Design and Control of a Tensegrity-Based Robotic Joint

Chapter · June 2019

DOI: 10.1007/978-3-030-20131-9_260

2 authors:

Andres Gonzalez
Universidad Latina de Costa Rica

A. Luo
Harbin Engineering University

Design and Control of a Tensegrity-Based Robotic Joint

Andres González and Ani Luo

Harbin Engineering University, Harbin HLJ 150001, China
andres@hrbeu.edu.cn

Abstract. Tensegrity structures are a new trend in the soft robotics field, especially for aerospace applications. However, many other applications, such as biomechanics, have not made full use of the advantages that this kind of structure presents yet. In this paper, we present a robotic joint based on a two-stage tensegrity structure. Using a marching procedure to find new stable positions, the control method calculates the required steps and actuates some of the cables until this new position is achieved. Preliminary experiments show that the structure can attain bending of 20º and maintain the equilibrium. This prototype shows that tensegrity structures can be effectively used for positioning in three dimensions.

Keywords: tensegrity, soft robotics, asymmetric motion.

1 Introduction

Defined by Bucky Fuller as "islands of compression inside of an ocean of tension" [1], the tensegrities are structural systems composed of compressive members (bars or struts) and tensile elements, where the forces are solely distributed as axial forces along the members, thus removing bending and other compound stresses. Amongst its main properties, being lightweight, the forces distribution, self-equilibrium, and the possibility to apply control methods for active reconfiguration have made the tensegrities a hot topic in the soft robotics field of study.

Another characteristic of tensegrities is that several authors have used them to model the structures of human cells, as Ingber [2] who argued that tensegrity is the fundamental architecture of life. Castro Arenas et al. [3] use the tensegrity structures to model the dynamic behavior of anatomical structures.

Human bodies have also been compared to tensegrity structures at a macro level. Silva et al. [4] compare the ability of tensegrity structures to re-acquire a force balance after being disturbed by mechanical forces from within or from its environment with the human body. Scarr [5] models the elbow joint as a tensegrity structure thanks to their similitude of maintaining equilibrium during active reconfiguration thanks to their continuous tension.

The main robotic developments have been made for aerospace applications, where the structures are actuated to compress from their original dimension to be located inside of rockets, and then deployed once in the space; deployment combined with light weight allows to reduce costs in the industry. Tensegrity masts [6] and deploya-

T. Uhl (ed.), *Advances in Mechanism and Machine Science*,
Mechanisms and Machine Science 73,
https://doi.org/10.1007/978-3-030-20131-9_260

ble antenna reflectors [7, 8] are just some examples of projects in this area. Similar designs have been made in the structural engineering field, with active deployable tensegrity bridges like the one proposed by Rhode-Barbarigos [9].

However, these proposals have in common that their static and dynamic properties are based on symmetry. Their members are divided into symmetry groups and it is supposed that all the members on each group present the same kind of motion. This assumption simplifies the static and dynamic models but, at the same time, limits the range of movement for the whole system. In the case of mast-like structures, this condition limits the reconfiguration to deployment motion in one axis only. Exceptions are the SUPERball by Caluwaerts et al. [10], or similar proposals by Koizumi et al. [11] and Ushigome et al. [12], rolling tensegrity robots that base their motion on creating asymmetry between its members to displace their mass center for allowing it to roll over one of its sides.

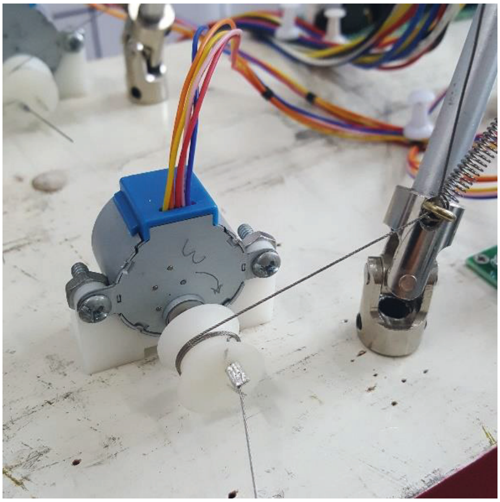
It is not until recent years that new research in biomechanics, especially in the so-called bio tensegrity subfield, has combined the discoveries in biology with the properties of tensegrities, and proposals like the tensegrity shoulder by Baltaxe et al. [13] have emerged. In this paper, we present a variation of the well-known two-stage tensegrity tower for asymmetric motion between the units. Using a marching procedure, modified from the originally presented by Micheletti and Williams [14] to calculate new stable positions, and fixing the bottom nodes to the ground, we show how the structure can bend to a new stable position, along different axes, by actuating some of its tensile members. The length change is done by stepper motors, and the control is done by an Arduino Mega 2560 board and MATLAB. If units of a tensegrity tower are allowed to have independent motion, as in a joint, tensegrity structures could have new applications, such as positioning or locomotion.

