations at the vertices in the desired configuration subject to constraints on force densities, which are the variables in the synthesis, respectively, and (ii) to obtain free lengths of all the members using the force densities. We use this method to synthesize a previously known semi-toroidal tensegrity arch with 24 bars and 102 cables and a hitherto unknown tensegrity of biconcave shape similar to that of a red blood cell comprising 24 bars and 112 cables. We also present static analysis of a tensegrity structure by minimizing the potential energy with unilateral constraints on the lengths of the cables, which cannot take compressive loads. We also extend the method to synthesize a tensegrity table of desired height and area with three bars and nine cables under a predefined load. The prototypes of all three synthesized tensegrities are made and tested.

Preliminary experiments are conducted to obtain static and viscoelastic force deflection responses of red blood cells (RBCs), so that a tensegrity model can be fit to those responses. For the experiment, the RBCs are isolated from the whole blood. The aspiration is carried out with a micro-pipette of diameter less than 2.5 μm (as the height and thickness of RBC is approximate 8 μm and 2.5 μm, respectively). It is observed that an RBC is aspirated at a single pressure step and thus making it difficult to obtain the needed force-deflection responses

.

Acknowledgment

I take this opportunity of acknowledging the people who have helped me directly or indirectly during my two years of stay in IISc and because of whom I am able to successfully complete the project.

Foremost, I would like to thank my Advisor, Prof. G. K. Ananthasuresh for his meticulous guidance, patience, and constant encouragement throughout the course of my work. I consider it a great honour to have worked with him and have been inspired by his sincerity and dedication towards work.

I would like to thank the examiners, Prof. C. S. Jog and Prof. Namrata Gundiah for their valuable suggestions and inputs during the course of my work.

I express my gratitude towards all faculty members of the department of Mechanical Engineering for the course work done with them. Their lectures have been very educative and inspiring. I would like to thank all the staff members of the department.

The M2D2, Multi-disciplinary and Multi-scale Devices and Design, Lab has always been conducive and friendly environment for technical discussions. I sincerely thank Santosh Bhargav, Suma, Shilpa, Srinath, Ramu and Bharat for their assistance in carrying out experimental work and fabrication of prototypes. I sincerely thank Rajesh at Health Center Lab, IISc, for providing me with blood samples for the experiments conducted.

Much I owe to my dear friends Snadeep, Subin, R. P. Singh and my lab-mates Jagdish, Puneet, Harish, Shantanu, Nandhini, Ramanath, Saurab, Nikhil, Gautam and Deepa with whom I worked, studied, learned, partied and enjoyed my stay at IISc. I got help whenever I asked them. Thank you all.

Words fail when it comes to acknowledging my family. Their constant, selfless love and encouragement made me what I am today.

I once again thank all the people who have made my stay at the Institute an enlightening one.

List of Figures

1	Tensegrity X and 3 bar prism	1
2	Equilibrium at a vertex	5
3	Tensegrity Arch	9
4	Cassini's oval	11
5	Biconcave Tensegrity structure (model, prototype)	12
6	Deformation of Tensegrity Table under loads	17
7	Fabricated Tensegrity Table	17
8	Aspirated and Crenated RBC	20
9	Bee-stinger micropipette and RBC	21
10	Micro-pipette Aspiration set-up	21

List of Tables

1	The coordinates of the vertices used to construct the Semi-torroidal Tensegrity Arch.	10
2	The connectivity of elements that make the Semi-torroidal Tensegrity Arch.	11
3	Force density and length of individual elements of Semi-torroidal Tensegrity Arch.	13
4	The coordinates of the vertices used to construct the RBC Biconcave model.	14
5	The connectivity of elements that make the RBC Biconcave model.	15
6	Force density and length of individual elements of RBC Biconcave model	16
7	The coordinates of the vertices used to construct the Tensegrity table.	18
8	The connectivity, force density and length of individual elements to make the Tensegrity table	18
9	Observations in micro-pipette aspiration of RBCs	20

Table of contents

Abstract i

Acknowledgement ii

1. Introduction 1
2. Towards Synthesis of Tensegrity Structures 5
3. Results and Discussion 9
4. Micro-pipette Aspiration of RBC 19
5. Summary and Conclusion 22

Appendix 23

A.1 Matlab program to synthesize a tensegrity prism 23
A.2 Matlab program to synthesize a semi-torroidal tensegrity arch 27
A.3 Matlab program to synthesize a biconcave shape of a RBC 37
A.4 Matlab program to synthesize a tensegrity table 42

Reference 51

Chapter 1
Introduction

Multiple models are proposed for simulating the mechanics of a *cytoskeleton*, i.e., the interior framework of biological cells comprising actin filaments, intermediate filaments, and microtubules. They include: simple elastic, viscoelastic, or poro-viscoelastic continuum [1], porous gel [2], soft glassy material [3], bi-phasic model [4], and *tensegrity* (a portmanteau that combines *tens*ion and int*egrity*) [5] network model incorporating discrete structural elements that bear tension and compression. In this project, the cytoskeleton is modeled as a tensegrity structure.

Tensegrity structures are made of tension and compression elements [6]. The tension elements (cables) take tensile loads and compressive elements (bars) take compressive loads. Shown in Fig. 1(a) – (b) are the simplest tensegrity structures in 2D and 3D. Tensegrity structures use internal preload to assume a stiff and stable equilibrium shape under no external loads. Thus, Cytoskeleton models based on the concept of tensegrity are used to model the mechanical response of biological cells [7]. A cytoskeleton too has pre-stress in its members, whose role is to confer stable shape to the cell. Both cytoskeletons and tensegrity structures show "action-at-a-distance" phenomenon: an action at a point is carried forth to a faraway point despite the seemingly discrete network of elements. Tensegrity structures are stabilized by the continuous tension carried by the structural members. Their structural stiffness increases in proportion with the level of pre-stress present in them. Experimental observations show proportional relationship between the cell stiffness and that of the cytoskeleton [7]; controlled experiments in which selected portions of actin filaments were broken with a laser beam that led to the collapse of the cell. Thus, tensegrity modeling is tenable for simulating cytoskeletal mechanics. Tensegrity structures also have applications in deployable devices used for example, in space structures [8]. They are used in the field of architecture, where different types of tensegrity domes and arches are built [9]. As the large deformations in a tensegrity are controllable they find application in robotic arms [10]. There are different classes of tensegrity structures; class m tensegrity structures are structures where m bars share a single vertex [6]. We consider class 1 tensegrity structures with and without external loads.

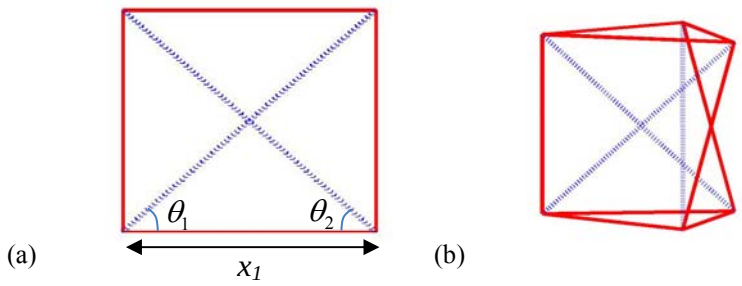


Fig.1. Simplest (a) 2D tensegrity structure known as tensegrity X and (b) 3D tensegrity known as ′3 bar prism′.

There are different methods to find the equilibrium state attained by a tensegrity structure for given connectivity and stiffness properties of the constituent bars and cables. These methods are generally known as *form-finding* methods [11]. They are classified into two categories: kinematic methods and static methods [6]. Some of the Kinematic methods of form-finding have analytical solutions [11]. Such methods are used for form-finding of prismatic tensegrities, where the cable lengths and the polygon face of the tensegrity prism are given, using which the lengths of the bars are calculated to get a stable tensegrity prism. Other Kinematic methods use nonlinear programming [12], by posing a constrained minimization problem, where connectivity and nodal points are known a priori and the lengths of the bars are maximized until a stable structure is obtained. The third Kinematic form-finding method uses dynamic relaxation [13] of different forms by varying the positions of the vertices of the system and solving the equilibrium equations at each node to find the unbalanced forces iteratively. The form is modified using the dynamic equations of motion. Although the kinematic methods have good convergence properties, they are only effective when the number of vertices in the structure is low.

Static methods of form-finding are of four types. In the force density method [14], for given connectivity of a tensegrity structure, the force density of each of the members is predefined. The nodal point coordinates are then calculated for the corresponding force densities that satisfy static equilibrium. In the analytical solution method [15], stable form of a rotationally symmetric tensegrity (e.g. a tensegrity prism) is found. Here, the nodal force balance is done to arrive at the angle subtended by the bars for given lengths of cables. In the energy method [16], the form of the tensegrity is found for which the total potential energy is minimum. Here the potential energy of both cables under tension and bars under compression are considered. In the reduced co-ordinate method [17], the bars are considered to be rigid and the equilibrium equations pertaining only to the cables are formulated with the help of the principle of virtual work.

In all the aforementioned methods the connectivity of the member elements and the stiffness property of the materials are given and we find the form that satisfies the equilibrium equations. It may be noted that form-finding addressed by these methods is a *forward problem* in the sense that the aim is to obtain a stable tensegrity form using given number of bars and cables and their connectivity and stiffness properties. The inverse problem is determining the geometry and force densities to obtain a prescribed shape. Form-finding of tensegrity structures with desired shape has been addressed by Masic [18], where the material stiffness properties and connectivity of the structure are known a priori. The difference between the coordinates of the vertices of the equilibrium form of the tensegrity and the coordinates of the prescribed vertices is minimized. The coordinates of the vertices and the force densities are the variables in this problem. Topology design of tensegrity design using mixed integer programming has been done by Ehara, S. and Kanno, Y. [19]. The formulation of the problem is given below:

$$
\begin{aligned}
\text{MIP1} \quad & \underset{\mathbf{q},\mathbf{x}}{Min} && -\textstyle\sum_{i\in N} x_i \\
& \textit{Subject to} && \mathbf{Hq} = 0 \\
& && -M\, x_i \le q_i \le M(1-x_i) - \varepsilon, && \forall i \in N_{bar} \\
& && \textstyle\sum_{i\in N(n_j)} x_i \le 1, && \forall j \in \mathbf{V} \\
& && x \in \{0,1\}^{|N|}.
\end{aligned}
$$

$$
\begin{aligned}
\text{MIP2} \quad & \underset{\mathbf{q},\mathbf{y}}{Min} && -\textstyle\sum_{i\in N_{cable}} y_i \\
& \textit{Subject to} && \mathbf{Hq} = 0 \\
& && q_i \le -\varepsilon, && \forall i \in N_{bar} \\
& && 0 \le q_i \le My_i, && \forall i \in N_{cable} \\
& && x \in \{0,1\}^{|N_{cable}|}.
\end{aligned}
\tag{1}
$$

Data $\mathbf{V}, M, \varepsilon$

where $\mathbf{V}$ is the set of all the vertices and M and ε are positive constants such that M is sufficiently large ($0 < \varepsilon << M$). It may be noted that $\mathbf{H}$ and $\mathbf{q}$ are the equilibrium matrix corresponding to the static equilibrium equations at the vertices and the force density vector respectively, and $\mathbf{x}$ and $\mathbf{y}$ are the vectors containing one or zero corresponding to the connections where bars and cables are present. Though the synthesis problem is solved, the structure we get is always statically indeterminate of order one or more.

The aforementioned are some of the few methods that address the inverse or the synthesis problem. In this project, we synthesize a tensegrity structure of desired shape, where the nodal points and connectivity are known a priori, and we find the stiffness properties of bars and cables of the tensegrity structure, which give the desired form for the tensegrity structure.

We synthesize a free-standing tensegrity structure (i.e., with no external loads) of desired shape by solving the force density method as an optimization problem with constraints on the force densities. We synthesize the tensegrity structures with external loads, which attain a predefined geometric constraint (e.g. height) under the application of the load, in two steps. The first step involves solving the force densities in an optimization problem to get the free-standing tensegrity. The second step involves static analysis, where we minimize the potential energy of the system under the externally applied loads with a reduced co-ordinate system. We use unilateral constraints on the cables in the form of sigmoid function to account for the slack condition of the cables under compressive loads. We check for the geometric constraint and update the tensegrity configuration and repeat the above steps until the geometric constraint is satisfied. We synthesize different desired shapes such as a semi-toroidal arch, biconcave tensegrity similar to the cytoskeleton of a red blood cell (RBC), and a table of desired height under the application of a prescribed external load.

Although we are representing the outer shape of RBC with tensegrity structures similar to that of a space-filling cytoskelton, the actual cytoskeleton of the RBC is a membrane skeletal network made of 2D protein polymer network (spectrin network) [20,21]. The membrane skeletal network of the RBC is made up of about 30000 junctional complexes [22]. Each junctional complex consists of a ~37 nm actin filaments (called protofilaments), six αβ spectrin heterodimers and six suspension complexes. The three pairs of αβ spectrin heterodimers spirally come down from the top of the protofilament and radiate outwards forming a hexagonal shape. These αβ spectrin heterodimers are attached to the lipid layer by a suspension complex, which are also made of proteins.

We have taken RBC for our study as it does not have a nucleus. The human RBCs do not have nucleus as it increases their oxygen carrying capacity. They generally act as vesicles carrying oxygen and carbon dioxide; so, they are not involved in cell division and growth and hence no need of a nucleus.

The mechanical properties of RBCs can be studied and measured experimentally by deforming them with a known force or stress and measuring the resulting deformation. Some of the widely used experimental techniques to study the mechanical properties of biological cells are atomic force microscope (AFM), the optical trap and micropipette aspiration (the experimental technique used in this project). Each of these experimental techniques has their own strengths and weaknesses. In AFM, the RBC surface is indented with a probe having constant velocity [23]. The probe being stiffer than the cell surface, the force produced in the AFM will be proportional to the indentation produced. The probe being stiffer small forces of the order 10-15 pN will be masked. In the optical trap, micron-sized refractile particles (like silica) are trapped in a laser beam focused through a high-numerical aperture microscope objective [24]. The trapped particle is touched to the surface of the RBC and the RBC is moved away from the adhered particle to get localized deformation of the cell membrane. The force is determined by the deflection of the particle in the trap perpendicular to the optical axis. This method has an advantage that no mechanical access to the cells is necessary. But optical traps do not give expected results when used to indent the RBC surface. In a micro-pipette aspiration experiment the cell membrane is pulled in the micro-pipette by controlled suction. One of the advantages of this experiment is that the cell can be either suspended in the solution or attached to a surface while the cell is bound to the micropipette and the negative pressure is applied. The micropipette aspiration can give measurements of three quantities: the cortical tension of the cell membrane; the cytoplasmic viscosity; and the cell elasticity [25]. In this project we try to aspirate the RBC.

The rest of the report is organized as follows. Chapter 2 contains the synthesis of tensegrity structures and the methodology followed in solving the problem. In chapter 3, the simulated examples with some of the fabricated tensegrity structures are shown. The micro-pipette aspiration experiment is discussed in chapter 4. Chapter 5 contains the summary and conclusions of the work.

Chapter 2
Towards Synthesis of Tensegrity Structures

For a free-standing tensegrity to be in static equilibrium, the net forces acting at each vertex should be zero. The internal and external forces in the member elements are resolved in the three directions and the sum of these resolved forces at each vertex is equated to zero to get the three equilibrium equations at each node. The equilibrium equations at vertex 1 in Fig. 2 can be written as:

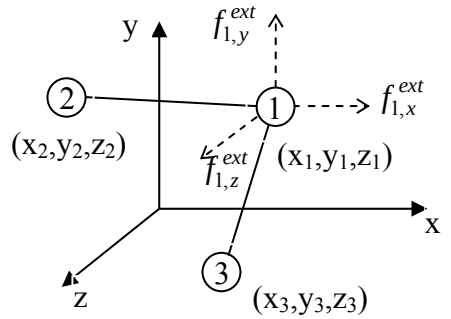


Fig.2: Equilibrium of vertex (node) 1in a tensegrity

$$\begin{aligned}
\frac{(x_1-x_2)f_{1-2}}{l_{1-2}}+\frac{(x_1-x_3)f_{1-3}}{l_{1-3}} &= f_{1,x}^{ext}\\
\frac{(y_1-y_2)f_{1-2}}{l_{1-2}}+\frac{(y_1-y_3)f_{1-3}}{l_{1-3}} &= f_{1,y}^{ext}\\
\frac{(z_1-z_2)f_{1-2}}{l_{1-2}}+\frac{(z_1-z_3)f_{1-3}}{l_{1-3}} &= f_{1,z}^{ext}
\end{aligned} \qquad (2)$$

where f_{i-j} and l_{i-j} are the internal force and length of an element connecting vertices i and j. For ease of calculation, we denote $\frac{f_{i-j}}{l_{i-j}}$ with q_{i-j}, where q_{i-j} is called the *force density*, i.e., force per unit length of the deformed element. Thus for the case where external loads are absent, we get:

$$\begin{aligned}
(x_1-x_2)q_{1-2}+(x_1-x_3)q_{1-3}&=0\\
(y_1-y_2)q_{1-2}+(y_1-y_3)q_{1-3}&=0\\
(z_1-z_2)q_{1-2}+(z_1-z_3)q_{1-3}&=0
\end{aligned} \qquad (3)$$

When there are n vertices, there will be $3n$ equations to be solved in m unknowns, m being the number of elements connecting n vertices. The number of elements in a tensegrity structure is selected by using Maxwell's rule [26], which states that the number of elements should be at least as many as $3n-6$ for a structure to be stiff. By Maxwell's rule it can be also seen that when $m>3n-6$, the structure will have $m-(3n-6)$ *states of self-stress*. Similarly when $m<3n-6$, the structure will have $(3n-6)-m$ *infinitesimal mode (mechanism).* A structure satisfying Maxwell's rule and having an infinitesimal mode also have a corresponding

state of self-stress or prestress. We choose m such that we have certain number of self-stress in our structure. As the tensegrity we are synthesizing are of class 1, no two bars will share a common vertex. Hence the number of bars in the structure will be $n/2$. The rest of the elements will constitute the cables of the structure.

