NTNU
Kunnskap for en bedre verden

Department of Mathematical Sciences

TMA4180 - Optimization 1

Form-finding of Tensegrity Structures Using an Optimization Based Model

Authors:
Aynalem Baymot Tamir
Peder Brekke
Jose Tomas Ulvik
Espen Bjørge Urheim

21.04.2023

Table of Contents

List of Figures

1 Abstract

Tensegrity structures are mechanical structures consisting of bars and cables connected in joints, that together guarantee the stability of the structure. Originally they were introduced as artworks, but are now becoming more prevalent in engineering as they allow for lightweight constructions. They have for example already been used in construction of several structures such as towers, bridges, stadium roofs and more.

Our focus is looking at an optimization based model for finding the form of a tensegrity structure given a configuration of bars and cables. We will find the form of different configurations using methods from numerical optimization.

2 Introduction

In this paper we seek to use mathematical models of tensegrity structures in different numerical optimization algorithms to find stable structures. In attempts of doing so, we first establish the mathematical framework used throughout the paper and some reasonable constraints for our problem, as well as deriving some necessary results.

In general, the structures we will look at are built out of two different elements; cables and bars. Connecting these elements with joints, we can model tensegrity structures. Depending on the stress applied to these elements, one may increase or decrease their internal energy. Using mathematical models for these internal energies in combination with potential energy of elements and weighted nodes, we can model the total energy of a tensegrity structure.

If we consider structures only containing of cables and weighted joints, we showed that the resulting cost function is convex. In contrast, all tensegrity structures containing bars was shown to be a non-convex problem. Furthermore, convexity of the cost function will affect the suitability of different numerical algorithms. The main algorithms used are the Conjugate Gradient-Method (CG) and the Broyden–Fletcher–Goldfarb–Shanno (BFGS) algorithm. We found that using BFGS in the convexity case and CG with Polak-Ribiére+ update for the non-convex case were the most effective methods. In general BFGS seemed to have the fastest convergence rate, but struggling to converge for non-convex cost functions. CG on the other hand could handle such problems, but in turn needs far more iterations to converge compared to BFGS.

3 Notation and Definitions

3.1 Mathematical Model of a Tensegrity-Structure

We define a tensegrity structure mathematically as a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, with nodes $\mathcal{V} = \{1, \dots, N\}$ and edges $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$. The edge from node i to node j may be a bar or a cable, which we denote $e_{ij} \in \mathcal{B}$ or $e_{ij} \in \mathcal{C}$, respectively, with $i > j$. We assume that all nodes are connected to at least one other node.

We describe the position of a node $x^{(i)}$ with Cartesian coordinates, that is $x^{(i)} = (x_1^{(i)}, x_2^{(i)}, x_3^{(i)})$. Since we will eventually look to minimize total energy with respect to the coordinates of all nodes, we concatenate the coordinates of each node into a single vector $X = (x^{(1)}, \dots, x^{(N)}) \in \mathbb{R}^{3N}$.

3.2 Energy Functions

The energy of the system can be split into the energy stored in bars, energy stored in cables and energy stored in external loads that may be put on nodes. We start by defining functions for the energy stored in bars.

Each bar has a resting length ℓ_{ij} and line density ρ. This results in the gravitational potential energy

$$E^{bar}_{grav}(e_{ij}) = \frac{\rho g \ell_{ij}}{2}(x_3^{(i)} + x_3^{(j)}),$$

where g denotes the gravitational acceleration on earth's surface. Additionally a bar may be stretched or compressed, which results in the elastic energy

$$E^{bar}_{elast}(e_{ij}) = \frac{c}{2\ell_{ij}^2}(\|x^{(i)} - x^{(j)}\| - \ell_{ij})^2.$$

Here, $c > 0$ is an elasticity constant. Hence, the total energy of a bar is the sum of its gravitational potential energy and elastic energy.

Each cable is assumed weightless, so the only energy contribution here is the elastic energy stored in a stretched cable. A cable that is "compressed" does not have any elastic energy. Similar to the bars, each cable has a resting length ℓ_{ij} and an elasticity constant $k > 0$. Using the same model as for the bars, the elastic energy function becomes

$$E^{cable}_{elast}(e_{ij}) = \begin{cases} \frac{k}{2\ell_{ij}^2}\big(\|x^{(i)} - x^{(j)}\| - \ell_{ij}\big)^2 & \text{if } \|x^{(i)} - x^{(j)}\| > \ell_{ij}, \\ 0 & \text{if } \|x^{(i)} - x^{(j)}\| \leq \ell_{ij}. \end{cases}$$

