

A Survey of the Computational Modeling and Control of Tensegrity Robots

Erik Komendera

Abstract

For decades, tensegrity structures have been recognized primarily as an art form and an architectural style. In the last 15 years, however, roboticists and biologists have taken a keen interest in tensegrity structures for both their desirable properties and their apparent ubiquitousness in nature. Likewise, the field of soft robotics has attracted similar amounts of interest recently for similar reasons. However, the soft robotics and tensegrity robotics communities do not currently overlap. This survey considers the computational modeling of the dynamics and controls of tensegrity robots. This survey also argues for merging tensegrity robotics with soft robotics. This survey defines a general mathematical model of tensegrity systems as a baseline for starting research. Finally, this survey suggests future avenues of research in the context of soft, mobile tensegrity robots.

1 Introduction

Tensegrity structures are structures in which compression members (struts) and tension members (cables) are held together in equilibrium. The compression and tension in each member cancel out globally, creating a rigid structure. The most cited definition of tensegrity structures comes from [1]:

A tensegrity system is established when a set of discontinuous compression components interacts with a set of continuous tensile components to define a stable volume in space.

A more poetic definition was given by Buckminster Fuller in his patent [2]:

Islands of compression inside an ocean of tension.

The word *tensegrity* itself came from Fuller as a portmanteau of “*tensile integrity*”. Yet another definition of tensegrity comes from [3], whose definition connects the rigidity of tensegrity structures with the state of self-stress within:

Tensegrity systems are systems whose rigidity is the result of a state of self-stressed equilibrium between cables under tension and compression elements.

All of these definitions capture the major feature of tensegrity structures: struts do not touch each other. Since cables can withstand tension, an art form quickly arose in which struts appear to be levitated above ground (Figure 1) embedded in a network of cables. This picture is the predominant image in tensegrity literature, and the foundation for most of the structural and control research, but the definition allows a much broader interpretation. The most common deviation from the disconnected strut model is the appearance of Class 2 tensegrity structures, in which multiple struts are allowed to touch, but tensioned elements are still a requirement for stability. But tensegrity can be further generalized. Replace rod-like and cable-like elements with soft surfaces that resist tension and compression, and the result is a soft robot that does not resemble the canonical art form. Replace those elements with microfilaments and microtubules, and the result is a cell [4]. Because tensegrity forms

are defined by the internal forces required to remain in form, any system whose structure embodies that requirement can be considered a tensegrity, making it far more widespread than its reputation of being a niche art and architectural form.

Interest in tensegrity outside of art and architecture exploded after Donald Ingber discovered that cellular cytoskeletons can be modeled exceptionally well with tensegrity structures, where tensional forces are carried by cytoskeletal microfilaments and intermediate filaments, balanced by interconnected compressional elements like internal microtubule struts and extracellular matrix adhesions [4–13]. Tensegrity has also been proposed as a model of animal movement [14,15], where tensioned members in the musculo-skeletal system store and distribute energy. Tensegrity, and soft robotics in general, may also be useful models of hyperredundant animal physiology. Consider a grasshopper: it has 6 degrees of freedom, but uses 270 different control muscles to be more energy efficient and robust to uncertainty [16]. Another example of a finely-controlled complex biological structure is the caterpillar: despite a small brain and limited control resources, they are still able to coordinate hundreds of muscles to perform highly complex tasks [17].

For in-depth discussions of the history of tensegrity structures, their benefits, descriptions of canonical physical forms, and the related mathematics, see [18–21].



Figure 1: From [22], the B-Tree sculpture by Kenneth Snelson, National Institutes of Health, Bethesda, MD.

1.1 Outline

This survey will first make an argument for merging the fields tensegrity robotics and soft robotics. Then, this survey will describe a mathematical model of tensegrity robotics, based on the Euler-Lagrange equations of motion, suitable for robotics. Then the methods for finding equilibrium configurations, known as form-finding, will be discussed. Following this, the approaches used in both perturbation control and path and shape change control will be discussed. Current applications of tensegrity structures will be enumerated. Finally, directions for future work will be suggested.

1.2 Tensegrity Structures in Soft Robotics

Soft robotics is a growing field that is becoming more popular as computational and manufacturing abilities have grown to facilitate their study. Soft robots are compliant, have dexterous mobility allowing them to work in unstructured and cluttered environments, and can easily deform, all of which allow them to safely interact with people and fragile objects [23]. Whereas traditional robots rely on rigid components and typically have only as many degrees of freedom as they require for their workspaces, soft robots have continuous deformation and therefore have infinite degrees of freedom. While this may appear to make the task of motion planning and manipulation intractable, its appearances in nature are ubiquitous. Tongues, elephant trunks, octopus arms, caterpillars, and slugs are just a few examples.

Possibly due to its reputation as a niche art form, the tensegrity robotics research community barely overlaps the soft robotics field as a whole, despite the very similar goals of recent tensegrity roboticists and soft roboticists. The soft robotics survey [23] does not include a single mention of tensegrity, and yet, modeling and control methods employed in tensegrity robotics research may be directly applied to the general problems of soft robotics.

Soft robots have few or no rigid members [23], so at first glance it is unclear how tensegrity robots can be considered soft. However, the lack of distinct rigid members does not mean there is no compression in a soft robot; a system entirely in tension would collapse. Because of this, any computational model of a soft robot must be an implicit tensegrity model, even if the distinction between cables and struts is not obvious. For example, an inflated balloon can be considered a tensegrity structure, where the surface is fully tensioned, but the compression force from the air prevents collapse. This means that robots composed of pneumatic cells [24] are, in the broad definition, tensegrity robots.

One advantage of tensegrity robots is that they can be modeled as a system of ordinary differential equations, without joint constraints, instead of requiring partial differential equations and their associated solution methods (ie. finite element method), which can require long computation times. A major challenge in soft robotics is the modeling, control, and path-planning of continuous deformable robots [23]. However, if these models can be transformed into general tensegrity models, suitable control schemes may arise. Much work has been done in controlling tensegrity robots using various techniques (Section 3.3); these methods may be applicable to soft robots in general. Casting a soft robotics problem as a tensegrity problem may not only open up research time for consideration of advanced control methods, but may allow such simulations to occur in real time onboard a soft robot, giving the robot the ability to plan its own path through state space.

The high degree of coupling in soft robotics leads to hyperredundancy and the capacity for communication between members, allowing for a distributed control system (which is itself another inspiration from the world of biology). These things are also true for tensegrity structures. Tensegrity elements may simultaneously actuate, sense, and provide feedback for control [25,26]. This notion of mechanical elements being able to sense, communicate, and provide feedback is known in the literature as *morphological computation* [26,27], and such structures are known as *smart structures* [28]. Harnessing these capabilities allows for the reduction in the number of actuators needed to perform movement and other tasks. For example, the structure shown in Figure 12 has 24 degrees of freedom and can be controlled by just 2 two-state actuators [27]. Another benefit of this dynamic coupling and actuation is that tensegrity robots could sustain multiple internal failures and still operate as intended. This gives tensegrity soft robots the desirable property of robustness.

1.3 Features of Tensegrity Robots

The favorable properties of tensegrity structures have fostered an increasing interest in using tensegrity structures in a wide range of applications. In particular, recent discoveries in biology have helped to spur a shift from the exploration of static tensegrity structures to its possible uses in robotic applications. These favorable properties, some of which are summarized in [19,29], are as follows:

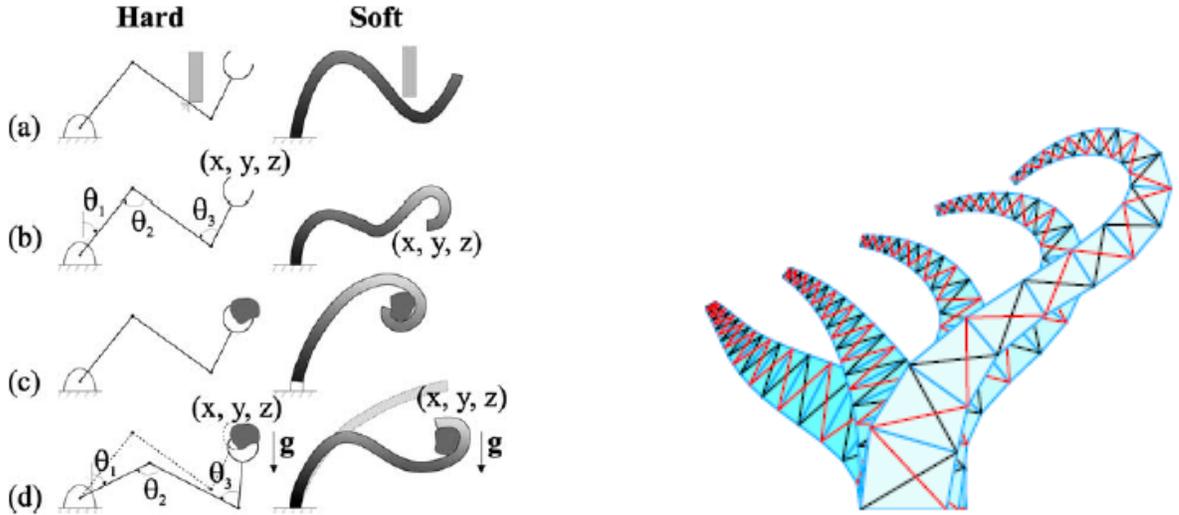


Figure 2: Comparison of hard and soft robotics, with a soft tensegrity robot analogue. Left, from [23]: Capabilities of hard and soft robots: (a) dexterity, (b) position sensing, (c) manipulation and (d) loading. Right, from [16]: Movement sequence of a planar tensegrity robot with fixed-length struts (cross-diagonal lines) and adjustable control tendons (all others).

Reconfigurability: Tensegrity structures are malleable and can change their structure to accomplish unusual tasks not allowed in the conventional robot [16]. Figure 2 shows a possible reconfiguration sequence for a planar tensegrity robot. This lends itself to unconventional locomotion principles [30]. Due to the property that a localized force on a tensegrity structure yields a global change of shape, independent from the position of the actuator, locomotion over rough terrain is possible with few actuators [30]. Tensegrity structures may also be collapsed, which has opened it up for orbital deployment applications, such as antennae and masts [31,32], where compact storage is the driving motivation for space structure design. For robots, this property makes tensegrity structures ideal for maneuvering around tight places, or deforming under large loads instead of failing.

Efficiency: The control of tensegrity structures requires less energy because some of the energy is stored in the structure itself [15]. Actuation releases some of the potential energy, which can be harnessed, and most of which is recaptured instead of being lost to friction. After movement, the pre-stress energy will still remain in the structure. Actuation of one part of a tensegrity structure affects every other member, allowing for fewer actuators [27].

Failure tolerance: As redundant, dynamically coupled structures, tensegrity robots can perform tasks reliably even when one or more elements fail [27, 33–36]. This is in contrast to conventional robots, in which the failure of a single element may lead to global failure.

Simplicity of design: Highly maneuverable, scalable, and redundant tensegrity robots can be composed of a minimum of two kinds of components: passive/actuated struts and passive/actuated cables. This eliminates the need to design each individual component separately. Many prototypical models choose one element type to be active while leaving the other passive. Other designs may consider a combination of passive and actuated elements of the same type in a structure.

Ease of modeling: Elements experience only axial forces; not bending moments or shear forces, which makes the computational modeling simpler to devise. Classical flexible structures require partial differential equations, which are difficult both analytically and numerically. Tensegrity models, which only need to consider axial forces, can be solved with ordinary differential equations [37].

2 Tensegrity Model

To create a computational model of a tensegrity robot, the mathematical models of the statics and dynamics must first be described. Not every combination of struts and cables yields a stable tensegrity structure, and indeed, finding new forms is nontrivial. This is known as the *form-finding* problem and has been the subject of extensive research, much of which has been surveyed by [29, 38, 39]. The form-finding process is an integral part of the control problem; most control problems are posed as a transformation from one equilibrium form to another, usually following a path made of equilibria.

Before describing the form-finding and control problems, this survey will define the mathematical models of tensegrity forms, in the contexts of both *force equilibrium* and *minimum potential energy*. Following this, the two most common dynamical models will be described: one in which tensegrity structures are composed of *point masses* at the nodes connected by springs, and one in which struts are *rigid bodies* connected by springs.

The equations of motion are explicitly derived from Lagrangians in many papers. However, in many cases, an off-the-shelf physics engine is used. In the latter case, the features provided by the library of choice are used [27, 34–36], including rigid bodies, springs, gravity, collision forces, and friction forces. The benefit of using a physics engine is that it permits researchers to quickly assemble prototypes as a basis for complicated AI control systems. One weakness of this approach is that the real-time focus of a physics engine like Open Dynamics Engine may not be suitable for high fidelity simulations of Hamiltonian systems.

2.1 Definitions

While the definition proposed by Fuller is a wide umbrella, many researchers choose to model tensegrity structures as collections of one-dimensional spring-like (or rigid) objects. This allows only compression and tension forces to be considered, and in turn, allows the use of ODEs, as opposed to PDEs, which are required for structures in general.

Strut: A strut is a compressed element, usually modeled as a rod or a rigid body. Nearly all research groups model struts as straight rods that do not experience bending, shear, or torsional forces, only compression. The only exception to this is in [30], which uses curved struts, but the internal forces are not modeled.

Cable: A cable is a tensioned element, usually modeled as a string or as a spring. Cables are almost universally modeled as strings that experience only tension forces. [40] expands the cable concept to include tension surfaces, which are modeled in ODE as soft bodies: lattices of point masses connected by springs.

The topology of tensegrity structures can be expressed as a graph. Tensegrity structures can be classified into a set of edges, which represent the elements of the structure, and nodes, which are the locations at which elements meet. Let the *abstract tensegrity framework* [41] be a graph $G(V, E)$ where the vertices $V = \{1, \dots, N_v\}$ is the set of nodes, and the edges $E = \{1, \dots, N_e\}$ are the set of elements, where $E = C \cup S$ if only cables and struts are considered, and $E = C \cup S \cup B$ if a third type of element called a *bar*, which is allowed to experience both compression and tension, is considered. Cables require an upper bound on their length, struts require a lower bound, and bars have a fixed length. Alternately, an edge $e = \{u, v\}$ may be defined as a pair of vertices.

A *tensegrity framework* is a pair (G, \mathbf{p}) with a position function $\mathbf{p}_i : v \in V \rightarrow \mathbb{R}^d$ mapping the vertices to a d -dimensional position. The vector of all positions, $\mathbf{p} = (\mathbf{p}_1^T, \dots, \mathbf{p}_{N_v}^T)$, is the *embedding* of the topological structure into Euclidean space. The *rigidity matrix* [42] \mathbf{R} is an $N_e \times dN_v$ matrix that maps embeddings to edges (Equation 1). For all edges $e \in E = \{u, v\}$, each edge row has d columns for each of the N_v vertices. The *equilibrium matrix* is the transpose of the rigidity matrix [43]. Tensegrity structures have *multiple embeddings* if each embedding cannot be transformed to another by a combination of translations, rotations, and reflections. If these multiple embeddings can be reached through continuous deformations that maintain all edge lengths, it is said to be a *mechanism*. Further definitions that are relevant to static analysis can be found in [29, 39].

