

Design and optimization of a general planar zero free length spring

Delissen, Arnoud A.T.M.; Radaelli, Giuseppe; Herder, Just L.

DOI

[10.1016/j.mechmachtheory.2017.07.002](https://doi.org/10.1016/j.mechmachtheory.2017.07.002)

Publication date

2017

Document Version

Accepted author manuscript

Published in

Mechanism and Machine Theory

Citation (APA)

Delissen, A. A. T. M., Radaelli, G., & Herder, J. L. (2017). Design and optimization of a general planar zero free length spring. *Mechanism and Machine Theory*, 117, 56-77.
<https://doi.org/10.1016/j.mechmachtheory.2017.07.002>

Important note

To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Copyright

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy

Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.

Design and Optimization of a General Planar Zero Free Length Spring

Authors

Arnoud A.T.M. Delissen, Giuseppe Radaelli, Just L. Herder

Precision and Microsystems Engineering

Delft University of Technology

Abstract

A zero free length (ZFL) spring is a spring with special properties, which is commonly used in static balancing. Existing methods to create ZFL springs all have their specific drawbacks, which rises to the need of a new method to create such a spring. A method is proposed to design planar ZFL springs with specified stiffness (250-750 N/m) within a certain range (up to 20 mm of displacement). Geometric non-linearities of a curved leaf spring are exploited by changing its shape. The shape is determined by a non-linear least squares algorithm, minimizing the force residuals from a non-linear numerical analysis. Constraints are introduced to help in preventing the spring from intersecting itself during deformation. For three types of springs with different boundary conditions, designs are found with characteristic shapes and maximum force errors less than 1%. A trend is observed between spring size, maximum stress and desired stiffness. New type of ZFL springs can now be designed, which can not only be used in existing applications, but also enables the use of ZFL springs in micro mechanisms.

Keywords

Zero free length spring; null-length spring; ideal spring; curved flexure; shape optimization; desired force-displacement

List of variables (in order of appearance)

F	Reaction/spring force
k	Spring stiffness
u	Displacement
L_0	Initial length of spring
x_i	Control points
N	Number of segments
l_i	Segment length
α_i	Segment relative angle
q	Design vector
w	Beam width
h	Beam height
F_e	External force vector
K	Stiffness matrix
K_t	Tangent stiffness matrix

$\mathbf{F}_i$	Internal force vector
F_x, F_y	Reaction force component in x and y -direction
F_A	Axial (reaction) force component (aligned with displacement direction)
F_T	Transverse (reaction) force component (force perpendicular to displacement direction)
M	Reaction moment
n_u	Number of displacement steps in one track
n_θ	Number of tracks in different directions
$u_{\min}$	Minimum displacement (displacement at first sample point)
$u_{\max}$	Maximum displacement range of the spring
$\tilde{\mathbf{F}}$	Desired force
r	Force residual
f_{obj}	Objective function
$\alpha_{\max}$	Maximum relative segment angle
$l_{\min}$	Minimum segment length
$l_{\max}$	Maximum segment length
$\mathbf{r}_0$	Direction vector from the light source node to the shadow segment
$\mathbf{r}_s$	Orientation vector of the shadow segment
$\mathbf{R}_{90}$	90° angle rotation matrix
$\mathbf{d}_0$	Normal vector of the shadow segment
$\hat{\mathbf{n}}_0$	Corrected normal vector of the shadow segment
$\mathbf{r}_1, \mathbf{r}_2$	Direction vectors from the light source node to the corners of the shadow segment
$\mathbf{d}_1, \mathbf{d}_2$	Normal vectors of the corners
$\mathbf{n}_1, \mathbf{n}_2$	Corrected normal vectors of the corners
μ	Direction parameter
$\mathbf{R}(\theta)$	Rotation matrix of angle θ
θ_e	Extra outwards rotation of normal vectors
$\hat{\mathbf{n}}_1, \hat{\mathbf{n}}_2$	Normal vectors of the shadow segment corners
f_0, f_1, f_2	Distances to the border of the shadow
g_I	Constraint equation for the shadow-method constraint to prevent self-intersection
N_c	Number of shadow constraint equations
g_{II}	Constraint equation for contact during displacement
r_g	Penalty function for constraint equations
c_I, c_{II}	Constraint constants for penalty functions
$f_{\text{obj},c}$	Constrained objective function
$\mathbf{J}$	Jacobian matrix
$\mathbf{p}$	Refined control points
ε_A	Maximum relative error of forces in axial direction
ε_T	Maximum relative error of forces in transverse direction
$\varepsilon_{\text{shape}}$	Error on the shape of the spring

1. Introduction

The spring is one of the most commonly used physical elements in engineering. A special sub-category in springs is the zero free length spring (ZFL spring), also null-length spring or ideal spring, which is - as the name implies - a spring with zero physical length when no forces are applied. This property results in a spring element which enables a range of special mechanisms to be realized, predominantly in the field of static balancing. To mention a few: Slow wave seismometer [1, 2], Anglepoise suspension [3] which are basically balanced arms with a mass [4], zero stiffness mechanisms [5, 6], mobile arm support for humans [7, 8], camera stabilizer apparatus [9] and in (robotic) manipulators [10, 11]. In all these applications any unwanted potential energy differences (resulting from gravity or elastic deformations) are counteracted by one or more ZFL springs, delivering the necessary forces to neutralize the undesired loads.

Before explaining the emerging implementation difficulties of a ZFL spring, it is important to note the properties which define the unique behavior of a ZFL spring. A ZFL spring gets its unique properties from the sole fact that the unstretched (free) length of a linear spring is zero. Observing **Figure 1**, a ZFL spring is shown on the left. The length being zero results in the spring pivot and endpoint being coincident. The spring force is now directly proportional to the displacement vector, i.e. the force is in the same direction and its magnitude is proportional to the extension length of the spring ($\mathbf{F} = k\mathbf{u}$). This in contrast to a normal (non-zero free length linear) spring, where the force is not only dependent on the displacement, but also on the initial position L_0 of the spring ($\mathbf{F} = k(L_0 + \mathbf{u})$). This is shown on the right in **Figure 1**. In one dimension there is no difference between the two springs, as the reference (zero) point can be chosen freely. But in two or three dimensions the zero point will always be the pivot point on which the spring is fixed.

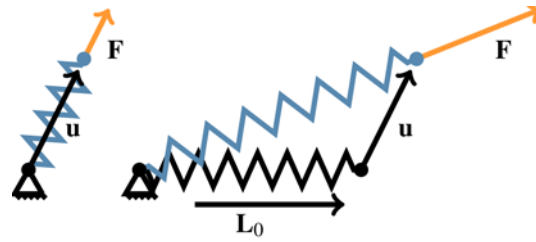


Figure 1 - Comparison of a ZFL spring (left) and a 'normal' spring (right). In the ZFL spring $\mathbf{F} = k\mathbf{u}$, while in the normal spring $\mathbf{F} = k(L_0 + \mathbf{u})$. The undeformed spring is shown in black, the deformed in blue.

Currently, a general ZFL spring is not an existing product, although various methods exist to approach the behavior of a ZFL spring at least in a certain range. The simplest ZFL spring is created by using a cantilever bar on its perpendicular plane. In all directions there is a linear proportional force from the origin. The plane of ZFL action for this spring is shown in **Figure 2a**. For large deformations however, it is not linear anymore, plus for some designs an out-of plane structure may not be desirable. Cawardine [3] proposed to create a pre-stressed spring using a special coiling process. By applying a pre-stress the effective undeformed length of the spring would become zero. In **Figure 2b** this effect is shown. A third method to create a ZFL spring is to use a material which exhibits a linear elastic behavior over a limited range, for instance implemented by a rubber band [8, 12]. The non-linear material properties of the material cause a linear proportional part in the force profile of the spring. This effect is schematically shown in **Figure 2c**. Another method to get a ZFL spring, would be to hide the length of a normal linear spring behind a guiding point. For instance, the spring can be connected to a wire, which is guided along a pulley. This guiding point then acts as a new zero point for the ZFL spring, which is shown in **Figure 2d**. Also other possible compound structures simulating ZFL springs exist [13, 14]. A recently explored method is to use an in-plane compliant mechanism. By the arrangement of leaf springs and links a desirable force-displacement curve can be obtained. It has been shown that it is possible to obtain ZFL behavior in one direction over large deflections [15].

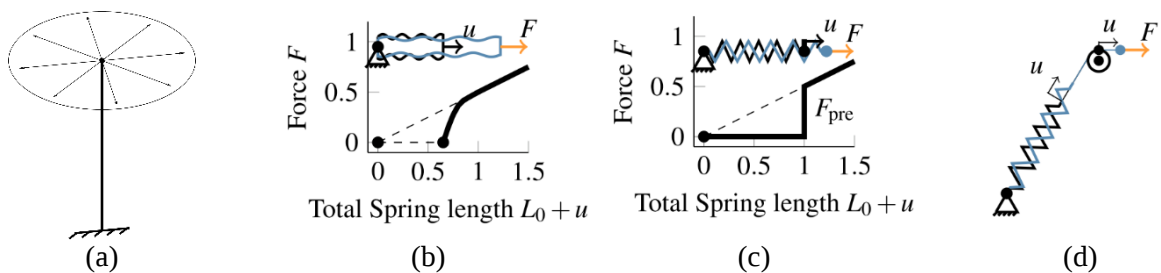


Figure 2 – Various existing methods to create ZFL springs. By using a cantilever spring (a), a pre-stressed spring (b), a spring made from non-linear rubber (c) or a compound spring (d).

The existing methods of creating ZFL springs have big disadvantages. For instance, the range of operation is limited by the range of linear material behavior in rubber or by the physical space a (complex) compound structure occupies. It is hard to produce a precise ZFL spring from a compound structure, as effects of friction and backlash are introduced by a guiding mechanism. Furthermore, it is not possible to create a spring on micro scale because fully three-dimensional structures, assembly methods and material choice are limited. The existing ZFL spring based on a mechanism [15] is unable to obtain ZFL behavior close to zero and is limited to one direction.

The goal of this research is to provide the designer with a method to create their own ZFL spring, tailored for any chosen spring stiffness. The spring is realized by a beam which is formed into a yet unknown shape. When displacing one end of the beam over a predetermined area of operation, the beam shape inherently causes a linear force-displacement proportionality in all in-plane directions. This method is applied on three cases with different boundary conditions, each behaving differently and being suitable for different applications.

The method starts off with describing the numerical model which is used for simulation, followed by an explanation how of the desired and actual force profiles are translated to a mathematical objective function. Furthermore, the optimization process is treated. The results section shows different designs and their performance, obtained from various initial designs and desired stiffness values. Finally the performance is analyzed and the optimization process, shape, possible applications and difficulties are discussed.

2. Method

In this section, a method is proposed which is able to obtain designs for ZFL springs. This research is limited to planar designs of curved flexures with constant thickness and height. The first step is to obtain a suitable numerical model. Then, the reaction forces can be calculated by numerically analyzing this model. An optimization problem is constructed, for which an objective function needs to be formulated from the reaction forces. Next, some constraints on the shape are introduced and incorporated into the objective function, to help prevent infeasible designs. Finally, an optimization algorithm is used to obtain designs for the ZFL spring.

2.1 Analysis

Before anything can be analyzed, the spring geometry needs to be parameterized. By using a (quadratic) B-spline [16] a smooth shape can be generated, dependent on a set of coordinates or control points $\mathbf{x}_i$ which form the scaffold of the curve. In this way complex shapes can be described by using relatively few parameters. As design parameters during optimization, the Cartesian control points will not directly be used, but instead they will be generated from N segments with lengths l_i and a relative angle α_i with respect to each other (**Figure 3**). This is to have better control on the relative positioning of the control points. The design parameters α_i and l_i are captured in the design vector $\mathbf{q}$. The first control point is chosen to be $\mathbf{x}_0 = \mathbf{0}$, any further coordinates can be calculated using **Equation 1**. Furthermore, the cross section of the beam is uniform over the whole curve and is also kept constant during optimization. The out-of-plane width is called w and the in-plane beam height h .

$$\mathbf{x}_i = \begin{bmatrix} x_i \\ y_i \end{bmatrix} = \mathbf{x}_0 + \sum_{p=1}^i l_p \begin{bmatrix} \cos(\sum_{q=1}^p \alpha_q) \\ \sin(\sum_{q=1}^p \alpha_q) \end{bmatrix} \quad (1)$$

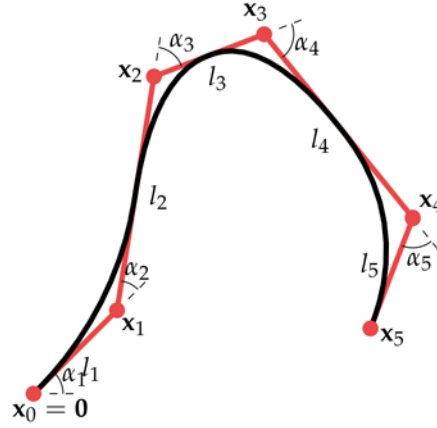


Figure 3 – An arbitrary curved spline with design variables being angles and lengths. The control point scaffold (red) and the resulting spline (black).

