Seed-and-Prune Approach for Rapid Discovery of Tensegrity-Like Structures of the Desired Shape

Sergei Savin

Seed-and-Prune Approach for Rapid Discovery of Tensegrity-like Structures of the Desired Shape*

Sergei Savin[1][0000−0001−7954−3144]

Innopolis University, Innopolis, 420500, Russia s.savin@innopolis.ru

Abstract. In this paper, a new tensegrity generation method, based on a combination of a random search, quadratic programming and connection pruning is proposed. The method exploits the structure of the static equilibrium equations of tensegrity structures with static nodes, allowing to form linear equality constraints. By abandoning the requirement of struts being disconnected we arrive at a simple convex program, where the existence of solution represents the existence of the sought structure. We propose a way to generate node positions and to prune the connections in order to shape the resulting tensegrity-like structures in the desired form.

Keywords: Generation of tensegrity structures · Force density method · Convex optimization

1 Introduction

Tensegrity structures are often defined as a collection of structural elements, where each element experiences axial forces: compression or tension. Elements experiencing only tension can be referred to as cables, while compressed elements are called struts [5]. These structures are of interest for a number of applications, but especially in robotics, where their properties are highly desirable. Those properties include folding, a good stiffness-to-mass ratio, collision resilience, and many others [2, 13]. Tensegrity structures have already been proposed for planetary exploration, underwater robotics, humanoid robotics, collaborative and other applications [1, 7, 9, 10].

Design of tensegrity structures is an especially interesting issue. The design process is significantly from the traditional engineering, allowing less freedom in "adding together" known structural elements, and instead requiring a generation of the entire structure, explicitly and simultaneously taking into account force balance and position of elements withing the structure. This process if often referred to as form finding [8, 12].

Form finding, and a related forward and inverse kinematics problem (finding stable element positions for the given element lengths, or finding element

* The research is supported by grant of the Russian Science Foundation (project No:19-79-10246)

lengths to achieve a given element position) can be of interest for a number of design-related tasks. One of those tasks is generation of a customized tensegrity structures. Those can be structures with a given shape, given number of elements, given limits on the tensile or compressive forces, etc. We will refer to it as a *form-finding design problem*. Existing methods already allow us to approach this problem. In this paper we show that with a new formulation, we can radically speed-up parts of the computational process, and facilitate the use of random search as a part of tensegrity design process. We use this methodology as a basis to build a pipeline for generation of shape-specific tensegrity structures, accounting for limits in terms of tensile and compressive forces, number of cables and connectivity patterns. Our main contributions are:

- Convex representation of a form-finding design problem with fixed nodes position, with a natural extension to include constrained modes and external forces.
- Seed-and-prune algorithm for generating connectivity patterns in accordance with constraints on the number of structural element, and on the relative connections between struts.
- A method for rapidly generating tensegrity structures of a specific shape, taking into account limits on the tensile or compressive forces.

The rest if the paper is organized as follows: Section 2 gives a description of the state of the art, section 3 provides a description of static equilibrium conditions of the tensegrity structure, written as a linear constraint, and section 4 presents seed-and-prune algorithm. Finally, section 5 demonstrates an example of the generated structure and its connectivity matrix.

2 State of the Art

Form finding for tensegrity structures has long become a diverse field of study with a number of sub-domains. There are works on general conditions of stability of tensegrity structures of certain class, such as [3] where group symmetry was used to define conditions for stability of tensegrity structures with dihedral symmetry. Another domain is building algorithms for finding connectivity patterns and the lengths of the structural elements for which tensegrity structure can remain stable. That problem is generally referred to as "form finding" [11], making it difficult to address other sub-domains in the field. Another set of problems can be presented as forward and inverse position problems; in the first, the task is to find stable position of the nodes while cable and strut lengths are known, and in the second the task is to find lengths of either cables or struts or both or of a subset of those [16]. Similar forward and inverse problems can be formulated for stiffness design and static force distribution for a given pose of the robot [14,15].

Form finding has been addressed a number of times before. A classification of methods provided in [17] includes two major categories: kinematic and static methods, different in the choice of variables; first category permits the strut lengths to change, while the second does the same for the cable lengths. Reviewed methods include two special-case analytical solutions, one method relying on a non-convex optimization, two equivalent force density methods and a dynamics simulation-like method. Force density methods are of particular interest, since they yield an algebraic formulation of the problem.