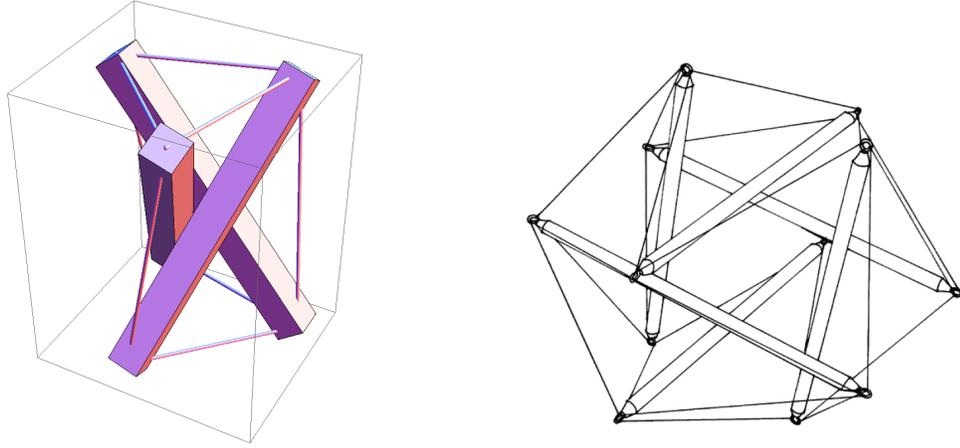


Figure 3: Left: the 3-prism: the simplest 3D tensegrity structure, with 3 struts (rectangular prisms) and 9 cables (narrow cylinders). Right: from [20], a tensegrity icosahedron with 6 struts and 24 cables.

$$\mathbf{R}_{ei} = \begin{cases} \mathbf{p}_u^T - \mathbf{p}_v^T & \text{If } i = u \\ \mathbf{p}_v^T - \mathbf{p}_u^T & \text{If } i = v \\ 0 & \text{Otherwise} \end{cases} \quad (1)$$

2.2 Forces in Equilibrium

In static equilibrium, the sum of the forces at each node of the structure must equal the external forces at each node, or zero if no external forces are acting on the structure:

$$\sum_{j=1}^{\|N_i\|} \mathbf{f}_{ij} = \mathbf{f}_i^{ext} \quad (2)$$

$$\mathbf{f}_{ij} = \begin{cases} f_{ij} \frac{\mathbf{p}_i - \mathbf{p}_j}{l_{ij0}} & \text{if } \{i, j\} \in E \\ 0 & \text{if } \{i, j\} \notin E \end{cases} \quad (3)$$

$$\begin{aligned} \forall \{i, j\} \in C, f_{ij} &\leq 0 \\ \forall \{i, j\} \in S, f_{ij} &\geq 0 \end{aligned} \quad (4)$$

Where l_{ij0} is the rest length of an element, and f_{ij} is the negative tension force if the element is a cable, and a positive compression force if the element is a strut. The convention here has tension forces pointing to the neighboring node, while compression forces point away from the neighboring node. If every equilibrium force is resolvable, this structure is *statically rigid* [41]. The above definition allows for forces to be modified, for example, by changing internal stress to compensate for external forces. This allows a structure to remain rigid under external forces. Alternately, f_{ij} can be defined more explicitly, such as by using Hooke's law for springs $F = -kx = -\frac{YA}{l_{ij0}}\Delta l$:

$$\mathbf{f}_{ij} = f_{ij} \frac{\mathbf{p}_i - \mathbf{p}_j}{l_{ij0}} = -\frac{Y_{ij}A_{ij}}{l_{ij0}} (\|\mathbf{p}_i - \mathbf{p}_j\| - l_{ij0}) \left(\frac{\mathbf{p}_i - \mathbf{p}_j}{\|\mathbf{p}_i - \mathbf{p}_j\|} \right) = -\frac{Y_{ij}A_{ij}}{l_{ij0}} \left(1 - \frac{l_{ij0}}{\|\mathbf{p}_i - \mathbf{p}_j\|} \right) (\mathbf{p}_i - \mathbf{p}_j) \quad (5)$$

Where Y_{ij} is the Young's modulus of the element, and A_{ij} is the cross sectional area of the element. With this model, if the Young's modulus, area, and rest length are fixed, structures will deform under

Static equilibrium definitions	
V	The set of nodes (joints)
E	The set of edges (struts or cables)
N_v	The number of nodes
N_e	The number of edges
d	The dimension (may be 2 or 3)
$C \in E$	The set of cables
$S \in E$	The set of struts
$B \in E$	The set of bars
$G(V, E)$	Abstract tensegrity framework
\mathbf{p}_i	Position of i th node
\mathbf{p}	Vector of positions
(G, \mathbf{p})	Tensegrity framework
\mathbf{R}	Rigidity matrix
\mathbf{R}^T	Equilibrium matrix
\mathbf{f}_{ij}	Force vector on node i from node j
f_{ij}	Magnitude of force on node i from node j
l_{ij0}	Rest length of edge between i, j , where $f_{ij} = 0$
\mathbf{f}_i^{ext}	External forces on node i
Y_{ij}	Young's modulus of edge between i, j
A_{ij}	Cross-sectional area of edge between i, j
q_{ij}	Force density: force per length of edge between i, j
\mathbf{q}	Stress vector of all force densities
V_{ij}	Potential energy of an edge i, j
$\mathbf{\Omega}$	Stress matrix
\mathbf{C}	Stiffness coefficient matrix
\mathbf{H}	Second derivative (Hessian) matrix

Table 1: Static equilibrium definitions.

external loads. The Hooke's law model of tensegrity structures is especially relevant to robotics. To enable movement, experiments modify only the rest lengths through actuation, and no other material properties of the structure are changed. Equation 5 is nonlinear, but [44] introduced a way to linearize the *force density* q_{ij} [44] of each element (force/length) into the system:

$$q_{ij} = \frac{f_{ij}}{l_{ij0}} \quad (6)$$

$$f_{ij} \frac{\mathbf{p}_i - \mathbf{p}_j}{l_{ij0}} = q_{ij}(\mathbf{p}_i - \mathbf{p}_j) \quad (7)$$

It is important to point out that equation 7 is only valid if the tendon rest length is 0, which does not agree with the Hookean model. The reason for this is that all q_{ij} are assumed to be constant, meaning the expression $q_{ij}(\mathbf{p}_i - \mathbf{p}_j) = 0$ only when $\mathbf{p}_i = \mathbf{p}_j$. Using the force density linearization, if we define the stress vector $\mathbf{q} \in \mathbb{R}^{N_e} = (\dots, q_{ij}, \dots)$ to be the vector of all force densities, then equation 2 can be rewritten as:

$$\mathbf{R}^T \mathbf{q} = \mathbf{f}^{ext} \quad (8)$$

This results in a linear system that uses the equilibrium matrix [44] and can be solved with linear algebra methods. For an in-depth analysis of forces in equilibrium, see [29].

Point mass dynamic model definitions	
m_i	Mass of node i
\mathbf{p}_i	Position of node i
f_{ij}^d	Spring damping force magnitude on node i from node j
c_{ij}	Coefficient of damping on edge i, j
Rigid body dynamic model definitions	
i	i th rigid body
ip	Node, identified by the i th rigid body, and $p \in \{1, 2\}$ is the location (1=top,2=bottom)
$\tilde{\mathbf{a}}$	Cross product matrix for vector \mathbf{a}
\mathbf{m}_i	Mass-inertia matrix
\mathbf{I}_i	Moment of inertia tensor
$\mathbf{I}_i^{\mathbf{b}}$	Moment of inertia tensor along principal axes
\mathbf{p}_i	Rigid body state vector $\{x_i, y_i, z_i, \psi_i, \theta_i, \phi_i\}$
\mathbf{x}_i	Position subset of rigid body state vector $\{x_i, y_i, z_i\}$
\mathbf{E}_i	Euler angle subset of rigid body state vector $\{\psi_i, \theta_i, \phi_i\}$
\mathbf{v}_i	Translational velocity $\{\dot{x}_i, \dot{y}_i, \dot{z}_i\}$
$\boldsymbol{\omega}_i$	Angular velocity $\{\omega_{ix}, \omega_{iy}, \omega_{iz}\}$
$\dot{\mathbf{E}}_i$	Euler angle derivative with respect to time $\{\dot{\psi}_i, \dot{\theta}_i, \dot{\phi}_i\}$
Γ_i	Matrix that transforms $\boldsymbol{\omega}_i$ to $\dot{\mathbf{E}}_i$
\mathbf{Q}_i	Rotation matrix for rigid body i
\mathbf{F}_i	Force vector for rigid body i
$\boldsymbol{\tau}_i$	Torque vector for rigid body i
C_T, C_R	Translational and rotational damping constant
h_i	Length of rigid body i
\mathbf{P}_{ip}	Position of node on rigid body i , location p
L_{ipjq}	Length of edge between node ip and node jq
L_{ipjq0}	Rest length of edge between node ip and node jq
\mathbf{F}_{ipjq}	Force acting on rigid body i at node ip from node jq
$\boldsymbol{\tau}_{ipjq}$	Torque acting on rigid body i at node ip from node jq

Table 2: Dynamics definitions.

2.3 Minimum Potential Energy

Tensegrity systems can also be modeled in terms of the potential energy stored within springs. A rigid tensegrity framework exists when the energy function is at a local minimum [42, 45]. Two conditions must be satisfied: the first derivative of the energy function must be 0 at the equilibrium point, and the second derivative must be positive at the equilibrium point. For a spring obeying Hooke's law, the potential energy of a spring is:

$$V_e = \frac{1}{2}kx^2 = \frac{YA}{2l_{e0}}\Delta l^2 \quad (9)$$

$$V_{ij} = \begin{cases} \frac{Y_{ij}A_{ij}}{2l_{ij0}}(\|\mathbf{p}_i - \mathbf{p}_j\| - l_{ij0})^2 & \text{if } \{i, j\} \in E \\ 0 & \text{if } \{i, j\} \notin E \end{cases} \quad (10)$$

$$V = \sum_{ij} \frac{Y_{ij}A_{ij}}{2l_{ij0}}(\|\mathbf{p}_i - \mathbf{p}_j\| - l_{ij0})^2 \quad (11)$$

The first derivative of the energy function must be 0 at the equilibrium point:

$$\frac{dV}{dt} = \sum_{ij} \frac{Y_{ij}A_{ij}(\|\mathbf{p}_i - \mathbf{p}_j\| - l_{ij0})}{l_{ij0}\|\mathbf{p}_i - \mathbf{p}_j\|}(\mathbf{p}_i - \mathbf{p}_j) \cdot (\dot{\mathbf{p}}_i - \dot{\mathbf{p}}_j) = 0 \quad (12)$$

The second derivative must be positive at the equilibrium point:

$$\begin{aligned} \frac{d^2V}{dt^2} = & \sum_{ij} \frac{Y_{ij}A_{ij}}{l_{ij0}} \left(\frac{((\mathbf{p}_i - \mathbf{p}_j) \cdot (\dot{\mathbf{p}}_i - \dot{\mathbf{p}}_j))^2}{\|\mathbf{p}_i - \mathbf{p}_j\|^2} \right. \\ & - \frac{(\|\mathbf{p}_i - \mathbf{p}_j\| - l_{ij0})(\mathbf{p}_i - \mathbf{p}_j) \cdot (\dot{\mathbf{p}}_i - \dot{\mathbf{p}}_j)}{\|\mathbf{p}_i - \mathbf{p}_j\|^3} \\ & \left. + \frac{(\|\mathbf{p}_i - \mathbf{p}_j\| - l_{ij0})\|\dot{\mathbf{p}}_i - \dot{\mathbf{p}}_j\|^2((\mathbf{p}_i - \mathbf{p}_j) \cdot (\dot{\mathbf{p}}_i - \dot{\mathbf{p}}_j))}{\|\mathbf{p}_i - \mathbf{p}_j\|} \right) > 0 \end{aligned} \quad (13)$$

The equilibrium equations listed here are more thoroughly explored in two previously published surveys ([29, 38]).

Equations 9-13 are nonlinear and difficult to work with. Therefore, first appearing in [42], these are simplified following the same linearization reasoning that resulted in equation 8. First, Hooke's law can be replaced with an arbitrary energy function v_{ij} of the squared distance between the nodes:

$$V = \sum_{ij} v_{ij}(\|\mathbf{p}_i - \mathbf{p}_j\|^2) \quad (14)$$

$$\frac{dV}{dt} = \sum_{ij} 2(\mathbf{p}_i - \mathbf{p}_j) \cdot (\dot{\mathbf{p}}_i - \dot{\mathbf{p}}_j) v_{ij}'(\|\mathbf{p}_i - \mathbf{p}_j\|^2) = 0 \quad (15)$$

$$\begin{aligned} \frac{d^2V}{dt^2} = & \sum_{ij} 4((\mathbf{p}_i - \mathbf{p}_j) \cdot (\dot{\mathbf{p}}_i - \dot{\mathbf{p}}_j))^2 v_{ij}''(\|\mathbf{p}_i - \mathbf{p}_j\|^2) \\ & + 2\|\dot{\mathbf{p}}_i - \dot{\mathbf{p}}_j\|^2 (\mathbf{p}_i - \mathbf{p}_j) \cdot (\dot{\mathbf{p}}_i - \dot{\mathbf{p}}_j) v_{ij}'(\|\mathbf{p}_i - \mathbf{p}_j\|^2) \geq 0 \end{aligned} \quad (16)$$

The $\ddot{\mathbf{p}}_x$ terms can be left out [29, 42], reducing equation 16 to:

$$\begin{aligned} \frac{d^2V}{dt^2} = & \sum_{ij} 4((\mathbf{p}_i - \mathbf{p}_j) \cdot (\dot{\mathbf{p}}_i - \dot{\mathbf{p}}_j))^2 v_{ij}''(\|\mathbf{p}_i - \mathbf{p}_j\|^2) \\ & + 2\|\dot{\mathbf{p}}_i - \dot{\mathbf{p}}_j\|^2 v_{ij}'(\|\mathbf{p}_i - \mathbf{p}_j\|^2) \geq 0 \end{aligned} \quad (17)$$

The term $v_{ij}'(\|\mathbf{p}_i - \mathbf{p}_j\|^2)$ can be considered a force density coefficient q_{ij} that populates the stress vector \mathbf{q} . Thus, equation 15 can be stated:

$$\frac{dV}{dt} = 2\mathbf{q}\mathbf{R}\dot{\mathbf{p}} = 0 \quad (18)$$

To find a similar matrix equation for equation 17, a few new definitions must be made. Let $\mathbf{\Omega}$ be a *stress matrix* such that:

$$\mathbf{\Omega}_{ij} = \begin{cases} -\omega_{ij} & \text{if } i \neq j \\ \sum_k \omega_{ik} & \text{if } i = j \end{cases} \quad (19)$$

Now let the *stiffness coefficient* be $c_{ij} = v_{ij}''(\|\mathbf{p}_i - \mathbf{p}_j\|^2)$, and \mathbf{C} be the full matrix of c_{ij} , then equation 17 can be stated as a Hessian matrix \mathbf{H} of second derivatives:

$$\mathbf{H} = \dot{\mathbf{p}}^T [2\mathbf{\Omega} + 4\mathbf{R}^T \mathbf{C} \mathbf{R}] \dot{\mathbf{p}} \quad (20)$$

When \mathbf{H} is positive definite, the energy will be at a minimum, indicating an equilibrium state. If equations 18 and 20 can be solved, an equilibrium form will be found. Solutions are not trivial and straightforward, but several form-finding methods have been found that work reasonably well in many cases.