The values of interest are the reaction forces. These are calculated using the non-linear equation

$$\mathbf{F}_e = \mathbf{K}(\mathbf{u})\mathbf{u} \quad (2)$$

The beam discretization is based on the Isogeometric Analysis (IGA) framework, developed by Hughes and Cottrell [17, 18]. Being very similar to Finite Element Analysis (FEA), it distinguishes itself by the fact that it is based on the exact design geometry as defined with splines. This means that a coarse (unrefined) mesh already exactly represents the shape of the beam. In FEA, the discretization into finite elements results in a poor geometrical continuity for coarse meshes, causing inaccuracies. With IGA this problem is overcome. Additionally, the IGA method is able to perform refinements to the discretization without changing the geometry. A Kirchhoff-Love rod formulation capable of finite strain, large rotations and large deformations was implemented into IGA by Nagy [19].

A linear elastic material model with isotropic behavior can be used since a metal will be chosen as the base material, operating in the linear regime. On account of occurring geometrical non-linearities resulting from large deformations, the system of equations is solved using an iterative solver with full Newton steps to find the static solution. **Equation 2** is iteratively solved by additive displacements resulting from the force imbalance:

$$\mathbf{K}_t(\mathbf{u})\Delta\mathbf{u} = \mathbf{F}_e - \mathbf{F}_i(\mathbf{u}) \quad (3)$$

Boundary conditions are applied on both ends of the beam by use of Lagrange multipliers [20]. Both rotations and displacements in two directions can independently be chosen constrained, displaced, or free. For different combinations of boundary conditions, distinct types of springs are distinguished. Three spring versions are identified as pinned-pinned, clamped-pinned and clamped-clamped, also seen in **Figure 4**. There is a fundamental difference between these types of springs. Because of the rotational freedom of the pinned-pinned spring, the force is always aligned along the two pinned points, shown in **Figure 4a**. This is comparable to a simple spring element as commonly used in engineering where the zero point or rotation point is at the base of the spring. The next type introduces a clamp on the base side, causing a reaction moment to be added. This allows for the direction of the force to be in any direction, as seen in **Figure 4b**. The result of this is that there is no rotation point, but only a zero force point at the end of the spring. The begin and endpoint do not need to coincide to get zero free length properties. Another clamp on the displaced side also introduces a reaction moment M at the displacement side in addition to the reaction force (**Figure 4c**). The force components for the clamped-pinned and clamped-clamped spring can be described as Cartesian (F_x and F_y) or as the axial and transverse force (F_A and F_T). The axial force is the component aligned with the displacement direction, while the transverse force is perpendicular to the displacement.

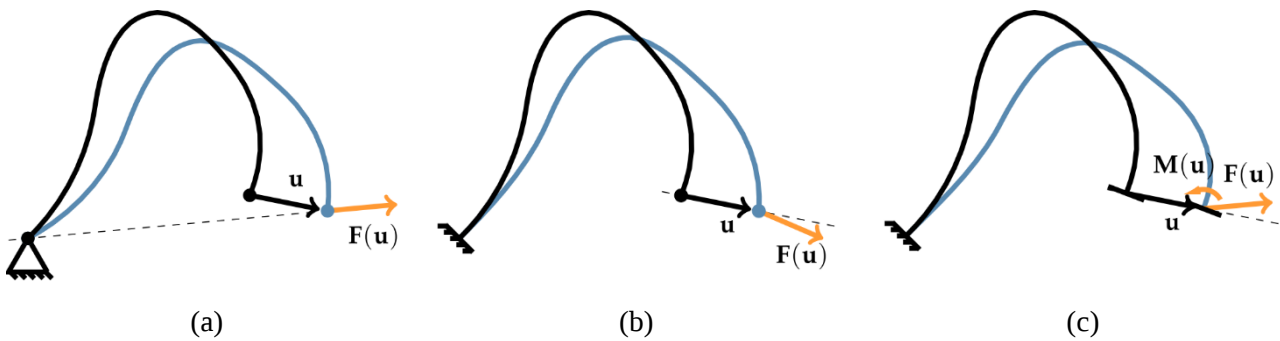


Figure 4 – Different boundary conditions give springs with different behaviors. The undeformed spring (black) and arbitrarily deformed state (blue). Distinguished are the pinned-pinned spring (a), the clamped-pinned (b) and the clamped-clamped spring (c).

Last in the analysis phase is to displace the endpoint over a certain track and calculate the corresponding reaction forces at several sample points on this track. Starting with the pinned-pinned spring, it can already be deduced that due to its rotational freedom, the smallest spring will be the one that exploits this freedom. The force-displacement profile will thus be axial symmetric around the rotation point. Therefore, the pinned-pinned spring will be displaced over a straight line in only one single direction. Because the spring's endpoint is not always near the base point, the spring's end first is pre-stressed to the first sample point, close to the spring base joint. Then, at a number (n_u) of sample points u_i on this displacement track the reaction loads are calculated. In the case of the pinned-pinned spring, displacement actually means distance of the end point to the base point.

The clamped springs do not have this rotational freedom, so being dependent on direction, multiple of straight tracks are made in different directions from the initial endpoint. A track is made in n_θ different directions, where on each line n_u sample points are taken between $u_{\min}$ and $u_{\max}$. An example of multiple straight tracks can be seen in **Figure 5**. For the clamped-clamped spring, it is chosen that the endpoint additionally keeps the same orientation - always staying parallel to its original angle. In some applications this angle might be chosen differently, resulting in a different force profile.

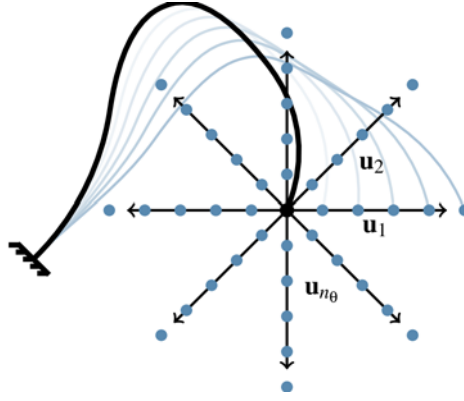


Figure 5 – A curved beam which is displaced over a straight line at discrete steps (sample points). At each of these locations the reaction loads are calculated.

2.2 Objective Function

Now being able to calculate the reaction loads (F_{ij} and M_{ij}) at the set of displacement sample points u_{ij} , the next step will be to compare these values with a desired force value ($\tilde{F}_{ij}$). In order to obtain a ZFL spring, the reaction force has to be linear proportional to the displacement, which means $\tilde{F}_{ij} = k u_{ij}$. These Cartesian forces are used during the optimization. For the pinned-pinned spring and the clamped-pinned spring only forces are acting on the endpoint. The clamped-clamped version also generates a torque at its endpoint. This moment is not taken into account during the optimization in this work. By taking the difference between the actual loads and the desired loads, residuals are formed as in **Equation 4** and **5**, illustrated in **Figure 6**. Since the displacements are taken constant during optimization, the loads are only dependent on the design q . Only the spline geometry will thus be optimized.

$$r_{x,ij}(q) = F_{x,ij}(q) - \tilde{F}_{x,ij} \quad (4)$$

$$r_{y,ij}(q) = F_{y,ij}(q) - \tilde{F}_{y,ij} \quad (5)$$

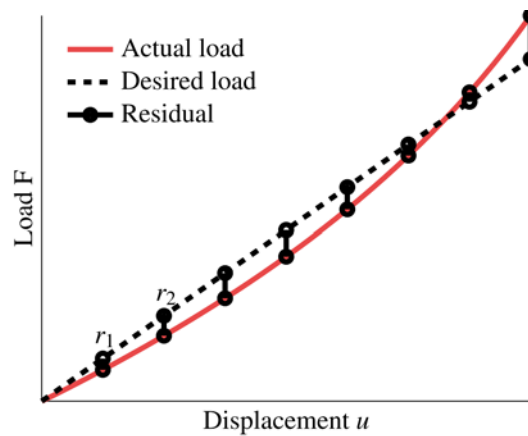


Figure 6 – An arbitrary force displacement plot corresponding with the displacement points on one track. The residuals indicated should be minimized.

For optimization purposes, it is favorable to have a single scalar component to optimize. This is done by combining all the residuals using the sum of squares formulation. Note that this step is done internally by the chosen optimization algorithm, which will be explained further in section 2.5.

$$f_{\text{obj}}(\mathbf{q}) = \sum_{i=1}^{n_\theta} \sum_{j=1}^{n_u} (r_{x,ij}(\mathbf{q})^2 + r_{y,ij}(\mathbf{q})^2) \quad (6)$$

By changing the design variables $\mathbf{q}$, the objective function is minimized by the algorithm.

$$\min_{\mathbf{q}} f_{\text{obj}}(\mathbf{q}) \quad (7)$$

2.3 Constraints

To help in generating feasible designs, three different constraint types are applied. The simplest constraints in the optimization are the bounds on the design parameters. By placing a lower bound on the lengths l_i of the segments, they are prevented from having negative or zero length. That could otherwise cause problems in the analysis. The upper length bound prevents the spring from being extremely large compared to the displacement range. Also the relative angles α_i are bounded, because numerically the use of angles introduces a periodicity in the design space. Additionally it is not favorable to have spring segments which are folded flat onto each other due to the beam thickness. The first angle α_1 is bounded between $-2\pi < \alpha_1 \leq 4\pi$, since it only changes the orientation of the spring instead of an angle between two segments. The initial orientation is chosen between $0 < \alpha_{1,\text{init}} \leq 2\pi$, to give enough freedom in orientation, not to be blocked immediately by the bounds. The optimizer now effectively has the possibility to go from 2π to 0 by moving towards 3π , preventing the optimizer getting stuck at the boundary. The only exception to this is the pinned-pinned type of spring for which the orientation is fixed at $\alpha_1 = \pi/2$ (since it is unaffected by orientation). Below, the bounds are summarized in equation form. They are enforced by the optimization algorithm to be discussed.

$$-2\pi < \alpha_1 \leq 4\pi \text{ (for PP: } \alpha_1 = \pi/2) \quad (8)$$

$$-\alpha_{\text{max}} \leq \alpha_i \leq \alpha_{\text{max}} \text{ for } i = 2, \dots, N \quad (9)$$

$$l_{\text{min}} \leq l_i \leq l_{\text{max}} \text{ for } i = 1, \dots, N \quad (10)$$

Even with bounds on the relative angles, a combination of design parameters could still cause the planar spring to intersect with itself. This is undesirable as it would be impossible to produce in a planar design. Not intersections of the spline itself, but crossings of the constructing scaffold lines are observed. Because the spline line is interpolating the scaffold lines, some cases may exist where the segments are not intersecting whereas the spline is, or the reverse. However, using the scaffolds leads to a simple and fast intersection prevention. Two subsequent segments are by definition already unable to intersect. All remaining unique combinations of the N segments make pairs that are able to intersect. No off the shelf solution exists for this constraint problem, so a new method is proposed.

Taking two random oriented segments i and j (with $j > i + 1$), their nodes $\mathbf{x}_i$, $\mathbf{x}_{i+1}$, $\mathbf{x}_j$ and $\mathbf{x}_{j+1}$ are placed, as illustrated in **Figure 7**. The infeasible area for point $\mathbf{x}_{j+1}$ looks like a shadow behind segment i cast by a light

source at $\mathbf{x}_j$. The constraint value in the infeasible domain will be defined as the shortest distance from the location of $\mathbf{x}_{j+1}$ to the feasible domain. First, the normal of segment i ($\mathbf{d}_0$) is determined, of which it is important it points towards the infeasible domain. The vector $\mathbf{r}_s$ aligned with segment i , is rotated 90° by using rotation matrix $\mathbf{R}_{90}$. The direction of this normal vector is corrected such that it points in the infeasible direction by aligning it with the vector pointing from $\mathbf{x}_j$ to the middle of the segment ($\mathbf{r}_0$). Doing this leads to the correct normal vector $\hat{\mathbf{n}}_0$.

$$\mathbf{r}_0 = \frac{\mathbf{x}_i + \mathbf{x}_{j+1}}{2} - \mathbf{x}_j \quad \mathbf{r}_s = \mathbf{x}_i - \mathbf{x}_{i+1} \quad (11)$$

$$\mathbf{R}_{90} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \quad (12)$$

$$\mathbf{d}_0 = \mathbf{R}_{90} \frac{\mathbf{r}_s}{\|\mathbf{r}_s\|} \quad \hat{\mathbf{n}}_0 = \frac{(\mathbf{r}_0 \cdot \mathbf{d}_0) \mathbf{d}_0}{\|(\mathbf{r}_0 \cdot \mathbf{d}_0) \mathbf{d}_0\|} \quad (13)$$

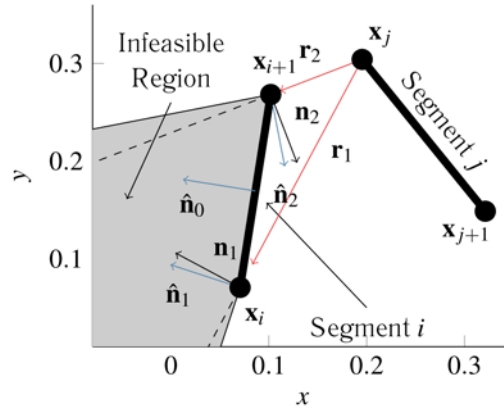


Figure 7 – The location of $\mathbf{x}_{j+1}$ results in an infeasible design if placed in the area shown by the filled contours. Constraint border normal are shown in black (unrotated) and blue (rotated).