Paper [6] introduced a way to generate tensegrity structures in a systematic way by solving a mixed-integer convex program. This method has a number of advantages, including the ability to limit the connections between the struts, following the definition of a tensegrity structure as a set of disconnected compressive elements (struts); however, if such a requirement is not of necessity, then there is a possibility to significantly simplify the generation process by eliminating integer variables from the problem design. Mixed-integer convex programs are solved using branch-and-bound methods, which limits the possibility of predicting the the computational load and termination time of the algorithm. This may lead to the imitations to the size of the problems tackled by such algorithm.

In this paper it is proposed to use a quadratic programming-based approach to tensegrity generation, abandoning the requirements for the discontinuity of the set of struts, but compensating for that in the ability to solve the generation problems for large numbers of nodes, and use a pruning method for shaping the resulting tensegrity structure.

3 Static equilibrium of a tensegrity structure as a constraint

We will model tensegrity structure as a set of nodes $\mathbf{r}_i$, where $1 \leq i \leq n$. In order to arrive at a description of the static equilibrium of a tensegrity structure as a linear constraint, we need to consider direction matrices $\mathbf{D}_i$ that indicate from which directions can the forces act on the node $\mathbf{r}_i$:

$$\mathbf{D}_i = \left[(\mathbf{r}_1 - \mathbf{r}_i),\, (\mathbf{r}_2 - \mathbf{r}_i),\, ...,\, (\mathbf{r}_n - \mathbf{r}_i) \right] \tag{1}$$

With that, we can formulate the static equilibrium condition as a question of the existence of the vectors $\mathbf{f}_i$ that deliver equality to the constraints:

$$\mathbf{D}_i \mathbf{f}_i = \mathbf{f}_i^{ext} \tag{2}$$

where $\mathbf{f}_i^{ext}$ is the sum of external forces acting n the node. Assuming that $\mathbf{F} = \left[\mathbf{f}_1, \mathbf{f}_2, ..., \mathbf{f}_n \right]$, we can add an additional constraint, respecting the fact that action equal reaction with the opposite sign:

$$\mathbf{F} = \mathbf{F}^\top \tag{3}$$

where the opposite sign should be encoded in the matrix (1).

Additionally, we note that some nodes do not have a connection between them. That can be integrated into the constraint as:

$$\mathbf{D}_i = \left[(\mathbf{r}_1 - \mathbf{r}_i)c_{1,i},\, (\mathbf{r}_2 - \mathbf{r}_i)c_{2,i},\, ...,\, (\mathbf{r}_n - \mathbf{r}_i)c_{n,i} \right] \tag{4}$$

where $c_{j,i}$ is a binary variable, which equals 0 if there is no connection between the nodes, and 1 otherwise. The matrix formed by there variables is called connectivity matrix, and is denoted as $\mathbf{C}$.

Together expressions (2) and (3) form constraints for a convex optimization problem. If the problem has a solution, the structure with the given nodes $\mathbf{r}_i$ is possible, and the sign of $f_{j,i}$ (elements of $\mathbf{F}$) determines whether each particular connection is a strut or a cable.

In order to chose some of the nodes as fixed (which can represents connective elements of a mounted structure), an additional decision variable can be added, acting as a reaction force. Assuming first w elements are fixed, the equation(2) for them becomes:

$$\mathbf{D}_i \mathbf{f}_i = \mathbf{f}_i^{ext} + \lambda_i, \; 1 \leq i \leq w \tag{5}$$

where λ_i is the reaction force implementing the imposed constraints. We should note that this type of additional constraints actually is a relaxation of the problem, as it increases its domain; it also provides the possibility to reduce the number of decision variables during a resolve stage, if a solver wrapper like CVX is used to automatically re-shape the problem [4].

4 Seed-and-prune algorithm

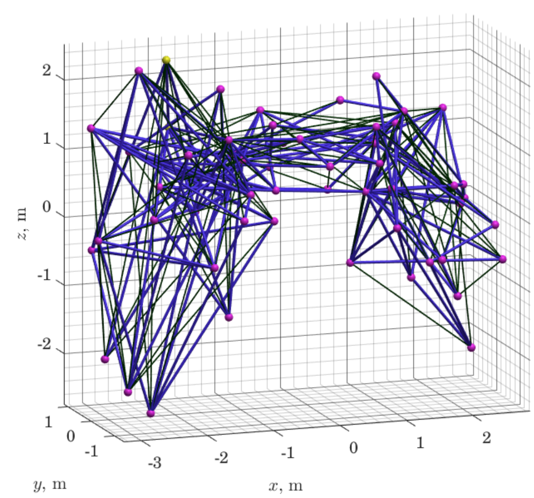

Fig. 1: Example of a tensegrity-like structure, generated using proposed method; the nodes are shown in purple, cables are shown in black and struts are shown in blue

Given m areas described by ellipsoids $\mathcal{E}_k = \{\mathbf{x} : \ ||\mathbf{E}_k \mathbf{x}|| \leq 1\}$, chosen such that their union represents a connected set, we generate P nodes, randomly distributed in the union of the ellipsoids. This stage represents *seeding*.