2.4 Point Mass Lagrangian Equations of Motion

In this model of tensegrity structures, each node is a point mass; struts and cables are modeled as springs. This model is developed thoroughly in [46]. This allows for a simplified model, one without rotational velocities or torques, and may only be a rough approximation. One benefit that the point mass model has over the rigid body model is that singularities arising from gimbal lock can be avoided [39]. The generalized coordinates \mathbf{q} can be represented by a vector $\mathbf{p}_i = \{p_{ix}, p_{iy}, p_{iz}\}$. For a point mass m_i , the kinetic energy is $\mathcal{T}_i = \frac{1}{2}m_i v_i^2$, the potential energy of a point mass due to gravity is

$\mathcal{V}_i = m_i g p_{iz}$, and the potential energy of a spring is $\mathcal{V}_{ij} = \frac{1}{2} k_{ij} \Delta l^2$. The dissipative forces may be defined many ways, including as damping on nodes only or damping on the springs, and friction may be included.

The Lagrangian of a system is the kinetic energy minus the potential energy:

$$\mathcal{L} = \mathcal{T} - \mathcal{V} \quad (21)$$

The Euler-Lagrange equations of motion, for a set of generalized coordinates $\mathbf{q} = \{q_1, q_2, \dots, q_n\}$, are:

$$\frac{d}{dt} \left(\frac{\delta \mathcal{L}}{\delta \dot{q}_i} \right) - \frac{\delta \mathcal{L}}{\delta q_i} = 0 \quad (22)$$

For a conservative nodal point mass tensegrity structure under gravity, the full Lagrangian \mathcal{L} of a system is:

$$\begin{aligned} \mathcal{L} = & \sum_i^{N_v} \frac{1}{2} m_i \dot{\mathbf{p}}_i \cdot \dot{\mathbf{p}}_i \\ & - \sum_i^{N_v} m_i g p_{iz} \\ & - \sum_{ij}^{N_e} \frac{1}{2} k_{ij} (\| \mathbf{p}_i - \mathbf{p}_j \| - l_{ij0})^2 \end{aligned} \quad (23)$$

Applying this system to equation 22 defines the *equations of motion* for a conservative nodal point mass tensegrity structure under gravity:

$$\begin{aligned} & \sum_i^{N_v} (m_i \ddot{\mathbf{p}}_i + m_i g \hat{\mathbf{z}}) \\ & + \sum_{ij}^{N_e} k_{ij} (\| \mathbf{p}_i - \mathbf{p}_j \| - l_{ij0}) \frac{\mathbf{p}_i - \mathbf{p}_j}{\| \mathbf{p}_i - \mathbf{p}_j \|} = \mathbf{0} \end{aligned} \quad (24)$$

Note that the spring forces in equation 24 are a restatement of the spring forces in equation 5. This system of equations can be inserted as-is into an integrator such as the Runge-Kutta 4th order solver. A fictional spring damping force may be applied, which can be stated as $F_d = -c \frac{dx}{dt}$. When recast as a function of $\| \mathbf{p}_i - \mathbf{p}_j \|$, the spring damping force becomes:

$$f_{ij}^d = -c_{ij} \frac{(\mathbf{p}_i - \mathbf{p}_j) \cdot (\dot{\mathbf{p}}_i - \dot{\mathbf{p}}_j)}{\| \mathbf{p}_i - \mathbf{p}_j \|} \quad (25)$$

Alternately, a point mass damping model can be applied:

$$f_i^d = -c_i \dot{\mathbf{p}}_i \quad (26)$$

Either way, the damping model serves only one purpose, and that is to allow tensegrity structures to return to a steady state after a perturbation. Thus, damping is a form of dynamic relaxation.

2.5 Rigid Body Lagrangian Equations of Motion

This model, explicitly defined in [25, 37, 39, 47] and implicit in [26, 27, 34–36], trades the coarse nodal point mass model for a model in which each strut is a *rigid body*, and each cable is modeled as a pair of equal but opposite forces acting at the tips of two rigid bodies. Rigid bodies are suitable for systems in which the struts deform so insignificantly as to be ignored, but they introduce additional issues. They are also more accurate because the centers of mass are located at the centers of struts, instead of at nodes. Variants of this model simplify the generalized coordinates wherever possible. A Newtonian model for tensegrity shells was converted to a non-linear system of equations where the first and second derivatives of strut lengths are constrained to be 0, making them rigid, but leaving out rotational inertia [48–50]. This model was expanded by [51–54] who added a nonlinear model of elasticity and the deformability of edges.

The model presented here will include no constraints: that is, every body is free to move and rotate along any axis. A rigid body has six position coordinates: three translational, and three angular. Euler angles provide a convenient minimized set of angular coordinates. The generalized coordinates for a

rigid body are its position and orientation in Euler angles $\mathbf{p}_i = (x_i, y_i, z_i, \psi_i, \theta_i, \phi_i)$. Since a body may rotate around the load axis ϕ , but none of the cable and strut forces act on that axis in most tensegrity models, ϕ_i is often dropped. However, a generalized strut (that is, something other than a rod) may require modeling of all states, so they will be kept here.

In the point mass notation, both nodes and point mass bodies could be represented by the same index i . Here, each rigid body is represented by an index i , and each rigid body has two nodes, which will be distinguished with an additional index p , so that a node is identified by the string ip , where i is the body, and $p \in \{1, 2\}$ is the specific location.

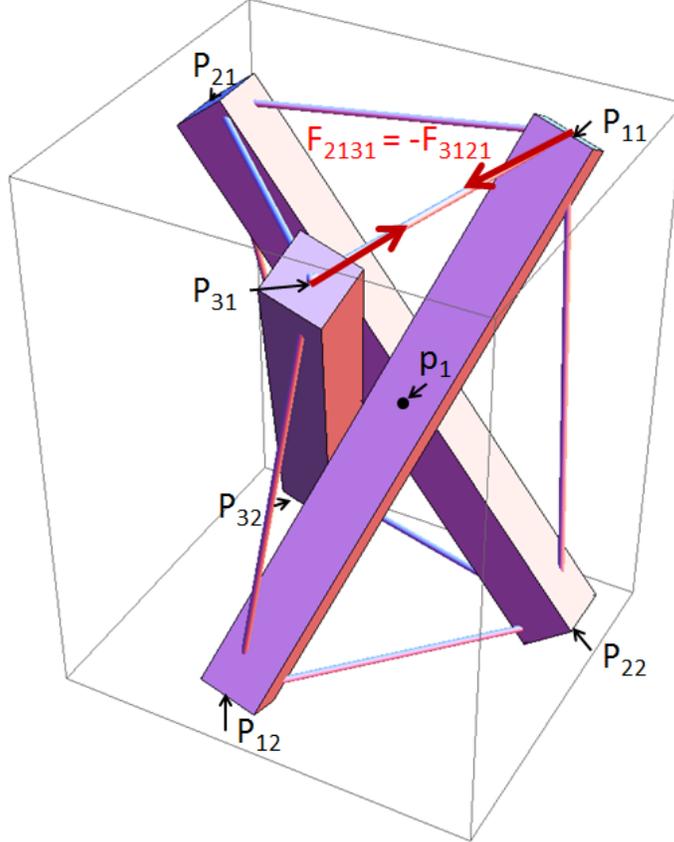


Figure 4: A labeled rigid body tensegrity 3-prism. All 6 nodes have positions labeled as \mathbf{P}_{ip} , where i is the identity of the body, and p identifies which of the two endpoints (top=1, bottom=2) is being considered. The center of body 1, \mathbf{p}_1 , is shown. The tension force vectors are shown in red, and are labeled \mathbf{F}_{ipjq} , where the first two digits identify the body and endpoint on which the force is acting, and the second two digits identify the body and endpoint on the opposite end.

A rigid body has a $2d \times 2d$ mass-inertia matrix \mathbf{m}_i :

$$\mathbf{m}_i = \begin{pmatrix} m_i & 0 & 0 & 0 & 0 & 0 \\ 0 & m_i & 0 & 0 & 0 & 0 \\ 0 & 0 & m_i & 0 & 0 & 0 \\ 0 & 0 & 0 & I_{ixx} & I_{ixy} & I_{ixz} \\ 0 & 0 & 0 & I_{iyx} & I_{iyy} & I_{iyz} \\ 0 & 0 & 0 & I_{izx} & I_{izy} & I_{izz} \end{pmatrix} \quad (27)$$

The moment of inertia tensor \mathbf{I}_i and its transformation along principal axes $\mathbf{I}_i^{\mathbf{p}}$ are:

$$\mathbf{I}_i = \begin{pmatrix} I_{ixx} & I_{ixy} & I_{ixz} \\ I_{iyx} & I_{iyy} & I_{iyz} \\ I_{izx} & I_{izy} & I_{izz} \end{pmatrix} \quad (28)$$

$$\mathbf{I}_i^{\mathbf{p}} = \begin{pmatrix} I_{i1} & 0 & 0 \\ 0 & I_{i2} & 0 \\ 0 & 0 & I_{i3} \end{pmatrix} \quad (29)$$

Before deriving the equations of motion, some notation needs to be explained. The cross product operation $\mathbf{a} \times \mathbf{b}$ can be rewritten as a matrix multiplication with a skew-symmetric matrix, becoming $\tilde{\mathbf{a}}\mathbf{b}$, where the tilde transforms a vector into a skew-symmetric matrix:

$$\widetilde{\{x, y, z\}} = \begin{pmatrix} 0 & -z & y \\ z & 0 & -x \\ -y & x & 0 \end{pmatrix} \quad (30)$$

For simplicity in understanding the subsequent equations, let \mathbf{x}_i and \mathbf{E}_i be the position and angle subsets of \mathbf{p}_i :

$$\mathbf{x}_i = \{x_i, y_i, z_i\} \quad (31)$$

$$\mathbf{E}_i = \{\psi_i, \theta_i, \phi_i\} \quad (32)$$

The time derivatives of position and angle are \mathbf{v}_i and $\boldsymbol{\omega}_i$, where the components of $\boldsymbol{\omega}_i$ are the angular velocity components around the x -axis, y -axis, and z -axis.

$$\mathbf{v}_i = \{\dot{x}_i, \dot{y}_i, \dot{z}_i\} \quad (33)$$

$$\boldsymbol{\omega}_i = \{\omega_{ix}, \omega_{iy}, \omega_{iz}\} \quad (34)$$

Since the angular velocity vector is defined in the orthonormal coordinates of the system, the derivative of the Euler angles must be found by a transformation with the $\boldsymbol{\Gamma}_i$ matrix:

$$\boldsymbol{\Gamma}_i = \begin{pmatrix} 1 & \sin(\psi_i) \tan(\theta_i) & -\cos(\psi_i) \tan(\theta_i) \\ 0 & \cos(\psi_i) & \sin(\psi_i) \\ 0 & -\sin(\psi_i) \sec(\theta_i) & \cos(\psi_i) \sec(\theta_i) \end{pmatrix} \quad (35)$$

$$\dot{\mathbf{E}}_i = \boldsymbol{\Gamma}_i \boldsymbol{\omega}_i \quad (36)$$

Let \mathbf{Q}_i be the rotation matrix associated with the rigid body i , in which the transformations happen in $X_1Y_2Z_3$ order:

$$\begin{aligned} \mathbf{Q}_i &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\psi_i) & -\sin(\psi_i) \\ 0 & \sin(\psi_i) & \cos(\psi_i) \end{pmatrix} \begin{pmatrix} \cos(\theta_i) & 0 & \sin(\theta_i) \\ 0 & 1 & 0 \\ -\sin(\theta_i) & 0 & \cos(\theta_i) \end{pmatrix} \begin{pmatrix} \cos(\phi_i) & -\sin(\phi_i) & 0 \\ \sin(\phi_i) & \cos(\phi_i) & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (37) \\ &= \begin{pmatrix} \cos(\theta_i) \cos(\phi_i) & -\cos(\theta_i) \sin(\phi_i) & \sin(\theta_i) \\ \cos(\psi_i) \sin(\phi_i) + \cos(\phi_i) \sin(\theta_i) \sin(\psi_i) & \cos(\phi_i) \cos(\psi_i) - \sin(\theta_i) \sin(\phi_i) \sin(\psi_i) & -\cos(\theta_i) \sin(\psi_i) \\ \sin(\phi_i) \sin(\psi_i) - \cos(\phi_i) \cos(\psi_i) \sin(\theta_i) & \cos(\psi_i) \sin(\theta_i) \sin(\phi_i) + \cos(\phi_i) \sin(\psi_i) & \cos(\theta_i) \cos(\psi_i) \end{pmatrix} \quad (38) \end{aligned}$$

The presence of cables attached to each end, and not to the center of mass of the rigid body, requires the definition of the endpoints of each strut. The forces from tension will act on the endpoints. Let \mathbf{P}_{ip} be the position of an endpoint, and let h_i be the length of the strut. The variable p is an index: $p = 1$ refers to the *top*, and $p = 2$ refers to the *bottom* when the Euler angles are $\mathbf{E} = \{0, 0, 0\}$:

$$\mathbf{P}_{ip} = \mathbf{Q}_i \left(\begin{array}{c} 0 \\ 0 \\ \left\{ \begin{array}{l} \frac{h_i}{2} \quad \text{if } p = 1 \\ -\frac{h_i}{2} \quad \text{if } p = 2 \end{array} \right. \end{array} \right) + \begin{pmatrix} x_i \\ y_i \\ z_i \end{pmatrix} \quad (39)$$

$$\mathbf{P}_{i1} = \begin{pmatrix} \frac{1}{2}h_i \sin(\theta_i) + x_i \\ y_i - \frac{1}{2}h_i \cos(\theta_i) \sin(\psi_i) \\ \frac{1}{2}h_i \cos(\theta_i) \cos(\psi_i) + z_i \end{pmatrix} \quad (40)$$

$$\mathbf{P}_{i2} = \begin{pmatrix} x_i - \frac{1}{2}h_i \sin(\theta_i) \\ \frac{1}{2}h_i \cos(\theta_i) \sin(\psi_i) + y_i \\ z_i - \frac{1}{2}h_i \cos(\theta_i) \cos(\psi_i) \end{pmatrix} \quad (41)$$

