ORIGINAL ARTICLE

lajss.org© Latin American Journal of Solids and Structures

www.lajss.org

An efficient Newton-Raphson based form-finding method for tensegrity structures with given strut forces and cable force density

Angelo Vumiliya[a], Ani Luo[a*], Andrés González-Fallas[b], Heping Liu[a]

[a]College of Mechanical and Electrical Engineering, Harbin Engineering University, Harbin, China. Email: vumiliiya@hrbeu.edu.cn, luoani@hrbeu.edu.cn, liuheping@hrbeu.edu.cn

[b]School of Mechanical Engineering, Universidad de Costa Rica, San José, Costa Rica. Email: ricardo.gonzalezfallas@ucr.ac.cr

* Corresponding author

https://doi.org/10.1590/1679-78258162

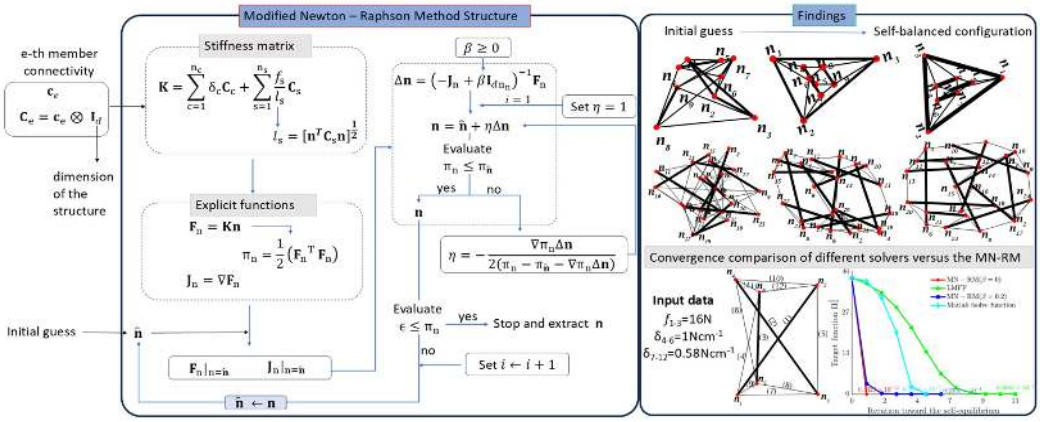
Abstract

Tensegrity structures are geometric nonlinear systems and statically and kinematically indeterminate structures that require an initial shape-finding procedure to establish a self-equilibrium state. This paper presents a shape-finding algorithm requiring structure topology, strut force, cable force density, and a random initial estimate of node coordinates as input. The equilibrium of the structure is achieved by zeroing the nonlinear static equilibrium in which the generalized nodal coordinates are chosen as variables. The modified Newton-Raphson method is used to solve the nonlinear equilibrium system by decreasing the nonlinear least square function to ensure global convergence. The stability of the self-balancing structure was evaluated using the properties of the geometric and tangent stiffness matrix. Various numerical examples are presented to illustrate the method's effectiveness for 2-*d* and 3-*d* tensegrity structures with multiple states of self-stress.

Keywords

Tensegrity, form-finding, shape optimization, nonlinear system, least square problem

Graphical Abstract



Received April 18, 2024. In revised form August 08, 2024. Accepted October 07, 2024. Available online October 08, 2024
https://doi.org/10.1590/1679-78258162

Latin American Journal of Solids and Structures. ISSN 1679-7825. Copyright © 2024. This is an Open Access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

1 INTRODUCTION

Tensegrity is a self-stable system consisting of a set of compressed components inside tensile components (Motro, 2003). The tensegrity's structural elements are designed to carry axial force, making the tensegrity an ideal structure for robotics (Paul et al., 2006), spatial (Tibert and Pellegrino, 2002), and civil engineering (Latteur et al., 2017; Feron et al., 2019) applications. Therefore, designing a structure that meets the requirements of various applications has become increasingly relevant.

Recently, form-finding methods have become a crucial aspect in the design of tensegrity structures. These methods differ regarding algorithms, variables, and initial input parameters. Some examples are methods based on iterative algorithms, such as the force density method originally used for cable-net structures (Schek, 1974). The method linearizes the nonlinear equilibrium equations of the system in terms of force per unit length and node position (Tran and Lee, 2010; Koohestani, 2017). The structure is obtained by simultaneous manipulation of the force density and equilibrium matrices such that the null spaces are the optimum geometrical position of nodes and force densities (Zhang and Ohsaki, 2006). Later, optimization methods based on minimizing the trace of the force density matrix were introduced in (Cai and Feng, 2015; Heng et al., 2021; Wang et al., 2021), with constraints such as the rank deficiency of force density, the group of members, and furthermore, the constraint on the properties of the structural components. The common feature of these algorithms is finding the acceptable set of force densities compatible with the geometric position of the nodes of the structure. Therefore, these methods are classified as static iterative algorithms in the review of the form-finding method by Tibert and Pellegrino (2011).

In addition to static methods, using nonlinear programming methods has played an essential role in geometrically determining the length of structural elements. These methods, as described in (Tibert and Pellegrino, 2011), aim to minimize or maximize a set of element lengths while keeping the lengths of other elements constrained. For example, in (Pellegrino, 1986), the length of struts was maximized while the length of cables was kept at a particular value. In addition, the stationary-based method presented in (Miki, 2011; Zhang et al., 2021) allowed the algorithm to minimize the length of the cables while constraining the length of the struts, resulting in various complex tensegrity. By taking the weight coefficient of members' length as the force density, the principles behind these methods are similar to the principle of virtual work as explained in (Miki, 2011).

Moreover, the method using the element properties (Zhang et al., 2014; Yuan et al., 2017; Xue et al., 2020; Ma et al., 2022a) has emerged, whereby solving the system's equilibrium minimizes the system's potential energy to achieve a stable structure. Inspired by the Lagrange method in which the node coordinates were variable, Ma et al. (2022a) studied the equilibrium and the stiffness of clustered tensegrity structures. Furthermore, a unified analysis method of tensegrity with interconnected rigid bodies is studied (Wang et al., 2023). The equilibrium and compatibility equations are derived through an energy approach combined with the Lagrange multiplier method achieving the computation of the self-stress state and mechanism mode. The drawbacks of algorithms that use the material properties are the rigid body motion that leads to the singularity of the structural stiffness matrix (Zhang et al., 2014), and the starting points of the developed method are close to the final solution of the structure (Gasparini et al., 2012; Xue et al., 2020).

However, there are cases when more control over the shape or prestress distribution of the configuration is necessary (Baudriller et al., 2006). In references (Xue et al., 2020; Khafizov and Savin, 2021), the method introduces the preferred strut direction into the problem statement to find the tensegrity with preferred orientation and shape. The structure is modeled as a collection of points representing the nodes (connecting struts and cables), where the equilibrium of each node is required for the overall system's stability. The design using the specified strut stiffness and cable forces in advance was developed in (Xue et al., 2020). The method requires the final length of the struts and searches for the minimum potential energy within the unknown final length of the cables. A modified dynamic relaxation-based form-finding is used to obtain a structure that presents an inextensible tensile member (He et al., 2024). The relationship between the number of struts and strings to satisfy the full-rank convexity criterion for a form-finding process is given in (Harish et al., 2023). Even though the above method yields self-equilibrium and stable tensegrity and controls the structural components of the tensegrity, none of the previous methods can directly control the struts' force distribution and optimize the elements' length when the cable force density is given. The present work aims to develop a numerical method to find the shape of tensegrities within a given topology, strut forces, force density in cables, and an initial random estimate of nodal position. To this end, we utilize the nonlinear system of equilibrium equations based on the given parameters to find a self-balanced configuration (the nodal positions, which means the node coordinates are taken as variables). The algorithm developed is based on the modified Newton-Raphson method in which the line search framework is used to decrease the least square function obtained through the nonlinear equilibrium to ensure a rapid convergence of the algorithm from a random initial guess.

The remainder of the paper is organized as follows: Section 2 outlines the notation used in this paper for the properties of tensegrity elements. Section 3 explains the nonlinear static equilibrium of the tensegrity structure based

on a set of forces in the struts and the force density in the cables. Section 4 introduces the shape-finding algorithm based on the modified Newton-Raphson method (MN-RM). Section 5 validates the accuracy and efficiency of the iterative algorithm implemented on different tensegrity structures. Section 6 thoroughly discusses the algorithm proposed in this paper, and Section 7 concludes the discussion.

2 ELEMENT PROPERTIES OF TENSEGRITY STRUCTURE

The tensegrity system is a structure consisting of a network of n_n nodes connected by a network of n_c tensile components and n_s struts. The elements are represented by vectors whose values come from the node vectors connected by these elements. The vector $\mathbf{n}_j=[x_j\ y_j\ z_j]^T$ denotes the coordinates of the *j*-th vertex in the cartesian frames x- y- and z-, so that the *j*-th vertex connected to an *e*-th component is denoted by $\mathbf{n}_{ej}$. For instance, the expression of a vector $\mathbf{b}_e$ of the *e*-th structural component connecting the (starting) node $\mathbf{n}_{ev}$ and the (end) node $\mathbf{n}_{eu}$ ($u\neq v$) in Figure 1 is as follows:

$$\mathbf{b}_e = \mathbf{n}_{eu} - \mathbf{n}_{ev} \in \mathbb{R}^{d\times 1} : d \text{ is structural dimension} \tag{1}$$

from which it follows the right side of equation (1) can be obtained using the following expression:

$$\mathbf{n}_{eu} - \mathbf{n}_{ev} = \mathbf{N}\mathbf{c}_e^{\,T} \tag{2}$$

where **N** and $\mathbf{c}_e$ are the node matrix and connectivity vector of the e-th component, respectively.