2 System Design

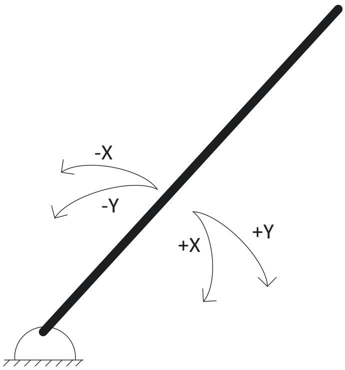
2.1 Structure

Our structure is based on a two-stage tensegrity system [15], composed of 6 bars and 18 tensile members. The later are divided into vertical, diagonal and saddle cables; the vertical cable connects a bottom node with a top node of the same unit, a diagonal cable connects a bottom node of an inferior unit with a bottom node of the next unit, saddle cables connect top nodes of an inferior unit with bottom nodes of a superior unit. In this project, the saddle cables have a fixed length, while vertical and diagonal cables will change their length. The span of the components, and location of the nodes is based on [6]. Because of the nature of the motion, the bottom nodes are fixed to the base and so the bottom cables are removed. The cables connecting the top nodes are also replaced by a stiff surface. The total height of the structure is 300 mm, with a height per unit of 200 mm and an overlap of 50%. The radius of the circumscribed circle connecting nodes at any level is 75 mm.

Since this is a first prototype, whose main goal is to verify the kinematic model and control scheme, we decided to use simple materials that were already available in our laboratory for the hardware components. The compressive elements are made of aluminum pipe of 8 mm in diameter and a wall thickness of 1.35 mm; the form finding process defines a required length of 238.5 mm per bar, for an overall weight of 85.86 grams. Holes of 3 mm were perforated on each bar to attach the cables using clamps. Because our design requires the bottom nodes to be fixed to the ground, universal joints are used to link the bars to the base, as shown in Fig. 1a, allowing the rotation described in Fig. 1b.



(a)



(b)

Fig. 1. Representation of a fixed node and its connection to the ground. (a) hardware of the system, along with the connection to the motor, (b) diagram of motion of a single bar.

For the upper surface, it is defined that the top nodes always maintain the same distance between them. At the same time, we needed a surface where to locate the motors for the upper cables, so we decided to use an acrylic sheet of 2 mm thick and 180 mm in diameter; the bars were also fixed to this surface using universal joints. The bases of the motors and the pulleys are 3D printed components made of PLA material and the weight is neglected. The overall weight on top of the structure is approximately 220gr and is evenly distributed to avoid bending forces.

The tensile members are divided into two types: cables and springs. The steel cables have a diameter of 0.5 mm and a tensile strength of 135 N; this small diameter allows to reduce the effect of the weight of the cables to the minimum while still maintaining enough strength. The springs have a constant factor of 0.11 N/mm. Because two types of tensile members change their length, but we only control one of them, we use springs to actively change the length of the other members. Fig. 1 shows how the cable goes through a ring before being rolled by the motor's pulley; this element maintains the proper alignment of the cable while redirecting the cable to the motor. Despite the possible friction, the involved forces are small and so we neglect its effect; also, no commercial pulleys of the required dimensions were found.

2.2 Motors

The stepper motors model 28BYJ-48 were chosen for driving the actuated cables; although being small, these unipolar stepper motors bring enough output torque for the desired application. Compared to other similar projects that use DC motors [16], we did not use this kind of motor because of the difficulty to precisely control the position of the shaft. Servo motors could also solve this issue, but their range of motion is limited, compared to stepper motors. Another advantage of the stepper motors is that they lock once they reach the desired position, keeping the cables under tension all the time.

Because the stepper motors require high current at the beginning of the rotation, and this condition can damage the Arduino integrated circuit, we use ULN2003AN drive boards. These Darlington transistor arrays for unipolar stepper motors can handle the 370 mA (measured value) that each motor require when loaded. The selected drive boards are powered by an external power supply.