If a line segment between two points lies entirely in one of the ellipsoids, we set the connection between the points as active $c_{j,i} = 1$, otherwise it is set to 0:

$$c_{j,i} = \begin{cases} 1 & \text{if } \exists k, \ ||\mathbf{E}_k \mathbf{r}_i|| \leq 1, \ ||\mathbf{E}_k \mathbf{r}_j|| \leq 1 \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

Next stage is pruning. Given number of connections p to prune, we chose p unique (irrespective of permutations) pairs of numbers (v_1, v_2), and set corresponding connections to zero:

$$c_{v_1(j), v_2(j)} = 0, \quad c_{v_2(j), v_1(j)} = 0, \quad j = 1, ..., p \tag{7}$$

Choosing the number of connections to prune we can control the complexity of the resulting structure.

After pruning we can formulate and solve quadratic program in order to find if the requested tensegrity structure is feasible:

$$\begin{aligned} \underset{\mathbf{x}}{\text{minimize}} \quad & ||\mathbf{F}|| \\ \text{subject to} \quad & \begin{cases} \Big[(\mathbf{r}_1 - \mathbf{r}_i) c_{1,i}, \, (\mathbf{r}_2 - \mathbf{r}_i) c_{2,i}, \, ..., \, (\mathbf{r}_n - \mathbf{r}_i) c_{n,i} \Big] \, \mathbf{f}_i = \mathbf{f}_i^{ext} \\ \mathbf{F} = \mathbf{F}^\top \end{cases} \end{aligned} \tag{8}$$

If there is no solution, the previous steps can be repeated for a different set of randomly distributed nodes.

5 Resulting structures

In this section we present example of a structure that can be discovered using proposed method. It is obtained using three ellipsoids, forming a *Π*-shaped structure. Fig. 1 shows an example of resulting structure. The elements that experience compressive loads are shown in blue, the elements that experience tensile loads are drawn as black lines. Nodes are drawn in red, except for fixed nodes that are yellow.

We can also demonstrate the connectivity matrix for the found structure. Fig. 2 shows the color map for the found structure. We can see blocks on the diagonal, that represent connections inside each of the three ellipsoids mentioned previously. We can notice that there are interconnections between the ellipsoids, represented by the the off-diagonal connections. This example did not include pruning, allowing to see dense number of connection formed inside each ellipsoid. This density is well-represented in the shape of the resulting structure, with a numerous elastic elements running close to one another, which in practice means a low volume work space, as the densely packed cables and struts will likely intersect as the structure deforms.

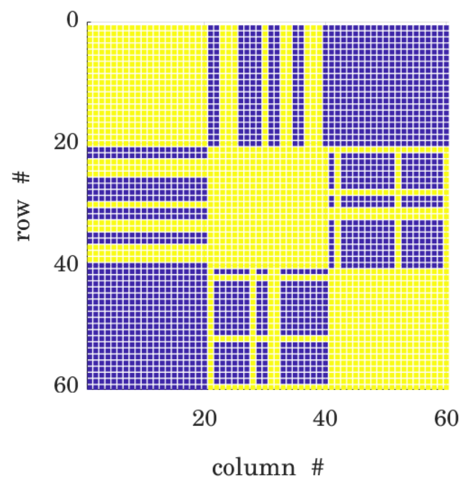

Fig. 2: Color map for the connectivity matrix of the tensegrity-like structure, generated using proposed method; blue correspond to no connection, yellow to a connection

6 Conclusions

In this work it was demonstrated that tensegrity generation problem can be formulated as a random search problem, solving a single quadratic program in order to identify the existence of the structure. The main advantage of the method is the low computational cost associated with solving quadratic programs, allowing for fast experimentation with the structure generation. The main downsides are the lack of constraint on the connectivity patterns, notably on the disconnectedness of the compressed elements; this is the result of the trade-off made to rid the problem of the integer variables, necessary to encode such a constraint. Also, same as other tensegrity topology generation methods shown in the literature, this one is based around the requirements of the existence of a static equilibrium under a possible pre-stress; however, there is also a separate issue of the stiffness of the resulting system, which needs not only to be sufficient for the chosen application, but also to allow the structure to avoid collapse under bounded external forces. It is of further interest to investigate stiffness-aware tensegrity generation, where structures that do not exhibit sufficient stiffness properties are discarded.

