A Better Way to Construct Tensegrities: Planar Embeddings Inform Tensegrity Assembly

Elizabeth Ricci John Rieffel (Advisor)

Computer Science Department, Union College

Introduction, Motivation, and Background

A tensegrity structure is composed of rigid struts under compression and tensile springs under tension. Originally studied through sculpture, tensegrities are of interest to the robotics community due to their strength, deformability and equilibrium form [2]. Due to these properties, several tensegrity-based robots have been designed, but there are a lack of robots designed using large or irregular tensegrity structures[1][3][4] [5][8] [10][11]. This is not due to a lack of larger, more-complex tensegrity structures, as Rieffel et al. have developed an evolutionary algorithm to represent and evolve tensegrity structures [9].

Due to the nature of tensegrity structures, it is difficult to construct them by hand using the naive approach, and while Liu et al. recently outlined how tensegrities can be constructed using shape memory polymers, we still lack a simple and cost-efficient way to construct arbitrary tensegrity structures [6]. We hypothesize that if there exists a better way to construct tensegrity structures it will encourage the use of more complex tensegrity structures in robotics.

As a basis for our concept, we consider that a given tensegrity can be deformed such that all of its struts are in a planar form, allowing for overlaps. If released, the tensegrity will revert back to its original, equilibrium state. We use this property to our benefit and detail a method for constructing tensegrities by first connecting the springs and then separating the struts. We consider a flat-packed model of a tensegrity that contains an arrangement of struts and interwoven pattern of springs.

Question

To what extent can we develop an algorithm, given a tensegrity structure and some constraints, that can be used to produce a flat-packed model of that tensegrity which meets the given constraints and will revert back to its equilibrium state upon release?

Methods

I have developed a black-box that takes a tensegrity and returns and graph and the overlap information necessary to physical construct it using our method. Figure 1 shows a high level overview of the computational process.

Tensegrity Attributes

3D Model: Open Dynamics Engine (ODE)

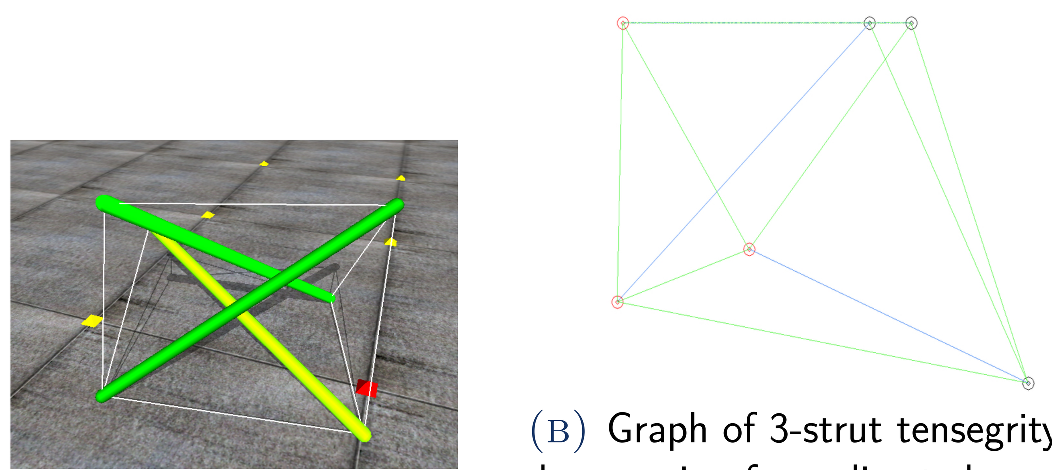
2D Model: Force Directed Graph Drawing

Flat-packed form of Tensegrity

Physical Tensegrity

Figure 1: Black-Box Steps.

We begin with the attributes of our given tensegrity, this is the number of struts, the number of springs and an array containing the information about the connections present. Next the tensegrity attributes are used to model the tensegrity in a physics simulator, ODE, Figure 2a. This model provides us with 3D coordinates of each strut endpoint. Next the tensegrity is modeled as a graph and a force directed graph drawing program is applied in an attempt to determine the best drawing of the graph, see figure 2b. At this point all of the intersects present in the graph are identified and the ordering of the elements involved is determined using the 3D coordinates from the 3Dd model. This information, in addition to the graph, form the flat-pack. The final step is to physically construct the tensegrity based on the flat-packed form, Figure 3.



(a) A 3-strut tensegrity simulated in ODE.

(b) Graph of 3-strut tensegrity drawn using force-directed graph drawing. The blue lines are struts and the green lines are springs. Note: One strut is overlapping with a spring.