Lastly, each node may be equipped with an external load. We denote the weight on node $x^{(i)}$ as m_i, and define the total external energy as

$$E_{ext}(X) = \sum_{i=1}^{N} m_i g x_3^{(i)}.$$

Summing up all the energy contributions we find the function for total energy in a tensegrity structure,

$$E(X) = \sum_{e_{ij} \in \mathcal{B}} \big(E^{bar}_{elast}(e_{ij}) + E^{bar}_{grav}(e_{ij})\big) + \sum_{e_{ij} \in \mathcal{C}} E^{cable}_{elast}(e_{ij}) + E_{ext}(X). \tag{1}$$

3.3 Constraints

If we try to minimize (1) without any constraints, the solution would be that all nodes tend to negative infinity. To avoid this we introduce two methods of constraining the problem; the first being to fix some of the nodes in place, and the second being to implement some sort of "floor" for the tensegrity-structure to stand on. To prove that these two methods admits a solution, we need to show lower semi-continuity and coercivity of (1) subject to both constraints.

To show that lower semi-continuity holds, it is sufficient to show that the energy function defined in (1) is continuous. It is known that the norm and polynomials are continuous functions, and therefore it follows that E^{bar}_{elast}, E^{bar}_{grav} and E_{ext} are continuous. Furthermore E^{cable}_{elast} is continuous in both parts of the split by the same argument, however we need to investigate the split.

$$\lim_{\|x^i - x^j\| \to l_{i,j}^+} E^{cable}_{elast}(e_{ij}) = \lim_{\|x^i - x^j\| \to l_{ij}^+} \frac{k}{2\ell_{ij}^2}\big(\|x^{(i)} - x^{(j)}\| - \ell_{ij}\big)^2 = 0 = \lim_{\|x^i - x^j\| \to l_{ij}^-} E^{cable}_{elast}(e_{ij}).$$

Hence the energy function is continuous, and thus also lower semi-continuous.

To show coercivity we have to investigate each of the constraints by themselves. We first consider fixing some nodes in place. We observe that E_{elast}^{bar} and E_{elast}^{cable} are second order terms, and the rest is first order terms. The driving terms are these second order terms, making it sufficient to show that they go to positive infinity when the coordinates of the free nodes goes to positive or negative infinity. The second order terms contains

$$(\|x^{(i)} - x^{(j)}\| - \hat{l}_{ij})^2,$$

where $x^{(i)}, x^{(j)}$ are nodes and $\hat{l}_{ij}$ is the edge length. This expression is always positive. Since we have at least one fixed node and the graph is connected, sending the coordinates of the free nodes to positive or negative infinity means the length of at least one of the edges goes to positive infinity. This means that at least one second order term is going towards positive infinity. Since this is a driving term, the energy function is coercive under the first constraint.

The second constraint is that we have some type of floor, that is $x_3^{(i)} \geq 0$ for all i. To check coercivity, consider sending all nodes to $\pm\infty$ in the (x_1, x_2)-plane while keeping the x_3-coordinates as finite and bounded below by the constraint. Then all edge lengths will be finite, and we have effectively only shifted the system in the (x_1, x_2)-plane. In this case the energy function does not diverge, and is therefore not coercive.

4 Cable-nets

We start by looking at cable-nets, which are tensegrity structures that only contain cables and nodes with external loads. Thus, the optimization problem we wish to solve is simplified to

$$\min_X E(X) = \sum_{e_{ij} \in \mathcal{C}} E_{elast}^{cable}(e_{ij}) + E_{ext}(X). \tag{2}$$

For this part, we constrain (2) by fixing some nodes in place, so that we only solve for the coordinates of the free nodes. Before implementing an optimization algorithm, however, we must look at the energy function and its properties in this situation to gain a better idea of what algorithms that are fit to use. First, we look at smoothness and convexity.

4.1 Smoothness and Convexity

We start by assessing the smoothness of the energy function. When looking at (2), we immediately see that $E_{ext}(X)$ is twice differentiable as it is a polynomial. By differentiating the function for elastic energy of a cable with respect to $x_n^{(i)}$, $n = 1, 2, 3$, we find that

$$\frac{\partial}{\partial x_n^{(i)}} E_{elast}^{cable}(e_{ij}) = \begin{cases} \frac{k}{\ell_{ij}^2}\left(1 - \frac{\ell_{ij}}{\|x^{(i)} - x^{(j)}\|}\right)(x_n^{(i)} - x_n^{(j)}) & \text{if } \|x^{(i)} - x^{(j)}\| > \ell_{ij}, \\ 0 & \text{if } \|x^{(i)} - x^{(j)}\| \leq \ell_{ij}. \end{cases} \tag{3}$$