The next two normals are the ones which define the edges of the shadow. First, the direction vector towards the segment's corners $\mathbf{r}_1$ and $\mathbf{r}_2$ are determined. By rotating these 90° , the normal vector is obtained. Since the shadow is behind the segment, the normals need to point inward. This is done by aligning the normals along the segment, which leads to the normal directions $\mathbf{n}_1$ and $\mathbf{n}_2$.

$$\mathbf{r}_1 = \mathbf{x}_i - \mathbf{x}_j \quad \mathbf{r}_2 = \mathbf{x}_{i+1} - \mathbf{x}_j \quad (14)$$

$$\mathbf{d}_1 = \mathbf{R}_{90} \frac{\mathbf{r}_1}{\|\mathbf{r}_1\|} \quad \mathbf{d}_2 = \mathbf{R}_{90} \frac{\mathbf{r}_2}{\|\mathbf{r}_2\|} \quad (15)$$

$$\mathbf{n}_1 = -\text{sgn}(\mathbf{d}_1 \cdot \mathbf{r}_s) \mathbf{d}_1 \quad \mathbf{n}_2 = \text{sgn}(\mathbf{d}_2 \cdot \mathbf{r}_s) \mathbf{d}_2 \quad (16)$$

An extra rotation of the normals is added to the shadow to prevent segment j to be placed directly through the nodes of i . The constraint function value would be exactly zero, so no penalty would be given for that case. The normals need to be turned away from the source $\mathbf{x}_j$. By using the determinant, the orientation of segment i with respect to the source ($\mathbf{x}_j$) can be determined. A rotation of $\theta_e = 10^\circ$ is added, acting as a buffer.

$$\mu = \det[\mathbf{r}_0 \quad \mathbf{r}_s] \quad \mathbf{R}(\theta) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \quad (17)$$

$$\hat{\mathbf{n}}_1 = \mathbf{R}(\text{sgn}(\mu)\theta_e)\mathbf{n}_1 \quad \hat{\mathbf{n}}_2 = \mathbf{R}(-\text{sgn}(\mu)\theta_e)\mathbf{n}_2 \quad (18)$$

Three individual shortest distances from point $\mathbf{x}_{j+1}$ to the feasible domain can be calculated as f_0 , f_1 and f_2 .

$$f_0 = \hat{\mathbf{n}}_0 \cdot (\mathbf{x}_{j+1} - \mathbf{x}_i) \quad (19)$$

$$f_1 = \hat{\mathbf{n}}_1 \cdot (\mathbf{x}_{j+1} - \mathbf{x}_i) \quad (20)$$

$$f_2 = \hat{\mathbf{n}}_2 \cdot (\mathbf{x}_{j+1} - \mathbf{x}_{i+1}) \quad (21)$$

The shortest of these three distances gives the constraint value:

$$g_{l,ij} = \min(f_0, f_1, f_2) \leq 0 \quad (22)$$

For each unique combination of two different segments, not being adjacent to each other, this constraint equation will be calculated. A curve with N segments thus gives N_c unique constraints.

$$N_c = \frac{(N-1)(N-2)}{2} \quad (23)$$

With the previous constraints, only the undeformed design has been subjected to intersection-prevention. During the displacement intersections could still occur, which are called displacement-contacts. It is very cumbersome to include intersections in the deformed spring: the model is refined for the analysis, which causes the (few) segments to refine into many smaller segments. Due to the quadratic relation (**Equation 23**) for the number of constraints, this approach would lead to an explosive amount of constraint equations. Therefore, a simple measure to prevent self-contact is proposed. The displacement of the endpoint is known prior to the analysis, of which the maximum is $u_{\max}$ in any direction. Although almost the whole spring is deforming, the base point $\mathbf{x}_0$ is defined at standstill. Therefore, $\mathbf{x}_0$ and the endpoint $\mathbf{x}_N$ have to be at least a distance of $u_{\max}$ apart from each other. Although this method does not prevent all cases of displacement-contact, it certainly helps. The constraint condition is mathematically written as:

$$g_{II} = u_{\max} - \|\mathbf{x}_0 - \mathbf{x}_N\| \leq 0 \quad (24)$$

Note that this constraint is only added for the clamped-clamped and clamped-pinned spring. For the pinned-pinned spring the distance between these two points needs to be smaller than the distance from the base point to the sample point, in order to realize a positive (pulling) force. Therefore, the constraint is reversed for pinned-pinned situations:

$$g_{II} = \|x_0 - x_N\| - u_{\min} \leq 0 \quad (25)$$

The non-linear constraints are incorporated into the objective function as a residual by a penalty formula $r_g = c \max(g, 0)$. The constant c is used for scaling or weighing of the constraint equations. Using all constraint equations, the constrained objective function can be written as

$$f_{\text{obj},c}(\mathbf{q}) = f_{\text{obj}}(\mathbf{q}) + \left(\sum_{i=3}^N \sum_{j=1}^{N-i+1} c_1^2 \max(g_{1,ij}(\mathbf{q}), 0)^2 \right) + c_{II}^2 \max(g_{II}(\mathbf{q}), 0)^2 \quad (26)$$

2.4 Design Sensitivities

Design sensitivities describe how sensitive an objective value or residual is to a change of its design parameters. They give a big advantage in the optimization procedure and computation time if they are derived analytically. For every residual (either force or constraint) the sensitivity can be calculated analytically, resulting in the Jacobian matrix $\mathbf{J} = \frac{dr}{dq}$. For the residuals they can be calculated using the chain rule:

$$\frac{dr}{dq} = \frac{dF}{dp} \frac{dp}{dx} \frac{dx}{dq} \quad (27)$$

In which $\frac{dF}{dp}$ are obtained using the direct sensitivity method. The refinement sensitivities $\frac{dp}{dx}$ describe the relation between changes of unrefined control points and refined coordinates [21]. Lastly, the term $\frac{dx}{dq}$ are the sensitivities between the Cartesian control points and the design variables - the derivatives of **Equation 1**.

2.5 Optimization Algorithm

The final stage is to obtain designs using an optimization algorithm, which minimizes the constrained objective function (**Equation 26**). The Matlab function for non-linear least squares optimization “*lsqnonlin*” is used, which minimizes the sum of squares value of all the residuals. This optimizer is able to use the Jacobian sensitivity matrix. In this optimization function the trust-region-reflective algorithm is used [22]. This optimization algorithm needs an initial design vector to start with. Due to the possible existence of local minima, multiple different initial designs are used. These are randomly generated in the entire design space, as defined by the bounds¹, by using a Latin hypercube [23]. This ensures the initial designs are spread randomly but evenly distributed over the design space (in terms of angles and lengths).

2.6 Properties and Constants

The material properties and geometry settings used for the optimization are listed in **Table 1**. The material properties used, are those of titanium alloy Ti-6Al-4V. This material has a high yield strength compared to its Young’s modulus, making it suitable for the large deformations and high internal stresses of the spring designs. Furthermore, the parameters which define the sample points and optimization settings for the different designs can be found in **Table 2**. The desired stiffness values k are chosen from the range {250, 375, 500, 625, 750} N/m.

Height h	1.00 mm
Width w	10.0 mm
Maximum segment angle $\alpha_{\max}$	0.9π rad
Minimum segment length $l_{\min}$	0.1 mm
Maximum segment length $l_{\max}$	200.0 mm
Number of segments N	4
Young's modulus E	113.8 GPa
Yield strength σ_{yield}	880 MPa

Table 1 – Geometry, bound limits and material constants

Design	PP	CP & CC
$u_{\min}$	5 mm	0 mm
$u_{\max}$	20 mm	20 mm
n_u	10	10
n_θ	1	8
c	10	10

Table 2 – Design and optimization parameters

3. Results

In this section the resulting designs from optimizations of the different spring types will be shown and analyzed. Starting with the pinned-pinned designs for various stiffness values, their performances are quantified using an error measure to enable comparison. Followed by clamped-pinned and clamped-clamped designs which have to cope with extra parasitic loads (forces in transverse direction of the displacement, which should be zero) and had to be optimized for more than one direction. Finally some general observations about the designs and their limitations are made.

For human interpretation it is more intuitive to speak in terms of axial and transverse forces instead of the Cartesian forces. The desired axial and transverse force respectively are $\tilde{\mathbf{F}}_{A,ij} = k\mathbf{u}_{A,ij}$ and $\tilde{\mathbf{F}}_{T,ij} = 0$, where the transverse force is called a parasitic force since it should be zero. The term k is the required stiffness and $\mathbf{u}_{A,ij}$ the axial displacement.

The performances of the springs are quantified using two error measures. The first one giving information about the forces in axial direction - the maximum relative error ε_A - is introduced as the maximum error between the actual force in axial direction $\mathbf{F}_{A,ij}$ and the desired force $\tilde{\mathbf{F}}_{A,ij}$. This measure gives the error with respect to the desired axial force in percentages:

$$\varepsilon_A = 100 \max \left(\left| \frac{\mathbf{F}_{A,ij}}{\tilde{\mathbf{F}}_{A,ij}} - 1 \right| \right) = 100 \max \left(\left| \frac{\mathbf{F}_{A,ij}}{\mathbf{u}_{A,ij}k} - 1 \right| \right) \quad (28)$$

Similarly, the performance of parasitic forces in transverse direction is quantified (if applicable). Since the desired transverse force is zero, the desired axial force will be used as a reference. This is now the error of the (parasitic) transverse force with respect to the desired axial force in percentages:

$$\varepsilon_T = 100 \max \left(\left| \frac{F_{T,ij}}{u_{A,ijk}} - 1 \right| \right) \quad (29)$$

Note that in obtaining these performance parameters, the final design is re-evaluated with more sample points than used in the optimization phase (twice the steps in axial direction and thrice the number of directions). This is to prevent (possibly large) errors at points between the sample points to go unnoticed.

All three type of springs were optimized using the same initial design vectors. The four initial designs used are shown in **Figure 8** and were generated using a Latin hypercube as described in the method. The results from each of these initial designs are given to see the influence of differing initial design. For ease of indicating the various designs, they are indicated using a special notation. The first two letters indicate the spring type - PP for pinned-pinned, CP for clamped-pinned and CC for clamped-clamped. Secondly, the initial design number is given and finally, the desired stiffness (separated with a dot). For instance, a clamped-pinned spring originating from the third initial design and optimized for a stiffness of 250 N/m is indicated with *CP3.250*.

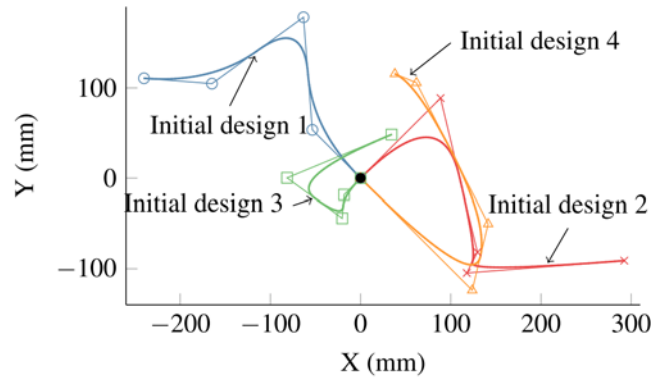


Figure 8 – The four initial designs used during optimization, generated by random Latin hypercube sampling.