The unknowns in Eq. (3) are the force densities $q_1, q_2, ..., q_m$. The resulting equations can be expressed in matrix form with the help of a connectivity matrix $\mathbf{C}$. Each row of matrix $\mathbf{C}$ represents the connectivity of the elements and each element in a particular row describes whether an element of the tensegrity structure is absent or an element begins or ends at that particular vertex. $\mathbf{C}$ is an $m \times n$ sparse matrix with m^{th} row having value 1 at the column corresponding to the vertex in which the member element emanates and -1 at the column corresponding to the node in which the element terminates with other elements of the row equal to zero. The sum of a row in matrix $\mathbf{C}$ should be zero. Thus we have:

$$\begin{aligned} \mathbf{C}^T diag(\mathbf{CX})\mathbf{q} &= 0 \\ \mathbf{C}^T diag(\mathbf{CY})\mathbf{q} &= 0 \\ \mathbf{C}^T diag(\mathbf{CZ})\mathbf{q} &= 0 \end{aligned} \tag{4}$$

where $\mathbf{X}, \mathbf{Y}$ and $\mathbf{Z}$ are $n \times 1$ vectors containing the x, y and z coordinates of the nodes respectively. $diag(\mathbf{CX})$ is an $m \times m$ diagonal matrix with vector $\mathbf{CX}$ constituting it's diagonal entries. Eq. (4) can be solved using singular value decomposition (SVD), but the resulting structure will be a tensegrity structure only for particular $\mathbf{C}$ matrices. In other cases although we may get solutions, it would merely result in trusses rather than a tensegrity structure. The tensegrity structure we solve for will be a linear combination of the solutions we get. Thus, we need to give constraints on the values of $\mathbf{q}$ so that the cables and bars have positive and negative values for $\mathbf{q}$ to ensure the tensegrity condition. So, we pose the problem as an optimization problem with constraints on the unknowns $\mathbf{q}$.

$$\begin{aligned} &\underset{\mathbf{q}}{Min} && (\mathbf{A} - \mathbf{f}^{ext})^2 \\ &where && \mathbf{A} = \begin{bmatrix} \mathbf{C}^T diag(\mathbf{CX})\mathbf{q} \\ \mathbf{C}^T diag(\mathbf{CY})\mathbf{q} \\ \mathbf{C}^T diag(\mathbf{CZ})\mathbf{q} \end{bmatrix}, \mathbf{f}^{ext} = \mathbf{0} \\ &Subject\ to && q_i < 0 && \forall i \in N_{bar} \\ & && -q_i < 0 && \forall i \in N_{cable} \\ &Data && \mathbf{C}, \mathbf{X}, \mathbf{Y}, \mathbf{Z}, \mathbf{f}^{ext}, N = N_{bars} \cup N_{cables} \end{aligned} \tag{5}$$

where N is the set of elements on the structure and N_{bar} is the set of elements that are bars and N_{cables} is the set of elements that are cables. After finding the force densities of each of the elements of the tensegrity structure we can find the free length of the individual elements for given cross-sectional area and Young's modulus of the material by fixing any two among the three of free length, cross-sectional area, and Young's modulus and finding the third one. We solve the minimization problem using the fmincon subroutine in MATLAB [27].

The second type of problem solved involves external forces. Here, we aim for the desired shape of the tensegrity under specified external load. The synthesis procedure for this is explained next.

Upon getting the desired structure, static analysis is done by minimization of potential energy approach. In static analysis of the tensegrity structure, we make following assumptions: (i) the bars are rigid and do not undergo any axial deformations or bending (ii) the strain energy of the cables and the work done by the external load contribute to the potential energy (iii) the cables can only take tensile load and become slack under compressive loads. We formulate the static problem as:

$$\begin{aligned} &\underset{g_i}{Min} \qquad PE = \sum_{k=1}^{n} \frac{1}{2S} K_k (L(g_i,l)_k - L_{0k})^2 - \sum_{m=1}^{r} F_m d(g_i,l)_m \\ &where \qquad S = 1 + e^{-sp\left(\frac{L(g_i)_k}{L_{0k}} - 1\right)} \\ &Data \qquad K, L_0, F, sp \end{aligned} \tag{6}$$

where K_i and L_{0i} are the stiffness and free lengths of cable in the tensegrity structure, l_i is the length of the bar, g_i is the generalized coordinate of the tensegrity structure in terms of the angles made by the bars with datum and final lengths of some cables. $L(g_i,l)$ and $d(g_i,l)$ are the final lengths of the cables and the displacements of the vertices at which the loads acts respectively. Both of them are functions of the generalized coordinates and the lengths of the bars. It may be noted that S is a sigmoid function used to account for the force-free slackness condition of the cables. The symbol sp is the steepness parameter of the sigmoid function.

The next step is to find the optimum height of a tensegrity structure which on application of a vertical load attains a desired shape. Equation (5) cannot be solved for all $\mathbf{f}^{ext} \neq 0$ (only a particular combination of the external load vector ($\mathbf{f}^{ext}$) can be solved) to get a tensegrity structure of desired shape. So, we pose the problem by imposing constraints on the force densities.

$$
\begin{aligned}
&\underset{a}{Min} && f = \left| h(a) - h^{*} \right| \\
&Subject\ to && \underset{q}{Min} && (\mathbf{A} - \mathbf{f}^{ext})^2 \\
& && Subject\ to && q_i < 0 && \forall i \in N_{bar} \\
& && && -q_i < 0 && \forall i \in N_{cable}
\end{aligned}
\tag{7}
$$

$$
\underset{x_i}{Min} \qquad PE = \sum_{k=1}^{n} \frac{1}{2S} K_k (L(x_i, l)_k - L_{0k})^2 - \sum_{m=1}^{r} F_m d(x_i, l)_m
$$

$$
where \quad S = 1 + e^{-sp\left(\frac{L(x_i)_k}{L_{0k}} - 1\right)}
$$

$$
Data \qquad \mathbf{C, X, Y, Z}, \mathbf{f}^{ext}, F, sp, N = N_{bars} \cup N_{cables}
$$

where K_i, L_{0i} and l_i are found from the output of Eq. (5) used in Eq. (7). h^{*} is the desired height the tensegrity structure should take under the application of load and a is the z coordinate of certain vertices of the initial free standing tensegrity. $h(a)$ is the height attained by the tensegrity structure which is a function of a. This is also an iterative process where the a changes until we get the required height.

Chapter 4
Results and Discussion

We use Eq. (5) to synthesize a previously known semi-toroidal tensegrity arch and an unknown tensegrity of biconcave shape of the cytoskeleton of a red blood cell. The semi-toroidal tensegrity arch (see Fig. 3) [28] is synthesized by first fixing the vertices according to the height and width of the arch. We take 48 vertices to make the outline of the arch. For a class 1 tensegrity there is no common vertex for two bars, leading to 24 bars. The connectivity is made such that three of the bars and nine cables form a unit tensegrity prism. With eight such prisms we can make the semi-toroidal tensegrity arch having 24 bars and 102 cables. We choose the cross-sectional areas of the bars and cables and the material (Young's modulus) used to fabricate them. Then, their free lengths can be solved using the force densities. We solve the optimization problem (Eq. (5)) to get the force densities and then use them to calculate the free lengths of the individual elements. The given data and results are indicated in Tables 1, 2 and 3. The assumed Young's modulus is 9 GPa and 35 GPa; and the area of cross-section is 3.85e-5 m^2 and 7.85e-7 m^2 for bars and cables respectively.

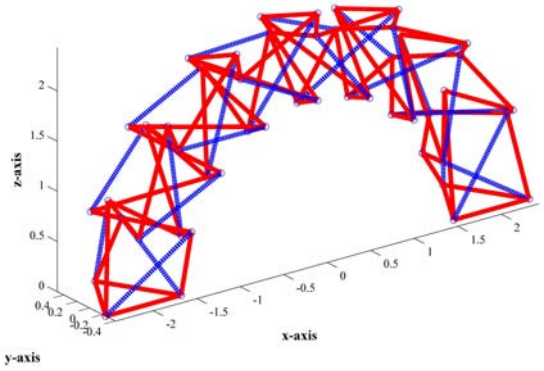


Fig. 3. A semi-toroidal tensegrity arch (blue dashed lines indicate the bars and the red lines indicate the cables).

Synthesis of the biconcave tensegrity (Fig. 5(a)) structure also involves the same steps as for the synthesis of the semi-toroidal arch. The nodal points are fixed on the surface generated by revolution of a Cassini's oval [29] corresponding to the size and shape of a red blood cell. The equation of a Cassini's oval is given by:

$$[(x-a)^2+y^2][(x+a)^2+y^2]=b^4$$
$$\text{where} \quad b=r_1 r_2$$
$$a=F_1+F_2 \tag{11}$$
$$\text{Diameter } d=2\sqrt{a^2+b^2}$$
$$\text{Thickness } t=\frac{b^2}{a}$$

where a and b are constants computed from the diameter and thickness data of a RBC. F_1 and F_2 are the two foci of the Cassini's oval and r_1 and r_2 are the distance between a point on the curve and the foci F_1 and F_2 respectively, ref. Fig. 4.

Table 1. The X, Y and Z coordinates (in m) of the 48 vertices used to construct the tensegrity arch.

SN	X	Y	Z	SN	X	Y	Z	SN	X	Y	Z
1	0.200	0.050	0.000	17	-0.093	-0.025	0.225	33	0.147	0.046	0.164
2	0.243	- 0.025	0.000	18	-0.060	-0.025	0.145	34	0.076	-0.040	0.217
3	0.157	-0.025	0.000	19	-0.141	0.050	0.141	35	0.050	-0.005	0.142
4	0.185	0.050	0.077	20	-0.172	-0.025	0.172	36	0.073	0.046	0.208
5	0.225	-0.025	0.093	21	-0.111	-0.025	0.111	37	-0.013	-0.040	0.229
6	0.145	-0.025	0.060	22	-0.185	0.050	0.077	38	-0.008	-0.005	0.150
7	0.141	0.050	0.141	23	-0.225	-0.025	0.093	39	-0.012	0.046	0.220
8	0.172	-0.025	0.172	24	-0.145	-0.025	0.060	40	-0.100	-0.040	0.207
9	0.111	-0.025	0.111	25	-0.200	0.050	0.000	41	-0.065	-0.005	0.135
10	0.077	0.050	0.185	26	-0.243	-0.025	0.000	42	-0.096	0.046	0.199
11	0.093	-0.025	0.225	27	-0.157	-0.025	0.000	43	-0.171	-0.040	0.153
12	0.060	-0.025	0.145	28	0.207	-0.040	0.100	44	-0.112	-0.005	0.100
13	0.000	0.050	0.200	29	0.135	-0.005	0.065	45	-0.164	0.046	0.147
14	0.000	-0.025	0.243	30	0.199	0.046	0.096	46	-0.217	-0.040	0.076
15	0.000	-0.025	0.157	31	0.153	-0.040	0.171	47	-0.142	-0.005	0.050
16	-0.077	0.050	0.185	32	0.100	-0.005	0.112	48	-0.208	0.046	0.073

We approximate the biconcave shape with 48 nodal points. A single Cassini's oval is approximated with 12 points as shown Fig. 4. The whole RBC model is made using four Cassini's ovals. We synthesize the biconcave tensegrity from three eight bar tensegrity prisms. Thus, for a biconcave tensegrity we have 24 bars and 112 cables. Tables 4 and 5 contain the data given to the problem. Table 6 contains the computed force densities and free lengths by solving the optimization problem of Eq. (5). Fig. 5(b) shows the prototype made of using wooden pencils and hemp thread. By fitting a taut triangular plane that replaces two tensioned cables and a bar, we can visualize the shape better than with the cables and bars. Such a model is shown in Fig. 5(c).

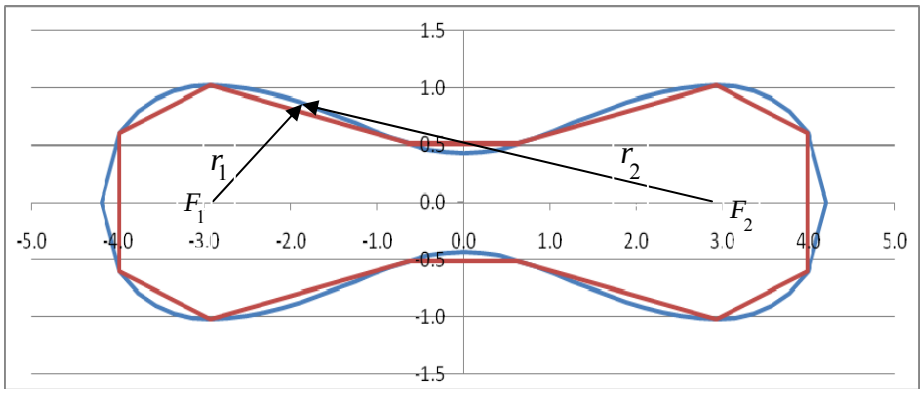


Fig. 4. A Cassini's oval modeled with diameter 8 μm and thickness 2 μm and the approximation curve made with 12 points.

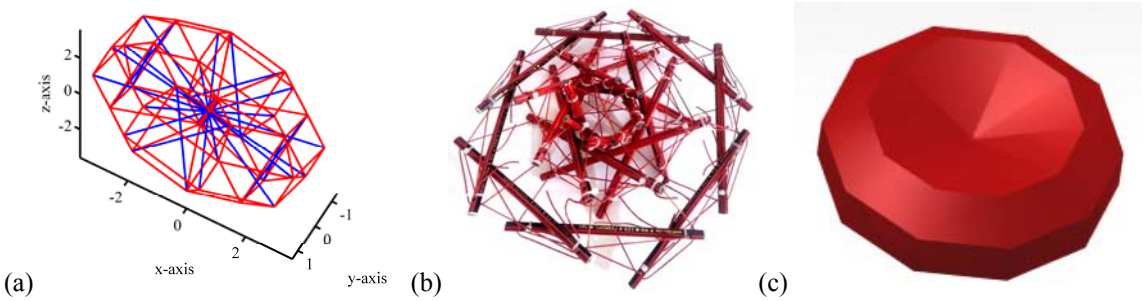


Fig. 5. (a) A biconcave tensegrity structure of a red blood cell (blue lines indicate the bars and the red lines indicate the cables). (b) A red blood cell prototype (scaled up) made with pencil and hemp threads. (c) Solid model of a red blood cell.

We also synthesized a tensegrity table that attains a desired height after application of load. The tensegrity table is made with three bars and nine cables and has six vertices. The vertices of the table are selected according to the area needed for the top of the table. The weight of a glass-top is the load. The heights (the z coordinates) of the first three vertices are selected to be as they rest on the ground (in x-y plane). The heights of the other three are randomly chosen for the initial analysis. The load on the table acts vertically down (i.e., in the $-z$ direction) at the three top vertices. The deformation of the table under load can be seen in the Figs. 6. (a) – (b). The table was synthesized such that the table has a height of 0.6 m for a load of 600 N. Tables 7 contains the coordinates of the vertices taken as an initial guess. Table 8 contains the connectivity data of the elements and the force densities and free lengths of the elements obtained after solving the optimization problem of Eq. (7).

Table 2. The connectivity of 96 elements that make the semi-torroidal tensegrity arch (last 24 elements (i.e. 103-126) are bars).

SN	Vertex 1	Vertex 2	SN	Vertex 1	Vertex 2	SN	Vertex 1	Vertex 2	SN	Vertex 1	Vertex 2
1	1	2	33	40	18	65	17	45	96	46	26
2	1	3	34	44	21	66	18	43	97	1	5
3	2	3	35	21	43	67	19	47	98	2	4
4	29	6	36	43	20	68	20	48	99	3	6
5	6	28	37	20	45	69	21	46	100	47	25
6	28	5	38	45	19	70	22	27	101	48	26
7	5	30	39	19	47	71	23	26	102	46	27
8	30	4	40	47	24	72	24	25	103	1	28
9	4	29	41	24	46	73	1	4	104	2	29
10	32	9	42	46	23	74	2	5	105	3	30
11	9	31	43	23	48	75	3	6	106	4	31
12	31	8	44	48	22	76	29	9	107	5	32
13	8	33	45	22	47	77	30	7	108	6	33
14	33	7	46	26	25	78	28	8	109	7	34
15	7	32	47	25	27	79	32	12	110	8	35
16	35	12	48	27	26	80	33	10	111	9	36
17	12	34	49	1	29	81	31	11	112	10	37
18	34	11	50	2	30	82	35	15	113	11	38
19	11	36	51	3	28	83	36	13	114	12	39
20	36	10	52	4	32	84	34	14	115	13	40
21	10	35	53	5	33	85	38	18	116	14	41
22	15	38	54	6	31	86	39	16	117	15	42
23	38	13	55	7	35	87	37	17	118	16	43
24	13	39	56	8	36	88	41	21	119	17	44
25	39	14	57	9	34	89	42	19	120	18	45
26	14	37	58	10	38	90	40	20	121	19	46
27	37	15	59	11	39	91	44	24	122	20	47
28	18	41	60	12	37	92	45	22	123	21	48
29	41	16	61	13	41	93	43	23	124	22	25
30	16	42	62	14	42	94	47	27	125	23	27
31	42	17	63	15	40	95	48	25	126	24	26
32	17	40	64	16	44						

Table 3. The force density and length of the individual elements of the semi-torroidal tensegrity arch prototype.