Looking at the intersection point $\|x^{(i)} - x^{(j)}\| = \ell_{ij}$, we find that both parts of the function are equal to zero, and since the norm is continuous, E is continuously differentiable for all i and n. Therefore the function E defined in (2) is C^1. If we differentiate again, this time with respect to $x_m^{(i)}$ with $m \neq n$, we find that

$$\frac{\partial^2}{\partial x_n^{(i)} \partial x_m^{(i)}} E_{elast}^{cable}(e_{ij}) = \begin{cases} \frac{k}{\ell_{ij}} \frac{(x_n^{(i)} - x_n^{(j)})(x_m^{(i)} - x_m^{(j)})}{\|x^{(i)} - x^{(j)}\|^3} & \text{if } \|x^{(i)} - x^{(j)}\| > \ell_{ij}, \\ 0 & \text{if } \|x^{(i)} - x^{(j)}\| \leq \ell_{ij}. \end{cases} \tag{4}$$

This means that the second derivatives of E not necessarily are continuous, as we see from (4) that continuity fails when $(x_n^{(i)}, x_m^{(i)}) \neq (x_n^{(j)}, x_m^{(j)})$ (with $m \neq n$), which definitely can happen. Thus we have proven that the energy function defined in (2) is C^1, but typically not C^2.

We now want to check if the total energy function in (2) is convex or not. Knowing that a sum of convex functions also is convex, we can check the summands in (2) independently. It is clear that $E_{ext}(X)$ is convex, as it is linear in its arguments. We therefore move to the summing term. To show convexity, we start by defining two functions $g : \mathbb{R}^3 \times \mathbb{R}^3 \to \mathbb{R}$ and $f : \mathbb{R} \to \mathbb{R}$ as

$$g(x^{(i)}, x^{(j)}) = f(\|x^{(i)} - x^{(j)}\|) \text{ , and } f(x) = \begin{cases} (x - \mu)^2 & \text{if } x > \mu, \\ 0 & \text{if } x \leq \mu. \end{cases}$$

Here μ is some positive constant. Having done so, we can now rewrite $E_{elast}^{cable}(e_{ij})$ using these functions,

$$E_{elast}^{cable}(e_{ij}) = \frac{k}{2\mu^2} g(x^{(i)}, x^{(j)}) \text{ when } \mu = \ell_{ij}. \tag{5}$$

We note that f is convex since it is a squared function, and g is convex because all norms are convex (from the criteria of fulfilling the triangle inequality). Furthermore, we want to show that f is increasing by looking at its first derivative.

$$\frac{d}{dx} f(x) = \begin{cases} 2(x - \mu) & \text{if } x > \mu, \\ 0 & \text{if } x \leq \mu. \end{cases} \quad \Rightarrow \quad \frac{d}{dx} f(x) \geq 0 \ \ \forall x \in \mathbb{R}.$$

This shows that f is increasing. We now use that the composite function $h \circ j$ is convex if j is convex and h is convex and increasing. From this, we directly get that g is a convex function, which in turn gives us the desired result that E_{elast}^{cable} is convex. This further implies, through our definition of E_{elast}^{cable} in (5), that E_{elast}^{cable} is convex. Thus all summands in (2) are convex, which means that the cost function in (2) itself is convex.

4.2 Optimality Conditions

We can reformulate the problem to a free optimization problem. Let P be the nodes with fixed positions and X_R be the free nodes. Then the vector of nodes will be $X = [P, X_R]^T$. We do not put any constraints on X_R and thus problem (2) is a free optimization problem with respect to X_R. Furthermore, we have shown that (2) is convex and differentiable. The first order necessary condition will be

$$(\nabla_{X_R} E(e_{ij}))_n^{(i)} = \sum_{e_{ij} \in \mathcal{C}} \frac{\partial}{\partial x_n^{(i)}} E_{cable}^{elast}(e_{ij}) + \frac{\partial}{\partial x_n^{(i)}} E_{ext}(X) = 0, \forall x^{(i)} \in \mathcal{V}, n = 1, 2, 3, \tag{6}$$

where the partial derivative of E_{cable}^{elast} is given in equation (3), and by

$$\frac{\partial}{\partial x_n^{(i)}} E_{ext}(X) = \begin{cases} 0 & \text{if } n = 1, 2, \\ m_i g & \text{if } n = 3. \end{cases} \tag{7}$$

As a consequence of the convexity of E defined in (2), the first order necessary optimality condition will be sufficient. Moreover, points where $\nabla_{X_R} E = 0$ will be global minimizers of (2).