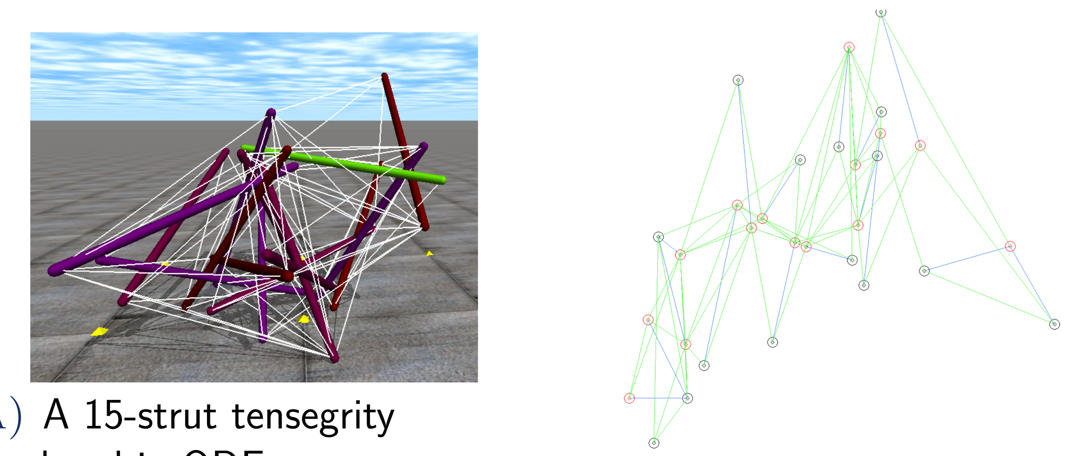
Figure 2: Models of the 3-strut tensegrity.

(a) Lasercut struts. (b) Struts with springs attached. (c) Final, equilibrium form.

Figure 3: Physical construction process.

Results

We have successfully developed and implemented an algorithm that given a tensegrity produces a flat-pack model of that tensegrity. The algorithm is robust enough to work on any tensegrity, although we are not guaranteed to generate a flat-packed form that is feasible to physically construct. Figure 4 shows outputs of the 3D and 2D modeling and reinforces that the process we have developed is able to produce results for any tensegrity, size and complexity are not limiting factors. This being said, we have not physically verified this process on a tensegrity containing more than 4 struts.



(a) A 15-strut tensegrity simulated in ODE.

(b) Graph of 15-strut tensegrity.

Figure 4: Models of the 15-strut tensegrity.

Future Work

Our future work includes improving the 2D graph drawing process to generate better drawing and to consider more limitations, such as maximum spring length. We would also like to write a program to translate a graph of struts to a file that can be used to lasercut the struts in the desired formation.This will enable us to physically test our findings more easily, and thus identify further physical constraints. Finally we consider the use of alternative materials to avoid attaching springs by hand.

Acknowledgments

I would like to thank the college for supporting this research through a SRG grant and Union's Evolutionary Robotics Lab.

References

[1] Ken Caluwaerts, Jérémie Despraz, Atıl Işçen, Andrew P Sabelhaus, Jonathan Bruce, Benjamin Schrauwen, and Vytas SunSpiral. Design and control of compliant tensegrity robots through simulation and hardware validation. *Journal of The Royal Society Interface*, 11(98):20140520, 2014.

[2] Robert Connelly and Allen Back. Mathematics and tensegrity: Group and representation theory make it possible to form a complete catalogue of "strut-cable" constructions with prescribed symmetries. *American Scientist*, 86(2):142–151, 1998.

[3] Jeffrey Friesen, Alexandra Pogue, Thomas Bewley, Mauricio de Oliveira, Robert Skelton, and Vytas Sunspiral. Ducutt: A tensegrity robot for exploring duct systems. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pages 4222–4228. IEEE, 2014.

[4] Mark Khazanov, Ben Humphreys, William D Keat, and John Rieffel. Exploiting dynamical complexity in a physical tensegrity robot to achieve locomotion. In *ECAL*, pages 965–972. Citeseer, 2013.

[5] Yuusuke Koizumi, Mizuho Shibata, and Shinichi Hirai. Rolling tensegrity driven by pneumatic soft actuators. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 1988–1993. IEEE, 2012.

[6] Ke Liu, Jiangtao Wu, Glaucio H Paulino, and H Jerry Qi. Programmable deployment of tensegrity structures by stimulus-responsive polymers. *Scientific Reports*, 7(1):3511, 2017.

[7] Michael J McGuffin. Simple algorithms for network visualization: A tutorial. *Tsinghua Science and Technology*, 17(4):383–398, 2012.

[8] Chandana Paul, Francisco J Valero-Cuevas, and Hod Lipson. Design and control of tensegrity robots for locomotion. *IEEE Transactions on Robotics*, 22(5):944–957, 2006.

[9] John Rieffel, Francisco Valero-Cuevas, and Hod Lipson. Automated discovery and optimization of large irregular tensegrity structures. *Computers & Structures*, 87(5-6):368 – 379, 2009.

[10] Andrew P Sabelhaus, Jonathan Bruce, Ken Caluwaerts, Pavlo Manovi, Roya Fallah Firoozi, Sarah Dobi, Alice M Agogino, and Vytas SunSpiral. System design and locomotion of superball, an untethered tensegrity robot. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pages 2867–2873. IEEE, 2015.

[11] Mizuho Shibata, Fumio Saijyo, and Shinichi Hirai. Crawling by body deformation of tensegrity structure robots. In *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, pages 4375–4380. IEEE, 2009.