SN	Force density (N/m)	Length (m)	SN	Force density (N/m)	Length (m)	SN	Force density (N/m)	Length (m)	SN	Force density (N/m)	Length (m)
1	0.014	0.086	33	0.035	0.073	65	0.027	0.124	96	0.005	0.090
2	0.017	0.085	34	0.193	0.019	66	0.031	0.109	97	0.008	0.121
3	0.011	0.086	35	0.026	0.073	67	0.027	0.105	98	0.004	0.122
4	0.174	0.020	36	0.149	0.022	68	0.006	0.126	99	0.015	0.060
5	0.028	0.073	37	0.034	0.073	69	0.023	0.110	100	0.012	0.067
6	0.201	0.021	38	0.166	0.021	70	0.038	0.074	101	0.002	0.098
7	0.046	0.072	39	0.019	0.074	71	0.009	0.098	102	0.005	0.120
8	0.204	0.020	40	0.002	0.023	72	0.020	0.066	103	-0.036	0.135
9	0.020	0.074	41	0.017	0.074	73	0.014	0.077	104	-0.027	0.128
10	0.228	0.019	42	0.099	0.023	74	0.013	0.094	105	-0.036	0.126
11	0.028	0.073	43	0.038	0.073	75	0.015	0.060	106	-0.051	0.135
12	0.225	0.021	44	0.111	0.022	76	0.014	0.055	107	-0.048	0.128
13	0.050	0.072	45	0.011	0.074	77	0.009	0.073	108	-0.045	0.126
14	0.216	0.020	46	0.009	0.086	78	0.019	0.081	109	-0.056	0.135
15	0.020	0.074	47	0.011	0.086	79	0.031	0.054	110	-0.055	0.128
16	0.245	0.018	48	0.012	0.086	80	0.018	0.072	111	-0.059	0.127
17	0.032	0.073	49	0.017	0.106	81	0.023	0.080	112	-0.060	0.135
18	0.245	0.020	50	0.002	0.096	82	0.023	0.054	113	-0.059	0.128
19	0.057	0.072	51	0.019	0.111	83	0.023	0.072	114	-0.060	0.127
20	0.255	0.019	52	0.049	0.103	84	0.023	0.080	115	-0.056	0.135
21	0.022	0.074	53	0.012	0.126	85	0.021	0.054	116	-0.060	0.128
22	0.235	0.019	54	0.040	0.109	86	0.023	0.072	117	-0.053	0.127
23	0.017	0.074	55	0.051	0.102	87	0.028	0.080	118	-0.043	0.135
24	0.253	0.020	56	0.018	0.125	88	0.019	0.055	119	-0.054	0.128
25	0.059	0.072	57	0.042	0.108	89	0.017	0.072	120	-0.049	0.127
26	0.257	0.020	58	0.057	0.102	90	0.015	0.081	121	-0.032	0.135
27	0.031	0.073	59	0.019	0.125	91	0.012	0.055	122	-0.024	0.128
28	0.231	0.019	60	0.049	0.108	92	0.033	0.071	123	-0.035	0.126
29	0.015	0.074	61	0.059	0.102	93	0.009	0.081	124	-0.021	0.120
30	0.208	0.020	62	0.019	0.125	94	0.001	0.081	125	-0.029	0.126
31	0.044	0.073	63	0.039	0.109	95	0.004	0.118	126	-0.015	0.123
32	0.224	0.021	64	0.048	0.103						

Table 4. The X, Y and Z coordinates (in m) of the 48 vertices used to construct the RBC biconcave model.

SN	X	Y	Z	SN	X	Y	Z	SN	X	Y	Z
1	0.057	-0.362	0	17	0.133	0.184	0.184	33	-0.053	0	0.008
2	0.133	-0.261	0	18	0.057	0.256	0.256	34	-0.053	0	-0.008
3	0.053	-0.008	0	19	-0.057	0.256	0.256	35	-0.133	0	-0.261
4	0.053	0.008	0	20	-0.133	0.184	0.184	36	-0.057	0	-0.362
5	0.133	0.261	0	21	-0.053	0.006	0.006	37	0.057	0.256	-0.256
6	0.057	0.362	0	22	-0.053	-0.006	-0.006	38	0.133	0.184	-0.184
7	-0.057	0.362	0	23	-0.133	-0.184	-0.184	39	0.053	0.006	-0.006
8	-0.133	0.261	0	24	-0.057	-0.256	-0.256	40	0.053	-0.006	0.006
9	-0.053	0.008	0	25	0.057	0.000	0	41	0.133	-0.184	0.184
10	-0.053	-0.008	0	26	0.133	0.000	0	42	0.057	-0.256	0.256
11	-0.133	-0.261	0	27	0.053	0.000	0	43	-0.057	-0.256	0.256
12	-0.057	-0.362	0	28	0.053	0.000	0	44	-0.133	-0.184	0.184
13	0.057	-0.256	-0.256	29	0.133	0.000	0	45	-0.053	-0.006	0.006
14	0.133	-0.184	-0.184	30	0.057	0.000	0	46	-0.053	0.006	-0.006
15	0.053	-0.006	-0.006	31	-0.057	0.000	0	47	-0.133	0.184	-0.184
16	0.053	0.006	0.006	32	-0.133	0.000	0	48	-0.057	0.256	-0.256

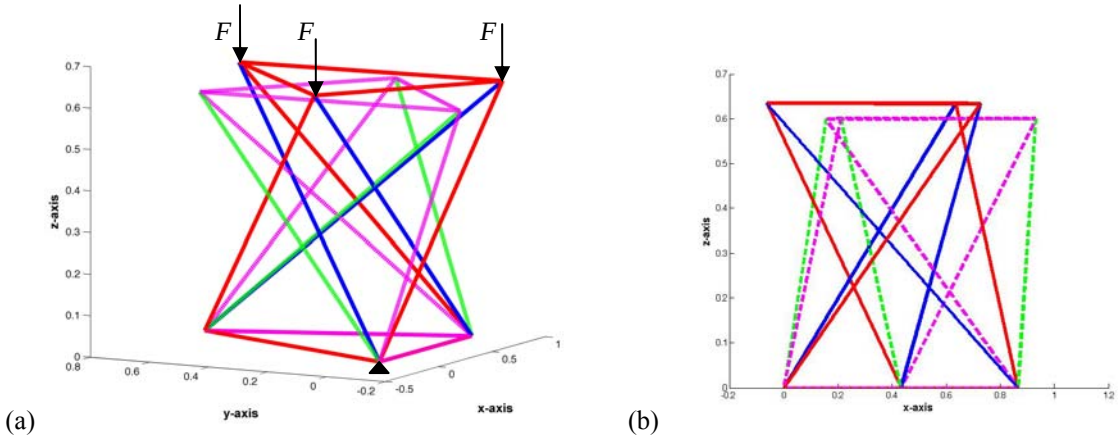
The designed tensegrity models are fabricated and tested. The semi-toroidal tensegrity arch and the biconcave tensegrity model of red blood cell are fabricated using wooden cylinders (pencil) and hemp thread (pencils act as bars and the hemp thread act as the cables). The tensegrity table is fabricated using aluminum rods for bars and braided stainless steel wire for cables. It is shown in Fig. 7(a-b). The table is stable and is currently used in our lab. As the weight of the glass top is nearly 40 kg, any small object (up-to a weight of 20 kg) placed on top of it does not disturb the stability of the structure.

Table 5. The connectivity of 136 elements to make the RBC biconcave model (last 24 elements (i.e. 113-136) are bars).

SN	vertex 1	vertex 2	SN	vertex 1	vertex 2	SN	vertex1	vertex 2	SN	vertex 1	vertex 2
1	1	2	35	35	36	69	5	38	103	34	22
2	2	3	36	36	25	70	38	26	104	22	10
3	3	4	37	37	38	71	26	14	105	3	10
4	4	5	38	38	39	72	14	2	106	40	45
5	5	6	39	39	40	73	11	44	107	28	33
6	6	7	40	40	41	74	44	32	108	16	21
7	7	8	41	41	42	75	32	20	109	4	9
8	8	9	42	42	43	76	20	8	110	39	46
9	9	10	43	43	44	77	8	47	111	27	34
10	10	11	44	44	45	78	47	35	112	15	22
11	11	12	45	45	46	79	35	23	113	1	43
12	12	1	46	46	47	80	23	11	114	42	31
13	13	14	47	47	48	81	11	2	115	30	19
14	14	15	48	48	37	82	41	44	116	18	7
15	15	16	49	1	42	83	32	29	117	6	48
16	16	17	50	42	30	84	17	20	118	37	36
17	17	18	51	30	18	85	5	8	119	25	24
18	18	19	52	18	6	86	47	38	120	13	12
19	19	20	53	6	37	87	35	26	121	2	44
20	20	21	54	37	25	88	14	23	122	41	32
21	21	22	55	25	13	89	3	40	123	29	20
22	22	23	56	13	1	90	40	28	124	8	17
23	23	24	57	12	43	91	28	16	125	5	47
24	24	13	58	43	31	92	16	4	126	35	38
25	25	26	59	31	19	93	4	39	127	26	23
26	26	27	60	19	7	94	39	27	128	11	14
27	27	28	61	7	48	95	27	15	129	45	3
28	28	29	62	48	36	96	15	3	130	40	33
29	29	30	63	36	24	97	10	45	131	28	21
30	30	31	64	24	12	98	45	33	132	16	9
31	31	32	65	2	41	99	33	21	133	4	46
32	32	33	66	41	29	100	21	9	134	39	34
33	33	34	67	29	17	101	9	46	135	27	22
34	34	35	68	17	5	102	46	34	136	15	10

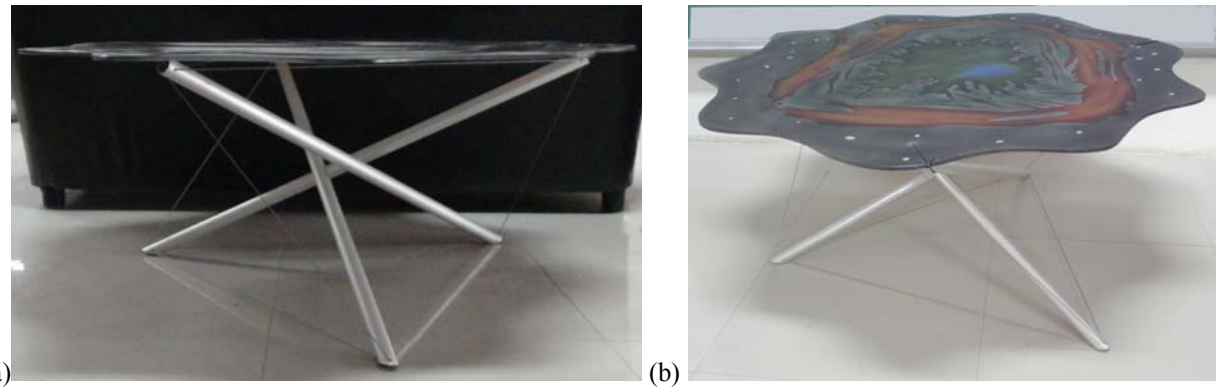
Table 6. The force density and free-length of the individual elements of the red blood cell biconcave prototype.

SN	Force density (N/m)	Length (m)	SN	Force density (N/m)	Length (m)	SN	Force density (N/m)	Length (m)	SN	Force density (N/m)	Length (m)
1	0.008	0.125	35	0.008	0.125	69	0.004	0.198	103	0.223	0.005
2	0.004	0.263	36	0.009	0.113	70	0.004	0.198	104	0.223	0.005
3	0.033	0.016	37	0.008	0.125	71	0.004	0.198	105	0.113	0.090
4	0.004	0.263	38	0.004	0.263	72	0.004	0.198	106	0.113	0.090
5	0.008	0.125	39	0.033	0.016	73	0.004	0.198	107	0.113	0.090
6	0.009	0.113	40	0.004	0.263	74	0.004	0.198	108	0.113	0.090
7	0.008	0.125	41	0.008	0.125	75	0.004	0.198	109	0.113	0.090
8	0.004	0.263	42	0.009	0.113	76	0.004	0.198	110	0.113	0.090
9	0.033	0.016	43	0.008	0.125	77	0.004	0.198	111	0.113	0.090
10	0.004	0.263	44	0.004	0.263	78	0.004	0.198	112	0.113	0.090
11	0.008	0.125	45	0.033	0.016	79	0.004	0.198	113	-0.003	0.300
12	0.009	0.113	46	0.004	0.263	80	0.004	0.198	114	-0.003	0.300
13	0.008	0.125	47	0.008	0.125	81	0.002	0.265	115	-0.003	0.300
14	0.004	0.263	48	0.009	0.113	82	0.002	0.265	116	-0.003	0.300
15	0.033	0.016	49	0.002	0.276	83	0.002	0.265	117	-0.003	0.300
16	0.004	0.263	50	0.002	0.276	84	0.002	0.265	118	-0.003	0.300
17	0.008	0.125	51	0.002	0.276	85	0.002	0.265	119	-0.003	0.300
18	0.009	0.113	52	0.002	0.276	86	0.002	0.265	120	-0.003	0.300
19	0.008	0.125	53	0.002	0.276	87	0.002	0.265	121	-0.005	0.332
20	0.004	0.263	54	0.002	0.276	88	0.002	0.265	122	-0.005	0.332
21	0.033	0.016	55	0.002	0.276	89	0.222	0.005	123	-0.005	0.332
22	0.004	0.263	56	0.002	0.276	90	0.222	0.005	124	-0.005	0.332
23	0.008	0.125	57	0.002	0.276	91	0.223	0.005	125	-0.005	0.332
24	0.009	0.113	58	0.002	0.276	92	0.223	0.005	126	-0.005	0.332
25	0.008	0.125	59	0.002	0.276	93	0.222	0.005	127	-0.005	0.332
26	0.004	0.263	60	0.002	0.276	94	0.223	0.005	128	-0.005	0.332
27	0.033	0.016	61	0.002	0.276	95	0.222	0.005	129	-0.109	0.108
28	0.004	0.263	62	0.002	0.276	96	0.222	0.005	130	-0.109	0.108
29	0.008	0.125	63	0.002	0.276	97	0.222	0.005	131	-0.109	0.108
30	0.009	0.113	64	0.002	0.276	98	0.222	0.005	132	-0.109	0.108
31	0.008	0.125	65	0.004	0.198	99	0.222	0.005	133	-0.109	0.108
32	0.004	0.263	66	0.004	0.198	100	0.223	0.005	134	-0.109	0.108
33	0.033	0.016	67	0.004	0.198	101	0.223	0.005	135	-0.109	0.108
34	0.004	0.263	68	0.004	0.198	102	0.223	0.005	136	-0.109	0.108



(a) (b)

Fig. 6. (a & b) Two views of the tensegrity table under the application of load. Free standing tensegrity table (blue lines indicate the bars and the red solid lines indicate the cables). The table under an applied external load (green dashed lines indicate the bars and the magenta dashed lines indicate the cables).



(a) (b)

Fig. 7. Two views of the tensegrity table. The entire weight of the glass-top of the stable is supported by the tensegrity structure.

Table 7. The X, Y and Z coordinates of the 6 vertices used to construct the tensegrity table. (Z coordinate of the vertices 4,5 and 6 are iterated)

SN	X(m)	Y(m)	Z(m)
1	0	0	0
2	0.866	0	0
3	0.433	0.750	0
4	0.636	0.707	0.7
5	-0.064	0.198	0.7
6	0.727	-0.155	0.7

Table 8. The connectivity of 12 elements (last 3(i.e. 10-12) are bars), force densities and initial lengths of individual elements to make the tensegrity table.

SN	vertex1	vertex2	Force Density (N/m)	Length (m)
1	1	2	0.200	0.863
2	2	3	0.200	0.863
3	3	1	0.200	0.863
4	4	5	0.200	0.863
5	5	6	0.200	0.863
6	6	4	0.200	0.863
7	1	5	0.363	0.664
8	2	6	0.363	0.664
9	3	4	0.363	0.663
10	1	4	-0.348	1.144
11	2	5	-0.348	1.144
12	3	6	-0.348	1.144

Chapter 4
Micropipette Aspiration of RBC

Aspiration of a cell into a glass micro-pipette using suction pressure is one of the techniques used to estimate the mechanical properties of cells. In this chapter we describe attempts made to aspirate RBCs, one of the smallest cells in the human body.

The first step involved in micro-pipette aspiration of RBC is to isolate them from the whole blood. Following steps are followed in the isolation of RBC from the whole blood received from the Health Center Lab, IISc:

1. To 400 µl of blood sample, 400 µl of histopaque-1077 (Sigma life science) is added and centrifuged at 2000 rpm for 10 min in the centrifuge (Hermle Inc.).
2. The supernatant is removed and 500 µl of saline is added to the pellet and centrifuged at 2000 rpm for 10 min. This step is repeated two more times.
3. 400 µl Percoll (Sigma life science) of density 1.1 g/ml (made from 1 ml percoll and 1.25 ml of 1.5 molar NaCl) is added to the remaining pellet and centrifuged at 2000 rpm for 10 min.
4. Once again the top layer is removed and 500 µl of saline is added to the pellets and centrifuged at 2000 rpm for 10 min. This step is repeated two more times.

Crenation (Fig. 8(b)) of RBC was observed when PBS (Phosphate Buffer Solution) was first used in place of saline. Crenation happens in RBCs when the solute concentration inside the cell is lower than that of the solute concentration in medium outside. The water in the cell diffuses out by osmosis and brings the solute concentration inside the cell equal to that of the outside medium. But, as the solution goes out, there is a reduction in the volume of the cytoplasm whereas the surface area of the cell remains constant. This prompts the cell surface to form abnormal notches.

Our setup for the micro-pipette aspiration consists of 3-degree-of-freedom (dof) XYZ micro-positioner (MP-285, Sutter Inc.), an inverted microscope (IX81, Olympus Inc.), a CCD (charged coupled device) camera (DP72, Olympus Inc.), a micro-injector (Xenoworks, Sutter Inc.), and a micro-pipette. All of these can be seen in Fig. 10. Pipettes are pulled using micro-pipette puller (Sutter Inc.) to get a bee-stinger micro-pipette. Bee-stinger micro-pipettes (Fig. 9 (a))are micro-pipettes having a shorter length and a sudden taper at the tip. They are useful in making micro-pipettes with smaller inner diameters with lesser drag during experiment when they are drawn through the solution due to their short height. Later, these micro-pipettes are forged to desired diameter using a micro-forge (Narshige Inc.). Once the pipettes are prepared, they are suitably positioned using the micro-positioner. The RBCs are aspirated under the inverted microscope. Fig 8(a) shows one such experiment in which the cell is drawn inside the pipette by applying a suitable suction pressure (in the order of hPa).

The suction force required to aspirate the cell into the pipette can be calculatred by:

$$F_0 = P(\pi R_p^2) \tag{8}$$

where F_0 is the suction force, P is the suction pressure and R_p is the inner radius of the micro-pipette. It is to be noted that the value of the force is assumed to be at a point but the actual pressure that the cell experiences is distributed in nature.

For a cell modeled as an elastic body, its Young's modulus E can be found out by [30]:

$$\Delta P = \frac{2\pi}{3} E \frac{L_p}{R_p} \phi \tag{9}$$

where ΔP is the pressure difference inside and outside the pipette, L_P is the length of the cell pulled into the pipette with radius R_P, and ϕ is a geometric constant with a value around 2.1 [30].