Let i, p be the body and endpoint indices for the body on which a force is acting, and j, q be the indices for another body whose forces are acting on body i . The length of a cable is represented by:

$$L_{ipjq} = \|\mathbf{P}_{ip} - \mathbf{P}_{jq}\| \quad (42)$$

The rest length is represented by:

$$L_{ipjq0} \quad (43)$$

With all of these definitions, the Lagrangian can be stated. The kinetic energy term is:

$$\mathcal{T} = \sum_i \frac{1}{2} \mathbf{v}_i^T m_i \mathbf{v}_i + \sum_i \frac{1}{2} \boldsymbol{\omega}_i^T \mathbf{I}_i \boldsymbol{\omega}_i \quad (44)$$

The potential energy term is the gravitational energy plus the stored energy of the springs:

$$\mathcal{V} = \sum_{ij} \begin{cases} \frac{Y_{ij} A_{ij}}{2L_{ipjq0}} (L_{ipjq} - L_{ipjq0})^2 & \text{if } L_{ipjq} > L_{ipjq0}, \text{ and cable exists between } \mathbf{P}_{ip}, \mathbf{P}_{jq} \\ 0 & \text{otherwise} \end{cases} \quad (45)$$

The full derivation is beyond the scope of this document, but it results in the Newton Euler equations. Euler's equations of motion for a rigid body under torque are:

$$\mathbf{I}_i \dot{\boldsymbol{\omega}}_i + \boldsymbol{\omega}_i \times (\mathbf{I}_i \boldsymbol{\omega}_i) = \boldsymbol{\tau}_i \quad (46)$$

Solving for $\dot{\boldsymbol{\omega}}_i$:

$$\dot{\boldsymbol{\omega}}_i = \mathbf{I}_i^{-1} (\boldsymbol{\tau}_i - \boldsymbol{\omega}_i \times (\mathbf{I}_i \boldsymbol{\omega}_i)) = \mathbf{I}_i^{-1} (\boldsymbol{\tau}_i - \tilde{\boldsymbol{\omega}}_i \mathbf{I}_i \boldsymbol{\omega}_i) \quad (47)$$

The transformations of the inertia tensor (and its inverse) from principal axes to system axes are:

$$\mathbf{I}_i = \mathbf{Q}_i \mathbf{I}_i^b \mathbf{Q}_i^T \quad (48)$$

$$\mathbf{I}_i^{-1} = \mathbf{Q}_i (\mathbf{I}_i^b)^{-1} \mathbf{Q}_i^T \quad (49)$$

From equations 31-49, the system of ODEs for body i with respect to the derivatives of the state vector \mathbf{p}_i can be found as a function of the principal inertia tensor parameter \mathbf{I}_i^b , the mass parameter m_i , the force vector \mathbf{F}_i , the torque vector $\boldsymbol{\tau}_i$, (and optional translation damping term C_T and rotation damping term C_R):

$$\dot{\mathbf{x}}_i = \mathbf{v}_i \quad (50)$$

$$\dot{\mathbf{v}}_i = \frac{1}{m_i} \mathbf{F}_i - C_T \mathbf{v} \quad (51)$$

$$\dot{\mathbf{E}}_i = \mathbf{\Gamma}_i \boldsymbol{\omega}_i \quad (52)$$

$$\dot{\boldsymbol{\omega}}_i = \mathbf{Q}_i(\mathbf{I}_i^b)^{-1} \mathbf{Q}_i^T (\boldsymbol{\tau}_i - \tilde{\boldsymbol{\omega}}_i \mathbf{Q}_i \mathbf{I}_i^b \mathbf{Q}_i^T \boldsymbol{\omega}_i) - C_R \boldsymbol{\omega}_i \quad (53)$$

Equations 50-53 are general solutions for any rigid body. Unfortunately, this derivation has a singularity at $\theta_i = \frac{\pi}{2}$, causing $\mathbf{\Gamma}_i$ to enter gimbal lock. There are a few ways of dealing with this: one is to use quaternions instead of Euler angles, but this adds a fourth variable to the angle specification, and unit constraint to be satisfied for the ODE solver; another way is to integrate the rotation matrix \mathbf{Q}_i directly, but this adds six additional variables and another unit constraint. For this model, $\theta_i = \frac{\pi}{2}$ corresponds to a strut being parallel to the XY plane, which, in practice, seldom occurs due to the low probability that for any time step in the integration routine, θ_i will be close enough to $\frac{\pi}{2}$ to trigger the singularity.

The force on body i from the cable spanning $\mathbf{P}_{ip}, \mathbf{P}_{jq}$ is:

$$\mathbf{F}_{ipjq} = \begin{cases} \frac{Y_{ij} A_{ij}}{L_{ipjq0}} \left(1 - \frac{L_{ipjq}}{L_{ipjq0}}\right) (\mathbf{P}_{jq} - \mathbf{P}_{ip}) & \text{if } L_{ipjq} > L_{ipjq0}, \text{ and cable exists between } \mathbf{P}_{ip}, \mathbf{P}_{jq} \\ 0 & \text{otherwise} \end{cases} \quad (54)$$

The torque on body i from the cable spanning $\mathbf{P}_{ip}, \mathbf{P}_{jq}$ is:

$$\boldsymbol{\tau}_{ipjq} = \begin{cases} (\mathbf{P}_{ip} - \mathbf{x}_i)^T \times \mathbf{F}_{ipjq} & \text{if } L_{ipjq} > L_{ipjq0}, \text{ and cable exists between } \mathbf{P}_{ip}, \mathbf{P}_{jq} \\ 0 & \text{otherwise} \end{cases} \quad (55)$$

The force terms \mathbf{F}_{ipjq} and torque terms $\boldsymbol{\tau}_{ipjq}$, alone, represent an uncontrolled tensegrity structure with no external forces. With positive C_T, C_R , any initial unstable configuration will eventually settle into a local minimal energy state as $t \rightarrow \infty$. This is a quick and useful way to perform the task of form-finding without implementing a specific algorithm, and closely resembles dynamic relaxation. While it does not provide an exact answer since convergence only comes at $t = \infty$, the calculated equilibrium structures can be arbitrarily close. It is useful for verifying tensile integrity: every cable should be longer than its rest length as $t \rightarrow \infty$; a valid tensegrity does not exist if at least one cable is slack.

2.6 Constraints

There are two ways of handling constraints: either including the constraints directly in the Lagrangian of the system with Lagrange multipliers [46], or creating penalty force pairs [55] that pushes objects in the desired direction without changing the total energy of the system. Such constraints include ground contact friction forces [26, 27], collision rebound forces [26, 27], and surface contact forces [55]. Nodes can be fixed in place by forcing $\mathbf{p}_i = \dot{\mathbf{p}}_i = \mathbf{0}$. This is a part of the model described in [16, 46, 56], and Lagrange multipliers are used to embed the constraints into the equations of motion. Instead of spring models, some researchers [48] employ length change constraints: $\frac{d\|\mathbf{P}_i - \mathbf{P}_j\|}{dt} = \frac{d^2\|\mathbf{P}_i - \mathbf{P}_j\|}{dt^2} = \mathbf{0}$

Rigid body physics libraries like Open Dynamics Engine allow for built-in collision detection between rigid bodies, which, like the forces derived in [55], applies elastic penalty forces to separate clipping objects. This approach is favored by [26, 27, 34–36], who simply incorporate ODE's built in collision system into their models.

3 Control

There are multiple ways of controlling a tensegrity robot, such as applying forces to nodes externally [57, 58], but the most frequently used control inputs are modifications to the rest lengths. This makes the edge rest lengths a function of time:

$$\begin{aligned} L_{ij0} &\rightarrow L_{ij0}(t) \\ L_{ipqj0} &\rightarrow L_{ipqj0}(t) \end{aligned} \quad (56)$$

Standard rigid robot designs have a number of actuators equal to the dimension of its workspace. For robots that are required to place an end effector at some location regardless of orientation, 3 actuators are needed. If orientation is considered, this number becomes 6. Additional degrees of freedom give robots redundancy, but also give inverse kinematics solvers multiple solutions. Paradoxically, tensegrity robots can be considered both under-actuated [34] and hyper-actuated [16], depending on the context. Underactuated tensegrity robots usually place actuators on a small number of edges not proportional to the number of edges in the structure. The property that tensegrity structures deform globally under a single force makes finding the workspace for a single actuator complicated, but may reveal potential control strategies would normally require multiple actuators on rigid robots. On the other hand, if the number of actuated edges is proportional to the number of edges, the system becomes hyperredundant. This means that methods that search for forms that place some reference point at a specific location may find multiple solutions, and multiple paths [16]. Finding such forms, the paths between them, and methods to reduce unwanted vibrations and errors, constitute the problem of control.

This section will first review the known methods for form-finding, which is important for finding specific locations for reference points and along paths. Then methods for controlling perturbations in the structure in the neighborhood of equilibrium forms will be discussed. Finally, this survey will document methods for changing the shape of a structure, and finding valid paths between configurations.

3.1 Form Finding

Form finding finds solutions to the force equilibrium and minimal energy problems. The task of identifying the form taken by a tensegrity structure is known as *form-finding*. As described in [59], and classified by [29,38], the approaches to this task fall under two major categories:

Static: these attempt to find solutions without iterating edge lengths. These also work well for simple and symmetric cases, but are intractable in the general case.

Kinematic: these methods consider modifying the lengths of either struts or cables until an equilibrium is found. They work well for simple and symmetric structures, but become intractable for the general case.

3.1.1 Static Form Finding

Analytical solutions: This is analogous to the kinematical method, and characterizes node relationships as explicit equations that must be solved [60].

Force-density method: Defined in equations 6-8, this is further elaborated in [38], which concludes that it is an excellent method for finding new tensegrity structures, especially when lengths are not known at the start, but not ideal for structures in which some element lengths must be fixed a priori. This method has been modified to include shape-based constraints [61]. The force-density method is the basis for the iterative algorithm presented in [62], which begins with the topology and iteratively calculates the force densities and geometry by using rank constraints on the stress matrix and rigidity matrix. Form finding is further explored in [63,64] with a numerical approach.

Energy method: Also assuming a spring model, this method tries to find local potential energy minima; [45]. This method is closely related to the force-density method. “Super stable” structures are defined using the energy method [65].

Reduced coordinates: This method, first introduced by [66], first identifies a set of generalized coordinates and equations for the struts, and then uses symbolic manipulation software like Mathematica to convert these equations into cable lengths. Structural symmetry is used to reduce the number of unknowns. This is tractable only when the structure is simple and symmetric.

Axial forces: This method first finds a set of axial forces that fit the topology of the structure, and then uses that information to find the node coordinates through successive approximation [67].

3.1.2 Kinematic Form Finding

Analytical solutions: This is for structures simple enough that the specific translational and angular relations can be concisely described in equations and first derivatives. If struts are varied, equilibrium is found at the conditions that maximize strut length. Likewise, if cables are varied, equilibrium occurs at minimal cable length. It is assumed that struts and cables, while variable overall, are all the same length within each category. This method is suitable to symmetrical structures [60].

Marching form-finding: This was introduced by [68] as an algorithm that begins with a stable placement of a given topology, and modifies it by solving a system of differential equations to change the lengths of members.

Nonlinear programming: This generalizes the analytical solution method by turning it into a constraint satisfaction problem [69]. In this case, maximizing the lengths of the struts is the objective function, and the constraints keep the lengths of cables constant, and ensure that all struts are the same. A function like `fmincon` in Matlab can be used to solve the problem. An advantage is that this allows software packages to be used, but as the structure grows, the constraint problem grows significantly.

Dynamic relaxation: First introduced into tensegrity analysis by [70]. Dynamic relaxation is a variant of straightforward integration of the systems of motion, in which, starting from a “guess”, forces at the nodes are calculated, and the nodes are accelerated in the direction of forces. This process iterates until forces at each node are in equilibrium. This is modified by [71] to include continuous cables that slide over pulleys. [70] concludes this method does not have good convergence properties for structures with large numbers of nodes. In [72], dynamic relaxation is combined with artificial neural networks to improve accuracy.

Genetic algorithms: Genetic algorithms may be used to evolve topologies by encoding the connectivity graph as a string that is subject to crossover and mutation. This approach has been used to find irregular and asymmetric structures [73–75], and shows a lot of promise for finding new categories of tensegrity structures, instead of building on forms found through intuition. Graph grammar descriptions of tensegrity connectivity, combined with genetic algorithms, were proposed by [75] and modified by [74] to generate very large irregular structures, and both claim that their methods produce much larger novel tensegrities than other literature. Genetic algorithms are further explored by [76], whose model is a slight variant, but with similar results.

Sequential quadratic programming: for additional material constraints such as buckling and yield stress, a quadratic penalty method may be applied to the search algorithm [77]. However, this may not always converge to an optimal solution.

Finite element method: closely related to dynamic relaxation, a pair of iterative algorithms converges a structure based on minimizing both displacements and energy [78].

3.2 Perturbation Control

Perturbation control, that is, the reduction of unwanted vibrations, has been a key area of research for static tensegrity structures. However, in the context of robotics, these studies are also useful, especially in systems in which undamped spring-like elements may continue to vibrate even after a desired control is performed. From a structural standpoint, controlling unwanted vibrations in the structure is important for ensuring long-term stability. Nearly all of the literature about perturbation control is active; only [48] examines passive control. However, active control requires little control energy [19], therefore there has been no pressing need to examine passive control.

Both the point-mass formulation and the rigid body formulation are high-dimensional nonlinear systems, and are therefore hard control problems. However, near the equilibrium points, these systems can be linearized as a function of the Jacobian matrix. This allows the system to be controlled using standard linear control schemes, such as PID control, optimal control with linear quadratic Gaussian control and Model Predictive Control, robust control using \mathcal{H}_2 and \mathcal{H}_∞ controllers, and intelligent controls. Feedback linearization can be used instead of full linearization. Linearization works well for

actively controlling unwanted vibrations, and can provide an approximation for the deformation of a tensegrity structure under a force perturbation [79]. Of all of the following methods, only [58, 80] actually tested these models in real structures [39].

Classical: The first studies on the perturbation of tensegrity structures was performed by [81, 82] whose model determined the transfer function between an input force and the oscillations around a linearized model of the tensegrity structure. In [80], two methods are used. One is called *local integral force feedback*, which is used in each actuated member to dampen its vibrations. This method only controls edges individually using only that edge’s tension as an input. The other method is called *acceleration feedback control*, builds upon a global information model found in [83]; this system performs a more effective dampening on the structure as a whole, but is harder to implement. [19, 37, 84, 85] also defined a simple prototype for an active control system based around linearization around the equilibrium. A two-part control law was devised by [57, 58] for tensegrity planar structures. A PI controller is used for filtering out high-frequency displacements. The other part is a robust control method, \mathcal{H}_∞ , accounts for errors in the dynamical system.

Feedback Linearization: Feedback linearization is used by [86, 87] to minimize tracking error while a tensegrity manipulator robot is undergoing a transformation along a path. It is successful at preventing forces within the cables from reaching an upper limit during a shape reconfiguration.

Optimal: Optimal control schemes seek to minimize changes in the equilibrium lengths of edges [88]. In [89], structures undergoing large displacements over time can be controlled via instantaneous optimal control using a linearized version of the model at each step in the transformation. However, the error of the linear system is increased due to the extra force being applied by the actuated members. The point mass model is used in [90] to describe a structure on which a linear quadratic controller is applied to control the linearized dynamics around an equilibrium point. The linear quadratic controller was found to have great performance. The authors also show that even large deformations can be reasonably approximated by a linear model.