4.3 Implementation

Now we implement a numerical method for the solution of (2). Since the problem function is convex, we know that Newton and Quasi-Newton methods work well. To avoid having to solve a linear set of equations each time we find the new step direction, p, we use the BFGS method, where we approximate the inverse of the Hessian in each step. As an initial approximation we simply use the identity matrix, and for the step size selection we use strong Wolfe conditions. Since the energy function in (2) is typically not C^2, however, we have no guarantee of convergence, and therefore we simply have to test our code and study the convergence rate numerically.

5 Tensegrity-domes

The next type of structures we will look at are called tensegrity-domes. Here we allow some of the edges to be bars, which means that the energy function we want to minimize is the total energy given in (1). Like in the case for cable-nets, we constrain the problem by fixing the first M nodes in place. The problem then becomes

$$\begin{aligned} \min_X E(X) = \sum_{e_{ij}\in\mathcal{B}} \left(E^{bar}_{elast}(e_{ij}) + E^{bar}_{grav}(e_{ij})\right) + \sum_{e_{ij}\in\mathcal{C}} E^{cable}_{elast}(e_{ij}) + E_{ext}(X) \\ \text{such that } x^{(i)} = p^{(i)}, \quad i = 1, \dots, M. \end{aligned} \tag{8}$$

First, we look at smoothness and convexity.

5.1 Smoothness and Convexity

We now want to look at the differentiability of the energy function in (8). We have already shown that the functions for elastic energy of a cable and total external energy are differentiable. Furthermore, since E^{bar}_{grav} is linear in its arguments, it is also differentiable. We therefore only need to check whether E^{bar}_{elast} is differentiable. Differentiating with respect to $x^{(i)}_n$, $n = 1, 2, 3$, we find that

$$\frac{\partial}{\partial x^{(i)}_n} E^{bar}_{elast}(e_{ij}) = \frac{c}{\ell^2_{ij}}\left(1 - \frac{\ell_{ij}}{\|x^{(i)} - x^{(j)}\|}\right)(x^{(i)}_n - x^{(j)}_n). \tag{9}$$

If we now let $x^{(i)}_z = x^{(j)}_z$ for all $z \neq n$, and then let $x^{(i)}_n$ approach $x^{(j)}_n$ both from above and below respectively, we see that

$$\frac{\partial}{\partial x^{(i)}_n} E^{bar}_{elast}(e_{ij}) = \begin{cases} -\frac{c}{\ell_{ij}} & \text{if } x^{(i)}_n \to (x^{(j)}_n)^+, \\ \frac{c}{\ell_{ij}} & \text{if } x^{(i)}_n \to (x^{(j)}_n)^-. \end{cases}$$

Thus we have shown that (8) is not differentiable in cases where $\mathcal{B} \neq \emptyset$. In practice, this does not pose any problems as the function is differentiable in all logical configurations of nodes. The only exception is the case where nodes are gathered in a single point, effectively meaning that bars are compressed to infinitesimal lengths. This cannot be a minima of our function in (8), as stretching one bar closer to its resting length will always reduce the total energy. In terms of numerical implementation, we will therefore never evaluate the gradient in this point unless we choose it in our initialization, which we obviously would not do.

Next we show that the cost function in (8) is non-convex in cases where $\mathcal{B} \neq \emptyset$. To do this we look specifically at the internal elastic energy function for a bar, where we consider X to be the

vector containing nodes $x^{(i)}$ and $x^{(j)}$ such that the bar e_{ij} is neither stretched or compressed, that is $\|x^{(i)} - x^{(j)}\| = \ell_{ij}$. Then we have that $E_{bar}^{elast}(X) = 0$. If we now let $Y = -X$, we get that

$$E_{bar}^{elast}(\lambda X + (1-\lambda)Y) = E_{bar}^{elast}((2\lambda - 1)X) \overset{\lambda=\frac{1}{2}}{=} E_{bar}^{elast}(0) = \frac{c}{2} > 0$$

Meanwhile we get from our definition of X and Y that

$$\lambda E_{bar}^{elast}(X) + (1-\lambda)E_{bar}^{elast}(Y) = 0.$$

Thus, we have shown that there exists some $X, Y \in \mathbb{R}^{3N}$ and a $0 < \lambda < 1$ such that

$$E_{bar}^{elast}(\lambda X + (1-\lambda)Y) > \lambda E_{bar}^{elast}(X) + (1-\lambda)E_{bar}^{elast}(Y),$$

which implies that the energy function $E(X)$ in (8) is a non-convex function when $\mathcal{B} \neq \emptyset$.