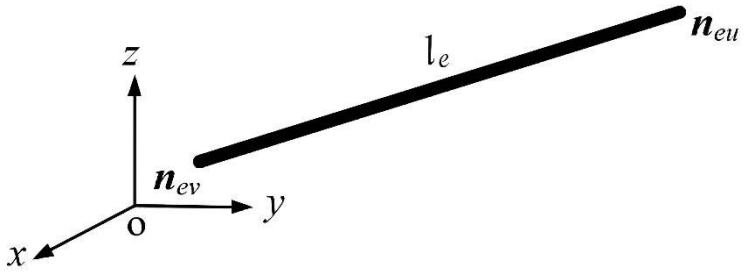


Figure 1: Geometrical representation of the e-th component in 3-*d* cartesian frame.

The node matrix consists of a sequence of the node vectors as

$$\mathbf{N} = [\mathbf{n}_1 \cdots \mathbf{n}_j \cdots \mathbf{n}_{n_n}] \ \in \mathbb{R}^{d\times n_n} \tag{3}$$

The vector $\mathbf{c}_e$ represents the connection between nodes $\mathbf{n}_{eu}$ and $\mathbf{n}_{ev}$, which are the *u*-th and *v*-th columns of the node matrix. To this, the vector $\mathbf{c}_e$ is expressed by a column vector whose *u*-th and *v*-th entries are 1 and -1, respectively, and zero otherwise. The connectivity vector of the *e*-th component is thus given as

$$\mathbf{c}_e = \begin{cases} -1 & v\text{-th column} \\ 1 & u\text{-th column} \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

The connectivity matrix is obtained by stacking all connectivity vectors into a matrix **C** as

$$\mathbf{C} = \left[\mathbf{c}_1^{\,T} \cdots \mathbf{c}_e^{\,T} \cdots \mathbf{c}_{n_m}^{\,T}\right]^T \in \mathbb{R}^{n_m \times n_n},\ n_m = n_c + n_s \tag{5}$$

2.1 Geometrical property of element

The node matrix defined in equation (3) is a set of *d* rows and the sequence of n_n columns of the coordinate vectors $\mathbf{n}_j$ (j=1,2, …,n_n); the node matrix can be vectorized as

$$\mathbf{n} = \begin{bmatrix} \cdots & \mathbf{n}_j & \cdots \end{bmatrix}^T \in \mathbb{R}^{dn_n \times 1} \tag{6}$$

Equation (6) represents the node vector, so-called generalized nodal coordinates. Using the generalized node coordinate, the e-th component length in Figure 1 that connects the nodes $\mathbf{n}_{ev}$ and $\mathbf{n}_{eu}$ is calculated as

$$l_e = \left\| \left(\mathbf{c}_e \otimes \mathbf{I}_d \right) \mathbf{n} \right\| = \left\{ \mathbf{n}^T \left[\left(\mathbf{c}_e{}^T \mathbf{c}_e \right) \otimes \mathbf{I}_d \right] \mathbf{n} \right\}^{\frac{1}{2}} \tag{7}$$

in which $\otimes$ is the Kronecker product and $\mathbf{I}_d$ is the identity matrix of size $d \times d$. Considering the sum of the product of the dimensional connectivity vector as a matrix $\mathbf{C}_e$, the expression of the length of the *e*-th component in equation (7) is simplified to

$$l_e = \left[\mathbf{n}^T \mathbf{C}_e \mathbf{n} \right]^{\frac{1}{2}} \tag{8}$$

with $\mathbf{C}_e$, being a symmetric and positive semidefinite matrix, written using the tensor product and the vector connectivity of the node matrix as

$$\mathbf{C}_e = \left(\mathbf{c}_e{}^T \mathbf{c}_e \right) \otimes \mathbf{I}_d \in \mathbb{R}^{dn_m \times dn_m} \tag{9}$$

2.2 Element's axial force

Element forces in tensegrity are typically tensile and compressive forces. They are inner forces of the structural elements, which act as external forces on the starting and ending nodes of elements. The nodes of elements are defined with the nodal position in a cartesian frame, so it is more suitable for equilibrium analysis of tensegrity to represent the inner force f_e of the *e*-th element on the starting and end node of the element as a force vector acting in *d*- directions as

$$\mathbf{f}_e = f_e \frac{\mathbf{C}_e \mathbf{n}}{l_e} \tag{10}$$

where $\mathbf{C}_e\mathbf{n}/l_e$ is the cosine direction of the *e*-th element and $\mathbf{f}_e$ is the force vector. Let us now take the ratio of force and length as force per unit length, noted as δ_e and given as

$$\delta_e = f_e l_e^{-1} \tag{11}$$

δ_e is known as the force density of the e-th member (Schek, 1974; Tran and Lee, 2010). Equation (11) into (10) yields

$$\mathbf{f}_e = \delta_e \mathbf{C}_e \mathbf{n} \tag{12}$$

The magnitude of $\mathbf{f}_e$ is proportional to the element vector $\mathbf{C}_e\mathbf{n}$ and acts on the starting and end nodes of the element in the opposite direction relative to the cartesian frame axis. The sign of the element vector defines the direction of the magnitudes of the force vectors. The magnitudes of the force vector are axial components in the *d*-direction of the cartesian frame, while the resultant force is axial to the element.

3 STATIC EQUILIBRIUM OF TENSEGRITY SYSTEM

This section presents the equilibrium equations of a general *d*-dimensional tensegrity structure based on the force density, generalized node coordinate, element forces, and element length. The static equilibrium of a typical node denoted by *j* in *d*-direction is expressed as follows:

$$\mathbf{f}_j = \sum_{e \ni j} \mathbf{f}_e \tag{13}$$

The notation $e \ni j$ denotes the set of forces of *e* members meeting at node *j*. The force $\mathbf{f}_j$ represents the residual nodal force of the sum of the magnitude of inner forces in *d*-direction; the residual nodal force is zero in each direction for a tensegrity in a self-equilibrium state.

Replacing the right side of equation (13) with the right side of equation (12), the nodal force of equation (13) is represented by a term combining the force density and the generalized nodal coordinate as

$$\mathbf{f}_j = \sum_{e \ni j} \left(\delta_e \mathbf{C}_e \right) \mathbf{n} \tag{14}$$

In our shape-finding approach, we use the strut forces instead of the force densities so that the residual nodal force expressed in equation (14) holds for

$$\mathbf{f}_j = \sum_{c \ni j}^{n_c \ni j} \delta_c \mathbf{C}_c \mathbf{n} - \sum_{s \ni j}^{n_s \ni j} \frac{f_s}{l_s} \mathbf{C}_s \mathbf{n} \tag{15}$$

Where $n_c \ni$ j represents a network of *c* tensile members of the force density δ_c connected to the node *j*. f_s and l_s are the force and length of the *s*-th strut meeting at the node *j*. The sign minus in equation (15) denotes the compression of struts.

Considering the whole system of n_n nodes (j=1,2, … ,n_n), the feasible set of the prestress vector for a self-balanced system is

$$\mathbf{F}_\mathbf{n} = \sum_{j=1}^{n_n} \mathbf{f}_j = \sum_{c=1}^{n_c} \delta_c \mathbf{C}_c \mathbf{n} - \sum_{s=1}^{n_s} \frac{f_s}{l_s} \mathbf{C}_s \mathbf{n} \tag{16}$$

The lengths of the struts are nonlinear functions of generalized coordinates, hence the system of the residual nodal force $\mathbf{F_n}$ (dn_n×1). The force is written as $\mathbf{F_n}$ to show its dependence on the position of node coordinates. A similar notation is adopted in all functions and equations whose values depend on the node coordinates, except for strut lengths, equilibrium, and geometric stiffness matrices. Equation (16) is divided into the tensile axial force and compressive axial force and can be simplified as follows:

$$\mathbf{F}_\mathbf{n} = \mathbf{K}_\mathrm{G} \mathbf{n} \tag{17}$$

where

$$\mathbf{K}_\mathrm{G} = \mathbf{K}_c + \mathbf{K}_s,\ \mathbf{K}_c = \sum_{c=1}^{n_c} \delta_c \mathbf{C}_c,\ \text{and}\ \mathbf{K}_s = -\sum_{s=1}^{n_s} \frac{f_s}{l_s} \mathbf{C}_s \tag{18}$$

Note that $\mathbf{K_G}$ ($dn_n \times dn_n$) is the symmetric matrix called the geometric stiffness matrix (Tran and Lee, 2010), which can be proved based on equations (9), (12) and (16). In addition, $\mathbf{K}_c$ is positive semidefinite, contrary to $\mathbf{K}_s$, which is

negative semidefinite due to the negative force densities expressing the compression in struts. To prove those assertations, let's consider a matrix **D** given as

$$\mathbf{D} = \sum_{e}^{n_m} \delta_e \left(\mathbf{c}_e{}^T \mathbf{c}_e \right) \in \mathbb{R}^{n_n \times n_n} \tag{19}$$

The matrix defined in equation (19) is a force density matrix in which the sum of the entries in each column is zero (Heng et al., 2021). Taking into account the relationship between $\mathbf{C}_e$ and $\mathbf{c}_e$, the force density matrix can be expanded to the geometric stiffness matrix as follows:

$$\mathbf{K}_{\mathrm{G}} = \mathbf{D} \otimes \mathbf{I}_d \tag{20}$$

Let's emphasize that equation (17) applies only for the nodal force $\mathbf{F_n}$, which does not take into account the elements' torsion and does not consider yielding and buckling constraints or the self-weight of the structural elements. The structure's scale depends mainly on the force densities (weight) (Miki, 2011; Koohestani, 2017). It is clear that, from equation (17), the self-equilibrium of the tensegrity structure is ensured by the relationship between the force densities and the nodal positions; in other words, the force densities and length of elements. Thus, the presence of a residual force ($\mathbf{F_n} \neq 0$) in a system that is in self-equilibrium shows the unbalance between the given force densities, the struts' member force, and the node positions.