Robust: Robust control accounts for errors in both the system and in the measurement of the system. A robust control technique, \mathcal{H}_∞ , makes up the second part of the control law in [57, 58]. Perturbations are reduced during shape reconfiguration in [91] by using an \mathcal{H}_2 controller to remove unwanted vibrations. The \mathcal{H}_2 controller is sufficient in keeping the actual path close to the reference path.

Machine Learning and Search Methods: A different approach examines the control of structures in the dynamical state space. A machine learning framework is combined with simulated annealing [92, 93] and the Probabilistic Global Search Lausanne (PGSL) [94] method to improve the performance of tensegrity structures over time. With other tensegrity search methods, the possible lengths are discretized. The discretized search function is combined with dynamic relaxation to find equilibrium points. Since each edge has a finite number of possibilities, the entire space of searchable structures is exponential with the number of elements, necessitating the inclusion of simulated annealing and PGSL. In [93], this approach is tested with a structure under various loads. A physical tensegrity structure under loads is compared to the numerical method, with the prediction closely matching the test structure. This approach can be used in real-time in [94], which also includes a machine learning method that stores good control commands to further speed up the search. Further tweaks have been made to reduce the search space, as in [95, 96], who include artificial neural networks to compensate for additional effects, and utilize additional criteria such as maximum stiffness and minimum stress.

3.3 Path and Shape Control

Path and shape control is the principal concern of the study of soft robotics control systems, and therefore is examined in great detail in this survey. Path and shape control concerns the transformation of a tensegrity structure from one feasible configuration to another by finding a path that maintains the feasibility of the structure throughout the process. The path requirements range from finding any feasible path, to finding a path that minimizes some cost function. This is important for tensegrity robots, as shape control is expected to be a consistent requirement, as opposed to static tensegrity

structures and single deployment structures like space antennae [37]. This is complicated by the fact that tensegrity robots may have obstacles in their workspaces, and the transformation from physical space to parameter space is a hard problem. The set of feasible tensegrity structures in equilibrium is a lower dimension than the space in which it is embedded; therefore, standard path-planning techniques, such as [97, 98] are not practical [39].

There are two very different approaches for judging the quality of a proposed control scheme. In one camp, the path as a whole is considered as a continuous function on which various control methods, such as open-loop control, optimal control, and Lyapunov control, are applied. The major difficulty in this is that closed-form solutions are not possible to derive except in highly simplified models, and in general, control of nonlinear systems is difficult, leading to these studies using the simplest symmetrical tensegrity structures. Generally, 2D structures are used in the continuous control literature: the smaller dimension allows for easier searching through state space for candidate solutions to the various problems proposed in the literature. With symmetry, this can be reduced to a low-dimensional problem.

The other camp discretizes the generation of paths. At various points along the path, including the start and end, form-finding methods are used to converge to equilibrium points, and the remainder of the path is generated by simply actuating to get from one length to the next. Discrete problems are suitable for graph search methods and artificial intelligence techniques. They also are more commonly used for irregular and large tensegrity structures. Three-dimensional structures, where considered, are usually simple and used for proofs of concept [55], due to the computational complexity of simulating large tensegrity structures with large parameter dimensions. The intelligent control methods used in [27, 99] buck this trend by showing that large and irregular structures can be controlled if the control specifications are simple enough (such as translation along a path).

3.3.1 Continuous Path Control

When the shape change control problem is considered as a continuous function, standard control theory techniques, ranging from open loop to PID to optimal control, are considered. The structures considered in these problems are highly symmetric and simple, reducing the parameter dimension considerably. While the result presented here are sufficient, it remains to be seen if these techniques can be used in more complicated structures.

Reconfiguration to a specific state, no path optimization: Slowly varying transformations are considered as an open loop quasi-static path control [100–103]. Their solution is to use slowly varying functions that move a structure from one equilibrium configuration to another along a path. By moving slowly between equilibrium configurations, the problem of unwanted nonlinear dynamics is ignored. By using a highly symmetric tensegrity system, the path has a closed form solution. Figure 5 shows the deployment process.

A PID controller is proposed in [104]. Here, Cartesian coordinates of the workspace are mapped to the parameter space of the tensegrity robot, and a PID controller is used to move the robot along the path. The error of the PID controller fluctuates, due in part to the 2-actuator system controlling a 3-DOF system, and damping is insufficient, leading to the PID controller to being a suboptimal choice.

Control based around a differential algebraic equation model of tensegrity structures with constraints is considered in [105, 106]. Differential algebraic equations are used to avoid the singularities that may occur in the rigid body ODE system. In [105, 106], a simple constrained tensegrity structure, essentially a guyed mast with a single strut and three cables, all attached to the ground, is controlled with two variants of nonlinear Lyapunov control, where the goal is to move the tip of the strut from one position to another, and the error is the distance between the current and goal positions. In the first variant, the system dynamics themselves are controlled, and in the second, the system is projected to an orthogonal space. Both approaches show convergence to the goal with exponential decay. The contributions of these papers are significant in that a nonlinear continuous control scheme is demonstrated for a three-dimensional tensegrity robot, whereas in other work nonlinear controllers are implemented as either discrete time steps or discretized choices [25, 33, 46, 91, 102, 107], or continuous controllers are used on two-dimensional tensegrity robots [86, 87]. A very detailed description of the differential

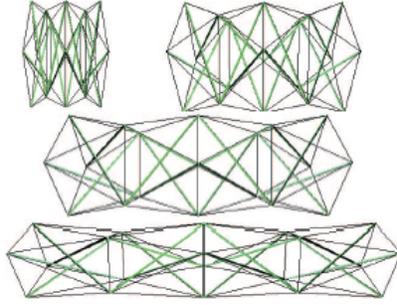


Figure 5: From [101]: The four configurations in the deployment of a compressed tensegrity structure into an extended one.

algebraic equation model with Lyapunov control can be found in [108].

Reconfiguration to a specific state, time optimization: Time-optimal control is considered in [16, 86, 87, 109, 110], where an optimized trajectory from position A to position B is found. The control law must obey an unsaturation requirement, where no cable is allowed to experience tension greater than the yield force; that is, $t_i < t_{MAX_i}$, and must also be robust to perturbations that might cause yield. Feedback linearization is used to control tracking error as the tensegrity robot moves along its path.

Reconfiguration to a specific state, time and average allowable force optimization: In [16], a hyper-actuated mechanical system (HAMS) is presented in which the number of control inputs exceeds the number of dimensions. One way of finding a unique path between configurations is by finding a time-energy optimal control law is derived from the center tracking control law. Here, *center tracking* refers to minimizing the distance between the cable tension and the central tension $c = \frac{t_{MIN_i} + t_{MAX_i}}{2}$, where a cable tension t_i must exist in the range $t_{MIN_i} \leq t_i \leq t_{MAX_i}$, and t_i must not go under the slack limit t_{MIN_i} and must not go over the breaking limit t_{MAX_i} . The time-energy optimal control is the unique solution to minimizing $\|t - c\|^2$ over all t .

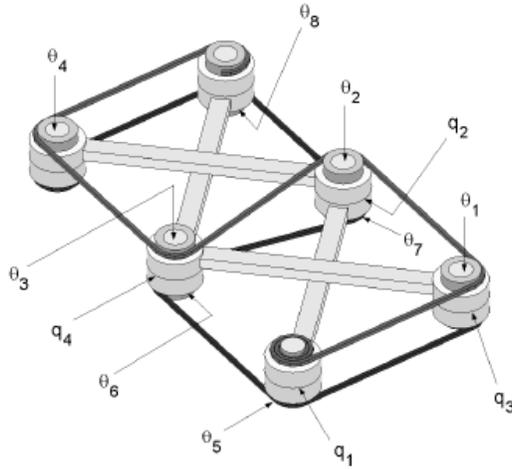


Figure 6: From [111]: Example of a tensegrity robot that uses a continuous cable connecting to struts by pulleys instead of ball joints.

Reconfiguration to a specific state, smallest maximum torque optimization: In [111], this control law is modified to be a minimal-effort (smallest maximum torque) backlash-free (no slack cables) control law. Here, motors drive cables, and motor torque is proportional to cable tension. The major difference from [16] is that no upper tension limit is used, but a lower tension limit is enforced. An optimal control law is derived that finds the trajectory that minimizes the maximum torque while ensuring that all cables remain taut. Using a control law that minimizes the maximum cable tension (expressed as motor torque) for path following, [111] briefly describes a method for generating non-obstructing paths for snakelike robots like the one shown in Figure 7. To do this, candidates were generated and ranked for fitness, where the fitness measure is the smallest angle: large small angles are said to be more controllable. Constraints applied to candidates included attaching nodes to splines. However, [16] does not provide additional details of this obstacle course maneuvering model due to space limitations. Using the system of equations for time-energy optimal control with center tracking described in [16], an iterative guessing algorithm is presented in which one of the parameters of the system is guessed, and if the system converges to the final end conditions (the robot has moved to its expected goal), the optimal trajectory is found, and if not, another guess is made and the process repeats.

Reconfiguration to a specific state, minimize maximum actuation length: Basing the design of tensegrity structures around the ability to control the shape is the main focus of [112]. Of particular interest is the selection of which edges should be given actuators, as giving all edges actuators is seen as costly. This paper presents mixed integer linear programming, with the objective function of minimizing the change in length of actuated struts, as an often suitable way to find a good path that minimizes actuation distance. Its main strength is the ability to be used on-line, with computation times roughly equivalent to the amount of time needed to follow the path. However, for rigorous treatment, mixed integer linear programming is inappropriate.

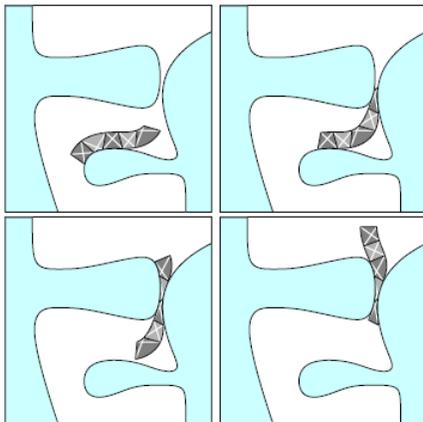


Figure 7: From [111]: Snakelike motion of a 2D planar tensegrity robot through a simulated obstacle course.

Reconfiguration to a specific state, avoid collision: In [56], a Lyapunov-based control system is used. Here, Model Predictive Control is used to estimate the future behavior of a moving tensegrity structure and choose a control input such that collisions are avoided up to a prediction time horizon. This approach is simulated for a 3-prism, with paths for the top 3 nodes shown in Figure 10. However, the main drawback is that there is no systematic method to find suitable Lyapunov functions for arbitrary tensegrity structures, and the time required to search for a suitable control input that reliably avoids collisions makes it impossible to use in real time.

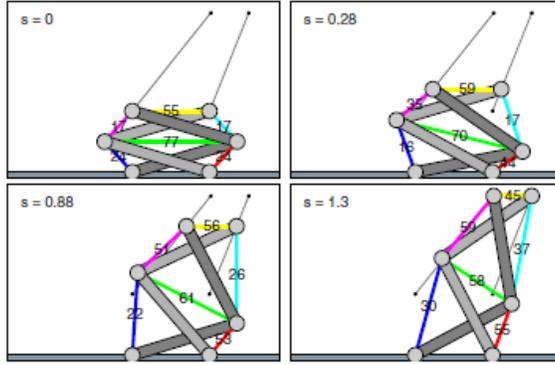


Figure 8: From [16]: A planar tensegrity manipulator that follows a path and minimizes the variation of forces (shown on the cables), with the admissible set being $0 \leq f \leq 100$.

3.3.2 Discrete Path Control

Due to the difficulty of form-finding, tensegrity robot shape control is often broken up into a discrete problem, where intermediate points on the path are solved with any of the previously described form-finding methods, and path control in between is just an interpolation of the equilibrium form values. By doing this, the space becomes searchable with graph-based techniques. This, in turn, allows for a path-finding algorithm or a controller to use optimization techniques in a large-dimensional space. If the number of discrete lengths for a single element is C_l , then the number of states is C_l^E in an E -dimensional length space. However, even with discretization, the search problem quickly becomes intractable for large structures. In the literature, discrete search lends itself particularly well to collision detection methods, due to collision forces being more easily modeled as a form-finding problem rather than a continuous transformation problem that must make continuous checks to see if a collision has occurred.

Reconfiguration to a specific state, no path optimization: In [59,113], the goal is to deflect a wing-like structure to match a desired goal shape: the flapping motion of a cownose ray. Here, the top surface of a tensegrity wing (see Figure 18) must match a curve as closely as possible. The control question is simply: which of the cables must be actuated? Only the start and end configurations are relevant; the intermediate steps in the path are not. The presented algorithm searches, using the MATLAB function *patternsearch*, for an optimization of the form-finding problem that places the top nodes of a structure as close as possible to the goal curve. When searching, if a good solution is found, future searches will look in the area of the previous best solution. The results are promising: the 18 3-prism cell structure shown in Figure 18 closely matches the upward deflection of a cownose wing. Figure 9 shows a plot of the goal curve and the top node points for a beam structure.

Reconfiguration to a specific state, average allowable force optimization: In [91,102], discretized search methods are used to find desired trajectories for the center of mass [102], or for nodes [91]. They used a nonlinear optimization technique to find intermediate equilibrium forms, and interpolated between forms. Shape control is modeled in [91] by first generating a path, then using an \mathcal{H}_2 controller to remove unwanted vibrations. Paths are generated as discrete state changes, where at each state, the forces within the tendons are minimized without becoming slack, and not allowing yield and buckling forces. During the shape change, the vibrations caused by the actuators are reduced by an \mathcal{H}_2 controller on a linearized version of the system dynamics at each step. The \mathcal{H}_2 controller is sufficient in keeping the actual path close to the reference path.

Reconfiguration to a specific state, minimize maximum actuation length: Discrete search is also used in [33] to actively control cable lengths to return a damaged tensegrity bridge to a state that

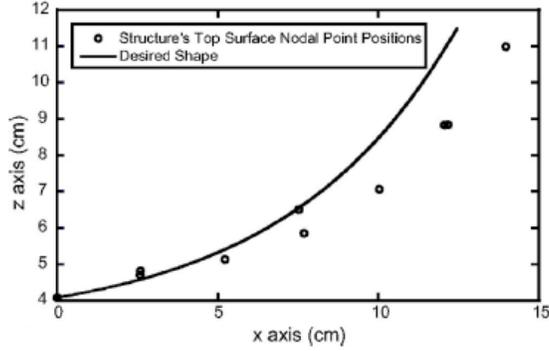


Figure 9: From [59]: The top node positions of a one-dimensional analogue to the cownose ray shown in Figure 18.

fits a safety requirement that the mid-span displacement must not exceed a certain length. The control system presented is a search through state space in which the objective function is minimizing the total actuation lengths of cables subject to structural safety constraints and cable/strut length constraints. If the cable lengths are discretized, the search space becomes exponential in the number of cables, thus a stochastic search method is necessary. Two stochastic search methods are used: Probabilistic Global Search Lausanne (PGSL), and Gradient Search. PGSL is based on the assumption that better solutions are close to good solutions. The result is that PGSL performs better at minimizing actuation length, but that both methods reduce mid-span displacement to safe levels. If cables are connected over pulleys, this also reduces the number of actuators needed to repair structures.