2.3 Integrated Circuit

For controlling the motors, we used an Arduino Mega 2560 integrated circuit board. This model was chosen because of the amount of ports that it presents, as the control of the 6 stepper motors requires 24 ports. The board is powered by the USB port of the computer and its ground is short-circuited to the ground of the power supply to avoid malfunctioning.

3 Control Scheme

The nonlinear nature of tensegrity structures makes the control of these systems a challenging task, and several different nonlinear schemes have been formulated, e.g. [17]. What is common for most of them is the use of a kinetic model, that requires the previous knowledge of physical properties such as stiffness and mass of the materials. To overcome this issue, we use differential geometry to obtain an open loop control strategy based on a kinematic model of tensegrity structures. The control system calculates a manifold of infinitesimal nodal displacements [18] along which the structure can move while maintaining its self-equilibrium, beginning from a known initial stable position. The changes in length of some of the m members are predefined while the length of other members is defined as fixed.

The complete kinematic model of a tensegrity structure with any number of degrees of freedom can be described by

$$\mathbf{A}(p, t)\,\dot{\mathbf{p}} = \boldsymbol{\delta} \tag{1}$$

where $\mathbf{A}$ is the $m \times m$ compatibility matrix that relates the initial and final nodes that compose any member, and $\boldsymbol{\delta}$ is the $m \times 1$ vector that contains the constant rates at which each of these distances between nodes change; $\dot{\mathbf{p}}$ is the $m \times 1$ vector of nodal velocities (x, y, z coordinates of each one of the nine free nodes).

The resulting system of first-order ordinary differential equations is programmed in MATLAB R20016b to be solved with the function ode45. This function uses the Runge Kutta method for estimating the solution of nonstiff ODEs [19]. This state-space representation requires the constant reformulation of A because it depends on the nodal coordinates at any given moment t.

After defining the cables to be actuated, this algorithm calculates the nodal coordinates for the new stable position. These coordinates are used to determine the new length of the cables, and the number and direction of steps for each stepper motor to achieve these lengths. Fig. 2 shows a graphic representation generated in MATLAB of the position of the nodes, and the change in length of all the cables. The numerical process to obtain this simulation takes 4.76 seconds; however, the generation of the graphical representation required almost half of this time.

With the required motion for each motor defined in the previous step, the desired routine is programmed in the Arduino IDE interphase. The native library included in Arduino is not able to control six stepper motors at the same time, so we use the library AccelStepper [20]. This library allows to control up to ten motors by first configuring the motion parameters for each stepper motor and then initializing the motion. The maximum speed for the stepper motors is defined as 300 steps per second, but the library computes the speed for each motor, so they all achieve the desired position at the same time.

The reason for applying this kind of linear motion to motors is the nature of the tensegrity structure itself. It is imperative that all the actuated cables reach the final length at, exactly, the same moment, otherwise some cables could be slack, resulting in the instability or collapse of the structure.

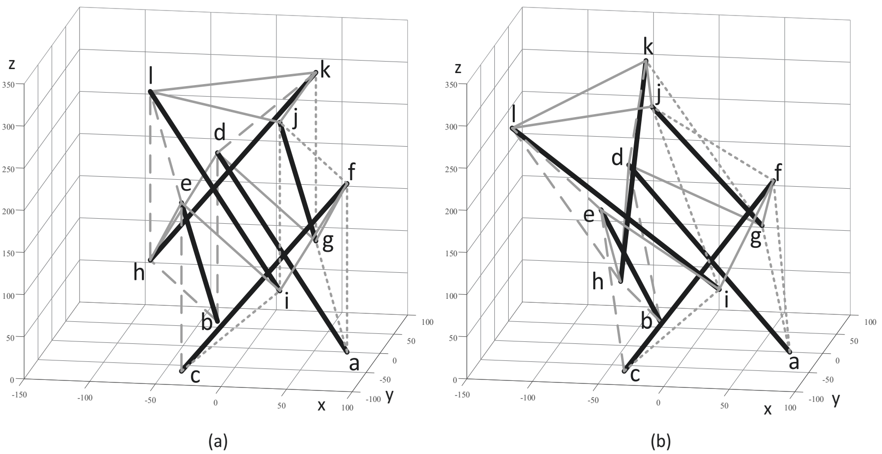


Fig. 2. Simulation of the bending of the structure. (a) original position, (b) after the form-finding process. Dotted lines represent the lengthening cables; dashed lines, the shortening cables; continuous lines, the cables that maintain the length. Thick lines represent the bars.