4 FORM-FINDING ALGORITHM

The form-finding method finds the generalized nodal coordinates **n** that balance the nonlinear equilibrium equation (17). The solution of the nonlinear system is obtained under the following assumptions:

- Assumption 1: The force density of cables and the forces of strut elements are given in advance.
- Assumption 2: There is no external force, such as example, the reaction force due to nodes' support.
- Assumption 3: The structure elements are considered as lines connecting two nodes such that the topology of the structure is given and remains unchangeable along the performance of the form-finding algorithm.

For the nonlinear system given in equation (17), the Newton's step and Newton-like approaches are the proposed strategic methods to find an approximate solution to similar problems (Al-Towaiq and Hour, 2016; Al-Towaiq and Abu Hour, 2017). However, these methods have significant drawbacks, such as starting with an initial estimate close to the zeros of the nonlinear system (Vetterling and Press, 1992) and finding an acceptable step length that ensures the algorithm's convergence. Based on the above assumptions, we cannot define the reference configuration close to the final solution because the lengths of struts and cables are unknown. The nonlinear static equilibrium is solved using the modified Newton-Raphson method and ensure the algorithm's convergence when the starting point is random.

4.1 Intermediary function: Nonlinear least square function

The feasibility of the iterative algorithm for solving equation (17) consists in reducing the least square function ($||\mathbf{F_n}||^2/2 = \mathbf{F_n}^T\mathbf{F_n}/2$) at each iteration, which can be considered as an approximately equal condition for trying to minimize the function

$$\Pi_{\mathbf{n}} = \frac{1}{2}\mathbf{n}^T \mathbf{K}_{\mathrm{G}}{}^2 \mathbf{n} \tag{21}$$

Equation (21) is the least square function obtained from the nonlinear equilibrium system of equation (17). There are effective techniques (and software implementations) that solve the least squares problem with tremendous accuracy and very high reliability (Boyd and Vandenberghe, 2004). However, there are circumstances when the minimization of equation (21) is not equivalent to explicitly zeroing $\mathbf{F_n}$; because it may occasionally fail by landing on a minimum that does not respond to the state of self-equilibrium.

The relative condition that the minimum of the least square function is the solution of $\mathbf{F_n}$ involves the first-order and the second-order optimality conditions, called gradient and Hessian, respectively. The vector **n** minimizes $\Pi_\mathbf{n}$ when the gradient respond to (Antoniou and Lu, 2007)

$$\nabla\Pi_{\mathbf{n}} = \frac{\partial\Pi_{\mathbf{n}}}{\partial\mathbf{n}} = 0 \tag{22}$$

which can be obtained as

$$\nabla\Pi_{\mathbf{n}} = \mathbf{J_n}^T\mathbf{F_n}; \tag{23a}$$

$$\nabla\Pi_{\mathbf{n}} = \mathbf{J_n}^T\mathbf{K}_{\mathrm{G}}\mathbf{n} \tag{23b}$$

where $\mathbf{J_n}$ is the Jacobian matrix. The derivation of equation (23) and the Jacobian's expression can be found in **Appendix**. If we consider in equation (23b), the Jacobian matrix non-singular and the non-zero vector **n**, the first order optimality condition can be fulfilled when the geometric stiffness matrix is singular. The further condition of vector **n** being the minima of $\Pi_\mathbf{n}$ requires the second-order optimality condition, so-called Hessian, to be positive semidefinite at the second-order derivative of $\Pi_\mathbf{n}$.

$$\begin{aligned}\nabla^2\Pi_{\mathbf{n}} &= \nabla(\mathbf{J_n}^T\mathbf{F_n}) \\ &= \nabla\mathbf{J_n}^T\mathbf{F_n} + \mathbf{J_n}^T\mathbf{J_n}\end{aligned} \tag{24}$$

The high term order ($\nabla\mathbf{J_n}^T=\nabla^2\mathbf{F_n}$) is relatively expensive to compute, particularly in high-dimensional problem, and add significant computational complexity and memory requirements. Thus, the Hessian can be approximated according to reference (Boyd and Vandenberghe, 2004) as

$$\nabla^2\Pi_{\mathbf{n}} \approx \mathbf{J_n}^T\mathbf{J_n} \tag{25}$$

According to the expression of the Jacobian in **Appendix**, the matrix $\mathbf{J_n}$ is square and symmetric $\mathbf{J_n}=\mathbf{J_n}^T$. The positive semi-definiteness of the Hessian in equation (25) is analyzed using the eigenvalue decomposition considering the symmetry of the Jacobi matrix. The eigenvalue decomposition of the Hessian is given as:

$$\begin{aligned}\nabla^2\Pi_{\mathbf{n}} &= \underbrace{\Theta\Psi\Theta^T}_{\mathbf{J_n}=\mathbf{J_n}^T}\underbrace{\Theta\Psi\Theta^T}_{\mathbf{J_n}} \\ &= \Theta\Psi^2\Theta^T\end{aligned} \tag{26}$$

where **Θ** is the orthogonal matrix and **Ψ** is the diagonal matrix that contains the eigenvalue of $\mathbf{J_n}$. All $\mathbf{\Psi}_{ii}^2=\Psi_i^2$ are positive in the Hessian matrix, indicating that the Hessian is positive semidefinite. For the optimum **n** that minimizes $\Pi_\mathbf{n}$ defines a set of self-equilibrium states. However, stability also ought to be considered. The stability is analyzed through the properties of the geometric stiffness matrix. The eigenvalue decomposition of the geometric stiffness matrix is

$$\mathbf{K}_{\mathrm{G}} = \Phi\lambda\Phi^T \tag{27}$$

where **Φ** ($dn_n \times dn_n$) is the matrix of eigenvector whose are the basis of $\mathbf{K}_\mathrm{G}$. **λ** ($dn_n \times dn_n$) is the diagonal matrix containing the eigenvalue whose *i*-th eigenvalue ($\lambda_i=\mathbf{\lambda}_{ii}$, with $\lambda_1 \le \ldots\ \lambda_i \le \ldots \le \lambda_{\mathrm{dnn}}$) corresponds to the *i*-th eigenvector. There are three cases related to the eigenvalue of $\mathbf{K}_\mathrm{G}$ that need to be considered:

- Case 1: The eigenvalues are in increasing order as:

$$\lambda_1 = \cdots = \lambda_r = 0, \lambda_{dn_n - r} \leq \cdots \leq \lambda_{dn_n} \tag{28}$$

where r is the number of zero eigenvalues of $\mathbf{K}_G$. The geometric stiffness matrix is positive semidefinite, and the structure is super-stable only if $r \geq d(d+1)$. This condition is equivalent to the condition related to the rank deficiency of the force density matrix (n_n-rank($\mathbf{D}$) $\geq d+1$) proposed in references (Connelly and Terrell, 1995; Connelly, 2002; Tran and Lee, 2010). Those conditions are related based on equation (20) due to the geometric stiffness matrix being an expansion of the force density matrix in dimension d so that the stability condition can be confirmed by the rank deficiency $d(n_n\text{-rank}(\mathbf{D})) \geq d(d+1)$.

- Case 2: The number of zero eigenvalues is $r \leq d^2$. $\mathbf{K}_G$ is positive semidefinite, but the obtained node position degenerates into a lower dimension. Similar to the first case, the rank deficiency of the force density matrix is lower than the requirement, so the obtained nodal coordinates for 2-d and 3-d structures are co-linear and co-planar, respectively. This relates to an inadequate scaling set of the force densities, especially for cables, which happens with structures with one degree of freedom of the self-stress state.
- Case 3: $\mathbf{K}_G$ is negative semidefinite and the eigenvalue are in increasing order

$$\lambda_1 \leq \cdots \leq \lambda_p < 0, \lambda_1 = \cdots = \lambda_r = 0, \lambda_1 \cdots \leq \lambda_{dn_n - (p+r)} \tag{29}$$

If $r \geq d(d+1)$ is satisfied, we evaluate the eigenvalues of the tangent stiffness matrix $\mathbf{K}_T$ of the tensegrity given in (Guest, 2006; Ohsaki and Zhang, 2006; Zhang and Ohsaki, 2006) as follows:

$$\mathbf{K}_{\mathrm{T}} = \mathbf{K}_{\mathrm{G}} + \mathbf{K}_{\mathrm{E}} \tag{30}$$

where $\mathbf{K}_E$ is the linear stiffness matrix given as:

$$\mathbf{K}_{\mathrm{E}} = \mathbf{A}\, diag\left(\frac{\mathbf{ea}}{\mathbf{L}}\right)\mathbf{A}^T \tag{31}$$

e, **a**, and **L** denote vectors of Young's moduli, cross-section areas, and prestressed lengths of n_m members, respectively. **A** is an equilibrium matrix obtained as in (Ma et al., 2022b).

$$\mathbf{A} = \mathbf{C}^T \oplus I_d \;\; b.d.\left(N\mathbf{C}^T\right) \tag{32}$$

in which *b.d.* (●) is the block diagonal of ●. The eigenvalues of the tangent stiffness matrix are in decreasing order as

$$\sigma_{dn_n} \geq \cdots \geq \sigma_{r_m - 1}, \sigma_{r_m} = \cdots = \sigma_1 = 0 \tag{33}$$

The zero r_m eigenvalues of the tangent stiffness matrix correspond to the rigid body motion and internal mechanism modes. To evaluate the system stability, it is necessary to check if the forces of structural elements can stiffen all the internal mechanism modes (Zhang and Ohsaki, 2006; Tran and Lee, 2010).