For liquid like flow of cells at pressures exceeding the cortical tension, the cyto-plasmic viscosity is calculated by [31]:

$$\eta = \frac{R_p \Delta P}{\left(\frac{dL_p}{dt}\right) m\left(1 - R_p/R\right)} \tag{10}$$

where η is the viscosity, R is the diameter of the cell outside the pipette, and m is a constant with a value around 9 [30].

The micro-pipette aspiration of RBC was conducted. Only 2 RBCs were aspirated and their values are given in table 9.

Table 9. Observations in micro-pipette aspiration of RBCs

cell	Pipette diameter (μm)	Pressure (hPa)
RBC	2.1	122
	2.1	155

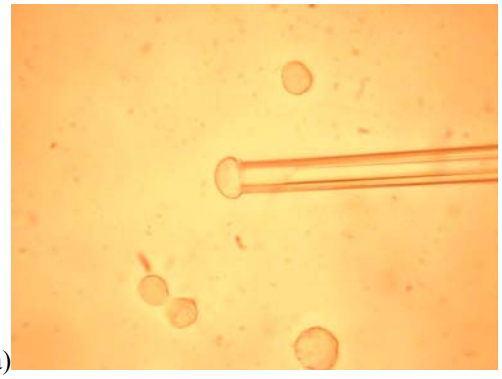
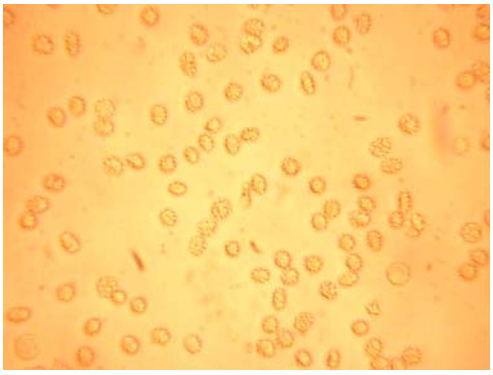
(a)  (b) 

Fig.8. (a) A RBC aspirated using a micro-pipette. (b) crenated RBC.

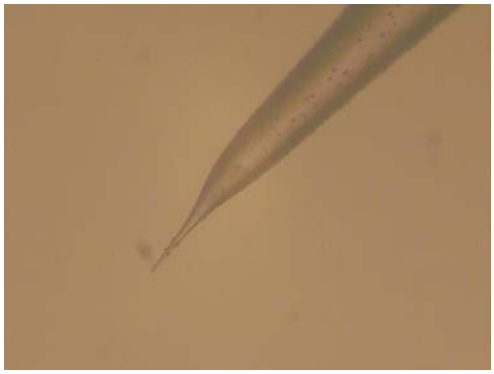
(a)  (b) 

Fig.9. (a) A bee-stinger micro-pipette under 5X zoom. (b) An RBC and the bee-stinger micro-pipette tip under (100x1.6) zoom.

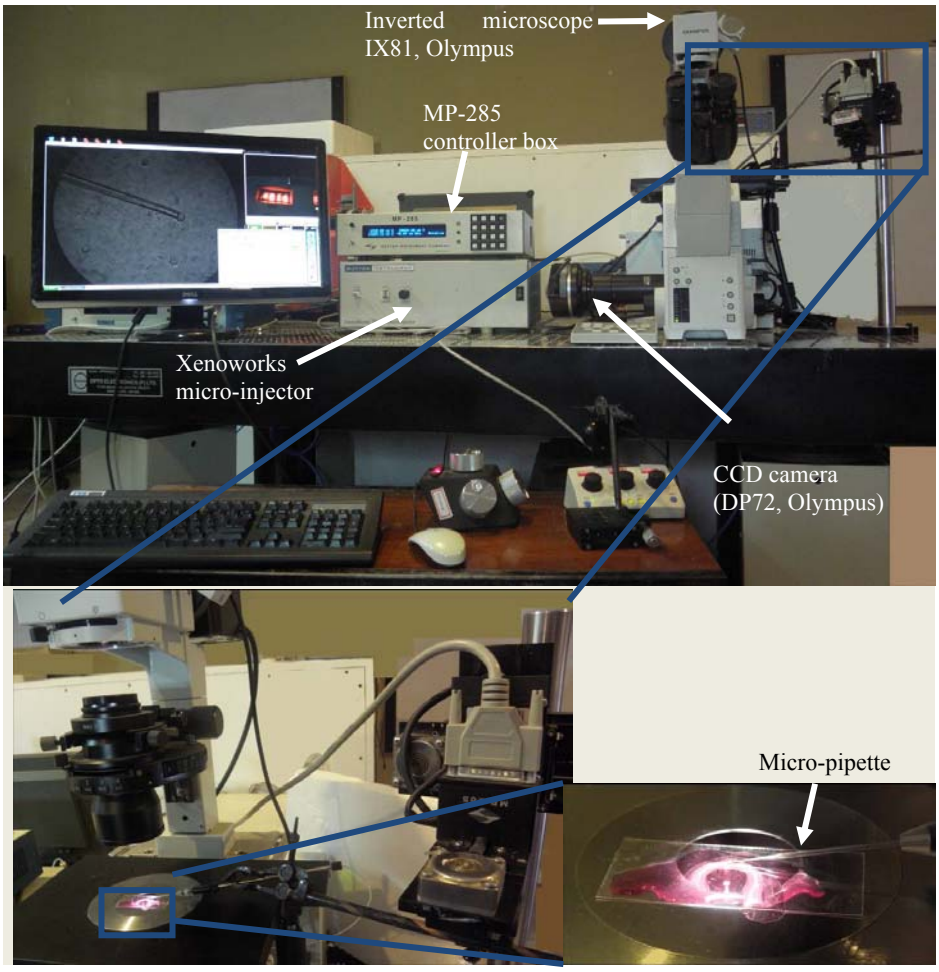


Fig.10. The set up of the micro-pipette aspiration experiment.

Chapter 5
Summary and conclusions

In this report, we developed an optimization method to synthesize tensegrity structures of desired shapes. The coordinates of the vertices and the connectivity of the elements are the inputs to the problem, which we use to solve for the force densities in each of the member elements using the equilibrium equations at the vertices. Constraints are imposed on the force densities to accommodate tensions in bars and compressions in cables. The following conclusions can be drawn from the work presented here.

- The solution is unique for a given set of positions of the vertices and their connectivity with bars and cables. The free lengths of the elements change according to different materials assigned but the force densities do not change.
- Solving the constrained optimization problem posed for the purpose of synthesis of the tensegrity structure of desired shape is equivalent to constructing the null-space of the combined connectivity-coordinate matrix. When the rank of the null-space is more than unity, tensegrity structures can be constructed using linear combinations of the basis vectors of the null-space.

The micro-pipette aspiration of RBC was conducted, for which the RBCs were isolated from the whole blood. It is noticed that the RBCs are aspirated into the pipette at a single pressure step i.e., there is no separate base pressure and initiation pressure for RBCs. The force needed to aspirate a RBC is about 1.92e-10 N [32]. When the diameter of the pipette is more than 1.55 μm, the suction pressure becomes less than 100 Pa. The micro-injector (Xenoworks, Sutter Inc.) has minimum pressure increment 100 Pa, thus the difference in the initiation pressure and base pressure cannot be calculated. When the pipette diameter is smaller than 1.55 μm, the adequate suction pressure needed to start the aspiration is not built.

Appendix

A. 1

```
%%% program to synthesize a tensegrity prism:
%%% Given Connectivity matrix, Coordinates and location of bars
clc;
clear all;
global Con X Y Z
%%% Connectivity matrix
Con = [1 -1  0  0  0  0;
    0  1 -1  0  0  0;
    1  0 -1  0  0  0;
    0  0  0  1 -1  0;
    0  0  0  0  1 -1;
    0  0  0  1  0 -1;
    1  0  0  0  0 -1;
    0  1  0 -1  0  0;
    0  0  1  0 -1  0;
    1  0  0  -1 0  0;
    0  1  0  0 -1  0;
    0  0  1  0  0  -1];
%%% coordinates of the structure
X = [  0  .433   -.433   0.2939 -0.4973  0.2034]';
Y = [ .5 -.25     -.25  -0.4045 -0.0523  0.4568]';
Z = [  0   0         0   .6    .6    .6]';
%%% force constraint matrices
A = [ -1 0 0 0 0 0 0 0 0 0 0 0;
    0 -1 0 0 0 0 0 0 0 0 0 0;
    0 0 -1 0 0 0 0 0 0 0 0 0;
    0 0 0 -1 0 0 0 0 0 0 0 0;
    0 0 0 0 -1 0 0 0 0 0 0 0;
    0 0 0 0 0 -1 0 0 0 0 0 0;
    0 0 0 0 0 0 -1 0 0 0 0 0;
    0 0 0 0 0 0 0 -1 0 0 0 0;
    0 0 0 0 0 0 0 0 -1 0 0 0;
    0 0 0 0 0 0 0 0 0 1  0  0;
    0 0 0 0 0 0 0 0 0  0 1  0;
    0 0 0 0 0 0 0 0 0  0 0  1];
b = [ 0 0 0 0 0 0 0 0 0 0 0 0 ]';
%%% Solving for force densities
options = optimset('Algorithm','interior-point');
x1 = [ 1 1 1 1 1 1 1 1 -1 -1 -1]';
[q,fval,exitflag,output,lambda,grad,hessian] =
fmincon(@tensegrity_table_points1,x1,A,b,[],[],[],[],[],options);
q
exitflag
fval
A1 = Con'*diag(Con*X);
A2 = Con'*diag(Con*Y);
A3 = Con'*diag(Con*Z);
C = vertcat(A1,A2,A3);
d = [ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]';
f1 = C*q-d
%%% Normalising q
```

```
tot = 0;
for i = 1:length(x1)
        tot = tot+(q(i))^2;
end
tot = sqrt(tot);
for i = 1:length(x1)
    norm_q(i) = q(i)/tot;
end
norm_q = norm_q'
%%% Area and Young's modulus values of the elements
A_strut = pi()*(0.032^2-0.0288^2)/4;
E_strut = 69e9;
A_cable = pi()*0.002^2/4;
E_cable = 125e9;
K_S = A_strut*E_strut;
K_C = A_cable*E_cable;
K = [ K_C K_C K_C K_C K_C K_C K_C K_C K_C K_S K_S K_S]';
I_strut = pi()*(0.032^4-0.0288^4)/64;
%%% finding length of the members
F_strut = zeros(3,1);
F_buckling = zeros(3,1);
len = length(Con);
length11 = ones(len,1);
for i = 1:len
    for j = 1:6
        if(Con(i,j)==1)
            x_c1 = X(j);
            y_c1 = Y(j);
            z_c1 = Z(j);
        end
        if(Con(i,j)==-1)
            x_c2 = X(j);
            y_c2 = Y(j);
            z_c2 = Z(j);
        end
    end
    length11(i) = sqrt((x_c1-x_c2)^2+(y_c1-y_c2)^2+(z_c1-z_c2)^2);
end
%%% allowable force calculation
FoS = 4;
strength_cable = 1.77e9;
stress_cable_allow = strength_cable/FoS;
F_cable_allow = stress_cable_allow*A_cable;
factr = norm_q(1);
for i = 1:length(x1)
    q(i) = F_cable_allow*norm_q(i)/(factr*length11(i));
end
%%% calculating the actual force from force densities
F = q.*length11
R_1 = F./K;
R = R_1+1;
length11
length01 = ones(len,1);
length01 = length11./R
F_act = zeros(12,1);
for i = 1:12
```

```
    if i<10
        F_act(i) = A_cable*E_cable*(length11(i)-length01(i))/length01(i);
    else
        F_act(i) = A_strut*E_strut*(length11(i)-length01(i))/length01(i);
    end
end
for i = 1:3
    F_strut(i) = F(i+9);
    F_buckling(i) = -1*(pi())^2*E_strut*I_strut/(length01(i+9))^2;
end
F_strut;
F_act
F_buckling
pre_stress = zeros(12,1);
for i = 1:12
    if i<10
        pre_stress(i) = F_act(i)/A_cable;
    else
        pre_stress(i) = F_act(i)/A_strut;
    end
end
pre_stress;
%% plots
X1=[X(1,1);X(2,1);X(3,1);X(1,1);X(6,1);X(5,1);X(4,1);X(6,1);X(5,1);X(3,1);X(2
,1);X(4,1)];
Y1=[Y(1,1);Y(2,1);Y(3,1);Y(1,1);Y(6,1);Y(5,1);Y(4,1);Y(6,1);Y(5,1);Y(3,1);Y(2
,1);Y(4,1)];
Z1=[Z(1,1);Z(2,1);Z(3,1);Z(1,1);Z(6,1);Z(5,1);Z(4,1);Z(6,1);Z(5,1);Z(3,1);Z(2
,1);Z(4,1)];
P1=[X(1,1);X(4,1)];Q1=[Y(1,1);Y(4,1)];R1=[Z(1,1);Z(4,1)];
P2=[X(2,1);X(5,1)];Q2=[Y(2,1);Y(5,1)];R2=[Z(2,1);Z(5,1)];
P3=[X(3,1);X(6,1)];Q3=[Y(3,1);Y(6,1)];R3=[Z(3,1);Z(6,1)];
plot3(X1,Y1,Z1,'Color','r','LineWidth',4);
hold on;
plot3(P1,Q1,R1,P2,Q2,R2,P3,Q3,R3,'Color','b','LineWidth',4);
hold on;
Radi = 0.6;
X2 = [-.6 : 0.01 : .6];
for i=1:length(X2)
    Y21(i) = sqrt(Radi^2-X2(i)^2);
    Y22(i) = -sqrt(Radi^2-X2(i)^2);
end
Z2 = 0.6*ones(length(X2),1);
plot3(X2,Y21,Z2,'Color','m','LineWidth',4);
hold on;
plot3(X2,Y22,Z2,'Color','m','LineWidth',4);
axis equal;
[U,S,V] = svd(C,0)
norm(C*norm_q)
alpha = V'*norm_q

%% Function to solve the static equilibrium equations at the nodes
function f = tensegrity_table_points1(x)
global Con X Y Z
%%% Constructing the static equilibrium matix for given connectivity and
%%% coordinates
```

```
A1 = Con'*diag(Con*X);
A2 = Con'*diag(Con*Y);
A3 = Con'*diag(Con*Z);
C = vertcat(A1,A2,A3);
%%% external force vector
d = [ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]';
f1 = C*x-d;
f = f1'*f1;
```