We can also show by means of an example that the energy function in (8) admits local minimizers in certain cases. For example, consider the case where we have four fixed nodes placed in a square in the (x_1, x_2)-plane, all having bars connected to one free node which is initialised to be above this square formation. We will then obtain a local minima when the combined compression forces are enough to support the combined gravitational downwards pull of the bars. Still, this will not be the global minima. Having the free node below the fixed nodes will give a lower potential energy compared to the other formation. This example is shown in the Jupyter notebook.

5.2 Optimality conditions

As in the Cable-Nets case, we can rewrite the optimization problem given by equation (8) as a free optimization problem with respect to X_R, where X_R are the free nodes and P are the fixed nodes. Our array of nodes can then be written as $X = \begin{bmatrix} P \\ X_R \end{bmatrix}$. The necessary optimality condition is that

$$(\nabla_{X_R} E(e_{ij}))_n^{(i)} = \sum_{e_{ij} \in \mathcal{B}} \frac{\partial}{\partial x_n^{(i)}} E_{bar}^{elast} + \frac{\partial}{\partial x_n^{(i)}} E_{bar}^{grav} + \sum_{e_{ij} \in \mathcal{C}} \frac{\partial}{\partial x_n^{(i)}} E_{cable}^{elast} + \frac{\partial}{\partial x_n^{(i)}} E_{ext} = 0,$$

this for all $x^{(i)} \in \mathcal{V}$ and values of n = 1,2,3. The partial derivatives of E_{cable}^{elast}, E_{ext} and E_{bar}^{elast} are already given in equation (3), (7) and (9) respectively. If we differentiate E_{bar}^{grav} in a similar manner as we have done earlier, we get that

$$\frac{\partial}{\partial x_n^{(i)}} E_{bar}^{grav}(e_{ij}) = \begin{cases} \frac{\rho g \ell_{ij}}{2} & \text{if } n = 3, \\ 0 & \text{else.} \end{cases}$$

Since we are now dealing with a non-convex cost function, the first order necessary optimality condition will not be sufficient for a local minimum. Neither can we talk about second order optimality conditions, as our cost function is not $\mathcal{C}^2$.

5.3 Implementation

Since the cost function we wish to minimize is non-convex, the BFGS method is not guaranteed to converge for all starting values, and therefore we implement another algorithm that is more suited. For this we choose the Conjugate Gradient method (CG) with Polak-Ribiére+ update. This should ensure a robust convergence at the cost of more iterations required.

6 Free-standing Structures

In this section we look at structures where no nodes are fixed, but where we introduce a "floor" for the free structures to stand on, $x_3^{(i)} \geq 0$. Our optimization problem then becomes

$$\begin{aligned} \min_X E(X) = \sum_{e_{ij}\in\mathcal{B}} \left(E_{elast}^{bar}(e_{ij}) + E_{grav}^{bar}(e_{ij})\right) + \sum_{e_{ij}\in\mathcal{C}} E_{elast}^{cable}(e_{ij}) + E_{ext}(X) \\ \text{s.t. } x_3^{(i)} \geq 0, \quad i = 1, \ldots, N. \end{aligned} \tag{10}$$

6.1 Optimality Conditions

Since the problem is constrained, the first order optimality requirements are the KKT-conditions. As we found in the previous section the cost function is non-convex, and therefore the conditions are necessary, but not sufficient for a local solution. We have N inequality constraints, all of the form $c_i(X) = x_3^{(i)} \geq 0$, and the Lagrangian is $\mathcal{L}(X, \lambda) = E(X) - \sum_{i=1}^N \lambda_i c_i(X) = E(X) - \sum_{i=1}^N \lambda_i x_3^{(i)}$ with $E(X)$ given in (1). Thus the first order optimality conditions for a local minimum X^* with Lagrange parameters $\lambda^* = \left(\lambda_1^*, \ldots, \lambda_N^*\right)$ are

$$\begin{aligned} \nabla_X \mathcal{L}(X^*, \lambda^*) = \nabla E(X^*) - \left((0,0,\lambda_1^*), \ldots, (0,0,\lambda_N^*)\right) = 0, \\ x_3^{(i)} \geq 0, \quad \lambda_i^* \geq 0, \qquad \lambda_i^* x_3^{(i)} = 0, \quad \text{for} \quad i = 1, \ldots, N. \end{aligned}$$

The assumptions for the first order optimality conditions are that $c_i(X)$ and $E(X)$ are continuously differentiable, and that the LICQ condition holds. $c_i(X)$ are linear functions, and hence they are differentiable, and although $E(X)$ is not differentiable, we discussed why this in practice pose no problem. The LICQ condition is that all $\nabla c_i(X)$ are linearly independent, and by looking at the gradients, we see that

$$\nabla c_i(X) = \nabla x_3^{(i)} = \big((0,0,0), \ldots, \underbrace{(0,0,1)}_{i\text{-th element}}, \ldots, (0,0,0)\big).$$

The gradients of the constraints are orthogonal, and thus definitely linearly independent, for all $X \in \mathbb{R}^{3N}$. This means that LICQ is satisfied, and the KKT-conditions (under the assumption that a non-smooth $E(X)$ is okay) are necessary conditions for a local solution of (10).