4.2 Modified Newton-Raphson method

Various techniques exist for solving the nonlinear least square function (Nocedal and Wright, 1999). The Gauss-Newton approach, a modified Newton's line search method, is the most simple and straightforward technique for solving equation (21). The benefit of the Gauss-Newton method is that it uses equation (25) to approximate the Hessian rather than equation (24) to determine the Gauss-Newton direction (Boyd and Vandenberghe, 2004; Yuan et al., 2017).

$$\mathbf{J_n}^T\mathbf{J_n}\Delta\mathbf{n} = \mathbf{J_n}^T\mathbf{F_n} \tag{34}$$

The Gauss-Newton search step in equation (34) is the value that minimizes the least square function. Although the consideration of using the step length in equation (34) requires a significant value of the eigenvalue Ψ_i to ensure an acceptable step $\Delta\mathbf{n}$ is a descendent direction ($\Pi_\mathbf{n} \leq \Pi_{\tilde{\mathbf{n}}}$). This has led the Levenberg-Marquardt method presented in (Yuan et al., 2017) to modify the Hessian heuristically approximated with the tangent stiffness matrix by adding a positive definite matrix to the Hessian; it becomes the descending direction of the least square function. However, an improper choice of the quantity added to the Hessian leads to small steps that slowly reduce the function $\Pi_\mathbf{n}$, with a decrease criterion that does not guarantee quadratic convergence. This section proposes a MN-RM for zeroing the nonlinear equilibrium system.

The algorithm uses the Newton step and establishes a decrease rate criteria within an acceptable step length. The Newton step can be calculated from equation (34) as

$$\begin{aligned}\Delta\mathbf{n} &= -\left(\mathbf{J_n}^T\mathbf{J_n}\right)^{-1}\mathbf{J_n}^T\mathbf{F_n} \\ &= -\mathbf{J_n}^{-1}\mathbf{F_n}\end{aligned} \tag{35}$$

When the step decreases the least square function, the Jacobian matrix tends to become singular as its value depends on the geometric stiffness matrix and leads to the largest step length. To ensure the Hessian is not singular as the node vector approaches the minimum of $\Pi_\mathbf{n}$, we add a positive-defined diagonal matrix so that the step is as follows:

$$\Delta\mathbf{n} = -\left(\mathbf{J_n} + \beta\mathbf{I}_{dn_n}\right)^{-1}\mathbf{F_n} \tag{36}$$

β is a positive quantity that guarantees the existence of the decedent step. If the value of β in equation (36) is small (i.e., β <1); $\Delta\mathbf{n}$ increases and decreases with β >1. In equation (36) the positive definite matrix does not change during the iteration of form-finding; it ensures that the Jacobi has the inverse. The acceptable Newton step is a descending direction of equation (21) with an initial rate of decrease given as

$$\begin{aligned}\nabla\Pi_\mathbf{n}\Delta\mathbf{n} &= -\mathbf{F_n}^T\mathbf{J_n}\mathbf{J_n}^{-1}\mathbf{F_n} \\ &= -\mathbf{F_n}^T\mathbf{F_n}\end{aligned} \tag{37}$$

The positive definite diagonal matrix is unequivocally disregarded as the decrease rate remains unaffected by the value of the variable β. As the function $\Pi_\mathbf{n}$ approaches the minimum, the nonlinear system equation (17) moves toward self-equilibrium; at this point, the step must further reduce $\Pi_\mathbf{n}$ while avoiding $\Delta\mathbf{n}$ to cause the divergence of the node vector from the solution. To control the node vector's divergence, the search direction is determined by the step $\Delta\mathbf{n}$, while the step size η is used to regulate the decrease rate so that the new position of the node vector is thus obtained as:

$$\mathbf{n} = \tilde{\mathbf{n}} + \eta\Delta\mathbf{n};\ \text{with: } 0 < \eta \leq 1 \tag{38}$$

From equation (38), it is possible to construct a sequence of steps in which $\Pi_\mathbf{n}$ decreases too slowly relative to the step length or the line of decreases in $\Pi_\mathbf{n}$ is too small relative to the initial rate of decrease equation (37) as it is explained in (Vetterling and Press, 1992). As a result, the algorithm may not converge to a minimum of $\Pi_\mathbf{n}$. For this purpose, we assume that the average acceptance rate of the decrease of the least-square function at the new position is at least a fraction of the initial rate.

$$\Pi_\mathbf{n} \leq \Pi_{\tilde{\mathbf{n}}} - \alpha\nabla\Pi_\mathbf{n}\Delta\mathbf{n} \quad \text{with} \quad 0 < \alpha < 1 \tag{39}$$

Note that the step length depends on the rate of decrease, and it must guarantee a decrease of $\Pi_{\mathbf{n}}$ at each iteration. To obtain the value of η that ensures the satisfaction of equation (39), we assume that the full Newton step length (η=1) will reduce $\Pi_{\mathbf{n}}$ to satisfy the criteria of decreasing the equation (39) at the first step.

Therefore, the value of η can be obtained using the quadratic approximation of the least square function at the nodal position given in equation (38). Let's redefine the least square function rewritten as a function Π_η of a step size η as:

$$\Pi_\eta = \underbrace{\Pi\left(\mathbf{n} + \eta\Delta\mathbf{n}\right)}_{\Pi_n} \tag{40}$$

The quadratic approximation of Π_η is calculated as

$$\Pi_\eta \approx \left(\underbrace{\Pi_{\mathbf{n}}}_{\Pi_\eta|_{\eta=1}} - \underbrace{\Pi_{\tilde{\mathbf{n}}}}_{\Pi_\eta|_{\eta=0}} - \nabla\Pi_{\mathbf{n}}\Delta\mathbf{n} \right)\eta^2 + \eta\nabla\Pi_{\mathbf{n}}\Delta\mathbf{n} + \Pi_{\tilde{\mathbf{n}}} \tag{41}$$

From which its minimum is (Vetterling and Press, 1992) given by the following equation:

$$\eta = -\frac{\nabla\Pi_{\mathbf{n}}\Delta\mathbf{n}}{2\left(\Pi_{\mathbf{n}} - \Pi_{\tilde{\mathbf{n}}} - \nabla\Pi_{\mathbf{n}}\Delta\mathbf{n}\right)} \tag{42}$$

The process from equation (36) to (42) is repeated with the setting of $\mathbf{n}$ to $\tilde{\mathbf{n}}$ till the design tolerance error ϵ is smaller or equal to the tolerance error of the target function.

4.3 MN-RM form-finding algorithm

To achieve the self-balance of the tensegrity system based on the nonlinear system of equation; the nodal force function, the least square function, and the Jacobian matrix are iteratively evaluated; for this purpose, we explicitly define these functions to assess them without their reconstruction at the given position. The algorithm consists of the following main parts:

(i) Input: the initial random starting point ($\tilde{\mathbf{n}}$); the topology of the structure $\mathbf{c}_e$ (e=1 … n_m), the cables forces densities, struts force, β and the design tolerance error ϵ.

(ii) Calculate $\mathbf{F_n}$ and $\mathbf{J_n}$ as explicit function, and set i=1.

(iii) Evaluate $\mathbf{F_n}$ and $\mathbf{J_n}$ at $\tilde{\mathbf{n}}$.

(iv) Calculate the Newton step $\Delta\mathbf{n}$ using equation (36).

(v) Update $\mathbf{n}$= $\tilde{\mathbf{n}}$ + $\eta\Delta\mathbf{n}$; use the full step at first iteration (i=1 ← η=1).

(vi) If the decrease criteria in equation (39) is satisfied, go to the next step; otherwise, calculate the minimum of η using equation (42) and return to step (v).

(vii) Evaluate the tolerance $\epsilon \leq \Pi_{\mathbf{n}}$ if satisfied, stop and extract $\mathbf{n}$. Otherwise, set i=1←i+1, $\tilde{\mathbf{n}} \leftarrow \mathbf{n}$ and return to step (iii).

5 NUMERICAL EXAMPLES

Numerical examples of various tensegrities are presented to capture the efficiency of the shape-finding algorithm presented in the section 4.3. The initial guess of the configuration with residual nodal forces is solved using the MN-RM algorithm to define the self-equilibrium of the structure. The stability of the obtained configuration is analyzed using the geometric and tangent stiffness matrix criteria for stability. In all examples, the initial parameters related to the rate of decrease and convergence criteria are set as α=10^{-4} and ϵ =10^{-4}, respectively.

5.1 X-module tensegrity

The X-modules are structures consisting of crossbars and four cables. The number of struts differs according to the class of the system; class one structures have two struts, while class four modules have four struts meeting at a node, respectively. The topology and self-equilibrium configurations related to class one and four X-modules are shown in Figure 2.

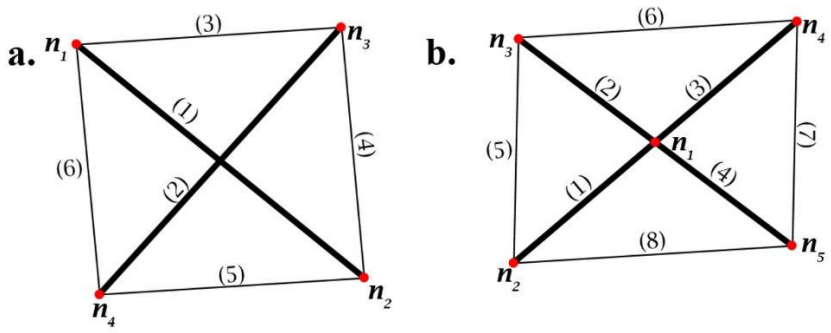


Figure 2: Obtained geometry of two X-module tensegrity structures.

The configuration in Figure 2.a is the class one structure consisting of two struts connected by four cables within the choice of cables' force density $\delta_{3\text{-}6}$=1.40Ncm^{-1} and the struts' force $f_{1\text{-}2}$=20.00N, the length of elements and r (=6) smallest eigenvalues that confirm the super stability of the structure are given in Table 1.