A final consideration is the use of an open-loop system. An advantage of the force density method, see [1], basis of the kinematic model here used, is the possibility to know the axial forces of the members at any given moment. If the maximum value of tension in the actuated cables is known, the stepper motors can be carefully sized in torque and speed to allow operation without slipping; if a smooth operation is assured, the feedback component can be omitted. The final control diagram is shown in Fig. 3.

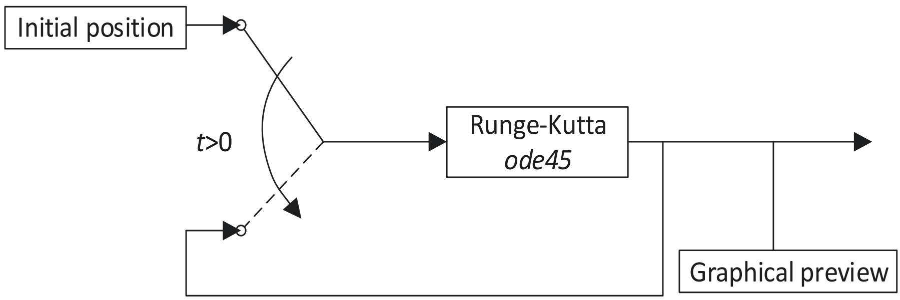


Fig. 3. Block diagram of the open-loop control system.

4 Experimental Results

4.1 One-axis Bending

To validate the physical components of the structure and the control scheme, we programed two motion routines. The first motion routine consists in bending the whole structure around the imaginary axis *bc* formed by the bottom nodes *b* and *c*, and then return it to the original straight position. To achieve this motion, some cables are lengthened while other cables are shortened, as defined in Fig. 2. After 2.5 seconds, the structure reaches the maximum displacement. At this point, the cables *af*, *gk* and *ij* are lengthened 9.52 mm, while the cables *bd*, *ce* and *hl* are shortened 5.13 mm. The displacement produced by these changes of length is shown in Fig. 4, where the red dots represent the location of the top nodes; the black lines represent a 5 mm × 5 mm grid.

Even though the Arduino library calculates the speed of each motor, so they all reach their final position at the same time, a pause is included in the code so the motors lock after each segment, to allow time for all the motors to be in position. After this pause of 1 second, the stepper motors do the inverse movement to return the structure to the initial position. The system remains stable during the whole process, and even during the pause before inverting the movement, no significative vibration is observed, and no slack cable is present.

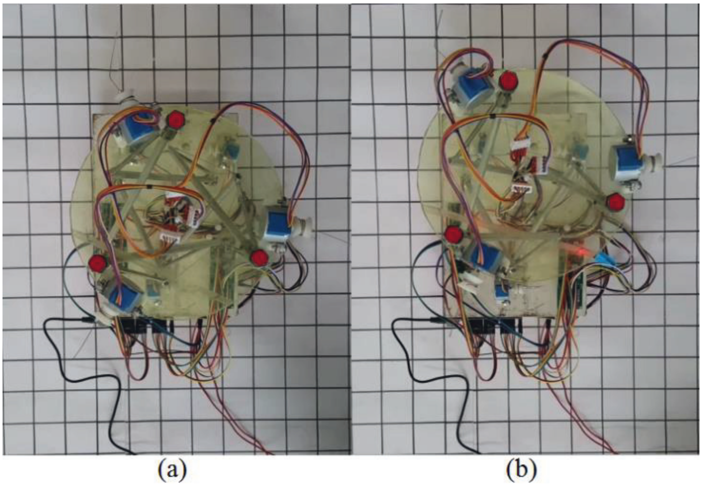


Fig. 4. Top view of the structure. (a) original position, (b) bent position. Red dots represent the position of the top nodes.

Fig. 5 displays the axial forces in each member, obtained from the numerical solution, for the initial and final positions. The orange lines that represent the forces in the initial position show that the different groups of members present the same force value thanks to the symmetry properties of the structure [21]. As expected, at the maximum displacement, the distribution of forces is different, with some tensile members under a bigger load than others; however, it is important to notice that all the members, tensile and compressive, remain loaded, and under the correct type of stress.