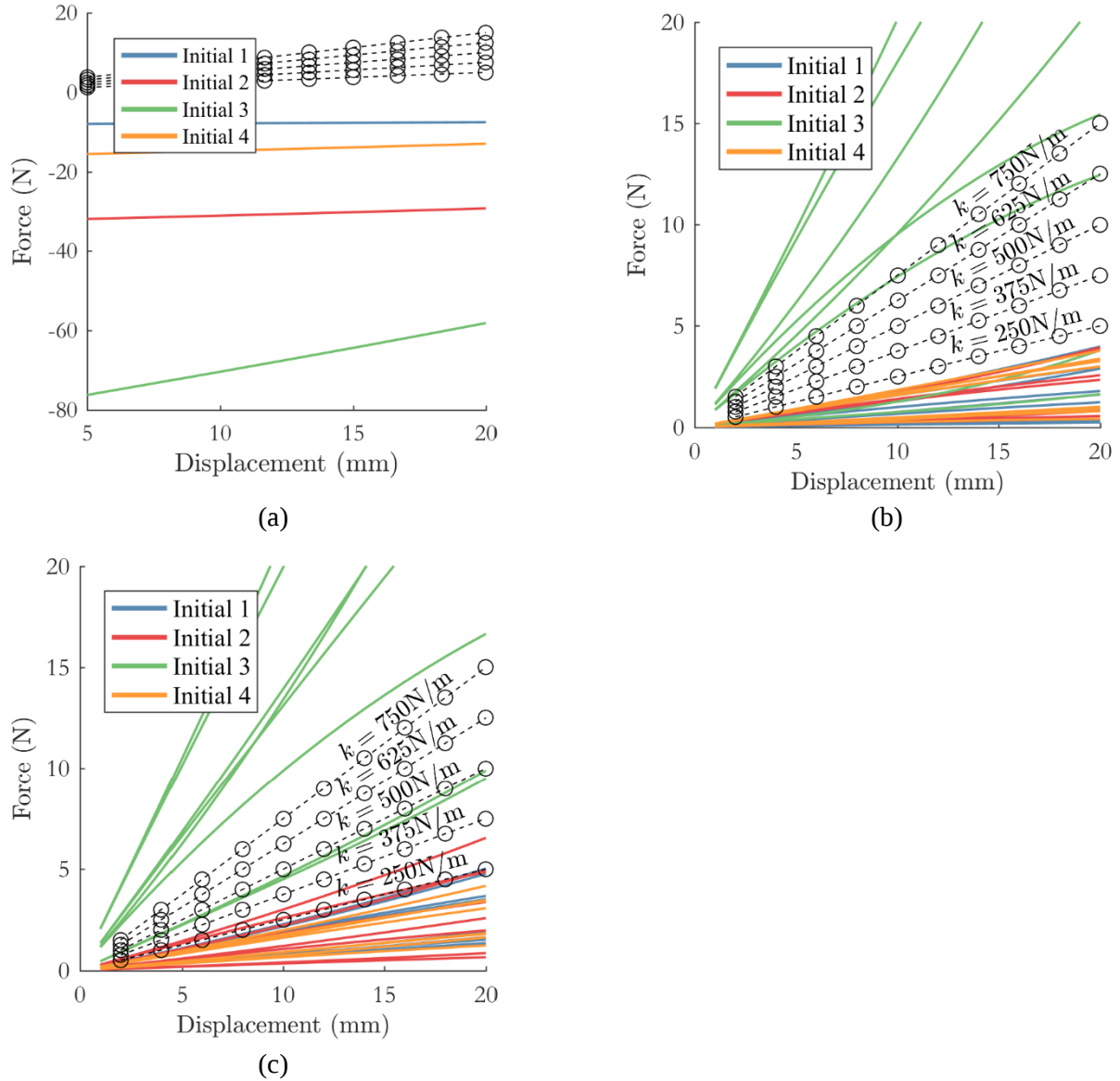


Figure 9 – The axial force profiles of the (un-optimized) initial designs. The same spring designs (indicated by differently colored lines, result in different behaviors for (a) the pinned-pinned spring, (b) the clamped-pinned spring and (c) the clamped-clamped spring. The maximum relative axial error ranges from 70% up to 800%.

3.1 Pinned-Pinned Spring

The pinned-pinned type of spring only had to be optimized for axial force as parasitic forces were absent. Optimizing for five different values of spring stiffness ($k = 250, \dots, 750$ N/m) resulted in a range of designs - of which three will be shown every time, for overview in the figures. The designs from initial design 2 are shown in **Figure 9a**, which generally perform the best of all initial designs. In **Figure 9b** the generally worst performing designs from initial design 4 are shown.

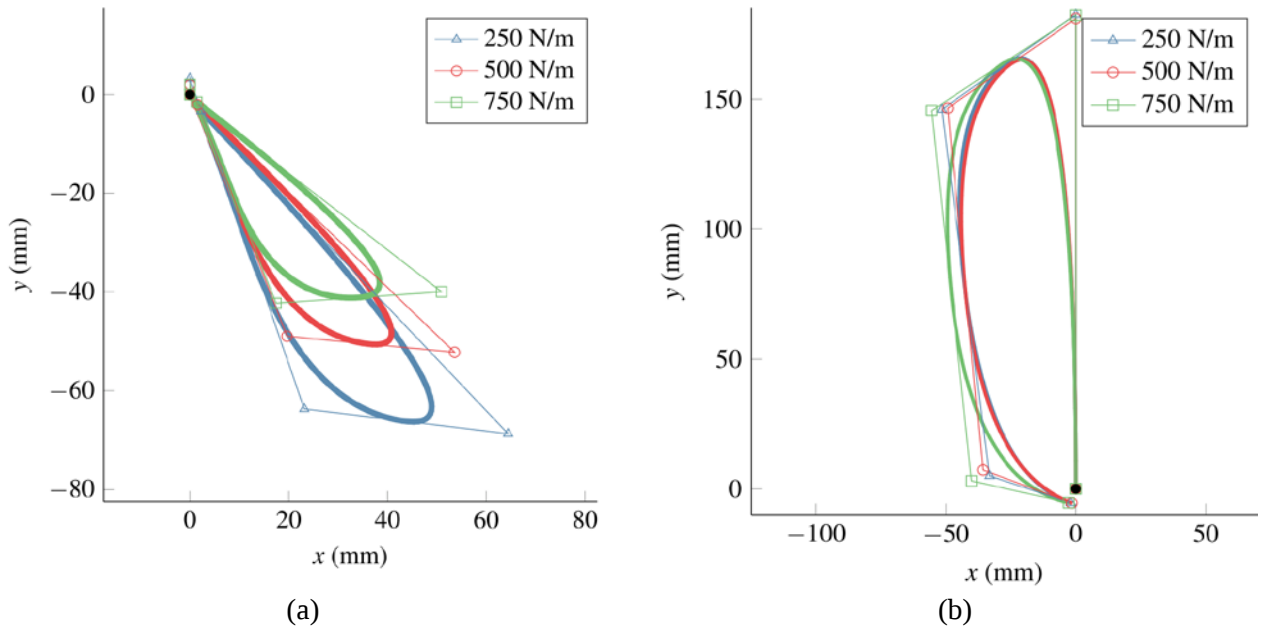


Figure 10 – Pinned-pinned spring designs performing best *PP2.250*, *PP2.500* and *PP2.750* (a) and worst *PP4.250*, *PP4.500* and *PP4.750* (b). The designs are rotated to align with each other.

The resulting shapes tend to a drop-like shape, with the begin and endpoint of the spring close to each other. For the different stiffness values the shapes resulting from the same initial design are comparable to each other. Only the size is different (**Figure 10a**). The springs with lowest stiffness were the largest, while stiffer springs were bigger. The axial force-profiles of the best and worst springs are shown in **Figure 11**. It can be seen that the best designs are seemingly spot on with the desired profile, while the worst set of designs are off more. No springs were found to be intersecting themselves during displacement.

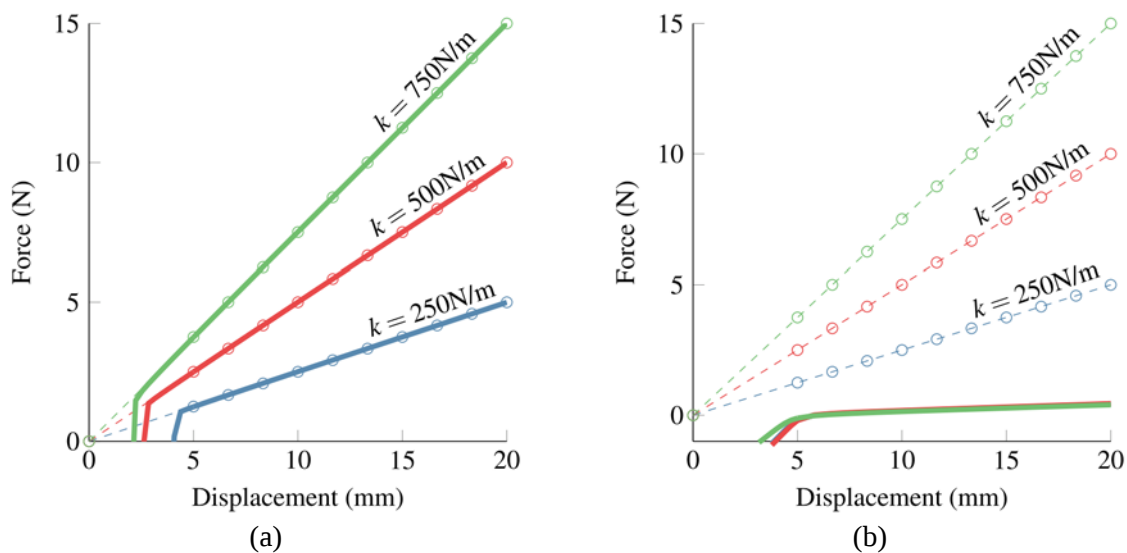


Figure 11 – The axial force profiles for (a) the best (*PP2.250*, *PP2.500* and *PP2.750*) and (b) the worst designs (*PP4.250*, *PP4.500* and *PP4.750*) compared to the desired force profile. The dashed lines are the desired forces, with circles at the sample points.

The axial errors were calculated for all the obtained designs on the refined number of sample points (**Table 3**). It can be seen that the results from the 3rd and 4th initial designs are all performing bad compared to the other designs, with errors over 100%. The designs resulting from initial design 2 were performing the best, all scoring $\varepsilon_A < 1\%$. Also the designs from initial design 1 perform well, with scores errors around 1%.

	ε_A (%)			
Stiffness (N/m)	Initial 1	Initial 2	Initial 3	Initial 4
250	0.954	0.711	55.507	114.063
375	1.515	0.423	69.968	95.668
500	1.174	0.157	79.592	107.907
625	1.091	0.314	84.284	116.491
750	0.815	0.279	100.916	102.962

Table 3 – The relative axial error ε_A for all optimized pinned-pinned designs. The colored values are the designs shown in **Figures 10** and **11**, the best performing spring in green and the worst in red.

3.2 Clamped-Pinned Spring

Two new difficulties in the spring design are introduced by adding a clamp to create a clamped-pinned spring. Paths in every direction had to be optimized and parasitic transverse forces were introduced. Again, the optimization was run with 4 different initial designs and for 5 stiffness values of which the best and worst performing designs are respectively shown in **Figures 12a** and **12b**. The results are (more or less) shaped like a spiral, with the pinned endpoint at the center of the spiral. By increasing the required stiffness, the spring design becomes smaller. The implemented constraints were not violated in any design. However, during displacement intersection occurs in two springs (*CP2.500* as shown in **Figure 12b** and *CP2.375*).

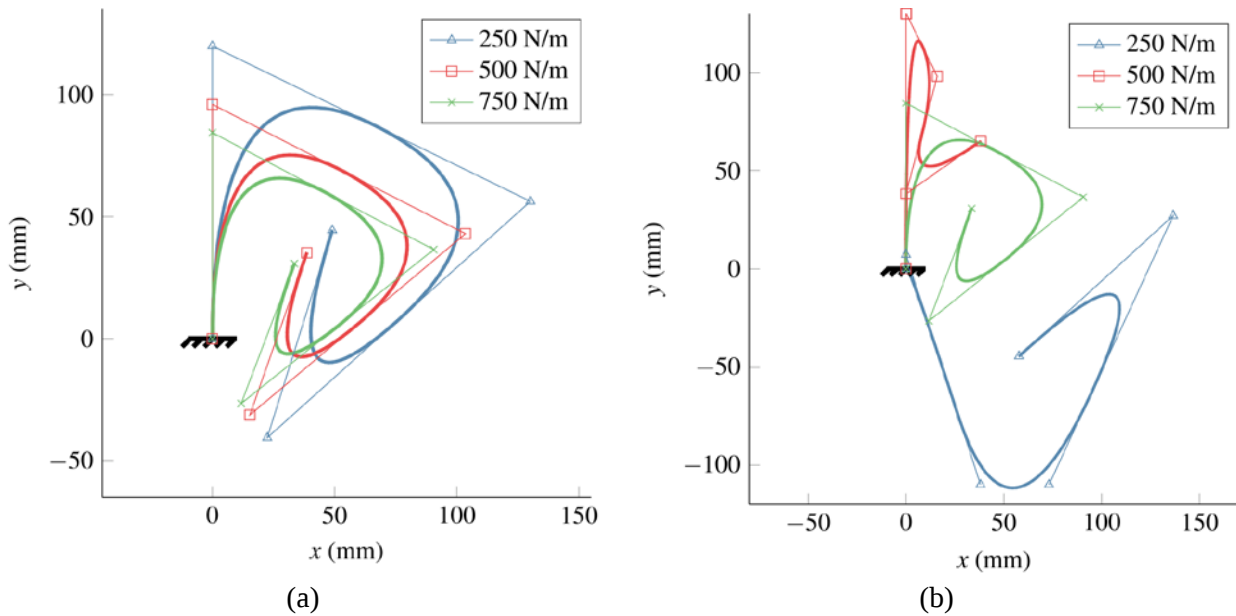


Figure 12 – Clamped-pinned spring designs performing best *CP3.250*, *CP3.500* and *CP3.750* (a) and worst *CP2.250*, *CP2.500* and *CP2.750* (b). Designs are rotated to align with each other.