Table 1 Final length and smallest eigenvalues of X-module tensegrity structure in Figure 2.

	i	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
	l_i [cm]	14.29	14.29	10.00	10.20	10.00	10.20		
Figure 2.a	$\lambda_i \times 10^{-5}$	-0.51	-0.51	-0.51	-0.51	0	0		
	l_i [cm]	10.00	10.00	10.00	10.00	14.11	14.18	14.12	14.18
Figure 2.b	σ_i	0	0	0	1.00	2.83	3.83	5.43	5.42

The self-equilibrium and the super stability states are obtained at the 20-th iteration within the target error 0.73×10^{-4} and β=3 (Figure 3). The force density in struts agrees with those obtained in (Tran and Lee, 2010; Zhang et al., 2014; Yuan et al., 2017).

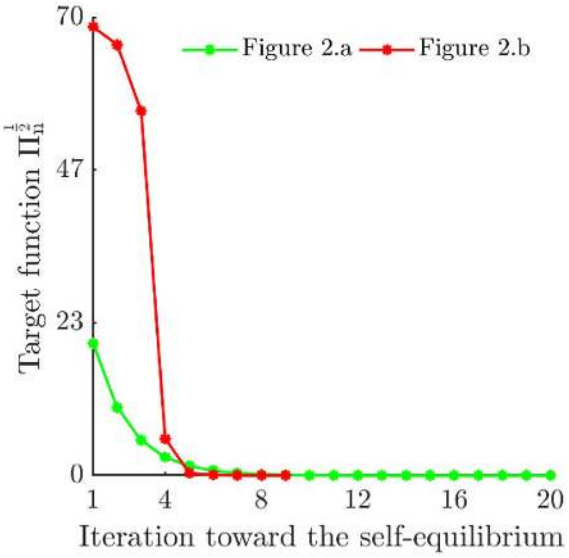


Figure 3: The convergence of the least square function of the X-modules towards the self-equilibrium.

The configuration of the class four tensegrity module obtained using the MN-RM algorithm is shown in Figure 2.b; the force density in cable members is $\delta_{5\text{-}8}$=1.00Ncm^{-1}, and the force in struts is $f_{1\text{-}4}$=20.00N. The converge graph plotted in Figure 3 shows the self-balanced configuration was obtained on the 9-th iteration within the error of 0.46×10^{-4} by

setting β=0. The eigenvalues presented in Table 1 are obtained after evaluating the tangent stiffness matrix with the members' axial stiffness ($e_i a_i$) set to 0.10. The three zero eigenvalues represent the infinitesimal mechanism mode and the rigid body motions that can be constrained with the prestress (Tran and Lee, 2010). The node position of the initial guess and the self-balanced structures shown in Figure 2 are given in Table 2.

Table 2 Initial and final node coordinates in configuration of tensgrity structures in Figure 2.

		n_i	1	2	3	4	5
Initial guess	Figure 2.a	*x*-	0.39	0.67	0.74	0.52	-
		y-	0.35	0.15	0.59	0.26	-
	Figure 2.b	*x*-	0.41	0.67	0.93	0.81	0.48
		y-	0.76	0.42	0.97	0.99	0.86
Self-equilibrium	Figure 2.a	*x*-	-5.24	6.40	4.61	-3.45	-
		y-	4.48	-3.80	6.24	-5.56	-
	Figure 2.b	*x*-	-58.33	-66.60	-64.00	-50.07	-52.66
		y-	148.85	143.22	157.09	154.48	140.61

5.2 Octahedral cell tensegrity

The systems in Figure 4 are class two structures with eight cables and five struts. The force densities of cables and strut forces related to different cases of the final configuration are given in Table 3. As shown in Figure 5, the MN-RM iterative procedure has converged on the 6-th iteration for case 1 and case 3 with a minimum of the target function of 0.67×10^{-4} and 0.62×10^{-4}, respectively. The convergence of the least squares function of case 2 towards self-equilibrium with a design error of 0.90×10^{-4} was obtained on the 12-th iteration.

Table 3 Forces density of cables and strut forces of cases considered for octahedral cell tensegrity.

element (*i*)	f_i [N]					δ_i [Ncm^{-1}]							
	1	2	3	4	5	6	7	8	9	10	11	12	13
nodes	n_1-n_2	n_2-n_3	n_3-n_4	n_1-n_4	n_5-n_6	n_1-n_5	n_2-n_5	n_3-n_5	n_4-n_5	n_1-n_6	n_2-n_6	n_3-n_6	n_4-n_6
Case 1	14.60	14.60	14.60	14.60	14.60	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Case 2	11.00	11.00	11.00	11.00	8.00	1.00	1.00	1.20	1.20	1.00	1.00	1.20	1.20
Case 3	15.00	11.00	15.00	11.00	21.00	0.72	0.72	0.72	0.72	1.20	1.20	1.20	1.20

After investigating the stability condition through eigenvalue decomposition in equation (27), there were r=12 zero eigenvalues. Still, the smallest eigenvalues were negative (-2.00, -2.07, and -1.92 for case 1, case 2, and case 3, respectively), indicating the structure was not super stable. To this end, the tangent stiffness matrix was investigated, where the axial stiffness was set to 0.50 for struts and 0.0005 for cables, and the structures have no infinitesimal mechanism after constraining their rigid body motion. The tangent stiffness matrix was positive, meaning the structures are in the prestressed stable configuration.

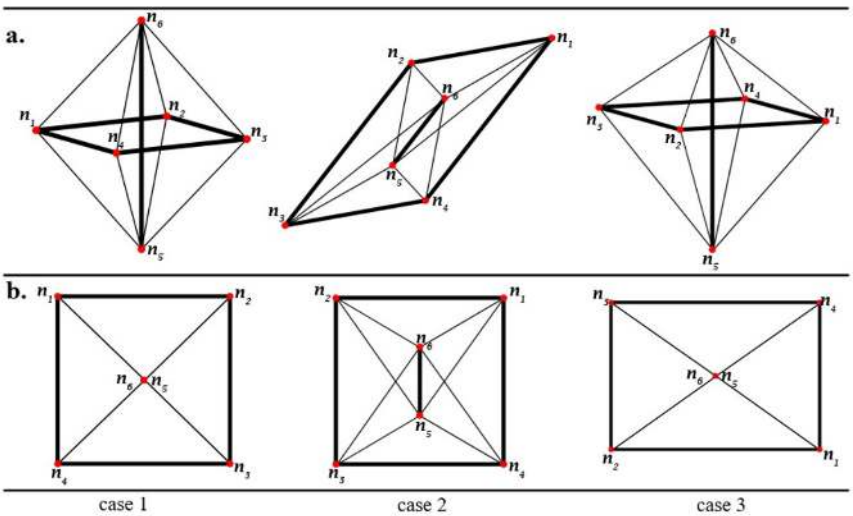


Figure 4: Self-equilibrium configurations of octahedral cell tensegrity. a. perspective view, b. top view of each case.

Table 4 contains the lengths of elements of the self-equilibrium configurations corresponding to the structural conditions given in Table 3. In addition, the force densities of struts of case 1 are consistent with the values found in (Tran and Lee, 2010).

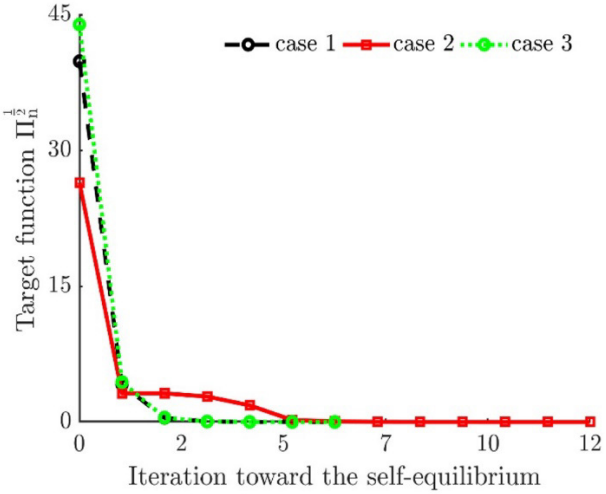


Figure 5: The convergence of the least square function of the octahedral cell

Table 4 Length of structural members on the self-equilibrium state of octahedral cell tensegrities.

l_i [cm]	1	2	3	4	5	6	7	8	9	10	11	12	13
Case 1	14.60	14.60	14.60	14.60	14.60	7.30	10.95	10.95	10.95	10.95	10.95	10.95	10.95
Case 2	10.38	10.63	10.63	10.63	4.38	9.12	9.12	6.06	6.06	9.12	9.12	6.06	6.06
Case 3	16.67	12.50	16.67	12.50	11.67	12.72	12.72	12.72	12.72	11.30	11.30	11.30	11.30

5.3 Tensegrity dome

By solving the linear static equilibrium, Tran and Lee (2010) have derived the normalized force densities and shown that the structure is stable regardless of the material properties. In this subsection, the tensegrity dome is solved using the MN-RM algorithm, and the stability state result from the analysis of the geometric stiffness matrix is compared to the stability state found in (Tran and Lee, 2010). The conditions taken for the cable force density are not the same as those in (Tran and Lee, 2010); the set of cable force density is grouped as $\delta_{7\text{-}12}$=1.46Ncm^{-1}, $\delta_{13\text{-}18}$=2.00Ncm^{-1} and strut forces $f_{1\text{-}6}$=20.00N. The plot in Figure 6 shows the nonlinear regressive curve of the convergence of the target function toward the self-equilibrium. The self-balanced model shown in Figure 7 is super stable regardless of the material properties, which Tran and Lee confirmed in (Tran and Lee, 2010). The elements' length of the self-stable tensegrity dome concerning the initial element properties is given in Table 5. Additionally, the related nodes coordinates are given in Table 6.