Appendix
A. 2

```
%%% program to synthesize a semi-torroidal tensegrity arch:
%%% Given Connectivity matrix, Coordinates and location of bars
clc;
clear all;
clf;
global X Y Z Con
%%% coordinates for constructing the arch
X = [  .2  .2433   .1567]';
X1 = [.2259  .15  .2241]';
Y = [ .05 -.025     -.025]';
Y1 = [-0.0428 -0.001  0.0438]';
Z = [  0   0         0]';
Z1 = [0    0    0]';

n_secs = 8;
n_ang = pi()/n_secs;
Cor = horzcat(X,Y,Z);
New_pt = zeros((n_secs-1)*3,3);

k = 0;
for i = 1:n_secs
    angl = i*n_ang;

    Tr_mat = [cos(angl) 0 -sin(angl);
        0 1 0;
        sin(angl) 0  cos(angl)];
    for j = 1:3
        M = Cor(j,:)';
        N = Tr_mat*M;
        k = k+1;
        New_pt(k,:) = N;
    end

end
New_pt = vertcat(Cor,New_pt);

ang = pi()/7;

Tr_mat = [cos(ang) 0 -sin(ang);
    0 1 0;
    sin(ang) 0  cos(ang)];
Cor = horzcat(X1,Y1,Z1);
New_Cor = Cor;
k=0;
for j = 1:3
    M = Cor(j,:)';
    N = Tr_mat*M;
    k = k+1;
    New_Cor(k,:) = N;
end
```

```
New_pt1 = zeros((n_secs-2)*3,3);
Cor = New_Cor;
k = 0;
for i = 1:(n_secs-2)
    angl = i*n_ang;

    Tr_mat = [cos(angl) 0 -sin(angl);
        0 1 0;
        sin(angl) 0  cos(angl)];
   for j = 1:3
        M = Cor(j,:)';
        N = Tr_mat*M;
        k = k+1;
        New_pt1(k,:) = N;
    end
 end

New_pt1 = vertcat(Cor,New_pt1);
New_pt = vertcat(New_pt,New_pt1);
length(New_pt)

for i = 1:length(New_pt)
    % for i = 22:24
    plot3(New_pt(i,1),New_pt(i,2),New_pt(i,3),'marker','o')
    hold on
end
X = New_pt(:,1);
Y = New_pt(:,2);
Z = New_pt(:,3);

X(25,1) = 0.0259-.2;
X(26,1) = -.25;
X(27,1) = 0.0241-.2;
Y(25,1) = -0.0428;
Y(26,1) = -0.001;
Y(27,1) = 0.0438;

%% connectivity matrix
Con = zeros(126,48);

Con(1,1) = 1; Con(1,2) = -1;
Con(2,1) = 1; Con(2,3) = -1;
Con(3,2) = 1; Con(3,3) = -1;

Con(4,29) = 1; Con(4,6) = -1;
Con(5,6) = 1; Con(5,28) = -1;
Con(6,28) = 1; Con(6,5) = -1;
Con(7,5) = 1; Con(7,30) = -1;
Con(8,30) = 1; Con(8,4) = -1;
Con(9,4) = 1; Con(9,29) = -1;

Con(10,32) = 1; Con(10,9) = -1;
Con(11,9) = 1; Con(11,31) = -1;
Con(12,31) = 1; Con(12,8) = -1;
Con(13,8) = 1; Con(13,33) = -1;
```

```
Con(14,33) = 1; Con(14,7) = -1;
Con(15,7) = 1; Con(15,32) = -1;

Con(16,35) = 1; Con(16,12) = -1;
Con(17,12) = 1; Con(17,34) = -1;
Con(18,34) = 1; Con(18,11) = -1;
Con(19,11) = 1; Con(19,36) = -1;
Con(20,36) = 1; Con(20,10) = -1;
Con(21,10) = 1; Con(21,35) = -1;

Con(22,15) = 1; Con(22,38) = -1;
Con(23,38) = 1; Con(23,13) = -1;
Con(24,13) = 1; Con(24,39) = -1;
Con(25,39) = 1; Con(25,14) = -1;
Con(26,14) = 1; Con(26,37) = -1;
Con(27,37) = 1; Con(27,15) = -1;

Con(28,18) = 1; Con(28,41) = -1;
Con(29,41) = 1; Con(29,16) = -1;
Con(30,16) = 1; Con(30,42) = -1;
Con(31,42) = 1; Con(31,17) = -1;
Con(32,17) = 1; Con(32,40) = -1;
Con(33,40) = 1; Con(33,18) = -1;

Con(34,44) = 1; Con(34,21) = -1;
Con(35,21) = 1; Con(35,43) = -1;
Con(36,43) = 1; Con(36,20) = -1;
Con(37,20) = 1; Con(37,45) = -1;
Con(38,45) = 1; Con(38,19) = -1;
Con(39,19) = 1; Con(39,44) = -1;

Con(40,47) = 1; Con(40,24) = -1;
Con(41,24) = 1; Con(41,46) = -1;
Con(42,46) = 1; Con(42,23) = -1;
Con(43,23) = 1; Con(43,48) = -1;
Con(44,48) = 1; Con(44,22) = -1;
Con(45,22) = 1; Con(45,47) = -1;

Con(46,26) = 1; Con(46,25) = -1;
Con(47,25) = 1; Con(47,27) = -1;
Con(48,27) = 1; Con(48,26) = -1;

Con(49,1) = 1; Con(49,29) = -1;
Con(40,2) = 1; Con(50,30) = -1;
Con(51,3) = 1; Con(51,28) = -1;
Con(52,4) = 1; Con(52,32) = -1;
Con(53,5) = 1; Con(53,33) = -1;
Con(54,6) = 1; Con(54,31) = -1;
Con(55,7) = 1; Con(55,35) = -1;
Con(56,8) = 1; Con(56,36) = -1;
Con(57,9) = 1; Con(57,34) = -1;
Con(58,10) = 1; Con(58,38) = -1;
Con(59,11) = 1; Con(59,39) = -1;
Con(60,12) = 1; Con(60,37) = -1;
Con(61,13) = 1; Con(61,41) = -1;
```

```
Con(62,14) = 1; Con(62,42) = -1;
Con(63,15) = 1; Con(63,40) = -1;
Con(64,16) = 1; Con(64,44) = -1;
Con(65,17) = 1; Con(65,45) = -1;
Con(66,18) = 1; Con(66,43) = -1;
Con(67,19) = 1; Con(67,47) = -1;
Con(68,20) = 1; Con(68,48) = -1;
Con(69,21) = 1; Con(69,46) = -1;

Con(70,22) = 1; Con(70,26) = -1;
Con(71,23) = 1; Con(71,27) = -1;
Con(72,24) = 1; Con(72,25) = -1;

Con(73,1) = 1; Con(73,4) = -1;
Con(74,2) = 1; Con(74,5) = -1;
Con(75,3) = 1; Con(75,6) = -1;
Con(76,29) = 1; Con(76,9) = -1;
Con(77,30) = 1; Con(77,7) = -1;
Con(78,28) = 1; Con(78,8) = -1;
Con(79,32) = 1; Con(79,12) = -1;
Con(80,33) = 1; Con(80,10) = -1;
Con(81,31) = 1; Con(81,11) = -1;
Con(82,35) = 1; Con(82,15) = -1;
Con(83,36) = 1; Con(83,13) = -1;
Con(84,34) = 1; Con(84,14) = -1;
Con(85,38) = 1; Con(85,18) = -1;
Con(86,39) = 1; Con(86,16) = -1;
Con(87,37) = 1; Con(87,17) = -1;
Con(88,41) = 1; Con(88,21) = -1;
Con(89,42) = 1; Con(89,19) = -1;
Con(90,40) = 1; Con(90,20) = -1;
Con(91,44) = 1; Con(91,24) = -1;
Con(92,45) = 1; Con(92,22) = -1;
Con(93,43) = 1; Con(93,23) = -1;
Con(94,47) = 1; Con(94,27) = -1;
Con(95,48) = 1; Con(95,25) = -1;
Con(96,46) = 1; Con(96,26) = -1;

Con(97,1) = 1; Con(97,5) = -1;
Con(98,2) = 1; Con(98,6) = -1;
Con(99,3) = 1; Con(99,4) = -1;
Con(100,46) = 1; Con(100,25) = -1;
Con(101,47) = 1; Con(101,26) = -1;
Con(102,48) = 1; Con(102,27) = -1;

% % bars
Con(103,1) = 1; Con(103,30) = -1;
Con(104,2) = 1; Con(104,28) = -1;
Con(105,3) = 1; Con(105,29) = -1;
Con(106,4) = 1; Con(106,33) = -1;
Con(107,5) = 1; Con(107,31) = -1;
Con(108,6) = 1; Con(108,32) = -1;
Con(109,7) = 1; Con(109,36) = -1;
Con(110,8) = 1; Con(110,34) = -1;
Con(111,9) = 1; Con(111,35) = -1;
```

```
Con(112,10) = 1; Con(112,39) = -1;
Con(113,11) = 1; Con(113,37) = -1;
Con(114,12) = 1; Con(114,38) = -1;
Con(115,13) = 1; Con(115,42) = -1;
Con(116,14) = 1; Con(116,40) = -1;
Con(117,15) = 1; Con(117,41) = -1;
Con(118,16) = 1; Con(118,45) = -1;
Con(119,17) = 1; Con(119,43) = -1;
Con(120,18) = 1; Con(120,44) = -1;
Con(121,19) = 1; Con(121,48) = -1;
Con(122,20) = 1; Con(122,46) = -1;
Con(123,21) = 1; Con(123,47) = -1;
Con(124,22) = 1; Con(124,25) = -1;
Con(125,23) = 1; Con(125,26) = -1;
Con(126,24) = 1; Con(126,27) = -1;


%% Analysis
%
lim = -1*ones(102,1);
lim1 = ones(24,1);
lim = vertcat(lim,lim1);

A = diag(lim);
b = zeros(126,1);
% Solving for force densities
% put MaxFunEvals = 150000 & defaultopt.TolX = 1e-5 in fmincon
options = optimset('Algorithm','interior-point');
x1 = 0.5.*-lim;
[q,fval,exitflag,output,lambda,grad,hessian] =
fmincon(@tensegrity_arch_8,x1,A,b,[],[],[],[],[],options);
q
fval
exitflag

% Normalising q
tot = 0;
for i = 1:length(x1)
    tot = tot+(q(i))^2;
end
tot = sqrt(tot);
for i = 1:length(x1)
    norm_q(i) = q(i)/tot;
end

norm_q = norm_q'

% Area and Young's modulus values of the elements
A_strut = pi()*(.007^2)/4;
E_strut = 9e9;
% E_strut = 69e4;
A_cable = pi()*0.001^2/4;
E_cable = 35e9;
% E_cable = 125e5;
K_S = A_strut*E_strut;
```

```
K_C = A_cable*E_cable;
K1 = K_C*ones(102,1);
K2 = K_S*ones(24,1);
K = vertcat(K1,K2);
I_strut = pi()*(0.007^4)/64;

% finding length of the members

F_strut = zeros(24,1);
F_buckling = zeros(24,1);

len = length(Con);
length11 = ones(len,1);

for i = 1:len
    for j = 1:48
        if(Con(i,j)==1)
            x_c1 = X(j);
            y_c1 = Y(j);
            z_c1 = Z(j);

        end
        if(Con(i,j)==-1)
            x_c2 = X(j);
            y_c2 = Y(j);
            z_c2 = Z(j);

        end
    end
    length11(i) = sqrt((x_c1-x_c2)^2+(y_c1-y_c2)^2+(z_c1-z_c2)^2);

end


% allowable force calculation
FoS = 4;
strength_cable = 1.77e9;
stress_cable_allow = strength_cable/FoS;
F_cable_allow = stress_cable_allow*A_cable;

factr = norm_q(1);

for i = 1:length(x1)
    q(i) = F_cable_allow*norm_q(i)/(factr*length11(i));
end
% calculating the actual force from force densities0
F = q.*length11;
length(F);
length(K);
nums = 1:126;
nums = nums';

F_1 = horzcat(nums,F)
```

```
R_1 = F./K;
R = R_1+1;
length11

length01 = ones(len,1);
length01 = length11./R
F_act = zeros(12,1);
for i = 1:126
    if i<103
        F_act(i) = A_cable*E_cable*(length11(i)-length01(i))/length01(i);
    else
        F_act(i) = A_strut*E_strut*(length11(i)-length01(i))/length01(i);
    end
end


for i = 1:24
    F_strut(i) = F(i+102);
    F_buckling(i) = -1*(pi())^2*E_strut*I_strut/length01(i+102);
end

F_strut;
F_act;
F_buckling

%% plot

P1=[X(1,1);X(28,1)];Q1=[Y(1,1);Y(28,1)];R1=[Z(1,1);Z(28,1)];
P2=[X(2,1);X(29,1)];Q2=[Y(2,1);Y(29,1)];R2=[Z(2,1);Z(29,1)];
P3=[X(3,1);X(30,1)];Q3=[Y(3,1);Y(30,1)];R3=[Z(3,1);Z(30,1)];
P4=[X(4,1);X(31,1)];Q4=[Y(4,1);Y(31,1)];R4=[Z(4,1);Z(31,1)];
P5=[X(5,1);X(32,1)];Q5=[Y(5,1);Y(32,1)];R5=[Z(5,1);Z(32,1)];
P6=[X(6,1);X(33,1)];Q6=[Y(6,1);Y(33,1)];R6=[Z(6,1);Z(33,1)];
P7=[X(7,1);X(34,1)];Q7=[Y(7,1);Y(34,1)];R7=[Z(7,1);Z(34,1)];
P8=[X(8,1);X(35,1)];Q8=[Y(8,1);Y(35,1)];R8=[Z(8,1);Z(35,1)];
P9=[X(9,1);X(36,1)];Q9=[Y(9,1);Y(36,1)];R9=[Z(9,1);Z(36,1)];
P10=[X(10,1);X(37,1)];Q10=[Y(10,1);Y(37,1)];R10=[Z(10,1);Z(37,1)];
P11=[X(11,1);X(38,1)];Q11=[Y(11,1);Y(38,1)];R11=[Z(11,1);Z(38,1)];
P12=[X(12,1);X(39,1)];Q12=[Y(12,1);Y(39,1)];R12=[Z(12,1);Z(39,1)];
P13=[X(13,1);X(40,1)];Q13=[Y(13,1);Y(40,1)];R13=[Z(13,1);Z(40,1)];
P14=[X(14,1);X(41,1)];Q14=[Y(14,1);Y(41,1)];R14=[Z(14,1);Z(41,1)];
P15=[X(15,1);X(42,1)];Q15=[Y(15,1);Y(42,1)];R15=[Z(15,1);Z(42,1)];
P16=[X(16,1);X(43,1)];Q16=[Y(16,1);Y(43,1)];R16=[Z(16,1);Z(43,1)];
P17=[X(17,1);X(44,1)];Q17=[Y(17,1);Y(44,1)];R17=[Z(17,1);Z(44,1)];
P18=[X(18,1);X(45,1)];Q18=[Y(18,1);Y(45,1)];R18=[Z(18,1);Z(45,1)];
P19=[X(19,1);X(46,1)];Q19=[Y(19,1);Y(46,1)];R19=[Z(19,1);Z(46,1)];
P20=[X(20,1);X(47,1)];Q20=[Y(20,1);Y(47,1)];R20=[Z(20,1);Z(47,1)];
P21=[X(21,1);X(48,1)];Q21=[Y(21,1);Y(48,1)];R21=[Z(21,1);Z(48,1)];
P22=[X(22,1);X(25,1)];Q22=[Y(22,1);Y(25,1)];R22=[Z(22,1);Z(25,1)];
P23=[X(23,1);X(26,1)];Q23=[Y(23,1);Y(26,1)];R23=[Z(23,1);Z(26,1)];
P24=[X(24,1);X(27,1)];Q24=[Y(24,1);Y(27,1)];R24=[Z(24,1);Z(27,1)];

X1=[X(1,1);X(30,1)];Y1=[Y(1,1);Y(30,1)];Z1=[Z(1,1);Z(30,1)];
X2=[X(2,1);X(28,1)];Y2=[Y(2,1);Y(28,1)];Z2=[Z(2,1);Z(28,1)];
X3=[X(3,1);X(29,1)];Y3=[Y(3,1);Y(29,1)];Z3=[Z(3,1);Z(29,1)];
```

```
X4=[X(4,1);X(33,1)];Y4=[Y(4,1);Y(33,1)];Z4=[Z(4,1);Z(33,1)];
X5=[X(5,1);X(31,1)];Y5=[Y(5,1);Y(31,1)];Z5=[Z(5,1);Z(31,1)];
X6=[X(6,1);X(32,1)];Y6=[Y(6,1);Y(32,1)];Z6=[Z(6,1);Z(32,1)];
X7=[X(7,1);X(36,1)];Y7=[Y(7,1);Y(36,1)];Z7=[Z(7,1);Z(36,1)];
X8=[X(8,1);X(34,1)];Y8=[Y(8,1);Y(34,1)];Z8=[Z(8,1);Z(34,1)];
X9=[X(9,1);X(35,1)];Y9=[Y(9,1);Y(35,1)];Z9=[Z(9,1);Z(35,1)];
X10=[X(10,1);X(39,1)];Y10=[Y(10,1);Y(39,1)];Z10=[Z(10,1);Z(39,1)];
X11=[X(11,1);X(37,1)];Y11=[Y(11,1);Y(37,1)];Z11=[Z(11,1);Z(37,1)];
X12=[X(12,1);X(38,1)];Y12=[Y(12,1);Y(38,1)];Z12=[Z(12,1);Z(38,1)];
X13=[X(13,1);X(42,1)];Y13=[Y(13,1);Y(42,1)];Z13=[Z(13,1);Z(42,1)];
X14=[X(14,1);X(40,1)];Y14=[Y(14,1);Y(40,1)];Z14=[Z(14,1);Z(40,1)];
X15=[X(15,1);X(41,1)];Y15=[Y(15,1);Y(41,1)];Z15=[Z(15,1);Z(41,1)];
X16=[X(16,1);X(45,1)];Y16=[Y(16,1);Y(45,1)];Z16=[Z(16,1);Z(45,1)];
X17=[X(17,1);X(43,1)];Y17=[Y(17,1);Y(43,1)];Z17=[Z(17,1);Z(43,1)];
X18=[X(18,1);X(44,1)];Y18=[Y(18,1);Y(44,1)];Z18=[Z(18,1);Z(44,1)];
X19=[X(19,1);X(48,1)];Y19=[Y(19,1);Y(48,1)];Z19=[Z(19,1);Z(48,1)];
X20=[X(20,1);X(46,1)];Y20=[Y(20,1);Y(46,1)];Z20=[Z(20,1);Z(46,1)];
X21=[X(21,1);X(47,1)];Y21=[Y(21,1);Y(47,1)];Z21=[Z(21,1);Z(47,1)];
X22=[X(22,1);X(26,1)];Y22=[Y(22,1);Y(26,1)];Z22=[Z(22,1);Z(26,1)];
X23=[X(23,1);X(27,1)];Y23=[Y(23,1);Y(27,1)];Z23=[Z(23,1);Z(27,1)];
X24=[X(24,1);X(25,1)];Y24=[Y(24,1);Y(25,1)];Z24=[Z(24,1);Z(25,1)];

X25=[X(3,1);X(1,1);X(2,1);X(3,1)];Y25=[Y(3,1);Y(1,1);Y(2,1);Y(3,1)];
Z25=[Z(3,1);Z(1,1);Z(2,1);Z(3,1)];
X26=[X(6,1);X(28,1);X(5,1);X(30,1);X(4,1);X(29,1);X(6,1)];
Y26=[Y(6,1);Y(28,1);Y(5,1);Y(30,1);Y(4,1);Y(29,1);Y(6,1)];
Z26=[Z(6,1);Z(28,1);Z(5,1);Z(30,1);Z(4,1);Z(29,1);Z(6,1)];
X27=[X(9,1);X(31,1);X(8,1);X(33,1);X(7,1);X(32,1);X(9,1)];
Y27=[Y(9,1);Y(31,1);Y(8,1);Y(33,1);Y(7,1);Y(32,1);Y(9,1)];
Z27=[Z(9,1);Z(31,1);Z(8,1);Z(33,1);Z(7,1);Z(32,1);Z(9,1)];
X28=[X(12,1);X(34,1);X(11,1);X(36,1);X(10,1);X(35,1);X(12,1)];
Y28=[Y(12,1);Y(34,1);Y(11,1);Y(36,1);Y(10,1);Y(35,1);Y(12,1)];
Z28=[Z(12,1);Z(34,1);Z(11,1);Z(36,1);Z(10,1);Z(35,1);Z(12,1)];
X29=[X(15,1);X(37,1);X(14,1);X(39,1);X(13,1);X(38,1);X(15,1)];
Y29=[Y(15,1);Y(37,1);Y(14,1);Y(39,1);Y(13,1);Y(38,1);Y(15,1)];
Z29=[Z(15,1);Z(37,1);Z(14,1);Z(39,1);Z(13,1);Z(38,1);Z(15,1)];
X30=[X(18,1);X(40,1);X(17,1);X(42,1);X(16,1);X(41,1);X(18,1)];
Y30=[Y(18,1);Y(40,1);Y(17,1);Y(42,1);Y(16,1);Y(41,1);Y(18,1)];
Z30=[Z(18,1);Z(40,1);Z(17,1);Z(42,1);Z(16,1);Z(41,1);Z(18,1)];
X31=[X(21,1);X(43,1);X(20,1);X(45,1);X(19,1);X(44,1);X(21,1)];
Y31=[Y(21,1);Y(43,1);Y(20,1);Y(45,1);Y(19,1);Y(44,1);Y(21,1)];
Z31=[Z(21,1);Z(43,1);Z(20,1);Z(45,1);Z(19,1);Z(44,1);Z(21,1)];
X32=[X(24,1);X(46,1);X(23,1);X(48,1);X(22,1);X(47,1);X(24,1)];
Y32=[Y(24,1);Y(46,1);Y(23,1);Y(48,1);Y(22,1);Y(47,1);Y(24,1)];
Z32=[Z(24,1);Z(46,1);Z(23,1);Z(48,1);Z(22,1);Z(47,1);Z(24,1)];
X33=[X(26,1);X(25,1);X(27,1);X(26,1)];
Y33=[Y(26,1);Y(25,1);Y(27,1);Y(26,1)];
Z33=[Z(26,1);Z(25,1);Z(27,1);Z(26,1)];

X34=[X(1,1);X(4,1)];Y34=[Y(1,1);Y(4,1)];Z34=[Z(1,1);Z(4,1)];
X35=[X(1,1);X(5,1)];Y35=[Y(1,1);Y(5,1)];Z35=[Z(1,1);Z(5,1)];
X36=[X(2,1);X(6,1)];Y36=[Y(2,1);Y(6,1)];Z36=[Z(2,1);Z(6,1)];
X37=[X(2,1);X(5,1)];Y37=[Y(2,1);Y(5,1)];Z37=[Z(2,1);Z(5,1)];
X38=[X(3,1);X(6,1)];Y38=[Y(3,1);Y(6,1)];Z38=[Z(3,1);Z(6,1)];
X39=[X(3,1);X(4,1)];Y39=[Y(3,1);Y(4,1)];Z39=[Z(3,1);Z(4,1)];
```

```
X40=[X(25,1);X(48,1)];Y40=[Y(25,1);Y(48,1)];Z40=[Z(25,1);Z(48,1)];
X41=[X(25,1);X(46,1)];Y41=[Y(25,1);Y(46,1)];Z41=[Z(25,1);Z(46,1)];
X42=[X(26,1);X(46,1)];Y42=[Y(26,1);Y(46,1)];Z42=[Z(26,1);Z(46,1)];
X43=[X(26,1);X(47,1)];Y43=[Y(26,1);Y(47,1)];Z43=[Z(26,1);Z(47,1)];
X44=[X(27,1);X(47,1)];Y44=[Y(27,1);Y(47,1)];Z44=[Z(27,1);Z(47,1)];
X45=[X(27,1);X(48,1)];Y45=[Y(27,1);Y(48,1)];Z45=[Z(27,1);Z(48,1)];

X46=[X(29,1);X(9,1)];Y46=[Y(29,1);Y(9,1)];Z46=[Z(29,1);Z(9,1)];
X47=[X(30,1);X(7,1)];Y47=[Y(30,1);Y(7,1)];Z47=[Z(30,1);Z(7,1)];
X48=[X(28,1);X(8,1)];Y48=[Y(28,1);Y(8,1)];Z48=[Z(28,1);Z(8,1)];

X49=[X(32,1);X(12,1)];Y49=[Y(32,1);Y(12,1)];Z49=[Z(32,1);Z(12,1)];
X50=[X(33,1);X(10,1)];Y50=[Y(33,1);Y(10,1)];Z50=[Z(33,1);Z(10,1)];
X51=[X(31,1);X(11,1)];Y51=[Y(31,1);Y(11,1)];Z51=[Z(31,1);Z(11,1)];
X52=[X(35,1);X(15,1)];Y52=[Y(35,1);Y(15,1)];Z52=[Z(35,1);Z(15,1)];
X53=[X(36,1);X(13,1)];Y53=[Y(36,1);Y(13,1)];Z53=[Z(36,1);Z(13,1)];
X54=[X(34,1);X(14,1)];Y54=[Y(34,1);Y(14,1)];Z54=[Z(34,1);Z(14,1)];
X55=[X(38,1);X(18,1)];Y55=[Y(38,1);Y(18,1)];Z55=[Z(38,1);Z(18,1)];
X56=[X(39,1);X(16,1)];Y56=[Y(39,1);Y(16,1)];Z56=[Z(39,1);Z(16,1)];
X57=[X(37,1);X(17,1)];Y57=[Y(37,1);Y(17,1)];Z57=[Z(37,1);Z(17,1)];
X58=[X(41,1);X(21,1)];Y58=[Y(41,1);Y(21,1)];Z58=[Z(41,1);Z(21,1)];
X59=[X(42,1);X(19,1)];Y59=[Y(42,1);Y(19,1)];Z59=[Z(42,1);Z(19,1)];
X60=[X(40,1);X(20,1)];Y60=[Y(40,1);Y(20,1)];Z60=[Z(40,1);Z(20,1)];
X61=[X(44,1);X(24,1)];Y61=[Y(44,1);Y(24,1)];Z61=[Z(44,1);Z(24,1)];
X62=[X(45,1);X(22,1)];Y62=[Y(45,1);Y(22,1)];Z62=[Z(45,1);Z(22,1)];
X63=[X(43,1);X(23,1)];Y63=[Y(43,1);Y(23,1)];Z63=[Z(43,1);Z(23,1)];

plot3(P1,Q1,R1,'--',P2,Q2,R2,'--',P3,Q3,R3,'--',P4,Q4,R4,'--',P5,Q5,R5,'--
',P6,Q6,R6,'--',P7,Q7,R7,'--',P8,Q8,R8,'--',P9,Q9,R9,'--
','Color','b','LineWidth',4);
hold on;
plot3(P10,Q10,R10,'--',P11,Q11,R11,'--',P12,Q12,R12,'--',P13,Q13,R13,'--
',P14,Q14,R14,'--',P15,Q15,R15,'--',P16,Q16,R16,'--
','Color','b','LineWidth',4);
hold on;
plot3(P17,Q17,R17,'--',P18,Q18,R18,'--',P19,Q19,R19,'--',P20,Q20,R20,'--
',P21,Q21,R21,'--',P22,Q22,R22,'--',P23,Q23,R23,'--',P24,Q24,R24,'--
','Color','b','LineWidth',4);
hold on;

plot3(X1,Y1,Z1,X2,Y2,Z2,X3,Y3,Z3,X4,Y4,Z4,X5,Y5,Z5,X6,Y6,Z6,X7,Y7,Z7,X8,Y8,Z8
,'Color','r','LineWidth',4);
hold on;
plot3(X9,Y9,Z9,X10,Y10,Z10,'Color','r','LineWidth',4);
hold on;
plot3(X11,Y11,Z11,X12,Y12,Z12,X13,Y13,Z13,X14,Y14,Z14,X15,Y15,Z15,X16,Y16,Z16
,X17,Y17,Z17,X18,Y18,Z18,'Color','r','LineWidth',4);
hold on;
plot3(X19,Y19,Z19,X20,Y20,Z20,X21,Y21,Z21,X22,Y22,Z22,X23,Y23,Z23,X24,Y24,Z24
,'Color','r','LineWidth',4);
hold on;
plot3(X25,Y25,Z25,X26,Y26,Z26,X27,Y27,Z27,X28,Y28,Z28,X29,Y29,Z29,X30,Y30,Z30
,X31,Y31,Z31,X32,Y32,Z32,X33,Y33,Z33,'Color','r','LineWidth',4);
hold on;
```

```
plot3(X34,Y34,Z34,X35,Y35,Z35,X36,Y36,Z36,X37,Y37,Z37,X38,Y38,Z38,X39,Y39,Z39
,'Color','r','LineWidth',4);
hold on;
plot3(X40,Y40,Z40,X41,Y41,Z41,X42,Y42,Z42,X43,Y43,Z43,X44,Y44,Z44,X45,Y45,Z45
,'Color','r','LineWidth',4);
hold on;
plot3(X46,Y46,Z46,X47,Y47,Z47,X48,Y48,Z48,'Color','r','LineWidth',4);
hold on;
plot3(X49,Y49,Z49,X50,Y50,Z50,X51,Y51,Z51,'Color','r','LineWidth',4);
hold on;
plot3(X52,Y52,Z52,X53,Y53,Z53,X54,Y54,Z54,'Color','r','LineWidth',4);
hold on;
plot3(X55,Y55,Z55,X56,Y56,Z56,X57,Y57,Z57,'Color','r','LineWidth',4);
hold on;
plot3(X58,Y58,Z58,X59,Y59,Z59,X60,Y60,Z60,'Color','r','LineWidth',4);
hold on;
plot3(X61,Y61,Z61,X62,Y62,Z62,X63,Y63,Z63,'Color','r','LineWidth',4);
hold on;
xlabel('x-axis','fontsize',12,'fontweight','b');
ylabel('y-axis','fontsize',12,'fontweight','b');
zlabel('z-axis','fontsize',12,'fontweight','b');
axis equal;
%% alpha calculation
A1 = Con'*diag(Con*X);
A2 = Con'*diag(Con*Y);
A3 = Con'*diag(Con*Z);
C = vertcat(A1,A2,A3);
F_ext = C*norm_q
[U,S,V] = svd(C,0);
S
alpha = V'*norm_q


%% minimization function
function f = tensegrity_arch_8(x)
global Con X Y Z
A1 = Con'*diag(Con*X);
A2 = Con'*diag(Con*Y);
A3 = Con'*diag(Con*Z);
%%% static equilibrium equations
C = vertcat(A1,A2,A3);
%%% external load vector
d = zeros(144,1);
f1 = C*x-d;
f=f1'*f1;
```