Just like for the pinned-pinned springs, the axial force-displacement profiles in different directions of the best and worst designs is shown in **Figure 13**. The best design shows axial force profiles very close to each other, while the worst design has force profiles scattered over a larger range.

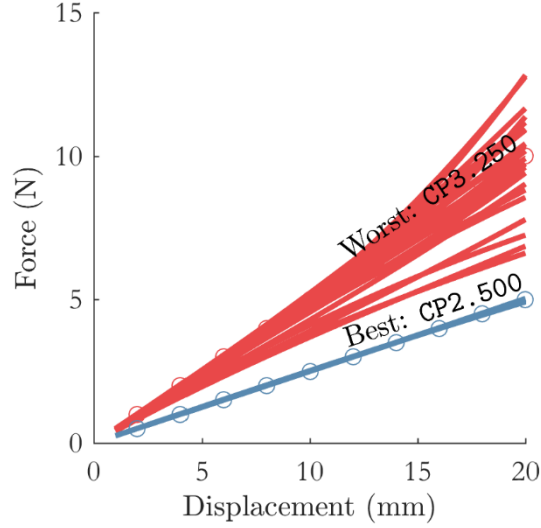


Figure 13 – The axial force-displacement profile for the best (CP2.500) and worst (CP3.250) clamped-pinned designs compared to the desired force profile. The desired forces at the sample points are indicated by circles.

Since the springs were optimized in 8 different track directions, it is interesting to see the force behavior at points in between the tracks. Therefore, the axial force-profile is shown for constant displacement length u_A . Each line shows the axial forces at locations on a circle around the undeformed pinned endpoint. These forces for the best spring CP3.250 and worst spring CP2.500 are shown in **Figure 14**. In the same manner the reaction moments in the base are shown in **Figure 15**.

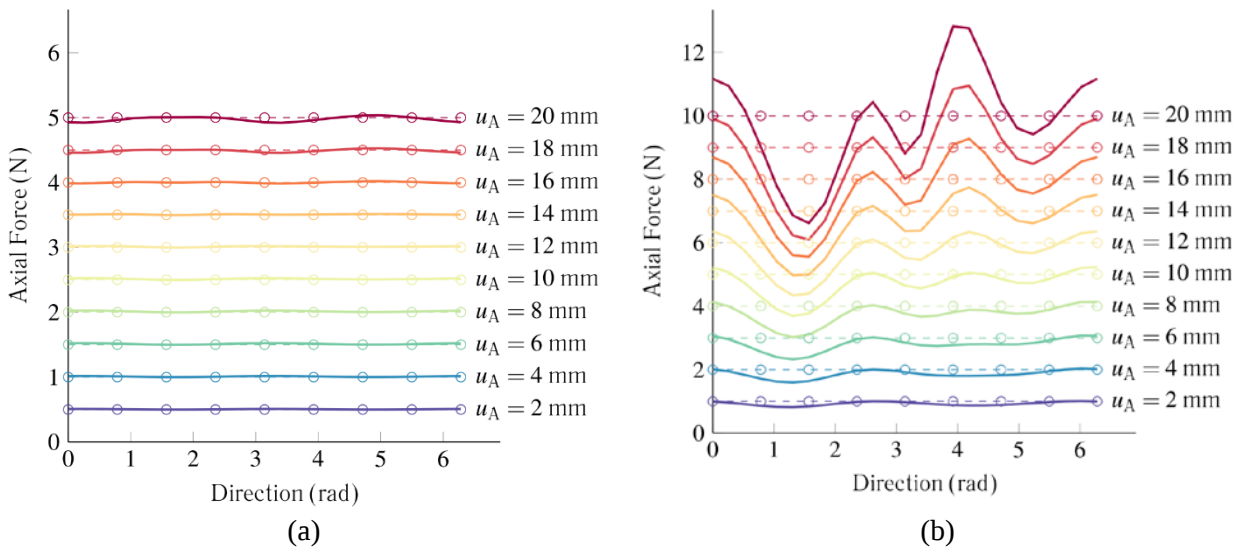


Figure 14 – Axial force profiles for the best *CP3.250* (a) and worst *CP2.500* (b) performing springs. The graphs represent force profile in circles around the zero point with different constant radii. The direction of displacement is shown on the horizontal axis.

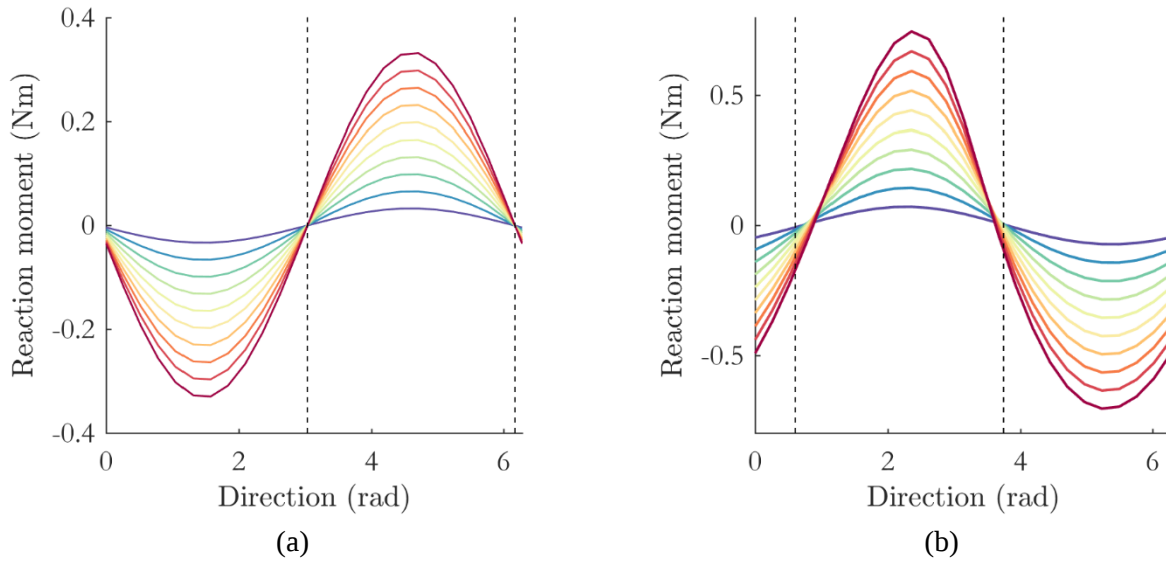


Figure 15 – The reaction moments at the base for the best *CP3.250* (a) and worst *CP2.500* (b) performing springs for displacements in different directions. The dashed lines indicate the direction in which the endpoint is displaced towards or away from the base point.

In **Table 4** both the axial and transverse errors can be seen for all designs. Only designs for stiffness $k = 250$ N/m managed to get a relative transverse error of $\varepsilon_T < 1\%$. No designs had an axial error of $\varepsilon_A < 1\%$. The results from initial designs 1, 3 and 4 result in good springs ($\varepsilon_A < 5\%$) compared to initial design 2 (with most designs $\varepsilon_A > 10\%$). Designs *CP2.625* and *CP2.750* do come to a good design, similar to the other springs of the same stiffness. It is noticed that the results found from initial designs 1, 3 and 4 were similar in shape and also have the same performance. The transverse errors follow the results of the axial errors - if the axial error is low, the transverse is consistently somewhat lower.

	Initial 1		Initial 2		Initial 3		Initial 4	
Stiffness (N/m)	ε_A (%)	ε_T (%)	ε_A (%)	ε_T (%)	ε_A (%)	ε_T (%)	ε_A (%)	ε_T (%)
250	1.821	0.961	10.742	10.125	1.817	0.957	1.822	0.961
375	2.355	1.237	24.328	14.597	2.350	1.233	2.355	1.237
500	2.816	1.473	33.820	21.261	2.816	1.473	2.816	1.473
625	3.242	1.693	3.235	1.687	3.235	1.687	3.242	1.693
750	3.623	1.886	3.629	1.891	3.623	1.886	3.624	1.887

Table 4 – The relative axial and transverse error ε_A and ε_T for all optimized clamped-pinned springs. The best and worst performing designs are highlighted in respectively green and red.

3.3 Clamped-Clamped Spring

The last new difficulty was introduced by also clamping the endpoint of the spring, which introduces extra reaction moments at the tip. Once more the optimization was run for 4 different initial designs and the range of stiffness values, of which the best and worst resulting designs are shown in **Figure 16**.

The resulting shapes either look like the shape of a horseshoe (*CC1.625*), a spiral which is less spiraling than the clamped-pinned springs (*CC1.250* and *CC1.750*) or an S-shape (*CC2.250* and *CC2.750*). Also here the design becomes smaller as a higher stiffness is required. Only in one case intersection during displacement occurs, which is for spring *CC2.625*.

Again, for the best and worst performing spring, the axial force-displacement profiles and the circular axial force profiles are respectively shown in **Figures 17** and **18**.

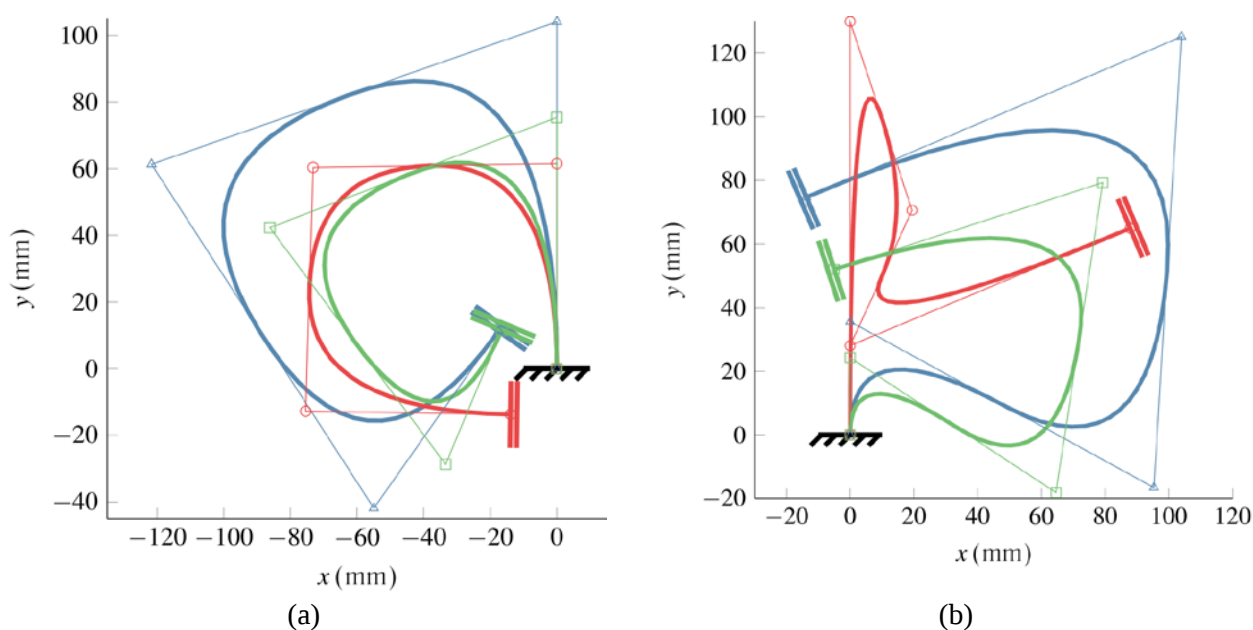


Figure 16 – Clamped-clamped spring designs performing best *CC1.250*, *CC1.625* and *CC1.750* (a) and worst *CC2.250*, *CC2.625* and *CC2.750* (b). Designs are rotated to align with each other.

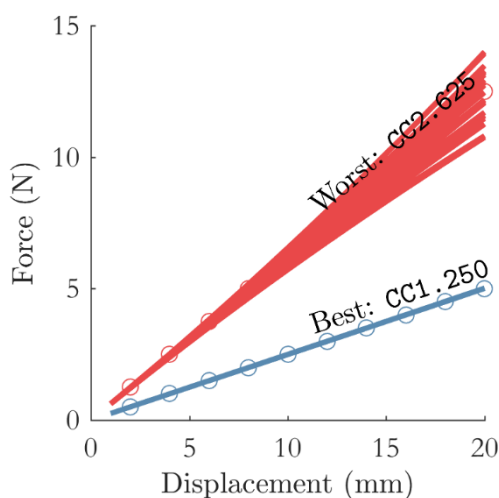


Figure 17 – The axial force-displacement profile for the best (CC1.250) and worst (CC2.625) clamped-clamped designs compared to the desired force profile. The desired forces at the sample points are indicated by circles.