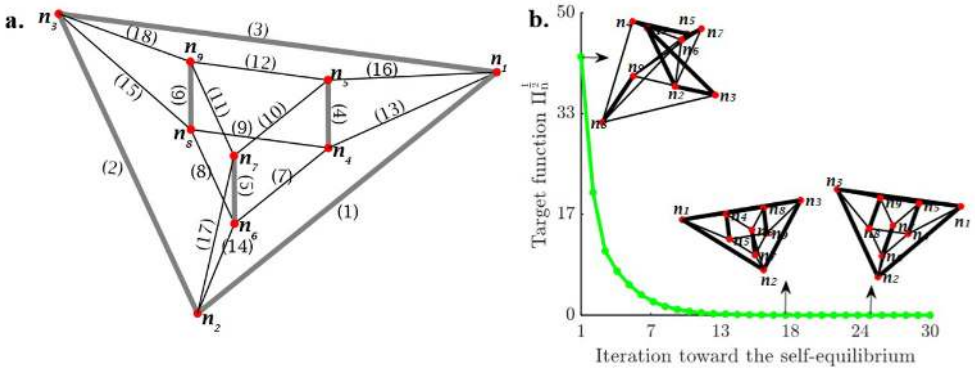


Figure 6: Tensegrity dome. a. Topology of the tensegrity dome, b. different configurations of the tensegrity dome obtained during the convergence of the target function toward the self-equilibrium.

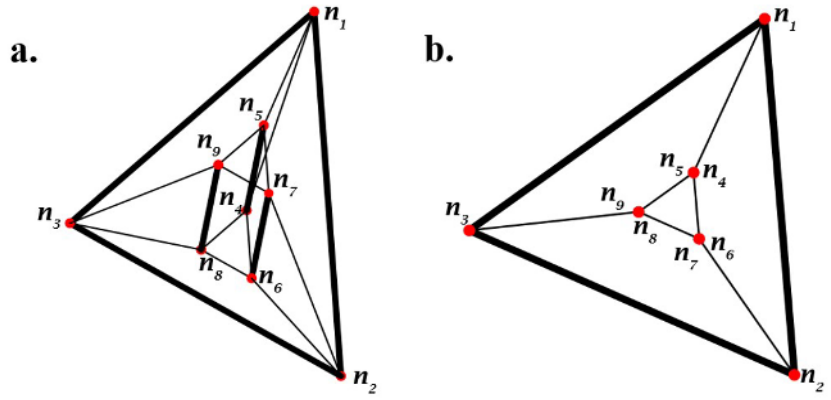


Figure 7: Obtained self-balanced configuration of the tensegrity dome. a. Perspective view, b. top view.

Table 5 Length of structural members on the self-equilibrium state of octahedral cell tensegrities.

i	1-3	4-6	7-12	13	14	15	16	17	18
l_i [cm]	26.23	24.00	8.23	17.07	15.09	15.29	14.58	16.62	16.35

Table 6 Initial and final node coordinates in configuration of the tensegrity dome.

	n_i	1	2	3	4	5	6	7	8	9
Initial guess	*x-*	0.32	0.55	0.98	0.55	0.33	0.62	0.36	0.76	0.41
	y-	0.49	0.69	0.97	0.33	0.84	0.74	0.95	0.03	0.36
	z-	0.66	0.28	0.23	0.71	0.62	0.59	0.66	0.05	0.35
Self-equilibrium	*x-*	-11.28	1.86	11.04	4.36	-10.70	8.49	-6.57	11.37	-3.69
	y-	-7.48	3.86	5.42	-10.93	7.06	-7.37	10.62	-6.88	11.11
	z-	5.40	-14.27	10.25	-0.50	4.53	-6.68	-1.65	1.02	6.05

5.4 Truncated tensegrity cells

We also consider the spherical tensegrity to enhance the integrity of the proposed form-finding algorithm. The general topology of spherical tensegrities consists of n_s struts and $n_c = 3n_s$ strings. To obtain a regular spherical shape, the strings are categorized as n_{ct} (=$2n_s$) string connecting the truncated edges and n_{cv} (=n_s) vertical strings. Accordingly, the force densities are categorized as δ_{ct}, the force density of the truncated string, and δ_{cv}, the force density of the vertical cables. Figure 8 shows the convergence of the truncated tetrahedral from a random initial guess to a self-equilibrium state. The structure in Figure 8.a is a truncated tetrahedral comprising 6 struts, 12 truncated strings, and 6 vertical cables. The properties of structural elements are that the cable force density was defined as δ_{ct}=0.87Ncm^{-1} and δ_{cv}=0.60Ncm^{-1} with struts force f_s=17.87N. The length of elements of the self-equilibrium configuration is obtained as l_{ct}=8.59cm, l_{cv}=19.54cm and l_b=29.80cm defining the super stability of the structure according to the analysis of the r smallest eigenvalue of the geometric stiffness matrix given in Table 7.

Table 7 Smallest eigenvalues of truncated tensegrity structure in Figure 8.

$\lambda_i \times 10^{-5}$	1-3	4-6	7-9	10-12	β	$\Pi_n^{1/2} \times 10^{-4}$	Iterations
Figure 8.a	-0.20	0	0.06	0.14	5.5	10.95	291
Figure 8.b	-0.12	0	0.27	0.68	5	9.12	6755
Figure 8.c	-0.75	0	0.03	0.72	10	12.72	429

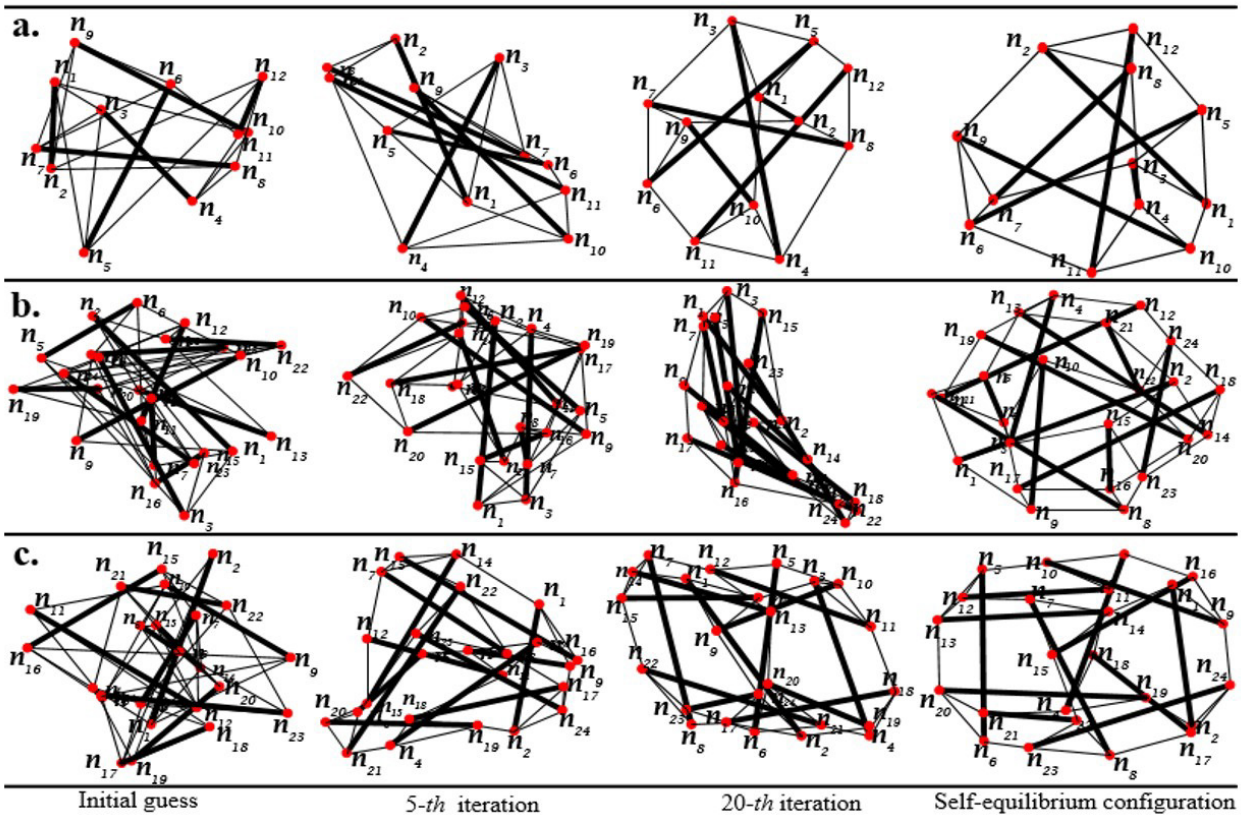


Figure 8: Different configurations of truncated tensegrities toward the self-equilibrium.

Figure 8.b-c are truncated octahedral and cubic configurations towards the self-equilibrium positions. The structural components consist of 24 and 12 struts, leading to 36 cables and 18 cables for the truncated octahedral and cubic structures, respectively. The force densities of truncated cables and vertical cables of the octahedral model are δ_{ct}=0.91Ncm^{-1} and δ_{cv}=0.86Ncm^{-1}, and the struts force f_b=18.00N. The self-equilibrium configuration has the element lengths as l_{ct}=16.17cm, l_{cv}=19.16cm and l_b=43.99 cm. The super stability state of the structure is concluded with the value of the smallest eigenvalues (in Table 7) of the geometric stiffness matrix. The force densities of truncated and vertical cables are δ_{ct}= δ_{cv}=1Ncm^{-1}, and the strut force is set to f_b=20.00N for the cubic truncated model. The balance of the structure obtained in Figure 8.c corresponds to strut lengths l_b=47.60cm and cable lengths l_{ct}=16.90cm, l_{cv}=19.51cm. According to the *r* smallest eigenvalue value in Table 7, the structure is stable regardless of the material property.

6 DISCUSSION

The MN-RM form-finding algorithm's accuracy and efficiency have been demonstrated in Section 5 with introductory examples of different tensegrities. In this section, we compare the performance of some form-finding algorithms (Yuan et al., 2017) that solve the system of nonlinear equilibrium equations for tensegrities.