Appendix
A. 3

```
%%% program to synthesize a biconcave rbc tensegrity structure:
%%% Given Connectivity matrix, Coordinates and location of bars
clc;
clear all;
global Con X Y Z q A len length11
%%% Coordinates on rbc surface
Y = [  0.0573  .1328  0.0529  0.0529  0.1328  0.0573 -0.0573 -.1328 -0.0529 -
0.0529 -.1328 -0.0573]';
X = [ -.3616 -.2606 -0.0084  0.0084  .2606  .3616  .3616  .2606  0.0084 -
0.0084 -.2606 -.3616]';
Z = [  0  0  0  0  0  0  0  0  0  0  0  0]';

Points = zeros(3,1);
Y_new = zeros(length(Y),3);
X_new = zeros(length(X),3);
Z_new = zeros(length(Z),3);
for j = 1:3
    alpha = pi()*j/4;
    Trans_mat = [ 1 0 0; 0 cos(alpha) -sin(alpha); 0 sin(alpha) cos(alpha)];
    for i = 1:12
    Points(1,1) = Y(i);
    Points(2,1) = X(i);
    Points(3,1) = Z(i);
    new_points = Trans_mat*Points;
    Y_new(i,j) = new_points(1,1);
    X_new(i,j) = new_points(2,1);
    Z_new(i,j) = new_points(3,1);
    end
end
Y_n = Y_new(:);
X_n = X_new(:);
Z_n = Z_new(:);
Y = vertcat(Y,Y_n);
X = vertcat(X,X_n);
Z = vertcat(Z,Z_n);
%% Connectivity Matrix
Con = zeros(126,48);
for i = 1:11
    Con(i,i) = 1;
    Con(i,i+1) = -1;
end
Con(12,1) = -1;
Con(12,12) = 1;
for i = 13:23
    Con(i,i) = 1;
    Con(i,i+1) = -1;
end
Con(24,13) = -1;
Con(24,24) = 1;
for i = 25:35
```

```
    Con(i,i) = 1;
    Con(i,i+1) = -1;
end
Con(36,25) = -1;
Con(36,36) = 1;
for i = 37:47
    Con(i,i) = 1;
    Con(i,i+1) = -1;
end
Con(48,37) = -1;
Con(48,48) = 1;
Con(49,1) = 1; Con(49,42) = -1;
Con(50,42) = 1; Con(50,30) = -1;
Con(51,30) = 1; Con(51,18) = -1;
Con(52,18) = 1; Con(52,6) = -1;
Con(53,6) = 1; Con(53,37) = -1;
Con(54,37) = 1; Con(54,25) = -1;
Con(55,25) = 1; Con(55,13) = -1;
Con(56,13) = 1; Con(56,1) = -1;
Con(57,12) = 1; Con(57,43) = -1;
Con(58,43) = 1; Con(58,31) = -1;
Con(59,31) = 1; Con(59,19) = -1;
Con(60,19) = 1; Con(60,7) = -1;
Con(61,7) = 1; Con(61,48) = -1;
Con(62,48) = 1; Con(62,36) = -1;
Con(63,36) = 1; Con(63,24) = -1;
Con(64,24) = 1; Con(64,12) = -1;

Con(65,2) = 1; Con(65,41) = -1;
Con(66,41) = 1; Con(66,29) = -1;
Con(67,29) = 1; Con(67,17) = -1;
Con(68,17) = 1; Con(68,5) = -1;
Con(69,5) = 1, Con(69,38) = -1;
Con(70,38) = 1; Con(70,26) = -1;
Con(71,26) = 1; Con(71,14) = -1;
Con(72,14) = 1; Con(72,2) = -1;
Con(73,11) = 1; Con(73,44) = -1;
Con(74,44) = 1; Con(74,32) = -1;
Con(75,32) = 1; Con(75,20) = -1;
Con(76,20) = 1; Con(76,8) = -1;
Con(77,8) = 1; Con(77,47) = -1;
Con(78,47) = 1; Con(78,35) = -1;
Con(79,35) = 1; Con(79,23) = -1;
Con(80,23) = 1; Con(80,11) = -1;
Con(81,2) = 1; Con(81,11) = -1;
Con(82,41) = 1; Con(82,44) = -1;
Con(83,32) = 1; Con(83,29) = -1;
Con(84,17) = 1; Con(84,20) = -1;
Con(85,5) = 1; Con(85,8) = -1;
Con(86,47) = 1; Con(86,38) = -1;
Con(87,35) = 1; Con(87,26) = -1;
Con(88,23) = 1; Con(88,14) = -1;

Con(89,3) = 1; Con(89,40) = -1;
Con(90,40) = 1; Con(90,28) = -1;
Con(91,28) = 1; Con(91,16) = -1;
```

```
Con(92,16) = 1; Con(92,4) = -1;
Con(93,4) = 1; Con(93,39) = -1;
Con(94,39) = 1; Con(94,27) = -1;
Con(95,27) = 1; Con(95,15) = -1;
Con(96,15) = 1; Con(96,3) = -1;
Con(97,10) = 1; Con(97,45) = -1;
Con(98,45) = 1; Con(98,33) = -1;
Con(99,33) = 1; Con(99,21) = -1;
Con(100,21) = 1; Con(100,9) = -1;
Con(101,9) = 1; Con(101,46) = -1;
Con(102,46) = 1; Con(102,34) = -1;
Con(103,34) = 1; Con(103,22) = -1;
Con(104,22) = 1; Con(104,10) = -1;
Con(105,10) = 1; Con(105,3) = -1;
Con(106,40) = 1; Con(106,45) = -1;
Con(107,28) = 1; Con(107,33) = -1;
Con(108,16) = 1; Con(108,21) = -1;
Con(109,4) = 1; Con(109,9) = -1;
Con(110,39) = 1; Con(110,46) = -1;
Con(111,27) = 1; Con(111,34) = -1;
Con(112,15) = 1; Con(112,22) = -1;

% bars
Con(113,1) = 1; Con(113,43) = -1;
Con(114,42) = 1; Con(114,31) = -1;
Con(115,30) = 1; Con(115,19) = -1;
Con(116,18) = 1; Con(116,7) = -1;
Con(117,6) = 1; Con(117,48) = -1;
Con(118,37) = 1; Con(118,36) = -1;
Con(119,25) = 1; Con(119,24) = -1;
Con(120,13) = 1; Con(120,12) = -1;
Con(121,2) = 1; Con(121,44) = -1;
Con(122,41) = 1; Con(122,32) = -1;
Con(123,29) = 1; Con(123,20) = -1;
Con(124,8) = 1; Con(124,17) = -1;
Con(125,5) = 1; Con(125,47) = -1;
Con(126,35) = 1; Con(126,38) = -1;
Con(127,26) = 1; Con(127,23) = -1;
Con(128,11) = 1; Con(128,14) = -1;
Con(129,45) = 1; Con(129,3) = -1;
Con(130,40) = 1; Con(130,33) = -1;
Con(131,28) = 1; Con(131,21) = -1;
Con(132,16) = 1; Con(132,9) = -1;
Con(133,4) = 1; Con(133,46) = -1;
Con(134,39) = 1; Con(134,34) = -1;
Con(135,27) = 1; Con(135,22) = -1;
Con(136,15) = 1; Con(136,10) = -1;
%% Synthesis
%%% constraint equations for forces
lim = -1*ones(112,1);
lim1 = ones(24,1);
lim = vertcat(lim,lim1);
A = diag(lim);
b = zeros(136,1);
% Solving for force densities
% put MaxFunEvals = 150000 & defaultopt.TolX = 1e-5 in fmincon
```

```
options = optimset('Algorithm','interior-point');
x1 = -lim;
[q,fval,exitflag,output,lambda,grad,hessian] =
fmincon(@tensegrity_biconcave_8,x1,A,b,[],[],[],[],[],options);
q
fval
exitflag
% Normalising q
tot = 0;
for i = 1:length(x1)
        tot = tot+(q(i))^2;
end
tot = sqrt(tot);
for i = 1:length(x1)
    norm_q(i) = q(i)/tot;
end
norm_q = norm_q'
% Area and Young's modulus values of the elements
A_strut = pi()*(.007^2)/4
E_strut = 9e9;
A_cable = pi()*0.001^2/4
E_cable = 35e9;
K_S = A_strut*E_strut;
K_C = A_cable*E_cable;
K1 = K_C*ones(112,1);
K2 = K_S*ones(24,1);
K = vertcat(K1,K2);
I_strut = pi()*(0.007^4)/64;
% finding length of the members
F_strut = zeros(24,1);
F_buckling = zeros(24,1);
len = length(Con);
length11 = ones(len,1);
for i = 1:len
    for j = 1:48
        if(Con(i,j)==1)
            x_c1 = X(j);
            y_c1 = Y(j);
            z_c1 = Z(j);
        end
        if(Con(i,j)==-1)
            x_c2 = X(j);
            y_c2 = Y(j);
            z_c2 = Z(j);
        end
    end
    length11(i) = sqrt((x_c1-x_c2)^2+(y_c1-y_c2)^2+(z_c1-z_c2)^2);
end
% allowable force calculation
FoS = 4;
strength_cable = 1.77e9;
stress_cable_allow = strength_cable/FoS;
F_cable_allow = stress_cable_allow*A_cable;
factr = norm_q(1);
for i = 1:length(x1)
    q(i) = F_cable_allow*norm_q(i)/(factr*length11(i));
```

```
end
% calculating the actual force from force densities
F = q.*length11;
length(F);
length(K);
nums = 1:136;
nums = nums';
F_1 = horzcat(nums,F)
R_1 = F./K;
R = R_1+1;
length11
length01 = ones(len,1);
length01 = length11./R
F_act = zeros(12,1);
for i = 1:136
    if i<113
        F_act(i) = A_cable*E_cable*(length11(i)-length01(i))/length01(i);
    else
        F_act(i) = A_strut*E_strut*(length11(i)-length01(i))/length01(i);
    end
end
for i = 1:24
    F_strut(i) = F(i+112);
    F_buckling(i) = -1*(pi())^2*E_strut*I_strut/length01(i+112);
end
F_strut;
F_act;
F_buckling
%% plots

P1=[X(1,1);X(43,1)];Q1=[Y(1,1);Y(43,1)];R1=[Z(1,1);Z(43,1)];
P2=[X(42,1);X(31,1)];Q2=[Y(42,1);Y(31,1)];R2=[Z(42,1);Z(31,1)];
P3=[X(30,1);X(19,1)];Q3=[Y(30,1);Y(19,1)];R3=[Z(30,1);Z(19,1)];
P4=[X(7,1);X(18,1)];Q4=[Y(7,1);Y(18,1)];R4=[Z(7,1);Z(18,1)];
P5=[X(6,1);X(48,1)];Q5=[Y(6,1);Y(48,1)];R5=[Z(6,1);Z(48,1)];
P6=[X(36,1);X(37,1)];Q6=[Y(36,1);Y(37,1)];R6=[Z(36,1);Z(37,1)];
P7=[X(25,1);X(24,1)];Q7=[Y(25,1);Y(24,1)];R7=[Z(25,1);Z(24,1)];
P8=[X(13,1);X(12,1)];Q8=[Y(13,1);Y(12,1)];R8=[Z(13,1);Z(12,1)];

P9=[X(2,1);X(44,1)];Q9=[Y(2,1);Y(44,1)];R9=[Z(2,1);Z(44,1)];
P10=[X(32,1);X(41,1)];Q10=[Y(32,1);Y(41,1)];R10=[Z(32,1);Z(41,1)];
P11=[X(29,1);X(20,1)];Q11=[Y(29,1);Y(20,1)];R11=[Z(29,1);Z(20,1)];
P12=[X(8,1);X(17,1)];Q12=[Y(8,1);Y(17,1)];R12=[Z(8,1);Z(17,1)];
P13=[X(5,1);X(47,1)];Q13=[Y(5,1);Y(47,1)];R13=[Z(5,1);Z(47,1)];
P14=[X(35,1);X(38,1)];Q14=[Y(35,1);Y(38,1)];R14=[Z(35,1);Z(38,1)];
P15=[X(11,1);X(14,1)];Q15=[Y(11,1);Y(14,1)];R15=[Z(11,1);Z(14,1)];
P16=[X(23,1);X(26,1)];Q16=[Y(23,1);Y(26,1)];R16=[Z(23,1);Z(26,1)];

P17=[X(3,1);X(45,1)];Q17=[Y(3,1);Y(45,1)];R17=[Z(3,1);Z(45,1)];
P18=[X(10,1);X(15,1)];Q18=[Y(10,1);Y(15,1)];R18=[Z(10,1);Z(15,1)];
P19=[X(40,1);X(33,1)];Q19=[Y(40,1);Y(33,1)];R19=[Z(40,1);Z(33,1)];
P20=[X(21,1);X(28,1)];Q20=[Y(21,1);Y(28,1)];R20=[Z(21,1);Z(28,1)];
P21=[X(9,1);X(16,1)];Q21=[Y(9,1);Y(16,1)];R21=[Z(9,1);Z(16,1)];
P22=[X(4,1);X(46,1)];Q22=[Y(4,1);Y(46,1)];R22=[Z(4,1);Z(46,1)];
P23=[X(39,1);X(34,1)];Q23=[Y(39,1);Y(34,1)];R23=[Z(39,1);Z(34,1)];
P24=[X(27,1);X(22,1)];Q24=[Y(27,1);Y(22,1)];R24=[Z(27,1);Z(22,1)];
```

```
X1=[X(1,1);X(2,1);X(3,1);X(4,1);X(5,1);X(6,1);X(7,1);X(8,1);X(9,1);X(10,1);X(
11,1);X(12,1);X(1,1)];
Y1=[Y(1,1);Y(2,1);Y(3,1);Y(4,1);Y(5,1);Y(6,1);Y(7,1);Y(8,1);Y(9,1);Y(10,1);Y(
11,1);Y(12,1);Y(1,1)];
Z1=[Z(1,1);Z(2,1);Z(3,1);Z(4,1);Z(5,1);Z(6,1);Z(7,1);Z(8,1);Z(9,1);Z(10,1);Z(
11,1);Z(12,1);Z(1,1)];
X2=[X(13,1);X(14,1);X(15,1);X(16,1);X(17,1);X(18,1);X(19,1);X(20,1);X(21,1);X
(22,1);X(23,1);X(24,1);X(13,1)];
Y2=[Y(13,1);Y(14,1);Y(15,1);Y(16,1);Y(17,1);Y(18,1);Y(19,1);Y(20,1);Y(21,1);Y
(22,1);Y(23,1);Y(24,1);Y(13,1)];
Z2=[Z(13,1);Z(14,1);Z(15,1);Z(16,1);Z(17,1);Z(18,1);Z(19,1);Z(20,1);Z(21,1);Z
(22,1);Z(23,1);Z(24,1);Z(13,1)];
X3=[X(25,1);X(26,1);X(27,1);X(28,1);X(29,1);X(30,1);X(31,1);X(32,1);X(33,1);X
(34,1);X(35,1);X(36,1);X(25,1)];
Y3=[Y(25,1);Y(26,1);Y(27,1);Y(28,1);Y(29,1);Y(30,1);Y(31,1);Y(32,1);Y(33,1);Y
(34,1);Y(35,1);Y(36,1);Y(25,1)];
Z3=[Z(25,1);Z(26,1);Z(27,1);Z(28,1);Z(29,1);Z(30,1);Z(31,1);Z(32,1);Z(33,1);Z
(34,1);Z(35,1);Z(36,1);Z(25,1)];
X4=[X(37,1);X(38,1);X(39,1);X(40,1);X(41,1);X(42,1);X(43,1);X(44,1);X(45,1);X
(46,1);X(47,1);X(48,1);X(37,1)];
Y4=[Y(37,1);Y(38,1);Y(39,1);Y(40,1);Y(41,1);Y(42,1);Y(43,1);Y(44,1);Y(45,1);Y
(46,1);Y(47,1);Y(48,1);Y(37,1)];
Z4=[Z(37,1);Z(38,1);Z(39,1);Z(40,1);Z(41,1);Z(42,1);Z(43,1);Z(44,1);Z(45,1);Z
(46,1);Z(47,1);Z(48,1);Z(37,1)];


X5=[X(2,1);X(41,1);X(29,1);X(17,1);X(5,1);X(38,1);X(26,1);X(14,1);X(2,1)];
Y5=[Y(2,1);Y(41,1);Y(29,1);Y(17,1);Y(5,1);Y(38,1);Y(26,1);Y(14,1);Y(2,1)];
Z5=[Z(2,1);Z(41,1);Z(29,1);Z(17,1);Z(5,1);Z(38,1);Z(26,1);Z(14,1);Z(2,1)];
X6=[X(11,1);X(44,1);X(32,1);X(20,1);X(8,1);X(47,1);X(35,1);X(23,1);X(11,1)];
Y6=[Y(11,1);Y(44,1);Y(32,1);Y(20,1);Y(8,1);Y(47,1);Y(35,1);Y(23,1);Y(11,1)];
Z6=[Z(11,1);Z(44,1);Z(32,1);Z(20,1);Z(8,1);Z(47,1);Z(35,1);Z(23,1);Z(11,1)];

X7=[X(3,1);X(15,1);X(27,1);X(39,1);X(4,1);X(16,1);X(28,1);X(40,1);X(3,1)];
Y7=[Y(3,1);Y(15,1);Y(27,1);Y(39,1);Y(4,1);Y(16,1);Y(28,1);Y(40,1);Y(3,1)];
Z7=[Z(3,1);Z(15,1);Z(27,1);Z(39,1);Z(4,1);Z(16,1);Z(28,1);Z(40,1);Z(3,1)];
X8=[X(10,1);X(22,1);X(34,1);X(46,1);X(9,1);X(21,1);X(33,1);X(45,1);X(10,1)];
Y8=[Y(10,1);Y(22,1);Y(34,1);Y(46,1);Y(9,1);Y(21,1);Y(33,1);Y(45,1);Y(10,1)];
Z8=[Z(10,1);Z(22,1);Z(34,1);Z(46,1);Z(9,1);Z(21,1);Z(33,1);Z(45,1);Z(10,1)];

X9=[X(1,1);X(13,1);X(25,1);X(37,1);X(6,1);X(18,1);X(30,1);X(42,1);X(1,1)];
Y9=[Y(1,1);Y(13,1);Y(25,1);Y(37,1);Y(6,1);Y(18,1);Y(30,1);Y(42,1);Y(1,1)];
Z9=[Z(1,1);Z(13,1);Z(25,1);Z(37,1);Z(6,1);Z(18,1);Z(30,1);Z(42,1);Z(1,1)];
X10=[X(12,1);X(24,1);X(36,1);X(48,1);X(7,1);X(19,1);X(31,1);X(43,1);X(12,1)];
Y10=[Y(12,1);Y(24,1);Y(36,1);Y(48,1);Y(7,1);Y(19,1);Y(31,1);Y(43,1);Y(12,1)];
Z10=[Z(12,1);Z(24,1);Z(36,1);Z(48,1);Z(7,1);Z(19,1);Z(31,1);Z(43,1);Z(12,1)];

X11=[X(11,1);X(2,1)];Y11=[Y(11,1);Y(2,1)];Z11=[Z(11,1);Z(2,1)];
X12=[X(14,1);X(23,1)];Y12=[Y(14,1);Y(23,1)];Z12=[Z(14,1);Z(23,1)];
X13=[X(35,1);X(26,1)];Y13=[Y(35,1);Y(26,1)];Z13=[Z(35,1);Z(26,1)];
X14=[X(38,1);X(47,1)];Y14=[Y(38,1);Y(47,1)];Z14=[Z(38,1);Z(47,1)];
X15=[X(8,1);X(5,1)];Y15=[Y(8,1);Y(5,1)];Z15=[Z(8,1);Z(5,1)];
X16=[X(17,1);X(20,1)];Y16=[Y(17,1);Y(20,1)];Z16=[Z(17,1);Z(20,1)];
X17=[X(32,1);X(29,1)];Y17=[Y(32,1);Y(29,1)];Z17=[Z(32,1);Z(29,1)];
```

```
X18=[X(41,1);X(44,1)];Y18=[Y(41,1);Y(44,1)];Z18=[Z(41,1);Z(44,1)];
X19=[X(10,1);X(3,1)];Y19=[Y(10,1);Y(3,1)];Z19=[Z(10,1);Z(3,1)];
X20=[X(15,1);X(22,1)];Y20=[Y(15,1);Y(22,1)];Z20=[Z(15,1);Z(22,1)];
X21=[X(34,1);X(27,1)];Y21=[Y(34,1);Y(27,1)];Z21=[Z(34,1);Z(27,1)];
X22=[X(39,1);X(46,1)];Y22=[Y(39,1);Y(46,1)];Z22=[Z(39,1);Z(46,1)];
X23=[X(9,1);X(4,1)];Y23=[Y(9,1);Y(4,1)];Z23=[Z(9,1);Z(4,1)];
X24=[X(16,1);X(21,1)];Y24=[Y(16,1);Y(21,1)];Z24=[Z(16,1);Z(21,1)];
X25=[X(33,1);X(28,1)];Y25=[Y(33,1);Y(28,1)];Z25=[Z(33,1);Z(28,1)];
X26=[X(40,1);X(45,1)];Y26=[Y(40,1);Y(45,1)];Z26=[Z(40,1);Z(45,1)];

plot3(P1,Q1,R1,P2,Q2,R2,P3,Q3,R3,P4,Q4,R4,P5,Q5,R5,P6,Q6,R6,P7,Q7,R7,P8,Q8,R8
,'Color','b','LineWidth',4);
hold on;
plot3(P9,Q9,R9,P10,Q10,R10,P11,Q11,R11,P12,Q12,R12,P13,Q13,R13,P14,Q14,R14,P1
5,Q15,R15,P16,Q16,R16,'Color','b','LineWidth',4);
hold on;
plot3(P17,Q17,R17,P18,Q18,R18,P19,Q19,R19,P20,Q20,R20,P21,Q21,R21,P22,Q22,R22
,P23,Q23,R23,P24,Q24,R24,'Color','b','LineWidth',4);
hold on;
plot3(X1,Y1,Z1,X2,Y2,Z2,X3,Y3,Z3,X4,Y4,Z4,X5,Y5,Z5,X6,Y6,Z6,X7,Y7,Z7,X8,Y8,Z8
,'Color','r','LineWidth',4);
hold on;
plot3(X9,Y9,Z9,X10,Y10,Z10,'Color','r','LineWidth',4);
hold on;
plot3(X11,Y11,Z11,X12,Y12,Z12,X13,Y13,Z13,X14,Y14,Z14,X15,Y15,Z15,X16,Y16,Z16
,X17,Y17,Z17,X18,Y18,Z18,'Color','r','LineWidth',4);
hold on;
plot3(X19,Y19,Z19,X20,Y20,Z20,X21,Y21,Z21,X22,Y22,Z22,X23,Y23,Z23,X24,Y24,Z24
,X25,Y25,Z25,X26,Y26,Z26,'Color','r','LineWidth',4);
hold on;
xlabel('x-axis','fontsize',12,'fontweight','b')
ylabel('y-axis','fontsize',12,'fontweight','b')
zlabel('z-axis','fontsize',12,'fontweight','b')
axis equal



%% Function to solve the static equilibrium equations at the nodes
function f = tensegrity_biconcave_8(x)
global Con X Y Z
%%% Constructing the static equilibrium matix for given connectivity and
%%% coordinates
A1 = Con'*diag(Con*X);
A2 = Con'*diag(Con*Y);
A3 = Con'*diag(Con*Z);
C = vertcat(A1,A2,A3);
%%% external force vector
d = zeros(144,1);
f1 = C*x-d;
f=f1'*f1;
```