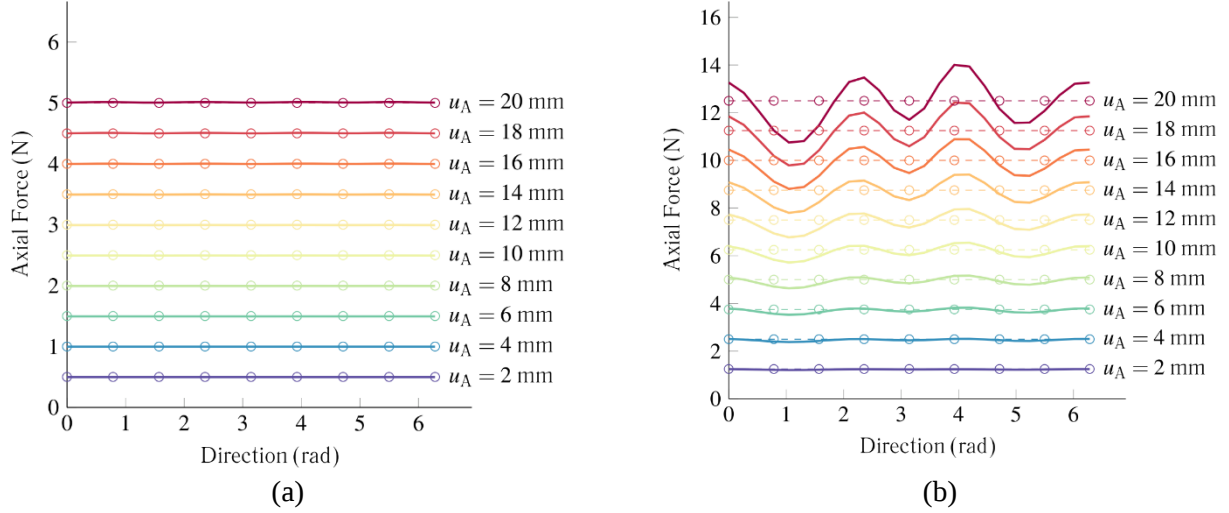


Figure 18 – The axial force profiles for the best *CC1.250* (a) and worst *CC2.625* (b) performing springs. The graphs represent force profile in circles around the zero point with different constant radii. The direction of displacement is shown on the horizontal axis.

The two kinds of errors (ε_A and ε_T) and the maximum reaction moments M are presented in **Table 5**. Most designs had a relative axial error and relative transverse error of $\varepsilon_A < 1\%$ and $\varepsilon_T < 1\%$. The results from initial design 1 are nearly identical to those of 4 (except *CC1.750* compared to *CC4.750*). Also initial design 2 and 3 are almost the same, with exception of *CC2.625*, which did not come to a good design. The reaction moments are in the range of 0.2 Nm to 0.9 Nm for all designs.

	Initial 1			Initial 2		
Stiffness (N/m)	ε_A (%)	ε_T (%)	M (Nm)	ε_A (%)	ε_T (%)	M (Nm)
250	0.406	0.102	0.211	0.660	0.287	0.368
375	0.723	0.199	0.414	0.808	0.361	0.482
500	0.859	0.242	0.507	0.953	0.430	0.587
625	0.979	0.282	0.594	13.998	10.007	0.906
750	0.839	0.270	0.375	1.164	0.547	0.770
	Initial 3			Initial 4		
Stiffness (N/m)	ε_A (%)	ε_T (%)	M (Nm)	ε_A (%)	ε_T (%)	M (Nm)
250	0.670	0.278	0.369	0.406	0.102	0.211
375	0.825	0.361	0.484	0.723	0.199	0.414
500	0.953	0.430	0.587	0.859	0.242	0.508
625	1.064	0.491	0.681	0.979	0.282	0.595
750	1.164	0.547	0.770	1.093	0.320	0.680

Table 5 – The relative axial and transverse errors ε_A and ε_T plus moments M for all optimized clamped-clamped springs. Best and worst values are indicated with respectively green and red.

3.4 Results | General Design Observations

To examine the deterioration of performance resulting from small deviations in the design, Monte-Carlo simulations were run. Small errors were introduced in each control point of the best clamped-clamped design (CC1.250). The beam design was refined by knot insertion to obtain 17 instead of 5 adjustable control points, resulting in a greater shape freedom. For each of these control points a random error in x and y -direction is added of maximum size $\sqrt{\Delta x^2 + \Delta y^2} < \epsilon_{\text{shape}}$ (see **Figure 19**). Two different sizes of shape errors are chosen as $\epsilon_{\text{shape}} = \{1.0, 5.0\}$ mm and for each of these, 1000 random variations are analyzed. This resulted in the data shown in **Figure 20**. In **Table 6** a summary of the errors is given.

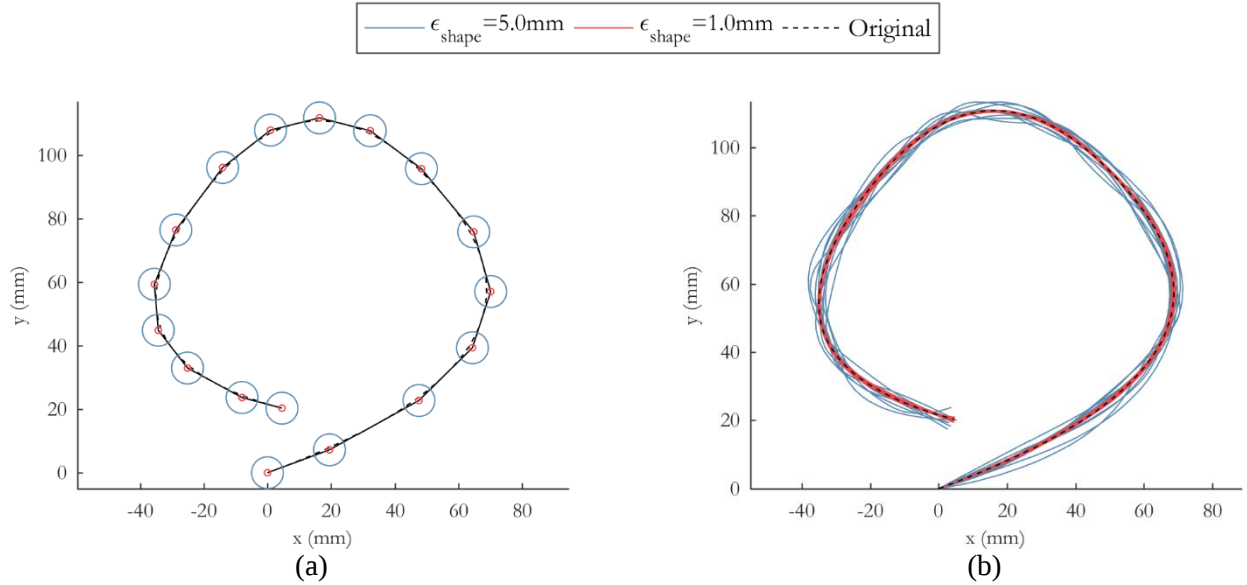


Figure 19 - The control points are randomly varied within the areas indicated in (a), of which several examples are shown in (b) for the different shape error sizes.

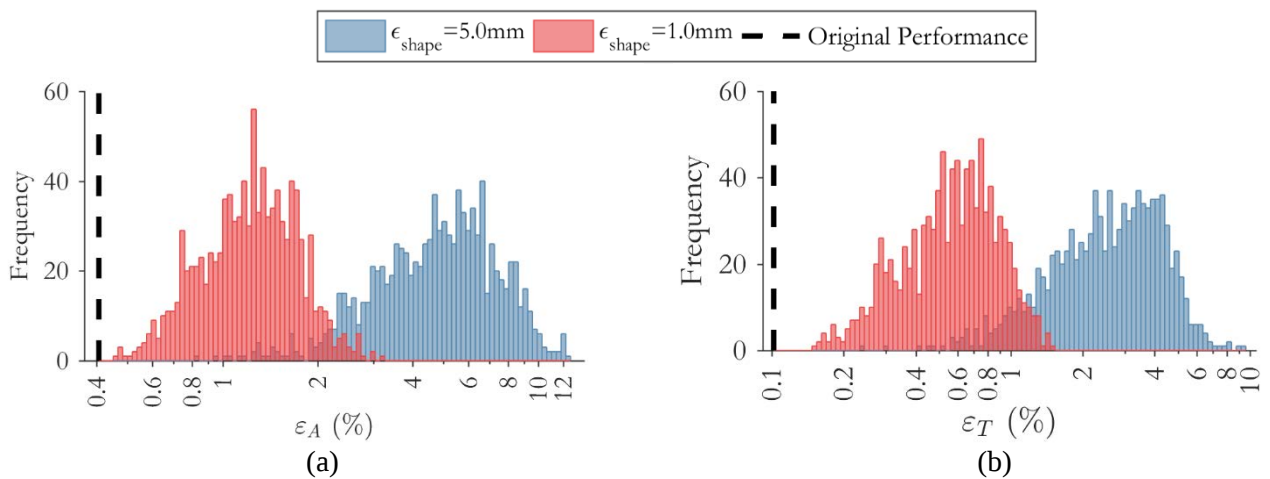


Figure 20 – The uncertainty range of spring design *CC1.250*, with the original best performance in a black dashed line. This uncertainty results in the maximum axial error distribution (a) and the maximum transverse error distribution (b) for 1000 samples in two different shape error regimes. The smaller shape error results are shown in red, the larger is shown in blue.

	Error size	Minimum error	Maximum error	Mean error
ε_A (%)	± 5.0 mm	0.823	12.64	5.146
	± 1.0 mm	0.465	3.253	1.287
	<i>Original</i>	0.406		
ε_T (%)	± 5.0 mm	0.283	9.615	1.287
	± 1.0 mm	0.149	1.485	0.610
	<i>Original</i>	0.102		

Table 6 – Summary of errors resulting from shape errors of the spring.

The maximum size measure of the spring is defined as the diameter of the smallest circle totally encompassing the spring design. This length measure is normalized by dividing with the maximum displacement range $u_{\max}$ and is shown in **Figure 21a** with the variation of normalized spring stiffness horizontally. The stiffness normalization is obtained by dividing with the Young's modulus E and the beam width w . A trend of smaller springs for a higher stiffness is observed for all kinds of springs. It is also noticed that the pinned-pinned springs are smaller than their clamped counterparts.

For each design in the maximum deformed position, the stresses were calculated. All optimizations resulted in springs which did not reach yield strength during deformation with the chosen stiffness values. Extracting the maximum occurring stresses, normalizing (by dividing with the yield strength σ_{yield}) and plotting them versus the normalized spring stiffness results in **Figure 21b**. The maximum stresses show an increasing trend when required stiffness is increased. This trend is valid for all spring versions. Finally, both the normalized maximum stresses are plotted against the normalized length measures in **Figure 22**.

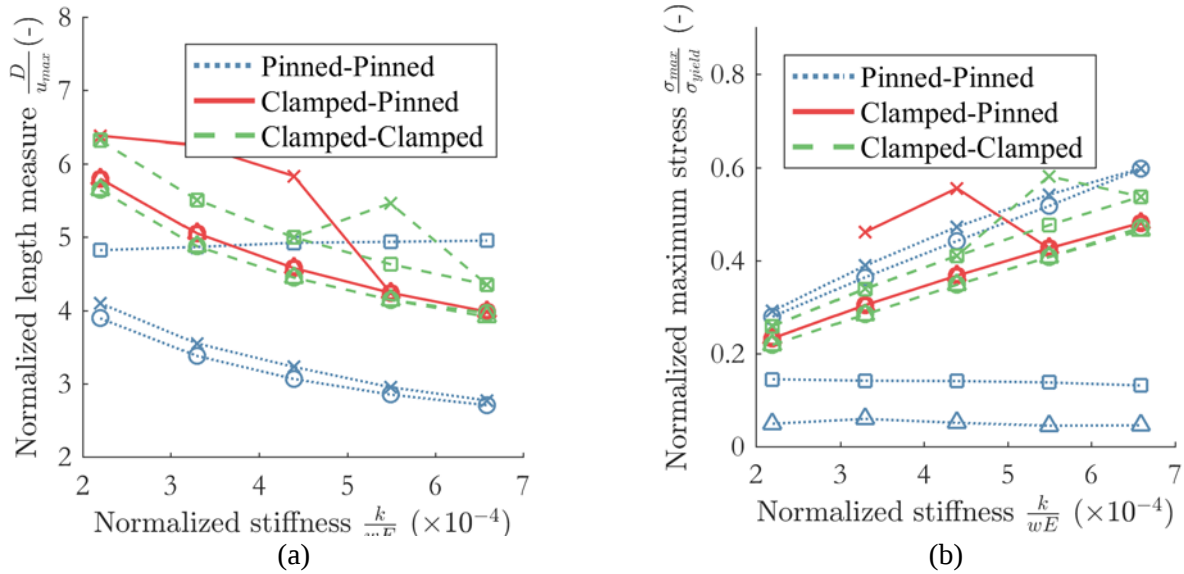


Figure 21 – (a) Different normalized length measures for different designs and (b) maximum normalized stresses for different designs plotted versus the normalized stiffness. The different line styles represent the different spring types, while the markers represent the used initial condition.