First, we do not need to constrain the rigid body motion or modify the positive definite diagonal matrix as in (Zhang et al., 2014; Yuan et al., 2017) because we use a step control to ensure that the step length is the descending direction of the least square function. This allows MN-RM to save time in terms of function evaluation at each iteration, unlike Levenberg-Marquardt form-finding (LMFF) (Yuan et al., 2017) where the least square function is reevaluated in any no-descending direction of the step, resulting in the sub-step that consumes time.

Second, the MN-RM algorithm does not require a large value β for the diagonal entry of the positive definite matrix. Specifying a large value of β is a drawback of the algorithm due to its quadratic convergence. With β=10 tested on the prism tensegrity structure; the algorithm MN-RM requires 35 more iterations than LMFF to converge to the same design tolerance error. In this performance, the value of β was fixed during the iterative LMFF; moreover, the Hessian in LMFF is approximated by equation (25).

Finally, we compare using Matlab software the performance and efficiency of the MN-RM algorithm, the LMFF algorithm and Matlab *fsolve* built-in function. The conditions for the comparison are that we use the same structure and the same initial estimate; we approximate the Hessian with equation (25) instead of using the tangent stiffness matrix as

the Hessian in LMFF algorithm. The numbering of the structural elements can be seen in Figure 9.a. The initial parameters of the considered tensegrity prism are the force density of the vertical cables, and the horizontal cables are $\delta_{4\text{-}6}$=1.00Ncm^{-1} and $\delta_{7\text{-}12}$=0.58Ncm^{-1}. The strut forces are $f_{1\text{-}3}$=16.00N. It is confirmed that the MN-RM outperforms the LMFF and the Matlab *fsolve* built-in function, as shown in Figure 9.b. In case of β=0, the MN-RM takes one iteration to converge with high accuracy (ϵ=0.1323×10^{-12}), whereas when β=0.2, the number of iterations required to achieve ϵ=0.2350×10^{-4} increase up to 6. This means, by setting β=0 the MN-RM is simply a Newton-Raphson with line search technique which presents the advantage for simple structure. However, for more complex structure the used of β>0 is necessary. The LMFF reaches the minimum of the least square function at the 11-th iteration with a design error of ϵ =0.5823×10^{-4} when β=100. In LMFF, the value of β=100 was observed to ensure the descending direction of Π_n, with β divided by 2 at each iteration. The Matlab built-in *fsolve* function converge at the 5-th iteration with the design error ϵ=0.7538×10^{-7}.

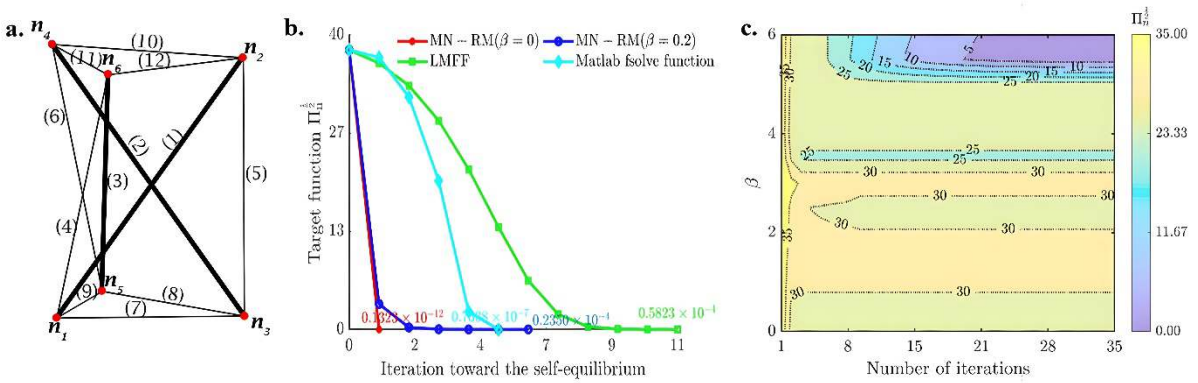


Figure 9: a. Configuration and structural numbering of tensegrity prism. b. Graph of convergence of the MN-RM, LMFF algorithms and Matlab *fsolve* function. (c) Contour plot of the value of β versus the number of iterations with respect to the target function.

Increasing the value of β in Figure 9.b increases the number of iterations, which is expected. On the other hand, in complex cases such as the tetrahedral tensegrity cell, the value of β might limit the method's ability to converge to the local minima of Π_n as shown in Figure 9.c. Thus, the value of β and backtracking procedures ensure that the MN-RM moves steadily towards the solution. Hence, β helps in such cases by refining the step size to ensure that the method adapts to the local curvature of the function, thus maintaining the convergence of the gradient toward the root.

7 CONCLUSION

The proposed design of a tensegrity structure uses the force density in the cables and the force of the struts with a specific structural topology. The nonlinear equilibrium system was developed using the cables' force density, the struts' length, and the strut force. For this purpose, the coordinates of the nodes were chosen as variables. The self-equilibrium system was obtained using the modified Newton-Raphson method, which ensures a reduction of the nonlinear least square function resulting from the nonlinear equilibrium. The design algorithm promises to be a robust and reliable solution for the intended application of form-finding. The decrease rate of the objective function was set to a fraction of the initial reduction and has shown the great convergence of the modified Newton-Raphson method. The choice of the positive diagonal matrix was crucial for the algorithm's convergence; with β>10, the algorithm converged in more than 40 iterations, in contrast to β<5, where convergence was achieved in less than 30 iterations.

ACKNOWLEDGMENTS

This work was supported by the National Natural Science Foundation of China (NFSC) [grant number 51875111], and the first, second, and fourth authors are grateful to the National Science Fund of Heilongjiang Province (No. LH2020E062).

Author's contributions: Conceptualization, A Vumiliya**; Methodology,** A Vumiliya, A Luo and A González-Fallas**; Investigation,** A Vumiliya, A Luo and A González-Fallas**; Writing - original draft,** A Vumiliya**; Writing - review & editing,** A Vumiliya, A Luo and A González-Fallas**; Funding acquisition,** H Liu**; Supervision,** A Luo and H Liu.

Editor: Marco L. Bittencourt

References

Al-Towaiq, M.H., Abu Hour, Y.S. (2017). Two improved classes of Broyden's methods for solving nonlinear systems of equations. Journal of Mathematics and Computer Science-JMCS 17:22–31

Al-Towaiq, M.H., Hour, Y.S.A. (2016). Two Improved Methods Based on Broyden's Newton Methods for the Solution of Nonlinear System of Equations. Journal of Engineering and Applied Sciences 11:2344–2348

Antoniou, A., Lu, W.-S. (2007). Practical optimization: algorithms and engineering applications. Springer

Baudriller, H., Maurin, B., Cañadas, P., et al. (2006). Form-finding of complex tensegrity structures: application to cell cytoskeleton modelling. Comptes rendus mécanique 334:662–668

Boyd, S.P., Vandenberghe, L. (2004). Convex optimization. Cambridge university press

Cai, J., Feng, J. (2015). Form-finding of tensegrity structures using an optimization method. Engineering Structures 104:126–132. https://doi.org/10.1016/j.engstruct.2015.09.028

Connelly, R. (2002). Tensegrity structures: why are they stable? Rigidity theory and applications 47–54

Connelly, R., Terrell, M. (1995). Globally rigid symmetric tensegrities. Struct Topol 1995 núm 21

Feron, J., Boucher, L., Denoël, V., Latteur, P. (2019). Optimization of footbridges composed of prismatic tensegrity modules. Journal of Bridge Engineering 24:4019112

Gasparini, D., Klinka, K.K., Arcaro, V.F. (2012). A finite element for form-finding and static analysis of tensegrity structures. Journal of Mechanics of Materials and Structures 6:1239–1254

Guest, S. (2006). The stiffness of prestressed frameworks: a unifying approach. International Journal of Solids and Structures 43:842–854

Harish, A.B., Nandurdikar, V., Deshpande, S., Andress, S.R. (2023). Mathematics of stable tensegrity structures. J Theor Comput Appl Mech

He, J., Wang, Y., Li, X., et al. (2024). A modified dynamic relaxation form-finding method for general tensegrity structures with inextensible tensile members. Comput. Struct. 291

Heng, T., Zhao, L., Liu, K., et al. (2021). An improved form-finding method for calculating force density with group theory. In: 2021 11th International Conference on Intelligent Control and Information Processing (ICICIP). pp 119–124

Khafizov, R., Savin, S. (2021). Length-constrained mixed-integer convex programming-based generation of tensegrity structures. 2021 6th IEEE International Conference on Advanced Robotics and Mechatronics, ICARM 2021 125–131. https://doi.org/10.1109/ICARM52023.2021.9536138

Koohestani, K. (2017). On the analytical form-finding of tensegrities. Composite Structures 166:114–119

Latteur, P., Feron, J., Denoel, V. (2017). A design methodology for lattice and tensegrity structures based on a stiffness and volume optimization algorithm using morphological indicators. International Journal of Space Structures 32:226–243

Ma, S., Chen, M., Peng, Z., et al. (2022a). The equilibrium and form-finding of general tensegrity systems with rigid bodies. Engineering Structures 266:114618

Ma, S., Chen, M., Skelton, R.E. (2022b). Tensegrity system dynamics based on finite element method. Composite Structures 280:114838

Miki, M. (2011). Three-term Method and Dual Estimate on Static Problems of Continuum Bodies. arXiv Prepr arXiv11081331

Motro, R. (2003). Tensegrity: Structural Systems for the Future. Butterworth-Heinemann, available at: https://doi.org/10.1016/B978-1-903996-37-9.X5028-8

Nocedal, J., Wright, S.J. (1999). Numerical optimization. Springer

Ohsaki, M., Zhang, J. (2006). Stability conditions of prestressed pin-jointed structures. International Journal of Non-Linear Mechanics 41:1109–1117