Appendix
A. 4

```
%% program to synthesize a tensegrity table of desired height.
clc;
clf;
clear all;

%%%% Initial guess in height
x2 = [ 0.5 0.5 0.5]';

%%% Initialising tolerance value
f1 = 1;
f2 = 1;
f3 = 1;

%%% specifying required height
reqd_ht = .6;

%% looping till getting required height
while (f1^2>=10e-7)  %&&(f2^2>=10e-8)&&(f3^2>=10e-8)


    global Con X Y Z

    %%% Connectivity matrix for the table
    Con = [1 -1 0  0  0  0;
        0 1 -1  0  0  0;
        1  0 -1  0  0  0;
        0  0  0  1 -1  0;
        0  0  0  0  1 -1;
        0  0  0  1  0 -1;
        1  0  0  0  -1 0;
        0  1  0  0  0 -1;
        0  0  1 -1  0  0;
        1  0  0  -1 0  0;
        0  1  0  0 -1  0;
        0  0  1  0  0  -1];


    %%%  Coordinates of the vertices of the table

        X = [  0  .5196   .2598    (0.1445+.2598) (-0.3+.2598)
(0.1554+.2598)]';
        Y = [  0   0     .45       (0.2629+.15) (-0.0063+.15)  (-
0.2566+.15)]';
        Z = [  0   0      0       x2(1)    x2(2)    x2(3)]';

    %%% force inequality constraint matrices

    A = [ -1 0 0 0 0 0 0 0 0 0 0 0;
```

```
        0 -1 0 0 0 0 0 0 0 0 0;
        0 0 -1 0 0 0 0 0 0 0 0;
        0 0 0 -1 0 0 0 0 0 0 0;
        0 0 0 0 -1 0 0 0 0 0 0;
        0 0 0 0 0 -1 0 0 0 0 0;
        0 0 0 0 0 0 -1 0 0 0 0;
        0 0 0 0 0 0 0 -1 0 0 0;
        0 0 0 0 0 0 0 0 -1 0 0;
        0 0 0 0 0 0 0 0 0 1  0;
        0 0 0 0 0 0 0 0 0  0 1;
        0 0 0 0 0 0 0 0 0  0 0];

    b = [ 0 0 0 0 0 0 0 0 0 0 0 ]';

    % Solving for force densities
    options = optimset('Algorithm','interior-point');
    x1 = [ 1 1 1 1 1 1 1 1 1 -1 -1 -1]';
    [q,fval,exitflag,output,lambda,grad,hessian] =
fmincon(@tensegrity_table_points1,x1,A,b,[],[],[],[],[],options);

    % Normalising force densities
    tot = 0;
    for i = 1:length(x1)
        tot = tot+(q(i))^2;
    end
    tot = sqrt(tot);
    norm_q = zeros(length(x1),1);
    for i = 1:length(x1)
        norm_q(i) = q(i)/tot;
    end
    % norm_q = q/norm(q)
    %%% Area and Young's modulus of the elements
    A_strut = pi()*(0.032^2-0.0288^2)/4;
    E_strut = 69e9;
    A_cable = pi()*0.002^2/4;
    E_cable = 125e9;
    K_S = A_strut*E_strut;
    K_C = A_cable*E_cable;
    K = [ K_C K_C K_C K_C K_C K_C K_C K_C K_C K_S K_S K_S]';
    I_strut = pi()*(0.032^4-0.0288^4)/64;

    % finding length of the members
    F_strut = zeros(3,1);
    F_buckling = zeros(3,1);

    len = length(Con);
    length11 = ones(len,1);
    for i = 1:len
        for j = 1:6
            if(Con(i,j)==1)
                x_c1 = X(j);
                y_c1 = Y(j);
                z_c1 = Z(j);
            end
            if(Con(i,j)==-1)
```

```
            x_c2 = X(j);
            y_c2 = Y(j);
            z_c2 = Z(j);
        end
    end
    length11(i) = sqrt((x_c1-x_c2)^2+(y_c1-y_c2)^2+(z_c1-z_c2)^2);

end

% allowable force calculation
FoS = 4;
strength_cable = 1.77e9;
stress_cable_allow = strength_cable/FoS;
F_cable_allow = stress_cable_allow*A_cable;

factr = norm_q(1);

for i = 1:length(x1)
    q(i) = F_cable_allow*norm_q(i)/(factr*length11(i));
end
% calculating the actual force from force densities
q;
F_1 = q.*length11;

R_1 = F_1./K;
R = R_1+1;

length01 = ones(len,1);
length01 = length11./R
F_act = zeros(12,1);
for i = 1:12
    if i<10
        F_act(i) = A_cable*E_cable*(length11(i)-length01(i))/length01(i);
    else
        F_act(i) = A_strut*E_strut*(length11(i)-length01(i))/length01(i);
    end
end


for i = 1:3
    F_strut(i) = F_1(i+9);
    F_buckling(i) = -1*(pi())^2*E_strut*I_strut/(length01(i+9))^2;
end

F_strut;
F_act;
F_buckling;

%% static analysis

global L1 L2 L3 L01 L02 L03 L04 L05 L06 L07 L08 L09 k1 k2 k3 k4 k5 k6 k7
k8 k9
global P21 P22 P23 F
```

```
    P11=[0 0 0];
    P12=[X(2) 0 0];
    P13=[X(3) Y(3) 0];
    P21=[X(4) Y(4) Z(4)];
    P22=[X(5) Y(5) Z(5)];
    P23=[X(6) Y(6) Z(6)];
    Pz1=P21(1,3);
    Pz2=P21(1,3);
    Pz3=P21(1,3);

    %%% allocating bar lengths, cable free lengths and cable stiffness
    L1 = length01(10)
    L2 = length01(11)
    L3 = length01(12)

    L01 = length01(1);
    L02 = length01(2);
    L03 = length01(3);
    L04 = length01(7);
    L05 = length01(8);
    L06 = length01(9);
    L07 = length01(4);
    L08 = length01(5);
    L09 = length01(6);

    k1=A_cable*E_cable/L01;
    k2=A_cable*E_cable/L02;
    k3=A_cable*E_cable/L03;
    k4=A_cable*E_cable/L07;
    k5=A_cable*E_cable/L08;
    k6=A_cable*E_cable/L09;
    k7=A_cable*E_cable/L04;
    k8=A_cable*E_cable/L05;
    k9=A_cable*E_cable/L06;


    %%% Initial guess and Force vector
    options = optimset('Algorithm','interior-point');
    x1 = [1,1,.5,.75,.5,.6,0.7,.2,.4];
    F=[0 0 -200 0 0 -200 0 0 -200];

    [x3,fval,exitflag,output,grad,hessian] =
fminunc(@tensegrity3D_static_sigmoid2_1,x1,options);

    x3
    %
    f1 = reqd_ht-L1*sin(x3(4))
    f2 = reqd_ht-L2*sin(x3(5))
    f3 = reqd_ht-L3*sin(x3(6))

    x2(1) = x2(1)+f1;
    x2(2) = x2(2)+f2;
    x2(3) = x2(3)+f3;
```

```
end
L1*sin(x3(4))
L2*sin(x3(5))
L3*sin(x3(6))

%% plots of deformed and undeformed configuraions

X1=[X(1,1);X(2,1);X(3,1);X(1,1);X(6,1);X(5,1);X(4,1);X(6,1);X(5,1);X(3,1);X(2
,1);X(4,1)];
Y1=[Y(1,1);Y(2,1);Y(3,1);Y(1,1);Y(6,1);Y(5,1);Y(4,1);Y(6,1);Y(5,1);Y(3,1);Y(2
,1);Y(4,1)];
Z1=[Z(1,1);Z(2,1);Z(3,1);Z(1,1);Z(6,1);Z(5,1);Z(4,1);Z(6,1);Z(5,1);Z(3,1);Z(2
,1);Z(4,1)];
P1=[X(1,1);X(4,1)];
Q1=[Y(1,1);Y(4,1)];
R1=[Z(1,1);Z(4,1)];
P2=[X(2,1);X(5,1)];
Q2=[Y(2,1);Y(5,1)];
R2=[Z(2,1);Z(5,1)];
P3=[X(3,1);X(6,1)];
Q3=[Y(3,1);Y(6,1)];
R3=[Z(3,1);Z(6,1)];
plot3(X1,Y1,Z1,'Color','r','LineWidth',4);
hold on;
plot3(P1,Q1,R1,P2,Q2,R2,P3,Q3,R3,'Color','b','LineWidth',4);
hold on;

X = [  0  x3(1)   x3(2)*cos(x3(3))    L1*cos(x3(4))*cos(x3(7))
x3(1)+L2*cos(x3(5))*cos(x3(8))
x3(2)*cos(x3(3))+L3*cos(x3(9))*cos(x3(6))]';
Y = [  0   0      x3(2)*sin(x3(3))    L1*cos(x3(4))*sin(x3(7))
L2*cos(x3(5))*sin(x3(8))
x3(2)*sin(x3(3))+L3*sin(x3(9))*cos(x3(6))]';
Z = [  0   0      0                   L1*sin(x3(4))
L2*sin(x3(5))                    L3*sin(x3(6))]';

X1=[X(1,1);X(2,1);X(3,1);X(1,1);X(6,1);X(5,1);X(4,1);X(6,1);X(5,1);X(3,1);X(2
,1);X(4,1)];
Y1=[Y(1,1);Y(2,1);Y(3,1);Y(1,1);Y(6,1);Y(5,1);Y(4,1);Y(6,1);Y(5,1);Y(3,1);Y(2
,1);Y(4,1)];
Z1=[Z(1,1);Z(2,1);Z(3,1);Z(1,1);Z(6,1);Z(5,1);Z(4,1);Z(6,1);Z(5,1);Z(3,1);Z(2
,1);Z(4,1)];
P1=[X(1,1);X(4,1)];
Q1=[Y(1,1);Y(4,1)];
R1=[Z(1,1);Z(4,1)];
P2=[X(2,1);X(5,1)];
Q2=[Y(2,1);Y(5,1)];
R2=[Z(2,1);Z(5,1)];
P3=[X(3,1);X(6,1)];
Q3=[Y(3,1);Y(6,1)];
R3=[Z(3,1);Z(6,1)];
plot3(X1,Y1,Z1,'--','Color','m','LineWidth',4);
hold on;
plot3(P1,Q1,R1,'--',P2,Q2,R2,'--',P3,Q3,R3,'--','Color','g','LineWidth',4);
hold on;
```

```
xlabel('x-axis','fontsize',12,'fontweight','b');
ylabel('y-axis','fontsize',12,'fontweight','b');
zlabel('z-axis','fontsize',12,'fontweight','b');

norm_q
length01


%% Function to solve the static equlilibrium equations at the nodes
function f = tensegrity_table_points1(x)
global Con X Y Z
%%% Constructing the equilibrium matrix from connectivity matrix and
%%% coordinates
A1 = Con'*diag(Con*X);
A2 = Con'*diag(Con*Y);
A3 = Con'*diag(Con*Z);
C = vertcat(A1,A2,A3);
%%% External force vector size 3n x 1
d = [ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]';
f1 = C*x-d;
f = f1'*f1;


%% Function to get the tensegrity configuration having least potential energy
function f = tensegrity3D_static_sigmoid2_1(x2)
global L1 L2 L3 L01 L02 L03 L04 L05 L06 L07 L08 L09 k1 k2 k3 k4 k5 k6 k7 k8
k9
global P21 P22 P23 F

%%% final length of the deformed cables
len1=x2(1);
len2=x2(2);
len3=sqrt((x2(1)-x2(2)*cos(x2(3)))^2+(x2(2)*sin(x2(3)))^2);
len4=sqrt((x2(2)*cos(x2(3))+L3*cos(x2(9))*cos(x2(6))-
x2(1))^2+(x2(2)*sin(x2(3))+L3*sin(x2(9))*cos(x2(6)))^2+(L3*sin(x2(6)))^2);
len5=sqrt((x2(1)+L2*cos(x2(5))*cos(x2(8)))^2+(L2*cos(x2(5))*sin(x2(8)))^2+(L2
*sin(x2(5)))^2);
len6=sqrt((L1*cos(x2(4))*cos(x2(7))-
x2(2)*cos(x2(3)))^2+(L1*cos(x2(4))*sin(x2(7))-
x2(2)*sin(x2(3)))^2+(L1*sin(x2(4)))^2);
len7=sqrt((x2(2)*cos(x2(3))+L3*cos(x2(9))*cos(x2(6))-
(x2(1)+L2*cos(x2(5))*cos(x2(8))))^2+(x2(2)*sin(x2(3))+L3*sin(x2(9))*cos(x2(6)
)-L2*cos(x2(5))*sin(x2(8)))^2+(L3*sin(x2(6))-L2*sin(x2(5)))^2);
len8=sqrt((x2(1)+L2*cos(x2(5))*cos(x2(8))-
L1*cos(x2(4))*cos(x2(7)))^2+(L2*cos(x2(5))*sin(x2(8))-
L1*cos(x2(4))*sin(x2(7)))^2+(L2*sin(x2(5))-L1*sin(x2(4)))^2);
len9=sqrt((x2(2)*cos(x2(3))+L3*cos(x2(9))*cos(x2(6))-
L1*cos(x2(4))*cos(x2(7)))^2+(x2(2)*sin(x2(3))+L3*sin(x2(9))*cos(x2(6))-
L1*cos(x2(4))*sin(x2(7)))^2+(L3*sin(x2(6))-L1*sin(x2(4)))^2);

%%% Work potential
ms=100;
a1=F(1,1)*(P21(1,1)-L1*cos(x2(4))*cos(x2(7)));
a2=F(1,2)*(P21(1,2)-L1*cos(x2(4))*sin(x2(7)));
a3=F(1,3)*(P21(1,3)-L1*sin(x2(4)));
```

```
a4=F(1,4)*(P22(1,1)-(x2(1)+L2*cos(x2(5))*cos(x2(8))));
a5=F(1,5)*(P22(1,2)-(L2*cos(x2(5))*sin(x2(8))));
a6=F(1,6)*(P22(1,3)-L2*sin(x2(5)));
a7=F(1,7)*(P23(1,1)-(x2(2)*cos(x2(3))+L3*cos(x2(9))*cos(x2(6))));
a8=F(1,8)*(P23(1,2)-(x2(2)*sin(x2(3))+L3*sin(x2(9))*cos(x2(6))));
a9=F(1,9)*(P23(1,3)-L3*sin(x2(6)));
a=(a1+a2+a3+a4+a5+a6+a7+a8+a9);

%%% strain energy in the cables after deformation including heavy side
%%% function
d=0.5*k4*(len4-L04)^2/(1+exp(-ms*((len4/L04)-1)));
e=0.5*k5*(len5-L05)^2/(1+exp(-ms*((len5/L05)-1)));
g=0.5*k6*(len6-L06)^2/(1+exp(-ms*((len6/L06)-1)));
h=0.5*k7*(len7-L07)^2/(1+exp(-ms*((len7/L07)-1)));
i=0.5*k8*(len8-L08)^2/(1+exp(-ms*((len8/L08)-1)));
j=0.5*k9*(len9-L09)^2/(1+exp(-ms*((len9/L09)-1)));
k=0.5*k1*(len1-L01)^2/(1+exp(-ms*((len1/L01)-1)));
l=0.5*k2*(len2-L02)^2/(1+exp(-ms*((len2/L02)-1)));
m=0.5*k3*(len3-L03)^2/(1+exp(-ms*((len3/L03)-1)));

f=a+d+e+g+h+i+j+k+l+m;

% potential enegy equation
```