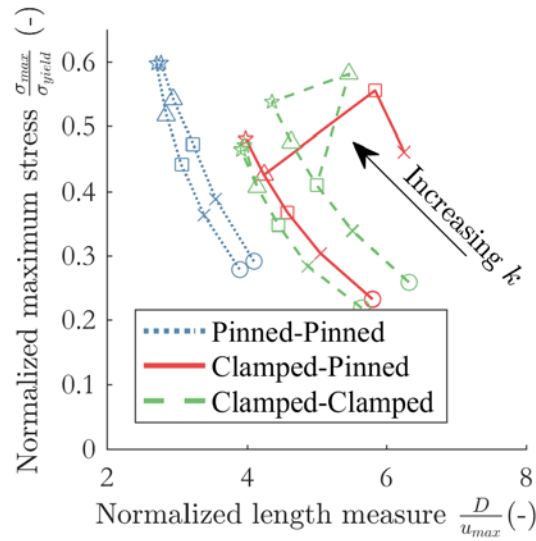


Figure 22 – The normalized length plotted against the normalized maximum stress. The different line styles indicate the different spring types and the markers represent the used initial design.

From a practical point of view it might be attractive to have a spring with a finite number of bends instead of a continuously curved beam, possibly leading to easier fabrication. Optimizations using straight beams were performed to create *CP250* and *CC250* springs. Linear interpolations with a set number of straight edges were made from the best resulting splines (*CP3.250* and *CC1.250*). The initial shapes are shown in **Figure 23 a** and the final shapes after optimization in **Figure 23 b**. Also their performance values are given in **Table 7**.

Initial design	No. of segments	ε_A (%)	ε_T (%)	M (Nm)
<i>CP3.250</i>	3	24.609	15.140	-
<i>CP3.250</i>	4	2.579	1.425	-
<i>CP3.250</i>	5	2.152	1.205	-
<i>CP3.250</i>	<i>Original spline</i>	<i>1.817</i>	<i>0.957</i>	-
<i>CC1.250</i>	3	1.824	0.612	0.402
<i>CC1.250</i>	4	0.637	0.409	0.242
<i>CC1.250</i>	5	0.453	0.113	0.229
<i>CC1.250</i>	<i>Original spline</i>	<i>0.406</i>	<i>0.102</i>	<i>0.211</i>

Table 7 - Overview of performance values for original design and optimized straight shapes.

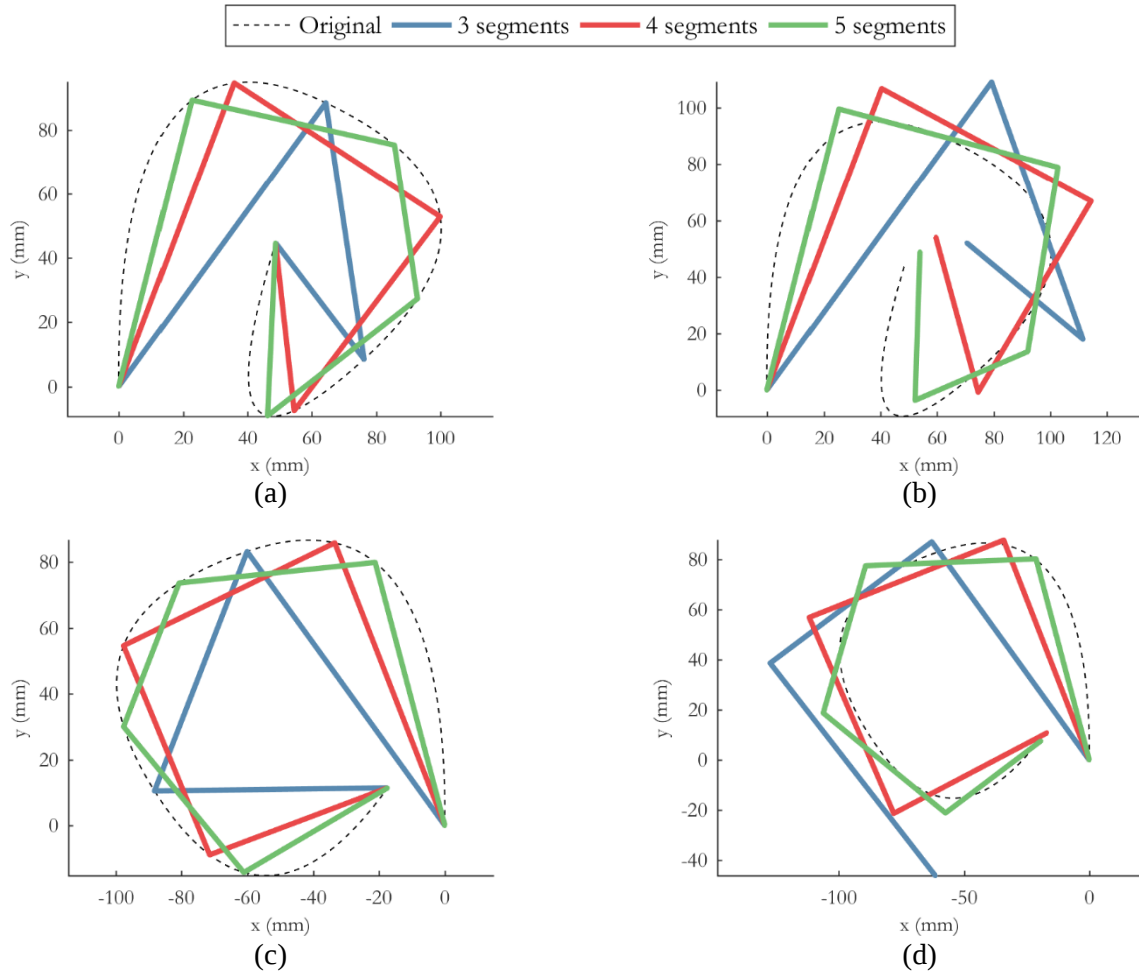


Figure 23 – The initial designs compared to the original spline shape and the optimized straight shapes for clamped-pinned springs, respectively (a) and (b). And for the clamped-clamped springs (c) and (d).

4. Discussion

Following the structure of the Results section, the findings are discussed in order of the springs defined by the three different types of boundary conditions. First, the optimization phase is treated for each spring type and secondly, the performance of the final designs. Next, the general shape of each type of spring is qualitatively analyzed and finally specific applications and restrictions of the different spring types are elaborated. The section is closed with some remarks about the general design of these springs.

4.1 Pinned-Pinned Spring

Because the pinned-pinned spring has a rotational freedom, the forces only need to be optimized over one track. In this case, the forces already point to the point of rotation, making the transverse forces automatically zero. This makes it the spring with least residuals to optimize. When looking at the performance values (**Table 3**) and comparing to the other spring types (**Tables 4** and **5**) this appears to result in lower errors (for the best performing springs). It also results in a smaller spring compared to the others (**Figure 21**).

For this spring there is great variation in the solutions found. Not every initial design gives good ZFL springs. Because this spring needs to move to the first sample point, a pre-stress was applied before starting the analysis. This gives difficulties in the cases of initial design 3 and 4 (**Table 3**). The spring starts as the initial design in a stretched configuration. To get to the first starting point a negative (pushing) force is applied, and this force remains negative over the entire displacement range from there (**Figure 9**). Now the optimizer seeks

to minimize the difference between the negative actual force and positive desired force. One option to do this is to move the endpoint closer to the begin point - helped by the additional constraint. The other is to lower the stiffness of the entire spring, thus lowering the overall forces (and hereby obtaining a lower objective value). Once the spring forces cross over to the positive side, the spring's compliance needs to be reduced again. By this time however, the optimizer also succeeded in bringing the two endpoints together, which introduces trouble. The fact that these two rotationally free points are (almost) coinciding causes a rigid body motion to occur, i.e. the solver cannot find one single solution anymore as all orientations are a valid solution, the stiffness matrix will become singular. Close to this point the design is also very sensitive in its force-response. A small change in design will cause the spring to behave entirely different. All in all, this means that the initial designs need to be chosen with care for this type of spring to prevent the optimizer to get stuck in a local minimum (10 out of 20 designs have bad performance). A possible solution for this is to choose an already designed zero free length spring with another stiffness close to the new desired stiffness as initial design.

Figure 11 shows that the force profile is smooth in between the sample points. This proves that the number of sample points is enough for this application. A sudden drop in force is observed close to a displacement of zero. This is caused by the fact that the spring is forced towards its base point instead of extending. In the final designs a tiny gap is left between the begin and endpoint of the springs. If the end point is displaced towards the begin point, the force will rapidly grow in negative direction.

The general shape of the springs is a droplet shape, where begin and endpoint are close together. To get linearity and a low enough stiffness, the spring needs length, which it generates by making the droplet shape. The short distance between begin and endpoint are a direct result of the rotational freedom of the spring and extra enforced by the helping constraint of the distance between begin and endpoint (which speeds up the optimization considerably). The shape of the best performing springs in **Figure 10a** has only three segments of considerable length, the first segment being very short (shorter than the width of the spring). This implies a redundancy in design variables, where the same spring shapes and performances might be obtained using less segments.

Possible applications for this spring are in the macro scale, where joints can be made such as weight balancing [13, 24], robotics [10, 11] and arm support [7, 8]. Especially for systems with low stiffness in transverse direction or rotation this would be a good spring, since there are no parasitic transverse forces or torques which could cause errors in the movement of the system.

4.2 Clamped-Pinned Spring

For the clamped-pinned springs multiple track directions were added, furthermore a second type of error was introduced here. Initial design 1, 3 and 4 result in good springs for all chosen stiffness values, while initial design 2 only produces two well performing springs. This shows the clamped-pinned is less inclined to get stuck in a local minimum than a pinned-pinned spring (only 6 out of 20 designs have bad performance compared to the others). Compared to the pinned-pinned springs the clamped-pinned are performing worse, which might be attributed to the added force requirements. On the other hand no convergence issues were encountered, since the newly introduced clamp prevents rigid-body motion.

Most of the designs found are around an error of 1-5% (**Table 4**), but three have errors beyond 10%. **Figure 14a** shows that the forces in between the sample points vary a little, however enough sample directions were used to capture the behavior in all directions as no strange peaks occur in between sample points. In **Figure 14b** the variation is bigger, but the error is still captured well by the chosen sample points. For larger displacements however it might be the case that the sample points are starting to move too far from each other. In this case, more sample directions or other sampling patterns might be used.

The general spring shape can be characterized as being spiral shaped (**Figures 12a** and **12b**). The low design stiffness gives need of generating enough flexure length. Combined with the fact that the stiffness needs to be equal in every direction, the material is distributed in a uniform shape with the same characteristics in every direction. To compensate the reaction moment in the base of the spring, the reaction forces at the endpoint are used. However, in the displacement direction towards and away from the base point this gives a problem. A perfect ZFL spring has in this direction only a force component aligned with the base point, thus having no moment arm to create a couple. The graph in **Figure 15** shows that the best design has negligible reaction moment in the direction aligned with the base point, while the worst design has a considerable moment. This is reflected in the spring design, which minimizes the bending moment at the base when displacing towards the base point. Also the spring becomes bigger in size due to the added directions (**Figure 21**) - in fact they are almost twice as big as the pinned-pinned springs, but also the desired force profile range is twice its size (one in forward direction and one in reverse direction).

The distance constraint (**Equation 24**) helps keeping the begin and endpoint apart from each other, to prevent contact during deformation. However, not all cases of contact during deformation are prevented by this. For instance, if the endpoint of design *CP2.500* is displaced to the left a contact point will occur. This constraint is only applied between the last and first node so it does not prevent any of the other nodes to make contact.

In practice this spring would also be used in macro applications due to the need of a joint like the pinned-pinned spring. In systems sensitive to transverse errors this spring type is less desirable. Still there are no parasitic torques possibly twisting the structure.

4.3 Clamped-Clamped Spring

The last spring type added an extra clamp, introducing a second reaction moment. Surprisingly by the addition of this clamp, the found designs performed better in axial and transverse force in comparison to their clamped-pinned counterparts (**Table 5** compared to **Table 4**) when comparing designs resulting from the same initial design and stiffness value. This could be explained by the extra reaction moment. To satisfy moment equilibrium, the moment in the clamped base has to be counter-acted. In the case of the clamped-pinned spring, there is only a reaction force counter-acting in the tip of the spring, while in the clamped-clamped spring a reaction moment is added. This gives more freedom in the optimization since the new reaction moment can help the forces in counter-acting the reaction moment in the base.