Paul, C., Valero-Cuevas, F.J., Lipson, H. (2006). Design and control of tensegrity robots for locomotion. IEEE Transactions on Robotics 22:944–957

Pellegrino, S. (1986). Mechanics of kinematically indeterminate structures. University of Cambridge

Schek, H.-J. (1974). The force density method for form finding and computation of general networks. Computer methods in applied mechanics and engineering 3:115–134

Tibert, A.G., Pellegrino, S. (2002). Deployable tensegrity reflectors for small satellites. Journal of Spacecraft and Rockets 39:701–709

Tibert, A.G., Pellegrino, S. (2011). Review of form-finding methods for tensegrity structures. International Journal of Space Structures 26:241–255

Tran, H.C., Lee, J. (2010). Advanced form-finding of tensegrity structures. Computers & structures 88:237–246

Vetterling, W.T., Press, W.H. (1992). Numerical recipes: The art of science computing-Second edition. Cambridge University Press

Wang, Y., Xu, X., Luo, Y. (2021). Form-finding of tensegrity structures via rank minimization of force density matrix. Engineering Structures 227:111419

Wang, Y., Xu, X., Luo, Y. (2023). Self-equilibrium, mechanism stiffness, and self-stress design of general tensegrity with rigid bodies or supports: A unified analysis approach. Journal of Applied Mechanics 90:81004

Xue, Y., Luo, Y., Xu, X. (2020). Form-finding of cable-strut structures with given cable forces and strut lengths. Mechanics Research Communications 106:103530. https://doi.org/10.1016/j.mechrescom.2020.103530

Yuan, X.-F., Ma, S., Jiang, S.-H. (2017). Form-finding of tensegrity structures based on the Levenberg--Marquardt method. Computers & Structures 192:171–180

Zhang, J.Y., Ohsaki, M. (2006). Adaptive force density method for form-finding problem of tensegrity structures. International journal of Solids and Structures 43:5658–5673

Zhang, L.-Y., Li, Y., Cao, Y.-P., Feng, X.-Q. (2014). Stiffness matrix based form-finding method of tensegrity structures. Engineering structures 58:36–48

Zhang, P., Zhou, J., Chen, J. (2021). Form-finding of complex tensegrity structures using constrained optimization method. Composite Structures 268:113971. https://doi.org/10.1016/j.compstruct.2021.113971

Appendix – The derivation of the relative condition for the mínimum of the least square function and the Jacobian's expression

The least-square function $\Pi_{\mathbf{n}}$ construct from the nonlinear equilibrium system is given in the simplified form of equation (23) and can be written as

$$\Pi_{\mathbf{n}} = \frac{1}{2}\mathbf{F}_{\mathbf{n}}{}^{T}\mathbf{F}_{\mathbf{n}} \tag{A.1}$$

The first-order derivative of the least-square function is as

$$\frac{\partial \Pi_{n}}{\partial \mathbf{n}} = \frac{1}{2}\frac{\partial}{\partial \mathbf{n}}\left(\mathbf{F}_{\mathbf{n}}{}^{T}\mathbf{F}_{\mathbf{n}}\right) \tag{A.2}$$

from which it follows that

$$\frac{\partial \Pi_{\mathbf{n}}}{\partial \mathbf{n}} = \frac{1}{2}\left(\frac{\partial \mathbf{F}_{\mathbf{n}}{}^{T}}{\partial \mathbf{n}}\mathbf{F}_{\mathbf{n}} + \mathbf{F}_{\mathbf{n}}{}^{\mathrm{T}}\frac{\partial \mathbf{F}_{\mathbf{n}}}{\partial \mathbf{n}}\right) = \frac{\partial \mathbf{F}_{\mathbf{n}}{}^{T}}{\partial \mathbf{n}}\mathbf{F}_{\mathbf{n}} \tag{A.3}$$

where

$$\nabla \Pi_{\mathbf{n}} = \frac{\partial \Pi_{\mathbf{n}}}{\partial \mathbf{n}} = \mathbf{J}_{\mathbf{n}}{}^{T}\mathbf{F}_{\mathbf{n}} \tag{A.4}$$

in which $\mathbf{J}_{\mathbf{n}}$ is the Jacobian matrix obtain as:

$$\mathbf{J}_{\mathbf{n}} = \frac{\partial \mathbf{F}_{\mathbf{n}}}{\partial \mathbf{n}} \tag{A.5}$$

For simplicity, we consider the jacobian matrix's *i*-th and *j*-th entries that correspond to the *i*-th and *j*-th entry of the nodal force and the node vector, the jacobian entries are determined as:

$$[\mathbf{J}_{\mathbf{n}}]_{ij} = \mathbf{R}_{i}\frac{\partial}{\partial [\mathbf{n}]_{j}}\left(\mathbf{K}_{\mathrm{G}}\mathbf{n}\right) \tag{A.6}$$

$\mathbf{R}_i$ is the column vector which the *i*-th entry is 1 and zero elsewhere. From equation (A.6), we obtain

$$[\mathbf{J}_{\mathbf{n}}]_{ij} = \mathbf{R}_{i}\left(\mathbf{K}_{\mathrm{G}}\frac{\partial \mathbf{n}}{\partial [\mathbf{n}]_{j}} + \frac{\partial \mathbf{K}_{\mathrm{G}}}{\partial [\mathbf{n}]_{j}}\mathbf{n}\right) \tag{A.7}$$

in which yields

$$[\mathbf{J}_{\mathbf{n}}]_{ij} = \mathbf{R}_{i}\left(\mathbf{K}_{\mathrm{G}}\mathbf{X}_{j} + \frac{\partial \mathbf{K}_{\mathrm{G}}}{\partial [\mathbf{n}]_{j}}\mathbf{n}\right) \tag{A.8}$$

where $\mathbf{X}_j$ is the vector given as

$$\mathbf{X}_j = \frac{\partial \mathbf{n}}{\partial [\mathbf{n}]_j} = \begin{bmatrix} 0 & \cdots & 1 & \cdots & 0 \end{bmatrix}^T \tag{A.9}$$

The *j*-th entry of the vector $\mathbf{X}_j$ is 1 and zero elsewhere. Considering the geometric stiffness matrix in equation (18) and the length of struts in equation (8), the term the partial derivative of the geometric stiffness matrix concerning the *j*-th entry of the node vector is as follows:

$$\frac{\partial \mathbf{K}_{\mathrm{G}}}{\partial [\mathbf{n}]_j} = \frac{\partial}{\partial [\mathbf{n}]_j} \left(\sum_{c=1}^{n_c} \delta_c \mathbf{C}_c - \sum_{s=1}^{n_s} \frac{f_s}{[\mathbf{n}^T \mathbf{C}_s \mathbf{n}]^{\frac{1}{2}}} \mathbf{C}_s \right) \tag{A.10}$$

Equation (A.10) can be simplified as

$$\frac{\partial \mathbf{K}_{\mathrm{G}}}{\partial [\mathbf{n}]_j} = -\frac{\partial}{\partial \mathbf{n}} \sum_{s=1}^{n_s} \left(\frac{f_s}{[\mathbf{n}^T \mathbf{C}_s \mathbf{n}]^{\frac{1}{2}}} \right) \mathbf{C}_s \tag{A.11}$$

From this, it follows that

$$f \frac{\partial \mathbf{K}_{\mathrm{G}}}{\partial [\mathbf{n}]_j} = -\sum_{s=1}^{n_s} \frac{\partial [\mathbf{n}^T \mathbf{C}_s \mathbf{n}]^{-\frac{1}{2}}}{\partial [\mathbf{n}]_j} f_s \mathbf{C}_s \tag{A.12}$$

$$\frac{\partial \mathbf{K}_{\mathrm{G}}}{\partial [\mathbf{n}]_j} = \frac{1}{2} \sum_{s=1}^{n_s} \frac{\mathbf{X}_j^T \mathbf{C}_s \mathbf{n} + \mathbf{n}^T \mathbf{C}_s \mathbf{X}_j}{[\mathbf{n}^T \mathbf{C}_s \mathbf{n}]^{\frac{3}{2}}} f_s \mathbf{C}_s \tag{A.13}$$

Equation (A.13) into (A.8) yields

$$[\mathbf{J}_{\mathbf{n}}]_{ij} = \mathbf{R}_i \left(\mathbf{K}_{\mathrm{G}} \mathbf{X}_j + \frac{1}{2} \sum_{s=1}^{n_s} \frac{\mathbf{X}_j^T \mathbf{C}_s \mathbf{n} + \mathbf{n}^T \mathbf{C}_s \mathbf{X}_j}{[\mathbf{n}^{\mathrm{T}} \mathbf{C}_s \mathbf{n}]^{\frac{3}{2}}} f_s \mathbf{C}_s \mathbf{n} \right) \tag{A.14}$$

The transpose of the $\mathbf{J_n}$ noted $\mathbf{J_n}^T$, can be obtained by transposition of the entries as

$$[\mathbf{J}_{\mathbf{n}}]_{ij}^{\ T} = [\mathbf{J}_{\mathbf{n}}]_{ji} = \left(\mathbf{X}_j^{\ T} \mathbf{K}_{\mathrm{G}} + \frac{1}{2} \sum_{s=1}^{n_s} \mathbf{n}^{\mathrm{T}} \frac{\mathbf{X}_j^T \mathbf{C}_s \mathbf{n} + \mathbf{n}^T \mathbf{C}_s \mathbf{X}_j}{[\mathbf{n}^T \mathbf{C}_s \mathbf{n}]^{\frac{3}{2}}} f_s \mathbf{C}_s \right) \mathbf{R}_i^{\ T} \tag{A.15}$$

which allows us to conclude

$$\mathbf{J}_{\mathbf{n}}^{\ T} = \mathbf{J}_{\mathbf{n}} \tag{A.16}$$