References

1. Karcher, H., Lammerding, J., Huang, H., Lee, R. T., Kamm, R. D., and Mofrad, M. R. K., "A Three-Dimensional Viscoelastic Model for Cell Deformation with Experimental Verification," *Biophysical Journal,* Vol. 85, November 2003, pp. 3336-3349.
2. Pollak, G. H., "The role of aqueous interfaces in the cell," *Advances in Colloid and Interface Science*, Vol. 103, 2003, pp. 173-196.
3. Bursac, P., Lenormand, G., Fabry, B., Oliver, M., Weitz, D. A., Viasnoff, V., Butler, J. P., and Fredberg, J. J., "Cytoskeletal remodeling and slow dynamics in the living cell," *nature materials*, Vol. 4, July 2005, pp. 557-561.
4. Mofrad, M. R. K. and Kamm, R. D., "Cytoskeleton Mechanics: models and measurements," Cambridge University Press, Cambridge 2006, pp. 84-102
5. Motro, R., "Tensegrity: Structural Systems for the future," Elsevier, Newyork 2003.
6. Skelton, R.E., and de Oliveira, M. C., *Tensegrity systems*: Springer, Berlin 2009.
7. Ingber, D. E., "Cellular tensegrity: defining new rules of biological design that govern the cytoskeleton," *Journal of Cell Science*, Vol. 104, 1993, pp 613-627.
8. Skelton, R.E., Adhikari, R., Pinaud, J.P., Chan, W., Helton, W.J., "An introduction to the mechanics of tensegrity structures," *Decision and Control, 2001. Proceedings of the 40th IEEE Conference*, Orlando, Florida December 2001, Vol. 5. IEEE, 2001.
9. Jáuregui, V. G., *Tensegrity structures and their application to architecture*: Diss. Queen's University Belfast, 2004.
10. Aldrich, J. B., Skelton, R. E., and Kreutz-Delgado, K., "Control synthesis for a class of light and agile robotic tensegrity structures," *Proceedings of the American Control Conference,* Denver, Colorado June 4-6, 2003 Vol. 6. IEEE, pp 5245 – 5251, 2003.
11. Tibert, A. G., and Pellegrino, S., "Review of form-finding methods for tensegrity structures," *International Journal of Space Structures,* Vol. 18(4), 2003, pp 209-223.
12. Pellegrino, S., *Mechanics of kinematically indeterminate structures*: Diss. University of Cambridge, 1986.
13. Motro, R. and Nooshin, H., "Forms and forces in tensegrity systems," *Proceedings of the Third International Conference on Space Structures*, Elsevier, Amsterdam, 1984, pp 180-185.
14. Gan, B.S., "Designing an Irregular Tensegrity as a Monumental Object," *World Academy of Science, Engineering and Technology*, Vol. 70, 2012, pp 634-640.
15. Connelly, R., and Terrell, M., "Globally rigid symmetric tensegrities," *Structural Topology 1995*, Vol. 21 , 1995.
16. Bayat, J.B., *Position Analysis of Planar Tensegrity Structures*: Diss. University of Florida. Gainesville Center for Intelligent Machines and Robotics, 2006.
17. Sultan, C., Corless, M., and Skelton, R.E., "The prestressability problem of tensegrity structures: some analytical solutions," *International Journal of Solids and Structures*,Vol.38.30, 2001, pp 5223-5252.
18. Masic, M., Skelton, R.E., and Gill., P.E., "Algebraic tensegrity form-finding," *International Journal of Solids and Structures*, Vol. 42.16, 2005, pp 4833-4858.

19. Ehara, S., and Kanno, Y., "Topology design of tensegrity structures via mixed integer programming." *International Journal of Solids and Structures*, Vol. 47.5, 2010, pp 571-579.
20. Sung, L. A. and Vera, C., "Protofilament and hexagon: a three-dimensional mechanical model for the junctional complex in the erythrocyte membrane skeleton," *Annals of biomedical engineering*, Vol. 31.11, 2003, pp 1314-1326.
21. Vera, C., Skelton, R. E., Bossens, F., and Sung, L. A., "3-D nanomechanics of an erythrocyte junctional complex in equibiaxial and anisotropic deformations," *Annals of biomedical engineering*, Vol. 33.10, 2005, pp 1387-1404.
22. Alloisio, N., Venezia, N. D., Rana, A., Andrabi, K., Texier, P., Gilsanz, F., Cartron, J. P., Delaunay, J., and Chishti, A. H., "Evidence that red blood cell protein p55 may participate in the skeleton-membrane linkage that involves protein 4.1 and glycophorin C," *Blood*, Vol 82.4, 1993, pp 1323-1327.
23. Radmacher, M., Fritz, M., Kacher, C. M., Cleveland, J. P., and Hansma, P. K., "Measuring the viscoelastic properties of human platelets with the atomic force microscope," *Biophysical Journal*, Vol. 70.1, 1996, pp 556-567.
24. Henon, S., Lenormand, G., Richert, A., and Gallet, F., "A new determination of the shear modulus of the human erythrocyte membrane using optical tweezers," *Biophysical Journal*, Vol.76. 2, 1999, pp 1145-1151.
25. Richelme, F., Benoliel, A. M., and Bongrand, P., "Dynamic study of cell mechanical and structural responses to rapid changes of calcium level," *Cell motility and the cytoskeleton*, Vol. 45.2, 2000, pp 93-105.
26. Calladine, C. R., "Buckminster Fuller's “tensegrity” structures and Clerk Maxwell's rules for the construction of stiff frames," *International Journal of Solids and Structures*, Vol.14.2, 1978, pp 161-172.
27. Guide, MATLAB User’s Manual, "The mathworks," *Inc., Natick, MA* 5 1998.
28. Burkhardt, R., "A practical guide to tensegrity design, 2nd edit." Cambridge, 2008.
29. Angelov, B. and Miladenov, I. M., "On the geometry of red blood cell, " *Geometry, Integrability and Quantization. (I. Mladenov and G.Naber, Eds.), Coral Press*. Sofia, pp 27-46, 2000.
30. Evans, E., and Yeung, A., "Apparent viscosity and cortical tension of blood granulocytes determined by micropipet aspiration," *Biophysical Journal*, Vol. 56.1, 1989, pp 151-160.
31. Mofrad, M. R. K., and Kamm, R. D., *Cytoskeletal Mechanics: Models and Measurements in Cell Mechanics*, Cambridge University Press, 2006.
32. Evans, E. A., "Minimum energy analysis of membrane deformation applied to pipet aspiration and surface adhesion of red blood cells," *Biophysical journal*, Vol. 30.2, 1980, pp 265-284.