References

1. Bliss, T., Iwasaki, T., Bart-Smith, H.: Central pattern generator control of a tensegrity swimmer. IEEE/ASME Transactions on Mechatronics **18**(2), 586–597 (2012)

2. Caluwaerts, K., Despraz, J., Işçen, A., Sabelhaus, A.P., Bruce, J., Schrauwen, B., SunSpiral, V.: Design and control of compliant tensegrity robots through simulation and hardware validation. Journal of the royal society interface **11**(98), 20140520 (2014)
3. Connelly, R., Terrell, M.: Globally rigid symmetric tensegrities. Structural Topology 1995 núm 21 (1995)
4. Grant, M., Boyd, S., Ye, Y.: Cvx: Matlab software for disciplined convex programming (2008)
5. Guest, S.D.: The stiffness of tensegrity structures. IMA Journal of Applied Mathematics **76**(1), 57–66 (2011)
6. Kanno, Y.: Exploring new tensegrity structures via mixed integer programming. Structural and Multidisciplinary Optimization **48**, 95–114 (2013)
7. Kim, K., Agogino, A.K., Moon, D., Taneja, L., Toghyan, A., Dehghani, B., SunSpiral, V., Agogino, A.M.: Rapid prototyping design and control of tensegrity soft robot for locomotion. In: 2014 IEEE International Conference on Robotics and Biomimetics (ROBIO 2014). pp. 7–14. IEEE (2014)
8. Kim, K., Agogino, A.K., Toghyan, A., Moon, D., Taneja, L., Agogino, A.M.: Robust learning of tensegrity robot control for locomotion through form-finding. In: 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 5824–5831. IEEE (2015)
9. Lessard, S., Bruce, J., Jung, E., Teodorescu, M., SunSpiral, V., Agogino, A.: A lightweight, multi-axis compliant tensegrity joint. In: 2016 IEEE International Conference on Robotics and Automation (ICRA). pp. 630–635. IEEE (2016)
10. Lessard, S., Castro, D., Asper, W., Chopra, S.D., Baltaxe-Admony, L.B., Teodorescu, M., SunSpiral, V., Agogino, A.: A bio-inspired tensegrity manipulator with multi-dof, structurally compliant joints. In: 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 5515–5520. IEEE (2016)
11. Paul, C., Lipson, H., Cuevas, F.J.V.: Evolutionary form-finding of tensegrity structures. In: Proceedings of the 7th annual conference on Genetic and evolutionary computation. pp. 3–10 (2005)
12. Paul, C., Roberts, J.W., Lipson, H., Cuevas, F.V.: Gait production in a tensegrity based robot. In: ICAR'05. Proceedings., 12th International Conference on Advanced Robotics, 2005. pp. 216–222. IEEE (2005)
13. Paul, C., Valero-Cuevas, F.J., Lipson, H.: Design and control of tensegrity robots for locomotion. IEEE Transactions on Robotics **22**(5), 944–957 (2006)
14. Sabelhaus, A.P., Li, A.H., Sover, K.A., Madden, J., Barkan, A., Agogino, A.K., Agogino, A.M.: Inverse statics optimization for compound tensegrity robots. arXiv preprint arXiv:1808.08252 (2018)
15. Savin, S., Balakhnov, O., Klimchik, A.: Convex optimization-based stiffness control for tensegrity robotic structures. In: 2020 28th Mediterranean Conference on Control and Automation (MED). pp. 990–995. IEEE (2020)
16. Savin, S., Balakhnov, O., Maloletov, A.: Linearization-based forward kinematic algorithm for tensegrity structures with compressible struts. In: Proceedings of 15th International Conference on Electromechanics and Robotics" Zavalishin's Readings". pp. 293–303. Springer (2020)
17. Tibert, A., Pellegrino, S.: Review of form-finding methods for tensegrity structures. International Journal of Space Structures **18**(4), 209–223 (2003)