6.2 Horizontal Shifts of Nodes

If all the nodes of the structure are shifted by an equal amount horizontally, the length of each cable and bar remains the same, and the gravitational potential energy of the external loads is unchanged as we do not shift the structure vertically. Thus the total energy of the system is unchanged, which means that (10) is invariant under horizontal shifts of the nodes. This may pose problems numerically. To fix this potential issue, we simply add equality constraints to the x_1- and x_2-coordinates of the first node, and let it move freely in the x_3-direction. We constrain these coordinates to be equal to one, i.e. $c_1^{\mathcal{E}}(X) = x_1^{(1)} - 1 = 0$ and $c_2^{\mathcal{E}}(X) = x_2^{(1)} - 1 = 0$.

6.3 Implementation

To solve the new constrained optimization problem we use a quadratic penalty method, that is we minimize the function

$$Q(X, \mu) = E(X) + \frac{\mu_1}{2} \sum_{i=1}^N \left(\left[x_3^{(i)}\right]^-\right)^2 + \frac{\mu_2}{2}(x_1^{(1)} - 1)^2 + \frac{\mu_2}{2}(x_2^{(1)} - 1)^2, \tag{11}$$

where $[x]^- = \max(0, -x)$. The larger we set the penalty parameters μ_1 and μ_2, the more we penalise solutions breaking the constraints, and thus the closer we get to the actual solution. However, a large μ results in an ill-conditioned problem, and therefore we gradually increase the penalty parameter by a factor γ as the error gets smaller. In each step, we use `BFGS()` to solve for the approximate minimizer of Q. Although we have no guarantee of convergence, we try with different initial values that we know are somewhat close to the exact solution.

7 Numerical Experiments

Now we implement optimization algorithms and solve the three problems described above, numerically. Some parameters vary in each subsection, but the ones that remain the same are:

- Those related to the strong Wolfe conditions: $c_1 = 0.01$, $c_2 = 0.9$, $\alpha_0 = 1$ (initial step-length), $\rho_W = 2$ (step length increase parameter).
- Error tolerance and maximum iterations in algorithms: $tol = 10^{-8}$, $maxiter = 1000$.

7.1 Cable-nets

We include one test example here, and leave one additional test in the Jupyter notebook. The parameters used are:

- $\ell_{ij} = 3$ for all edges with cables (shown in Figure 1).
- $k = 3$, $m_i g = \frac{1}{6}$ for $i = 5, 6, 7, 8$, i.e. for the free nodes.
- $X_{init} = \big((5,5,0), (-5,5,0), (-5,-5,0), (5,5,0), (1,1,0), (-1,1,0), (-1,-1,0), (1,1,0)\big)$, first four nodes are fixed ($M = 4$).

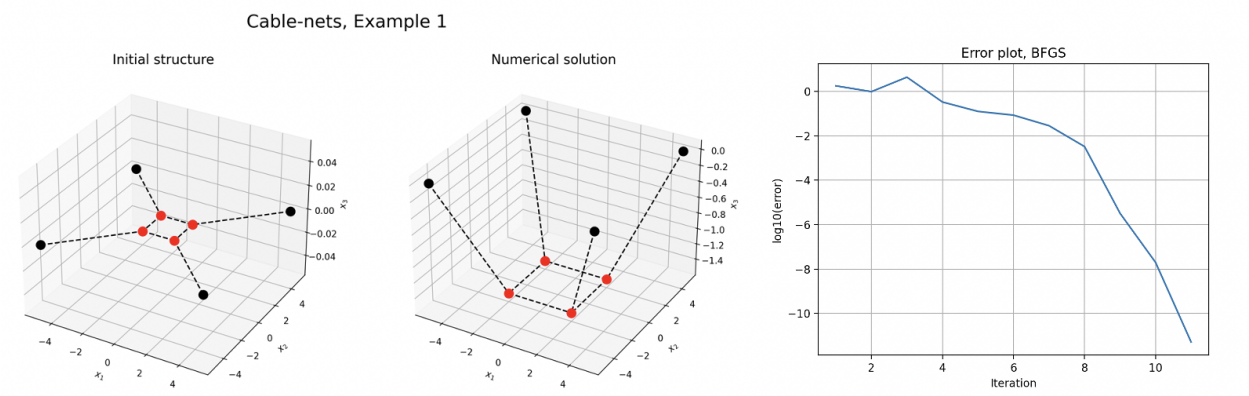

Figure 1: *3D-plot and convergence plot of the convex cable-nets problem with 4 fixed nodes. Fixed nodes are colored black, free nodes are red. Dashed lines represent cables.*