Reconfiguration to a specific state, minimize tracking error integral: For a constrained point mass model, finding the path that minimizes the tracking error integral (the distance between desired state and current state over time), [46] finds that a general solution is impossible, and so a discretization is performed in which the path is divided into several parts, and a constant control input is used for each part, called a “quasi-static” solution. The control inputs for each part were determined by using an artificial neural network. The quasi-static method is compared to a nonlinear gain scheduling control algorithm that also uses artificial neural networks for determination of control inputs, and the gain scheduling algorithm is found to reduce both the mean square tracking error and the tracking time. The authors conclude using neural networks to approximate the inverse input-output function is reliably robust.

Reconfiguration to a specific state, avoid collision: To solve the collision problem [55], an unconstrained problem is posed where the constraints are replaced by penalty forces that increase for every iteration of the algorithm. Minimizing the forces will tend to avoid situations where large penalties are applied, thus resulting in collision-free paths. This is represented by:

$$\min_{\mathbf{x}} f_{cost}(\mathbf{x}) + c \sum_i \|h_i(\mathbf{x})\|^2 + \sum_i \frac{1}{c} (e^{-cg_i(\mathbf{x})} - 1) \quad (57)$$

Where \mathbf{x} is the vector of optimization variables, $h_i(\mathbf{x})$ are the equality constraints, $g_i(\mathbf{x})$ are the inequality constraints, $f_{cost}(\mathbf{x})$ is the cost function in terms of elastic energy, and c is the positive penalty factor increased after each iteration. Since c grows for each iteration a constraint is not satisfied, the offending bodies are eventually forced apart. Minimizing this optimization function results in a configuration that does not have collisions. Figure 10 shows the result of using penalty forces to prevent struts from intersecting one another. This collision avoidance method is applied in [56] for use with Lyapunov controls. Collision avoidance is also considered in [114]. In this one, a path starts as a start/end pair configuration, and the path is divided into multiple segments, with

intermediate positions called “relay points”, if there is no single actuation path that leads to the goal without intersection. This paper only considers a planar robot without self-intersections.

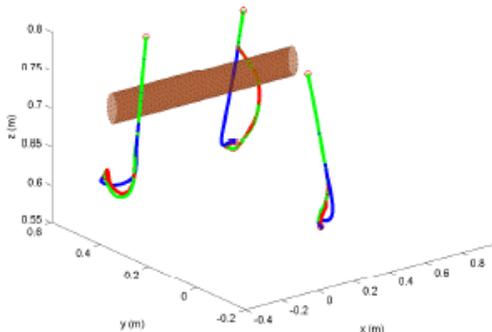


Figure 10: From [56]: starting from a compressed state, the top 3 nodes move without collision avoidance (blue), with external collision avoidance (green), and both internal and external collision avoidance (red).

Move along a line, no optimization: A tensegrity model of a caterpillar is studied in [17, 115], inspired by [26, 99]. The cable spring model is nonlinear to mimic the stretching limits of muscles: the spring stiffness changes depending on length. The caterpillar model is a chain of tensegrity cells, with two control schemes: a low level local controller, and a high level central controller. The low level controller reacts to sensed forces to maintain the integrity of cells, while the high level controller issues simple commands to the cells to cause movement. This simple soft robot model is useful for rough terrain. It also embodies a decentralized approach in that the high level controller does not have to directly control each cell, but can rely on distributed low level controllers. The control strategy here is further developed in [116].

Move along a line, greedy choice: A brute force control approach is explored in [47, 107], based on the rigid body model developed in [25]. The lengths of the edges are discretized, creating a searchable space of possible configurations. Using a very simple 3-prism with 3 struts and 9 cables, and allowing each cable to shorten by a discrete length, stay the same length, or lengthen by a discrete length over a time interval Δt , there are 3^9 possible states for each interval. To move the center of mass of the 3-prism along a straight line, the best option was greedily chosen at each interval. This method was used to simulate a walking 3-prism.

Move along a line, maximum speed: A genetic algorithm is derived for generating high-fitness periodic gaits of a 4-prism with 4 actuated cables [27, 34–36], with a fitness function of how far the center of mass of a structure traveled. Cable actuation is modeled as instantaneous changes in rest length, causing discontinuous increases and decreases in force applied to nodes. A gait cycle is a set of at most one shortening/lengthening pair for each cable in the model. The genome string for each gait included the overall gait period, and the amplitude, phase, and duration of cable contraction for each cable. Every 200-agent generation chose the 100 most fit genomes and further divided those into two groups of 50 for crossover and 50 for mutation. This algorithm was run over 200 generations. With all actuators working, the average fitness was 2.59 meters over 1 second. With one damaged actuator, this was reduced to 2.21 meters, and with two damaged actuators, 1.62 meters. A jumping gait was found for no damaged actuators and one damaged actuator, but not for two damaged actuators. Figures 12 and 13 show samples of a jumping undamaged structure, and a walking damaged structure. An interesting note: while the gaits shown in Figures 12 and 13 are periodic, the force and ground contact plots are not. Further examination would reveal whether these plots become periodic, or if chaotic behavior prevents this. While the generation of locomotion of a 24-DOF structure using only

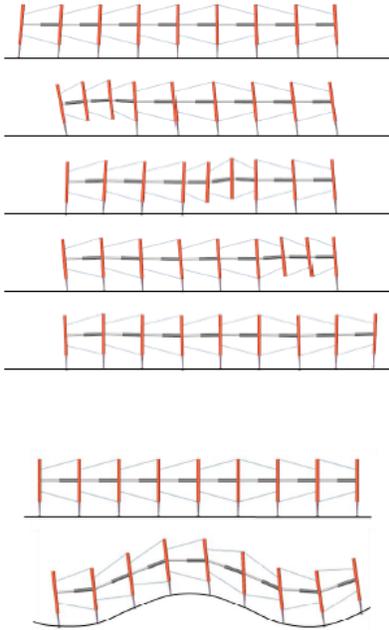


Figure 11: From [115]: Examples of a 2D caterpillar-inspired tensegrity structure walking by (top) propagating a wave forward and lifting legs, and (bottom) maneuvering over hilly terrain.

two actuators works very well for the 4-prism, the authors acknowledge that it may not work for all high-DOF high-dynamic-coupling structures.

This work was developed further by [26,99] in the context of decentralized control. In the interest of doing away with central control, each strut is its own computer and can make decisions only on the forces it senses. Each strut controls its adjacent cables with a spiking neural network that accepts only the tensions at the ends as inputs. Controllers were evolved that chose both the location of the cable actuators, and the weights for each spiking neural network. The goal is to develop gaits. The evolved gaits differ depending on the speed of actuation, which is evidence of the deep coupling between gaits and system dynamics. A more important result is that the dynamic coupling of the modules leads to the *emergence of communication* between struts through the variation of tensions. This leads to the general result that communication through tension between distributed computational struts allows for complex movement. Models such as the one in [26,99] show that the computation necessary for complex dynamic and mechanical behavior can be done in the mechanism itself, lending credence to the notion that similar phenomena can be found in biological systems, and making soft robotic control computation tractable by embedding it within the structure.

4 Applications

Tensegrity robots have been envisioned as many things, ranging from manipulators meant to compete with pick-and-place machines, to locomotors to be used in studying animal locomotion. One common trend is that these robots are compliant and potentially safe for use around humans. These generally parallel the applications of soft robots, but differ in that these robots are mostly implementations of simple tensegrity structures. With the recent studies into caterpillar locomotion and simulation of nervous system control [17,99,115], the door is open for more crossovers into soft robotics.

Manipulators: A parallel manipulator was designed by [117,118] as a tensegrity structure. This

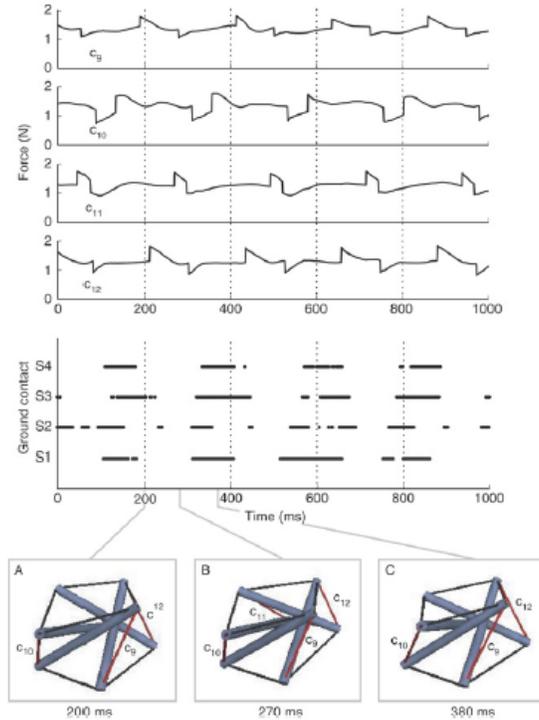


Figure 12: From [27]: A jumping gait is achieved by a 4-cable-actuator-controller found by genetic algorithms. The ground contact plot (bottom) shows contact durations for the bottom four nodes, with several time gaps in which all nodes are airborne. Cable actuations are shown by spikes in the force plot (top).

was extended by [16, 86, 87, 111], who used continuous cables sliding over pulleys on the ends of struts, and showed that such a manipulator has a much larger workspace than an equivalent Stewart robotic platform. Figure 8 shows an example of the structure of such a proposed model. A planar manipulator is presented in [104] with 3 degrees of freedom and only 2 actuators, and resembles the planar mechanisms found in [16, 86, 87, 111]. In three dimensions, [119] describes a manipulator based on the 3-prism, in which the base is fixed. In this version, the manipulator is restricted so that the top surface only translates.

Rolling Locomotors: [30] describes a simple rolling tensegrity structure (Figure 15) that uses *curved struts* — uniquely in the literature — but provides only a physical prototype with no computational model. With actuators on the struts, movement happens when actuation causes the structure to tip over, rolling into a new configuration. A cubic variant, with straight struts that meet at the center, demonstrates the same rolling-by-tipping principle. A very similar robot (Figure 16), based on the tensegrity icosahedron, rolls through selective deformation of struts [120, 121].

Walking, Running, and Jumping Locomotors: Locomoting robots that are capable of walking, running, and jumping, can be used in a wide range of applications unsuitable to traditional wheeled robots. However, where in the former a single joint/member failure leads to system-wide failure, robust tensegrity robots can keep going with graceful degradation in performance [35]. In traditional walkers, the physical complexity of robots requires numerous elements, whereas tensegrity walkers can be comprised by only a few base elements: actuated cables, actuated struts, passive cables of varying lengths, and passive struts of varying lengths. Even the simplest tensegrity structure, the 3-prism, can be made to walk and jump by changing cable lengths (see Figure 14). These kinds of tensegrity

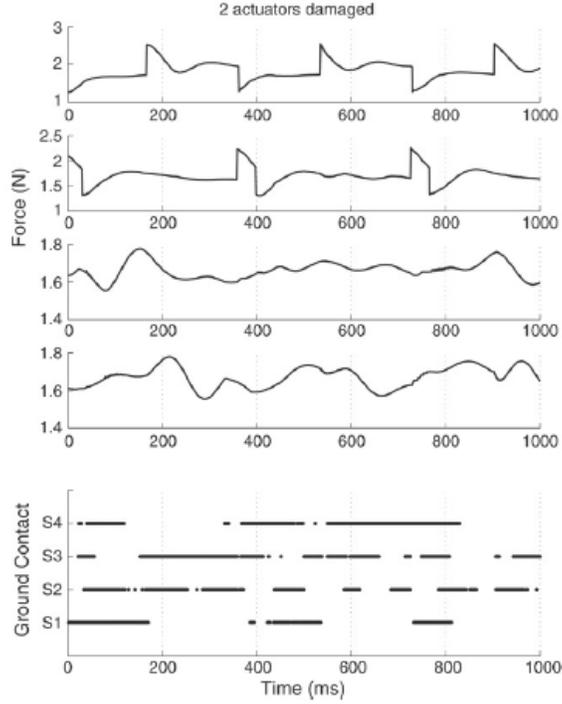


Figure 13: From [27]: With two actuators damaged, a less optimal walking trajectory is still found.

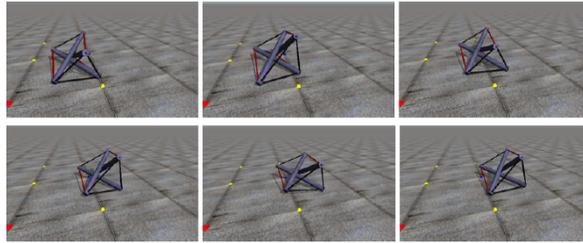


Figure 14: From [35]: Snapshots of the locomotion of a 3-prism.

locomotors are the focus of several works, including [26, 27, 34–36, 107]. Studies into musculoskeletal biology show that there is plenty of promise for the improvement of animal-like locomotion [14, 15]. Fast moving animals have tensional elements in their musculoskeletal system that can help store and distribute energy.

Snake-like, Worm-like, and Caterpillar-like Locomotors: [111] presents a model in which a 2D tensegrity robot can deform and maneuver through an obstacle course. Figure 7 shows one such possible maneuver sequence of a snakelike robot through tight spaces. [101] presents a model in which a 3D tensegrity robot, made of 4-simplices connected strut-to-strut (Class 2), can locomote by applying a transverse wave through the structure, generating forward movement, similar to worms. The result of this is shown in Figure 17. Another variation of this kind of mobility comes in the form of caterpillars, which are well-known examples of high-mobility soft structures. [115] describes a simple 2D tensegrity structure that can maneuver in a similar fashion, including over terrain. A tensegrity-inspired robot

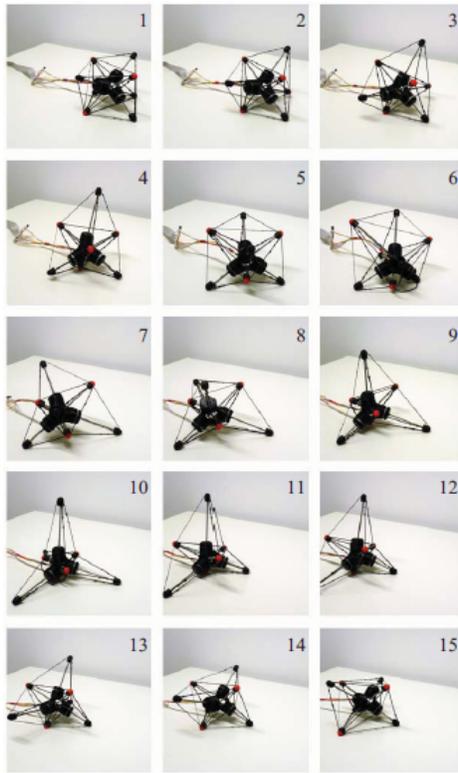


Figure 15: From [30]: Sequence of deformations that lead to a rolling motion.