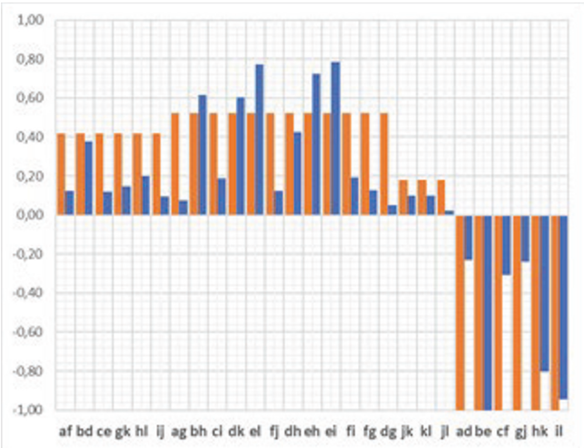


Fig. 5. Axial forces of the members of the system. Forces are normalized with respect to the maximum compression.

During the experimental run, no slack cables were found. This result validates the sign of the values of force densities obtained from the form finding algorithm; the run also confirms that the control system and mechanical components work properly.

4.2 Circuit Motion

The second routine moves the structure between three stable positions and then returns it to the initial configuration. The purpose of this experiment is to validate that the mechanism can move between different positions and maintain the equilibrium; at the same time, to verify that all the components work properly under routines involving different motion for each component.

Table 1. Length changes of actuated cables.

Cable	Position			
	1 (mm)	2 (mm)	3 (mm)	4 (mm)
af	9.52	-14.65	0.00	14.65
bd	-5.13	14.65	-14.65	0.00
ce	-5.13	0.00	14.65	-14.65
gk	9.52	0.00	-14.65	14.65
hl	-5.13	14.65	0.00	-14.65
ij	9.52	-14.65	14.65	0.00

The first of these positions coincides with the position achieved before, a bending around the axis *bc*; the next location corresponds to a bending around the axis *ca*, and the third place is produced by a bending around the axis *ab*. With the nodal coordinates obtained from the form finding process for the three different positions (pos1, pos2, pos3), the cable lengths for each configuration are determined, and used to calculate the changes in length between the four segments of the routine (pos1-pos2, pos2-pos3, pos3-pos4, pos4-pos1). These values, shown in Table 1, are used to program the motion for the corresponding stepper motors.

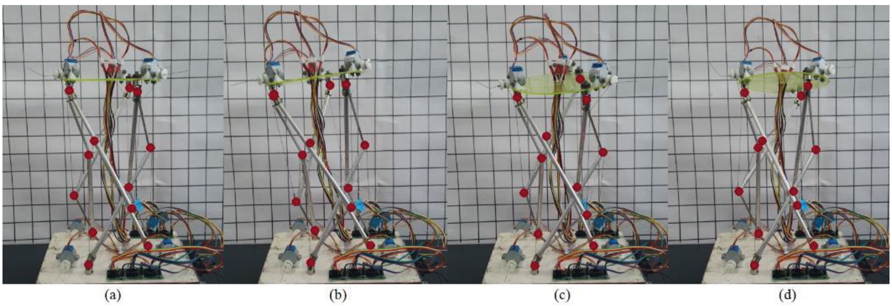


Fig. 6. Four different positions of the structure. (a) initial position, (b) bent around bc (c) bent around ca, (d) bent around ab. The location of the nodes is represented by red dots.

Fig. 6-a shows the structure at the resting position, being this the initial and final configurations. From Fig. 6-b, it can be seen how the system bends to the left, as cables *af*, *gk* and *ij* lengthen. In Fig. 6-c, the top surface seems to bend towards the reader, as it rotates around the axis *ca*. The opposite effect is noticed in Fig. 6-d, where the top surface seems to move away from the reader, because the structure is rotating around the axis *ab*. This whole routine takes a total time of 12.4 seconds.

5 Conclusion

In this paper we presented a new application for tensegrities in soft robotics. The structure consists of a two-stage tensegrity with some of its cables actuated by stepper motors and springs, the control process is carried out by the integrated circuit Arduino Mega 2560 and MATLAB. It is demonstrated how, by actuating some of the cables, a tensegrity structure can deform sideways to a new stable position.