The nodes all start at $x_3^{(i)} = 0$, and fall down to the equilibrium point where energy is minimized. As we see from the convergence plot the BFGS algorithm solves the convex problem quickly, using only 11 iterations to reach an error of less than 10^{-10}. The numerical solution converges to the analytical solution, $x^{(5)} = (2, 2, -\frac{3}{2})$, $x^{(6)} = (-2, 2, -\frac{3}{2})$, $x^{(7)} = (-2, -2, -\frac{3}{2})$ and $x^{(8)} = (2, -2, -\frac{3}{2})$. A printout of this is shown in the notebook.

7.2 Tensegrity-domes

We include one test example here, and leave one additional test in the Jupyter notebook. The parameters used in the first test example are:

- $\ell_{15} = \ell_{26} = \ell_{37} = \ell_{48} = 10$ (bar lengths).
- $\ell_{18} = \ell_{25} = \ell_{36} = \ell_{47} = 8$ and $\ell_{56} = \ell_{67} = \ell_{78} = \ell_{58} = 1$ (cable lengths).
- $k = 0.1$, $c = 1$, $\rho g = 0$, $m_i g = 0$ for $i = 5, 6, 7, 8$.
- $X_{init} = \big((1,1,0), (-1,1,0), (-1,-1,0), (1,1,0), (1,1,1), (-1,1,1), (-1,-1,1), (1,1,1)\big)$, first four nodes are fixed ($M = 4$).

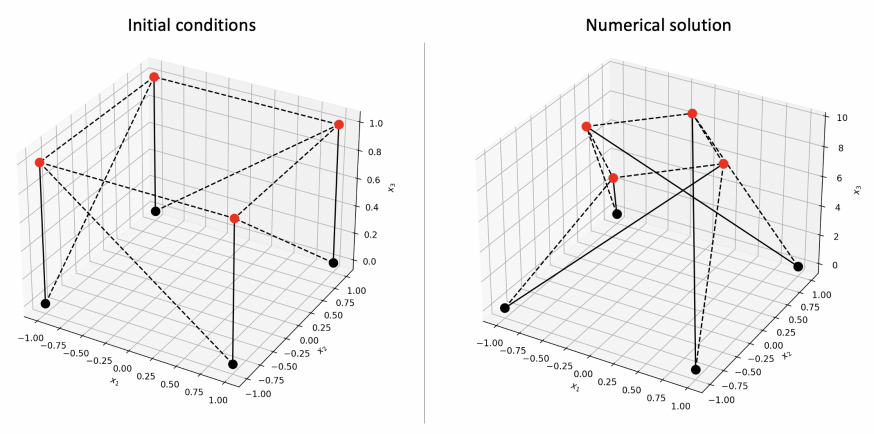

Figure 2: *3D-plots of initial structure and numerical solution of a tensegrity-dome. Black dots are fixed nodes, red dots are free. Dashed lines are cables and solid lines are bars.*

The exact numerical solution is shown in a printout in the notebook. It coheres with the analytical solution given the parameters presented above.

We compare convergence of `BFGS()` and `CG()`, and also combine the two in a final experiment to try to achieve smooth and relatively fast convergence. To do this we let `CG()` solve to a precision of 10^{-4}, and then use `BFGS()` to achieve an error of size 10^{-8}.