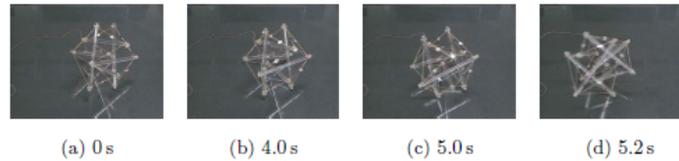


Figure 16: From [121]: An icosahedron tensegrity robot can also roll by deforming its struts.

has also been shown to move through rough terrains using only system vibration from a harmonic oscillator [122]. The tensegrity basis allows the robot to deform if necessary.

Wings: [59] considers using tensegrity structures as aerodynamic/hydrodynamic surfaces to mimic the propulsion of eagle rays. Such structures may one day be used for underwater locomotion, but this has not been tested yet.

Structure Repair: Tensegrity structures are robust: therefore, a failure of a single member does not mean the failure of the entire structure. In [33], bridges made of tensegrity can heal from the failure of several cables.

Other Uses: [28] describes a smart sensor made of local force sensors. [123] described a flight simulator as an alternative to Stewart platform flight simulators. [124] describes a potential use as a shape memory: by having multiple stable states, some with higher potential energy than others, a deformation may be “remembered”, and later, a suitable force may push the tensegrity structure back

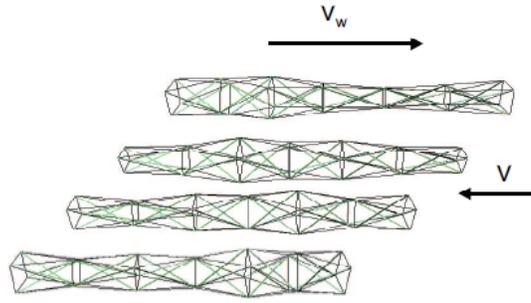


Figure 17: From [101]: Worm-like locomotion of a Class 2 tensegrity structure by applying a wave transformation that travels opposite (V_w) to the direction of movement of the structure (V).

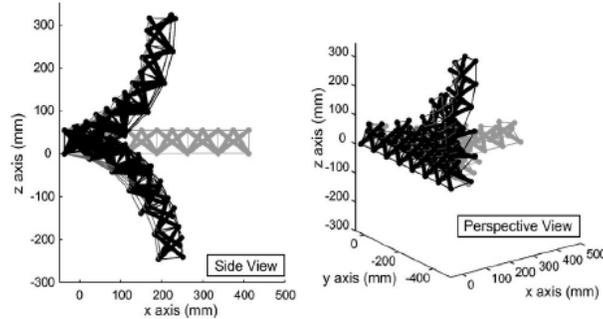


Figure 18: From [59]: Deflection of a tensegrity structure modeled after the motion of a cownose ray.

into its original state.

5 Conclusions and Directions for Future Research

While the field of literature in tensegrity has been very well explored by several groups using several methods for both form-finding and control, the current state of the art is still tied to the artistic vision that arose when tensegrity was discovered. However, many of the techniques may be applicable for control problems in soft robotics, and many methods in soft robotics may be applicable to tensegrity robots. This author believes that eventually, the two disparate fields will be united, thereby advancing both concepts. To do this, several avenues for research are possible:

To unite all the various approaches used in tensegrity robotics, it is necessary to perform a rigorous qualitative comparison between control schemes, both mathematically and in simulation. With few exceptions, mostly within research groups, tensegrity roboticists tend to cite one another, but do not provide qualitative comparisons between their work and the work of other groups. Such comparisons could serve as a springboard for further improving control systems, and save research groups just beginning explorations into tensegrity the effort of trying to determine which schemes work best. This can also be a springboard to improve upon standard control theoretical methods, for example, testing optimal and robust control methods on three-dimensional, large, irregular tensegrity structures.

Tensegrity models may also be expanded to include surfaces, continuum volumes, and nonlinear actuators, thereby merging with soft robotics. Many desirable soft robots are not composed of simple

cables and struts; however, surfaces can be modeled as meshes of members separated by springs, and has been quite successful in the simulation of cloth dynamics. Work toward this goal has been started in [40], but to date, nothing else has occurred. Expanding the classes of struts and cables could lead to a variety of interesting capabilities, and would better reflect what has been studied in biology. A current challenge in soft robotics is to find accurate models and controllers that describe soft robotics, as current methods are too computationally expensive to be used for real-time control [23]. By casting a soft robotics model as a tensegrity model, this might become tractable.

Research into distributed control systems should also be expanded. This has been done in [17,115], and biologically-inspired distributed control systems were investigated [26,27]. Combined with an expanded model of tensegrity structures to include surfaces, many interesting and useful techniques may be found to advance the state of soft robotics. With embedded systems, distributed control systems may be devised that can run in real time, and can react to local disturbances such as obstacles, grippable objects, and localized failure. This is also an opportunity to further explore the space of path planning and control to develop robots that can reliably maneuver through rough terrain, grip soft objects, and be compliant.

One very noticeable hole in the literature is path and shape motion through non-equilibrium states; that is, fast movement. This happens in [26,27,34–36,99] as an artifact of real-time physics libraries, but specific studies have never been performed. Thus, another option is to explore control systems that actively use the dynamics of systems to move efficiently; that is, instead of planning paths as a slow reconfiguration, reconfigure by taking advantage of swift dynamics, in order to emulate animalistic behavior such as running, jumping, and climbing. Since the dynamics are nonlinear, it is very likely that all tensegrity systems not in equilibrium are chaotic. However, instead of the trying to eliminate the dynamics as all perturbation control schemes are intended to do, it may be beneficial to apply chaotic control methods to these systems.

In general, all of the above goals are intended to divert focus from standard symmetrical tensegrity structures like prisms and domes to soft robotic structures, irregular structures, animal-like structures, and cell-like structures that may find practical use. Some recent work toward that goal has been done, but tensegrity robots, currently, have a very limited scope despite the enormous potential that could come with combination with soft robotics, but this author believes that it will soon change.

References

- [1] A. Pugh, *An Introduction to Tensegrity*. University of California Press, 1976.
- [2] R. B. Fuller, “Tensile-integrity structures,” *United States Patent 3063521*, November 1962.
- [3] R. Motro, “Tensegrity systems: the state of the art,” *Journal of Space Structures*, vol. 7, no. 2, pp. 75–83, 1992.
- [4] D. Ingber, “Cellular tensegrity: defining new rules for biological design that govern the cytoskeleton,” *Journal of Cell Science*, vol. 104, pp. 613–627, 1993.
- [5] D. Ingber, “Architecture of life,” *Scientific American*, vol. 52, pp. 48–57, 1998.
- [6] D. Ingber, “Tensegrity i. cell structure and hierarchical systems biology,” *Journal of Cell Science*, vol. 116, pp. 1157–1173, 2003.
- [7] D. Ingber, “Tensegrity ii. how structural networks influence cellular information processing networks,” *Journal of Cell Science*, vol. 116, pp. 1397–1408, 2003.
- [8] R. Beer, H. Chiel, R. Quinn, and R. Ritzmann, “Biorobotic approaches to the study of motor systems,” *Current Opinion in Neurobiology*, vol. 8, no. 6, p. 777782, 1998.
- [9] C. Chen and D. Ingber, “Tensegrity and mechanoregulation: from skeleton to cytoskeleton,” *Osteoarthritis Cartilage*, vol. 7, no. 1, pp. 81–94, 1999.

- [10] N. Wang, K. Naruse, D. Stamenovic, J. Fredberg, S. Mijailovich, I. Tolic-Norrelykke, T. Polte, R. Mannix, and D. Ingber, “Mechanical behavior in living cells consistent with the tensegrity model,” in *Proceedings of the National Academy of Sciences*, vol. 98, pp. 7765–7770, 2001.
- [11] C. Sultan, D. Stamenovic, and D. Ingber, “A computational tensegrity model predicts dynamic rheological behaviors in living cells,” *Annals of Biomedical Engineering*, vol. 32, no. 4, pp. 520–530, 2004.
- [12] T. Liedl, B. Högberg, J. Tytell, D. Ingber, and W. Shih, “Self-assembly of three-dimensional prestressed tensegrity structures from dna,” *Nature Nanotechnology*, vol. 5, pp. 520–524, 2010.
- [13] D. Stamenovic, J. Fredberg, N. Wang, J. Butler, and D. Ingber, “A microstructural approach to cytoskeletal mechanics based on tensegrity,” *Journal of Theoretical Biology*, vol. 181, pp. 125–136, 1996.
- [14] S. Vogel, *Cats Paws and Catapults: Mechanical Worlds of Nature and People*. W.W. Norton and Company, 2000.
- [15] S. Levin, “The tensegrity-truss as a model for spinal mechanics: biotensegrity,” *Journal of Mechanics in Medicine and Biology*, vol. 2, no. 3, 2002.
- [16] J. B. Aldrich and R. E. Skelton, “Time-energy optimal control of hyper-actuated mechanical systems with geometric path constraints,” in *Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference 2005*, pp. 8246–8253, December 2005.
- [17] O. Shai, T. Tehori, A. Bronfeld, M. Slavutin, and U. Ben-Hanan, “Adjustible tensegrity robot based on assur graph principle,” in *Proceedings of the ASME 2009 International Mechanical Engineering Congress and Exposition*, 2009.
- [18] R. Motro, *Tensegrity: Structural Systems for the Future*. Kogan Page, 2003.
- [19] R. Skelton, J. Helton, R. Adhikari, J. Pinaud, and W. Chan, *An Introduction to the Mechanics of Tensegrity Structures*. CRC Press, 2002.
- [20] R. W. Burkhardt, *A Practical Guide to Tensegrity Design*. Online, 2008.
- [21] R. Skelton and M. de Oliveira, *Tensegrity Systems*. Springer, 2009.
- [22] K. Snelson, “Photograph of b-tree,” <http://kennethsnelson.net/category/sculptures/outdoor-works/>, 2012.
- [23] D. Trivedi, C. Rahn, W. Kier, and I. Walker, “Soft robotics: Biological inspiration, state of the art, and future research,” *Applied Bionics and Biomechanics*, vol. 5, no. 3, pp. 99–117, 2008.
- [24] N. Correll, C. Onal, H. Liang, E. Schoenfeld, and D. Rus, “Soft autonomous materials using programmed elasticity and embedded distributed computation,” in *International Symposium on Experimental Robotics*, 2010.
- [25] J. M. Mirats Tur, S. Hernandez, and A. G. Rovira, “Dynamic equations of motion for a 3-bar tensegrity based mobile robot,” in *IEEE Conference on Emerging Technologies and Factory Automation*, pp. 1334–1339, September 2007.
- [26] J. Rieffel, F. Valero-Cuevas, and H. Lipson, “Morphological communication: exploiting coupled dynamics in a complex mechanical structure to achieve locomotion,” *Journal of the Royal Society Interface*, vol. 7, no. 45, pp. 613–621, 2009.
- [27] C. Paul, “Morphological computation: A basis for the analysis of morphology and control requirements,” *Robotics and Autonomous Systems*, vol. 54, pp. 619–630, 2006.

- [28] C. Sultan and R. Skelton, “A force and torque tensegrity sensor,” *Sensors and Actuators*, vol. 11, no. 2, pp. 220–231, 2004.
- [29] S. Hernandez and J. M. Mirats Tur, “Tensegrity frameworks: Static analysis review,” *Mechanism and Machine Theory*, vol. 43, no. 1, pp. 859–881, 2008.
- [30] V. Bohm, A. Jentzsch, T. Kaufhold, F. Schneider, and K. Zimmerman, “An approach to compliant locomotion systems based on tensegrity structures,” in *Proceedings of the 56th International Scientific Colloquium*, September 2011.
- [31] H. Furuya, “Concept of deployable tensegrity structures in space applications,” *Journal of Space Structures*, vol. 7, no. 2, pp. 143–151, 1992.
- [32] A. Tibert, *Deployable Tensegrity Structures for Space Applications*. PhD thesis, Royal Institute of Technology, 2003.
- [33] S. Korkmaz, N. Bel Hadj Ali, and I. F. C. Smith, “Self-repair of a tensegrity pedestrian bridge through grouped actuation,” in *Proceedings of the International Conference in computing and Building Engineering*, 2010.
- [34] C. Paul, F. Valero-Cuevas, and H. Lipson, “Design and control of tensegrity robots for locomotion,” *Robotics, IEEE Transactions on*, vol. 22, pp. 944–957, October 2006.
- [35] C. Paul, J. W. Roberts, H. Lipson, and F. Valero-Cuevas, “Gait production in a tensegrity based robot,” in *Proceedings of the International Conference on Advanced Robotics, 2005*, pp. 216–222, July 2005.
- [36] C. Paul and H. Lipson, “Redundancy in the control of robots with highly coupled mechanical structures,” pp. 3585–3591, August 2005.
- [37] C. Sultan, *Modeling, design and control of tensegrity structures with applications*. PhD thesis, Purdue University, 1999.
- [38] A. Tibert and S. Pellegrino, “Review of form-finding methods for tensegrity structures,” *International Journal Solids and Structures*, vol. 18, no. 4, pp. 209–223, 2003.
- [39] J. Mirats Tur and S. Hernandez, “Tensegrity frameworks: Dynamic analysis review and open problems,” *Mechanism and Machine Theory*, vol. 44, no. 1, pp. 1–18, 2009.
- [40] C. Yu, K. Haller, D. Ingber, and R. Nagpal, “Morpho: A self-deformable modular robot inspired by cellular structure,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems, 2008*, pp. 3571–3578, September 2008.
- [41] B. Roth and W. Whiteley, “Tensegrity frameworks,” *Transactions of the American Mathematical Society*, vol. 265, pp. 419–446, 1981.
- [42] R. Connelly and G. Whiteley, “Second order rigidity and prestress stability for tensegrity frameworks,” *Discrete Mathematics*, vol. 9, no. 3, pp. 453–491, 1996.
- [43] S. Pellegrino and C. Calladine, “Two step matrix analysis of prestressed cable nets,” in *Proceedings of the Third International Conference of Space Structures*, pp. 744–749, 1984.
- [44] H.-J. Schek, “The force density method for form finding and computation of general networks,” *Computer Methods in Applied Mechanics and Engineering*, vol. 3, pp. 115–134, 1974.
- [45] R. Connelly, “Rigidity and energy,” *Inventiones Mathematicae*, vol. 66, no. 1, pp. 11–33, 1982.
- [46] N. Kanchanasaratool and D. Williamson, “Motion control of a tensegrity platform,” *Communications in Information and Systems*, vol. 2, no. 3, pp. 299–324, 2002.