With further research, and thanks to the properties of lightweight and axial force distribution, this kind of structure can be suitable for biomechanical applications in the form of joints such as legs or arms. Our future work will include the effect of external forces on the structure.

Acknowledgements

This paper is funded by the International Exchange Program of Harbin Engineering University for Innovation-Oriented Talents Cultivation and National Natural Science Foundation of China (NSFC) 51605111, 51675114.

References

1. Motro, R.: Tensegrity: Structural Systems for the Future. 1st edn. Kogan Page Science, London (2003).
2. Ingber, D.: Tensegrity I. Cell structure and hierarchical systems biology. Journal of Cell Science (11), 1157-1173 (2003).
3. Castro Arenas, C., Ghersi, I., Miralles, M.: Biomechanics and Biotensegrity: Study Method and Frequency Response of the Simplex and 3-bar-SVD Tensegrity Configurations. Journal of Physics: Conference Series (705), 1-10 (2016).
4. Silva, P., Fonseca, S., Turvey, M.: Is tensegrity the functional architecture of the equilibrium point hypothesis? Motor Control (14), 35-40 (2010).
5. Scarr, G.: A consideration of the elbow as a tensegrity. International Journal of Osteopathic Medicine (15), 53-65 (2012).
6. Tibert, G.: Deployable tensegrity structures for space applications. Ph.D. dissertation, Department of Mechanics, Royal Institute of Technology, Stockholm (2002).
7. Zolesi, V. S., Ganga, P. L, Scolamiero, L., Micheletti, A., Podio-Guidugli, P., Tibert, G. et al. On an innovative deployment concept for large space structures. In: 42nd International Conference on Environmental Systems, San Diego (2012).

8. Tibert, G., Pellegrino, S. Deployable tensegrity reflectors for small satellites. Journal of Spacecraft and Rockets (39), 701-709 (2002).
9. Rhode-Barbarigos, L.: An active deployable tensegrity structure. Ph.D. dissertation, Faculty of Natural Environment, Architecture and Construction, École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland (2012).
10. Caluwaerts, K., Despraz, J., Iscen, A., Sabelhaus, A., Bruce, J., Schrauwen, B., et al.: Design and control of compliant tensegrity robots through simulation and hardware validation. Journal of the Royal Society Interface (11), (2014).
11. Koizumi, Y., Shibata, M., Hirai, S. Rolling Tensegrity Driven by Pneumatic Soft Actuators. In: IEEE International Conference on Robotics and Automation, pp. 1988-1993. Minnesota (2012).
12. Ushigome, Y., Niiyama, R., Nishimura, K., Tanikawa, T., Hirose, M. Archi/e machina: Interactive architecture based on tensegrity. In: 16th International Conference on Virtual Systems and Multimedia, pp. 55-62. Seoul, Korea (2010).
13. Baltaxe-Admony, L., Robbins, A., Jung, E., Lessard, S., Teodorescu, M., SunSpiral, V., et al. Simulating the human shoulder through active tensegrity structures. In: Proc. ASME Int. Design Eng. Tech. Conf., Charlotte (2016).
14. Micheletti, A., Williams, W.: A marching procedure for form-finding for tensegrity structures. Journal of Mechanics of Materials and Structures (2), 857-882 (2007).
15. Skelton, R., De Oliveira, M.: Tensegrity Systems. 1st edn. Springer, New York (2009).
16. Sabelhaus, A., Bruce, J., Caluwaerts, K., Chen, Y., Lu, D., Liu, Y., et al. Hardware design and testing of SUPERball, a modular tensegrity robot. In: 6th World Conference of the International Association for Structural Control and Monitoring, Barcelona (2014).
17. Sultan, C.: Tensegrity deployment using infinitesimal mechanisms. International Journal of Solids and Structures (51), 3653-3668 (2014).
18. Robbin, J., Salamon, D.: Introduction to differential geometry. ETH Zurich, Zurich (2018).
19. Shampine, L., Gladwell, I., Thompson, S.: Solving ODEs with Matlab. Cambridge University Press, New York (2003).
20. McCauley, M. (2018). AccelStepper: AccelStepper library for Arduino, http://www.airspayce.com/mikem/arduino/AccelStepper/, last accessed 2018/06/05.
21. Zhang, J., Guest, S., Ohsaki, M.: Symmetric prismatic tensegrity structures: Part I. Configuration and stability. International Journal of Solids and Structures (46), 1-14 (2009).