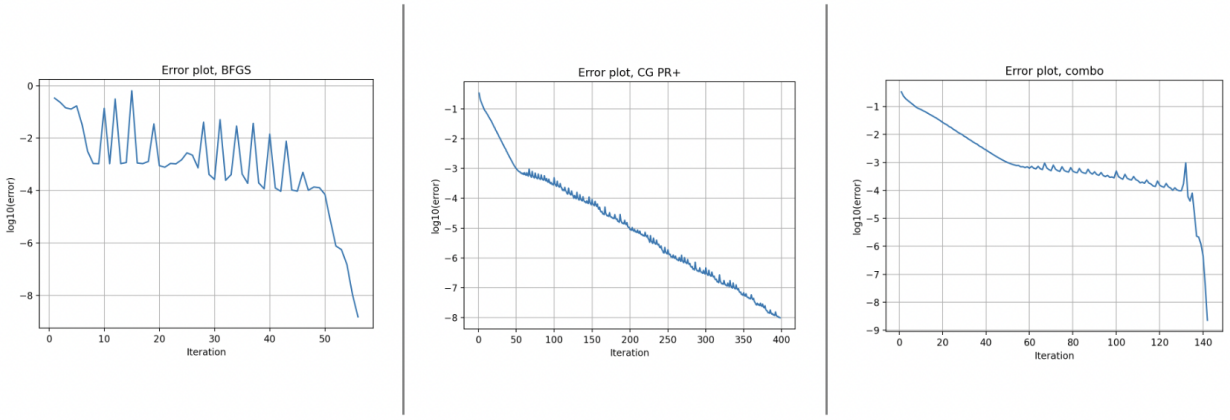

Figure 3: *Convergence plots using `BFGS()`, `CG()` and a combination of both.*

We see that BFGS solves the problem quickly, using about 55 steps, but that the convergence plot is jagged. This can be explained by the non-convexity of the energy function, as discussed

earlier. Conjugate gradient uses about 400 iterations to converge to the same error of 10^{-8}, but is as expected more robust. The combination of the two seems more robust, but slower than BFGS and faster than CG, using about 140 steps in total. The initial "jump" once BFGS takes over may be due to the initial approximation of the inverse of the Hessian, where we use the identity matrix.

7.3 Free-standing Structures

As before, we show one numerical experiment here and leave an additional one in the notebook. We use the same parameters as in the previous section with exception of/in addition to:

- $\rho g = 10^{-5}$, $\mu_{1,init} = 10^{-3}$, $\mu_{2,init} = 10^{-5}$, $\gamma = \begin{cases} 3 & \text{if number of iterations } < 500, \\ 1.5 & \text{otherwise.} \end{cases}$.
- $X_{init} = \big((1,1,0),(-1,1,0),(-1,-1,0),(1,1,0),(-1,0,10),(0,-1,10),(1,0,10),(0,1,10)\big)$.

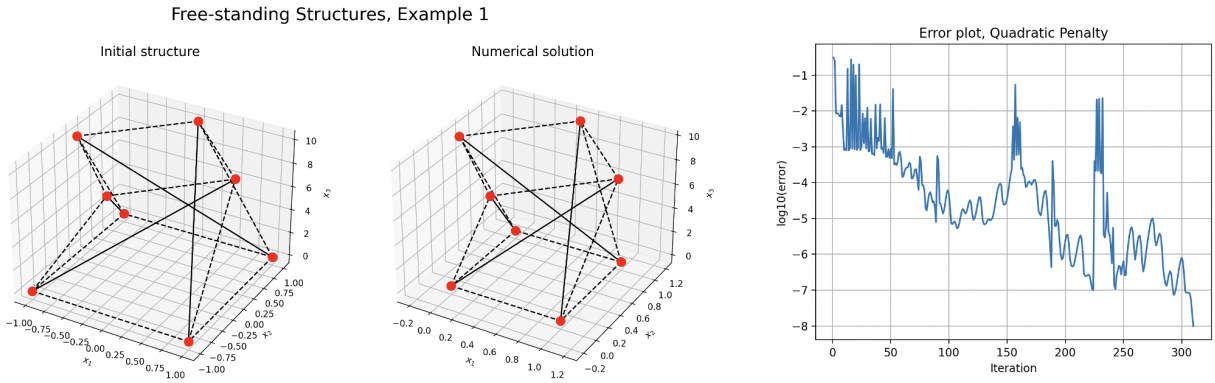

Figure 4: *Initial structure, numerical solution and convergence plot for a free-standing structure.*

All nodes are (nearly) above zero in the x_3-direction, $x_1^{(1)} \approx 1$ and $x_2^{(1)} \approx 1$, which implies that the penalty parameters are sufficiently large. The numerical solution is similar to that of Figure 2, but a bit slimmer at the bottom, which makes sense as the cables will pull the free nodes inwards. The convergence plot shows a very jagged convergence that eventually reaches an error of 10^{-8}, which may be explained by the ill-conditioned problem and the fact that we reset the Hessian approximation for each call to `BFGS()`.

8 Conclusion

We have shown that our model of cable nets results in a convex cost function, while tensegrity-domes give a non-convex problem. The convexity-property of the cost function will affect the suitability of different numerical algorithms. BFGS seemed to have the fastest convergence rate, but struggling to converge for non-convex cost functions. CG on the other hand could handle such problems, but in turn needs far more iterations to converge compared to BFGS. However, when we are dealing with ill-conditioned non-convex problems, such as the quadratic penalty problem in Free-standing structures, BFGS works if the initial guess is sufficiently close to a solution and the penalty parameters are sufficiently small.