- [47] A. G. Rovira and J. M. Mirats Tur, “Control and simulation of a tensegrity-based mobile robot,” *Robotics and Autonomous Systems*, vol. 57, pp. 526–535, 2009.
- [48] R. Skelton, J. Pinaud, and D. Mingori, “Dynamics of the shell class of tensegrity structures,” *Journal of the Franklin Institute*, vol. 338, pp. 255–320, 2001.
- [49] R. Skelton, “Dynamics and control of tensegrity systems,” in *IUTAM Symposium on Vibration Control of Nonlinear Mechanisms and Structures*, pp. 309–318, 2005.
- [50] R. Skelton, “Dynamics of tensegrity systems: compact forms,” in *45th IEEE Conference on Decision and Control*, p. 22762281, 2006.
- [51] H. Murakami, “Static and dynamic analysis of tensegrity structures. part 1: nonlinear equations of motion,” *International Journal of Solids and Structures*, vol. 38, pp. 3599–3613, 2001.
- [52] H. Murakami, “Static and dynamic analysis of tensegrity structures. part 2: quasi-static analysis,” *International Journal of Solids and Structures*, vol. 38, pp. 3615–3629, 2001.
- [53] H. Murakami and Y. Nishimura, “Static and dynamic characterization of some tensegrity modules,” *Journal of Applied Mechanics*, vol. 68, pp. 19–27, 2001.
- [54] H. Murakami and Y. Nishimura, “Static and dynamic characterization of regular truncated icosahedral and dodecahedral tensegrity modules,” *International Journal of Solids and Structures*, vol. 38, no. 50-51, pp. 9359–9381, 2001.
- [55] S. Hernandez and J. M. Mirats Tur, “A method to generate stable, collision free configurations for tensegrity based robots,” in *Proceedings of the 2008 IEEE/RSJ International conference on Intelligent Robots and Systems*, pp. 3769–3774, September 2008.
- [56] S. Hernandez, R. E. Skelton, and J. M. Mirats Tur, “Dynamically stable collision avoidance for tensegrity based robots,” in *Proceedings of the 2009 ASME/IFTOMM International Conference on Reconfigurable Mechanisms and Robots*, pp. 315–322, June 2009.
- [57] J. Averseng and B. Crosnier, “Static and dynamic robust control of tensegrity systems,” in *IASS 2004 Symposium on Shell and Spatial Structures From Models to Realization*, vol. 45, pp. 169–174, 2004.
- [58] J. Averseng, J. Dube, B. Crosnier, and R. Motro, “Active control of a tensegrity plane grid,” in *44th IEEE Conference on Decision and Control*, pp. 6830–6834, 2005.
- [59] K. W. Moored and H. Bart-Smith, “The analysis of tensegrity structures for the design of a morphing wing,” *Transactions of the ASME*, vol. 74, pp. 668–676, July 2007.
- [60] R. Connelly and M. Terrell, “Globally rigid symmetric tensegrities,” *Structural Topology*, vol. 21, pp. 59–78, 1995.
- [61] M. Masic, R. E. Skelton, and P. E. Gill, “Algebraic tensegrity form-finding,” *International Journal of Solids and Structures*, vol. 42, no. 1617, pp. 4833 – 4858, 2005.
- [62] G. Estrada, H. Bungartz, and C. Mohrdieck, “Numerical form-finding of tensegrity structures,” *International Journal of Solids and Structures*, vol. 43, pp. 6855–6868, 2006.
- [63] H. Tran and J. Lee, “Advanced form-finding of tensegrity structures,” *Computers and Structures*, vol. 88, pp. 237–246, 2010.
- [64] H. Tran and J. Lee, “Determination of a unique configuration of free-form tensegrity structures,” *Acta Mechanica Sinica*, vol. 220, pp. 331–348, 2011.

- [65] R. Connelly, “Tensegrity structures: why are they stable?,” in *Rigidity theory and applications*, pp. 47–54, Plenum Press, New York, 1999.
- [66] C. Sultan, M. Corless, and R. Skelton, “Reduced prestressability conditions for tensegrity structures,” in *Proceedings of the 40th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*, April 1999.
- [67] J. Zhang, M. Ohsaki, and Y. Kanno, “A direct approach to design of geometry and forces of tensegrity systems,” *International Journal of Solids and Structures*, vol. 43, no. 7-8, pp. 2260–2278, 2006.
- [68] A. Michelletti and W. Williams, “A marching procedure for form-finding for tensegrity structures,” *Journal of Mechanics of Materials and Structures*, vol. 2, no. 5, pp. 857–885, 2007.
- [69] S. Pellegrino, *Mechanics of kinematically indeterminate structures*. PhD thesis, University of Cambridge, 1986.
- [70] R. Motro, “Forms and forces in tensegrity systems,” in *Proceedings of the Third International Conference on Space Structures*, pp. 180–185, 1984.
- [71] N. B. H. Ali, L. Rhode-Barbarigos, and I. F. C. Smith, “Analysis of clustered tensegrity structures using a modified dynamic relaxation algorithm,” *International Journal of Solids and Structures*, vol. 48, no. 5, pp. 637 – 647, 2011.
- [72] B. Domer, E. Fest, and V. Lalit, “Combining dynamic relaxation method with artificial neural networks to enhance simulation of tensegrity structures,” *Journal of Structural Engineering*, vol. 129, no. 5, pp. 672–681, 2003.
- [73] C. Paul, H. Lipson, and F. Valero-Cuevas, “Evolutionary form-finding of tensegrity structures,” in *Proceedings of the 2005 Conference on Genetic and Evolutionary Computation*, 2005.
- [74] D. Lobo and F. Vico, “Evolutionary development of tensegrity structures,” *BioSystems*, vol. 101, pp. 167–176, 2010.
- [75] J. Rieffel, F. Valero-Cuevas, and H. Lipson, “Automated discovery and optimization of large irregular tensegrity structures,” *Computers and Structures*, vol. 87, pp. 368–379, 2009.
- [76] X. Xu and Y. Luo, “Form-finding of nonregular tensegrities using a genetic algorithm,” *Mechanics Research Communications*, vol. 37, pp. 85–91, 2010.
- [77] M. Masic, R. E. Skelton, and P. E. Gill, “Optimization of tensegrity structures,” *International Journal of Solids and Structures*, vol. 43, pp. 4687 – 4703, 2006.
- [78] M. Pagitz and J. M. Tur, “Finite element based form-finding algorithm for tensegrity structures,” *International Journal of Solids and Structures*, vol. 46, pp. 3235–3240, 2009.
- [79] B. de Jager and R. Skelton, “Inputoutput selection for planar tensegrity models,” *IEEE Transactions on Control Systems Technology*, vol. 13, no. 5, 2005.
- [80] W. Chan, D. Arbelaez, F. Bossens, and R. Skelton, “Active vibration control of a three-stage tensegrity structure,” in *SPIE 11th Annual International Symposium on Smart Structures and Materials*, March 2004.
- [81] R. Motro, S. Najari, and P. Jouanna, “Static and dynamic analysis of tensegrity systems,” in *Proceedings of the ASCE International Symposium on Shell and Spatial Structures: Computational Aspects*, pp. 270–279, 1986.

- [82] R. Motro, S. Najari, and P. Jouanna, “Tensegrity systems. from design to realization,” in *Proceedings of the First International conference on Lightweight structures in architecture*, 1986.
- [83] F. Bossens, R. de Callafon, and R. Skelton, “Experimental modeling and modal analysis of a tensegrity structure,” *International Journal of Solids and Structures*, 2003.
- [84] C. Sultan, M. Corless, and R. Skelton, “The prestressability problem of tensegrity structures: some analytical solutions,” *International Journal of Solids and Structures*, vol. 38, pp. 5223–5252, 2001.
- [85] C. Sultan, M. Corless, and R. Skelton, “Linear dynamics of tensegrity structures,” *Journal of Engineering structures*, vol. 24, pp. 671–685, 2002.
- [86] J. Aldrich, R. Skelton, and K. Kreutz-Delgado, “Control synthesis for a class of light and agile robotic tensegrity structures,” in *Proceedings of the American Control Conference*, pp. 5245 – 5251, june 2003.
- [87] J. Aldrich and R. Skelton, “Control/structure optimization approach for minimum-time reconfiguration of tensegrity systems,” in *Proceedings of SPIE: Smart Structures and Materials 2003: Modeling, Signal Processing, and Control*, pp. 448–459, 2003.
- [88] M. Raja and S. Narayanan, “Active control of tensegrity structures under random excitation,” *Journal of Smart Materials and Structures*, vol. 16, no. 3, pp. 809–817, 2007.
- [89] S. Djouadi, R. Motro, J. Pons, and B. Crosnier, “Active control of tensegrity systems,” *Journal of Aerospace Engineering*, vol. 11, no. 2, pp. 37–44, 1998.
- [90] N. Kanchanasaratool and D. Williamson, “Modelling and control of class nsp tensegrity structures,” *International Journal of Control*, vol. 75, no. 2, pp. 123–139, 2002.
- [91] J. van de Wijdeven and B. de Jager, “Shape change of tensegrity structures: design and control,” in *American Control Conference, 2005. Proceedings of the 2005*, pp. 2522 – 2527 vol. 4, june 2005.
- [92] K. Shea, E. Fest, and I. Smith, “Developing intelligent tensegrity structures with stochastic search,” *Journal of Advanced Engineering Informatics*, vol. 16, no. 1, pp. 21–40, 2002.
- [93] E. Fest, K. Shea, and B. Domer, “Adjustable tensegrity structures,” *Journal of Structural Engineering*, vol. 129, pp. 515–526, 2003.
- [94] E. Fest, K. Shea, and B. Domer, “Active tensegrity structure,” *Journal of Structural Engineering*, vol. 130, no. 10, pp. 1454–1465, 2004.
- [95] B. Domer and F. Smith, “An active structure that learns,” *Journal of Computing in Civil Engineering*, vol. 19, no. 16, 2004.
- [96] B. Adam and I. Smith, “Tensegrity active control: multiobjective approach,” *Journal of Computing in Civil Engineering*, vol. 21, no. 3, 2007.
- [97] L. Kavraki, P. Svestka, J. Latombe, and M. Overmars, “Probabilistic roadmaps for path planning in high dimensional configuration spaces,” *IEEE Transactions on Robotics and Automation*, vol. 12, 1996.
- [98] S. LaValle, “Rapidly-exploring random trees: A new tool for path planning,” tech. rep., Iowa State University, 1998.
- [99] J. Rieffel, B. Trimmer, and H. Lipson, “Mechanism as mind: What tensegrities and caterpillars can teach us about soft robotics,” *Artificial Life*, vol. XI, pp. 506–512, 2008.

- [100] M. Masic and R. Skelton, “Open-loop shape control of stable unit tensegrity structures,” in *Proceedings of the Third World Conference on Structural Control*, pp. 439–447, 2003.
- [101] M. Masic and R. E. Skelton, “Open-loop control of class-2 tensegrity towers,” in *SPIE’s 11th Annual International Symposium on Smart Structures and Materials*, March 2004.
- [102] J. Pinaud, M. Masic, and R. Skelton, “Path planning for the deployment of tensegrity structures,” in *SPIE 10th Annual International Symposium on Smart Structures and Materials*, 2003.
- [103] J. Pinaud, S. Solari, and R. Skelton, “Deployment of a class 2 tensegrity boom,” in *Proceedings of SPIE Smart Structures and Materials*, vol. 5390, pp. 155–162, 2004.
- [104] M. Arsenault and C. M. Gosselin, “Kinematic, static and dynamic analysis of a planar 2-dof tensegrity mechanism,” *Mechanism and Machine Theory*, vol. 41, no. 9, pp. 1072 – 1089, 2006.
- [105] A. Wroldsen, M. de Oliveira, and R. Skelton, “A discussion on control of tensegrity systems,” in *Proceedings of the 45th IEEE Conference on Decision and Control*, 2006.
- [106] A. Wroldsen, M. de Oliveira, and R. Skelton, “Modelling and control of non-minimal non-linear realisations of tensegrity systems,” *International Journal of Control*, vol. 82, no. 3, pp. 389–407, 2009.
- [107] J. M. Mirats Tur, “On the movement of tensegrity structures,” *International Journal of Space Structures*, vol. 25, no. 1, pp. 1–14, 2010.
- [108] A. Wroldsen, *Modelling and Control of Tensegrity Structures*. PhD thesis, Norwegian University of Science and Technology, 2007.
- [109] C. Sultan and R. Skelton, “Deployment of tensegrity structures,” *International Journal of Solids and Structures*, vol. 40, no. 18, p. 46374657, 2003.
- [110] C. Sultan, M. Corless, and R. Skelton, “Symmetrical reconfiguration of tensegrity structures,” *International Journal of Solids and Structures*, vol. 32, no. 8, pp. 2215–2234, 2002.
- [111] J. B. Aldrich and R. E. Skelton, “Backlash-free motion control of robotic manipulators driven by tensegrity motor networks,” in *Proceedings of the 45th IEEE Conference on Decision and Control*, pp. 2300–2306, December 2006.
- [112] B. de Jager, “Design for shape control of tensegrities,” in *Proceedings of the 2006 American Control Conference*, pp. 2528–2533, June 2006.
- [113] K. W. Moored, S. A. Taylor, and H. Bart-Smith, “Optimization of a tensegrity wing for biomimetic applications,” in *Proceedings of SPIE*, vol. 6173, 2006.
- [114] X. Xu and Y. Luo, “Collision-free shape control of a plane tensegrity structure using an incremental dynamic relaxation method and a trial-and-error process,” *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, pp. 1–7, 2012.
- [115] O. Orki, O. Shai, A. Ayali, and U. Ben-Hanan, “A model of caterpillar locomotion based on assure tensegrity structures,” in *Proceedings of ASME 2011 International Design Engineering Technical Conferences and Computers and Information in Engineering*, August 2011.
- [116] A. Bronfeld, “Shape change algorithm for a tensegrity device,” Master’s thesis, Tel Aviv University, 2010.
- [117] T. Tran, C. Crane, and J. Duffy, “The reverse displacement analysis of a tensegrity based parallel mechanism,” in *Proceedings of the Fifth Biannual World Automation Congress*, 2002.

- [118] M. Marshall and C. Crane, “Design and analysis of a hybrid parallel platform that incorporates tensegrity,” in *Proceedings of the ASME 2004 Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, 2004.
- [119] C. A. Mohr and M. Arsenault, “Kinematic analysis of a translational 3-dof tensegrity mechanism,” *Transactions of the Canadian Society for Mechanical Engineering*, vol. 35, no. 4, pp. 573–584, 2011.
- [120] M. Shibata and S. Hirai, “Rolling locomotion of deformable tensegrity structure,” in *WSPC Proceedings*, 2009.
- [121] M. Shibata, F. Saijyo, and S. Hirai, “Crawling by body deformation of tensegrity structure robots,” in *2009 IEEE International Conference on Robotics and Automation*, 2009.
- [122] Y. Hagiwara and M. Oda, “Tetrabot: Resonance based locomotion for harsh environments,” in *The 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2431–2436, October 2010.
- [123] C. Sultan, M. Corless, and R. Skelton, “Tensegrity flight simulator,” *Journal of Guidance, Control, and Dynamics*, vol. 23, no. 6, pp. 1055–1064, 2000.
- [124] M. Defossez, “Shape memory effect in tensegrity structures,” *Journal of Mechanics Research Communications*, vol. 30, no. 4, pp. 311–316, 2003.