It is difficult to say whether the reaction moments (**Table 5**) are big or small since there is no reference. If the connected structure is stiff in rotation direction this influence will of course be smaller than a very compliant structure in rotational direction. This causes the need of the spring to be designed in conjunction with the entire system it will be incorporated in. For instance, a balancer with a mass on an arm [25] makes a rotation during operation. The designed spring was optimized for no rotations, so if a rotation were to be imposed,

larger errors would occur. In such a case the (usually known) rotation would need to be used in the optimization procedure, possibly resulting in different designs than presented in this work.

Compared to the other spring types, the clamped-clamped spring results are very close together, with only one spring design (out of 20) which has much higher errors than the others. This indicates very little influence of local minima far away from the optimum. There are three main shapes found (**Figure 16**), which can be identified as three local minima. The first being the shape of a horse-shoe (*CC1.625*), the second a spiral (*CC1.250* and *CC1.750*) - which is less spiraling than the clamped-pinned springs - or the third, an S-shape (*CC2.250* and *CC2.750*). Similar to previous designs also here the length in the loop of the spring is used to gain a low stiffness in every direction. The symmetry in the designs can be explained by the fact that the boundary conditions are identical on both ends. A symmetric design has equal, but opposite reaction moments in both ends (if displaced in a symmetric manner). When the ends are moving directly from and to each other, thus keeping symmetry, there is no need for the spring reaction forces to generate a couple.

This kind of spring is very useful for applications where friction is a big issue as no joints are needed. This also results in application possibilities in systems on micro-scale.

4.4 General

In all cases spring designs were found, having a large improvement in performance on the initial design (compare **Figure 9** to **Figures 11, 13** and **17**). From the optimization results and previous discussion, it can be noted that the pinned-pinned spring is hardest to optimize, followed by the clamped-pinned and the clamped-clamped spring. For future research, (analytical) methods should be developed which can provide better initial designs. For instance, the linear stiffness in different directions can be optimized first before optimizing the non-linear force-displacement response.

The process of designing a spring starts with the requirements given on spring stiffness k and displacement range $u_{\max}$. Then, depending on application, the spring type can be chosen. After that, a suitable material is chosen. The parameters of the beam cross-section are chosen by technical capabilities (e.g. laser cutting), material type (e.g. sheet metal) and/or available design space. Furthermore, the maximum segment length can be adjusted to the available design space. With this information the optimization can be executed. If the results are not as desired, iterations can be done by changing the beam cross-section. Thicker beams will result in springs with higher stiffness, but also higher stresses. **Figure 21a** shows that the spring size is increasing for lower values of stiffness. The stiffness the designer is able to obtain is bounded below by limitations in space within or around the structure and the upper bound is determined by the minimal design area (e.g. minimum feature size). The yield strength also determines the upper bound for the stiffness, since the stresses are increasing for springs with higher stiffness (**Figure 21b**). In future research, focus should be on determining suitable cross-section parameters before starting optimization. If approximations can be made for maximum stresses or spring size, this would reduce the number of design iterations to be made.

Constraints were added to prevent intersections from happening. One constraint focused on preventing intersections in the initial (undeformed) state, while another focused on preventing the end-point from being too close (or too far away in the case of the pinned-pinned spring). From the results, only a few spring designs were found to intersect during displacement despite the constraints used. In the pinned-pinned case no intersections were detected, for the clamped-pinned spring only two (out of 20) springs and for the clamped-

clamped spring only one. These were coincidentally also bad-performing springs compared to the others. This leads to the thought that the constraints might obstruct the design from getting better by blocking certain directions. This is a problem that has to be looked into in future research. Also, some designs might be excluded from the feasible design set due to the constraints. Cases exist where the spline scaffold is intersecting, but the spline itself is not.

No spring can be manufactured in exactly the dimensions given. So from a manufacturing point of view it is interesting to see the performance decay as the spring dimensions are affected by slight random variations. From the Monte-Carlo simulations, a shape error of $\pm 1.0\text{mm}$ gave axial errors of average 1.29%. An error of $\pm 5.0\text{mm}$ averaged to an axial error of 5.15%, compared to the axial error of the original spline being 0.41%. Although small errors in the design do not directly yield large errors in performance, depending on the application, the shape needs to be controlled to the right amount to prevent the errors from getting too large.

A way which might make production easier is to use a finite number of bends instead of a continuously curved spring. The effect of this was analyzed using a second optimization using straight segments, starting from a linear interpolation of the continuous spline. The straight beams tend to keep the global form of the spline, with performance getting closer to the original spring as the number of segments is increased. This behavior was showed for both the clamped-pinned and the clamped-clamped spring. It might well be that these final straight springs have a sub-optimal behavior in their entire shape freedom, but using the spline design is a reliable way to start the optimization. By using enough bends, fabrication might become a lot easier while still maintaining good performance.

Fabrication proved to be very difficult, as different approaches did not yield a satisfactory result. Three methods were investigated, the first being laser cutting the spring out of PTFE. Creating the curved shapes using laser cutting was successful, however the material properties of PTFE are less than optimal for a spring. Effects such as hysteresis and creep dramatically distorted the force-displacement profile that no linear behavior could be observed. The second method was by bending a strip of spring steel. However, due to the high yield strength of this material (which is needed to obtain a spring capable of large deflection) the springback was very large and we did not succeed in creating the proper form needed. Finally we tried to produce a spring by wire EDM out of Titanium Ti6Al4V. The long and slender shape introduced too much vibrations for the shape to be cut out properly.

Not in all applications a full circular range will be needed. Some systems only require one path or an oddly shaped area to have zero free length properties. Using this fact, application specific spring could be designed, behaving more precise in the required area. This way for systems which are more critical with respect to transverse errors or reaction moments, a customized spring can be made.

5. Conclusions

A zero free length spring is an engineering element which is useful, yet difficult to create. A new method of designing in-plane compliant zero free length springs is proposed, based on the given desired force characteristics within a certain range (up to 20 mm displacement). This is done for three different types of springs, which differ in boundary conditions; the pinned-pinned, clamped-pinned and clamped-clamped springs. The first two are most suitable for macro applications. The latter for micro applications, since it has no need for joints. From a set of initial designs, non-linear least squares optimization is applied to obtain

optimized designs for different values of desired stiffness (250-750 N/m). Constraints are introduced to prevent the spring from intersecting with itself (during deformation).

The clamped-clamped spring is least affected by the choice of initial design, since 19 out of 20 designs yield good results compared to each other. For the clamped-pinned this is 14 and for the pinned-pinned 10, making it more sensitive to the choice of initial design. The constraints work well, since no constraints are violated. However, some designs still have intersections during deformation (2 of type CP and 1 for CC). Still some improvement on the constraint functions can be made, because some feasible solutions are considered infeasible by the constraints, and vice versa. For each type of spring, characteristic shapes are identified from the resulting designs and their shape is analyzed. The pinned-pinned spring is shaped like a droplet, with the begin and endpoint close together. The clamped-pinned spring prefers a spiral shape and the clamped-clamped spring a symmetric horseshoe shape. Considering only the well performing springs, the pinned-pinned and clamped-clamped springs perform best in general with maximum axial force errors of less than 1%. The worst performing springs are the clamped-pinned springs with axial force errors of 1-2%. The clamped-pinned springs have problems compensating the reaction moment, which the clamped-clamped springs can counteract with a moment in the other end of the spring, causing better performance. The pinned-pinned springs have good performance because of the small number of residuals to optimize. They lack transverse errors, which the other two types have.

The feasible values of stiffness are bounded by design area, minimum feature size and yield strength of the material. For increasing stiffness with constant beam cross-section, the maximum stresses increase, while the spring size becomes smaller. More defined relations between cross-section, maximum stresses and spring size are to be established in further research. Finally, it is observed that small errors in the design yield errors in the obtained force profile. Shape errors in the design of ± 1.0 mm and ± 5.0 mm in the best optimized clamped-clamped spring, result in an average performance deterioration of maximum axial relative error of respectively 1.3% and 5.1%, compared to an original error of 0.4%. Also the introduction of straight segments instead of a continuously curved beam, introduces a deterioration of error, diminishing as more segments are used.

References

- [1] L. J. B. LaCoste, "A New Type Long Period Vertical Seismograph," *Journal of Applied Physics*, vol. 5, no. 7, p. 178, 1934. DOI: 10.1063/1.1745248
- [2] L. J. B. LaCoste and A. Romberg, "Force measuring device," U.S. Patent US2293437, August 18, 1942.
- [3] G. Cawardine, "Improvements in Elastic Equipointing Mechanisms," U.K. Patent GB404615, Jan 4, 1934.
- [4] D. A. Streit and E. Shin, "Equilibrators for Planar Linkages," *Journal of Mechanical Design*, vol. 115, no. 3, pp. 604–611, 1993. DOI: 10.1115/1.2919233
- [5] J. L. Herder, "Design of spring force compensation systems," *Mechanism and Machine Theory*, vol. 33, no. 1-2, pp. 151–161, 1998. DOI: 10.1016/S0094-114X(97)00027-X
- [6] M. Schenk, S. Guest, and J. Herder, "Zero stiffness tensegrity structures," *International Journal of Solids and Structures*, vol. 44, pp. 6569–6583, oct 2007. DOI: 10.1016/j.ijsolstr.2007.02.041
- [7] B. Mastenbroek, E. de Haan, M. van den Berg, and J. L. Herder, "Development of a Mobile Arm Support (Armon): Design Evolution and Preliminary User Experience," in 2007 IEEE 10th International Conference on Rehabilitation Robotics, pp. 1114–1120, IEEE, jun 2007. DOI: 10.1109/ICORR.2007.4428563

- [8] A. G. Dunning, "Slender spring systems," PhD dissertation, Delft University of Technology, 2016. DOI: 10.4233/uuid:50404d39-d6aa-4fd4-bb20-04794ec01b7b
- [9] G. Brown and A. O. DiGuilio, "Support Apparatus," U.S. Patent US4208028, June 17, 1980.
- [10] C. M. Gosselin and J. Wang, "Static balancing of spatial six-degree-of-freedom parallel mechanisms with revolute actuators," *Journal of Robotic Systems*, vol. 17, no. 3, pp. 159–170, 2000. DOI: 10.1002/(SICI)1097-4563(200003)17:3<159::AID-ROB3>3.0.CO;2-J
- [11] M. Vermeulen and M. Wisse, "Intrinsically Safe Robot Arm: Adjustable Static Balancing and Low Power Actuation," *International Journal of Social Robotics*, vol. 2, pp. 275–288, sep 2010. DOI: 10.1007/s12369-010-0048-9
- [12] P. N. Kooren, A. G. Dunning, M. M. H. P. Janssen, J. Lobo-Prat, B. F. J. M. Koopman, M. I. Paalman, I. J. M. de Groot, and J. L. Herder, "Design and pilot validation of A-gear: a novel wearable dynamic arm support," *Journal of NeuroEngineering and Rehabilitation*, vol. 12, p. 83, dec 2015. DOI: 10.1186/s12984-015-0072-y
- [13] J. Herder, *Energy-free systems: theory, conception, and design of statically balanced spring mechanisms*. PhD dissertation, Delft University of Technology, 2001. DOI: 10.13140/RG.2.1.3942.8966
- [14] R. Barents, M. Schenk, W. D. van Dorsser, B. M. Wisse, and J. L. Herder, "Spring-to-Spring Balancing as Energy-Free Adjustment Method in Gravity Equilibrators," *Journal of Mechanical Design*, vol. 133, no. 6, 2011. DOI: 10.1115/1.4004101
- [15] L. C. Leishman and M. B. Colton, "A Pseudo Rigid-Body Model Approach for the Design of Compliant Mechanism Springs for Prescribed Force Deflections," *ASME 2011 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference*, no. October, 2011. DOI: 10.1115/DETC2011-47590
- [16] L. Piegl and W. Tiller, *The NURBS Book*. Berlin Heidelberg: Springer-Verlag, 2 ed., 1996.
- [17] T. J. R. Hughes, J. A. Cottrell, and Y. Bazilevs, "Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement," *Computer Methods in Applied Mechanics and Engineering*, vol. 194, no. 39-41, pp. 4135–4195, 2005. DOI: 10.1016/j.cma.2004.10.008
- [18] J. A. Cottrell, T. J. R. Hughes, and Y. Bazilevs, *Isogeometric Analysis: Toward Integration of CAD and FEA*. Chichester: John Wiley & Sons, Ltd, 2009. DOI: 10.1002/9780470749081
- [19] A. P. Nagy, *Isogeometric Design optimisation*. Phd dissertation, Delft University of Technology, 2011.
- [20] G. Radaelli and J. L. Herder, "Isogeometric Shape Optimization for Compliant Mechanisms With Prescribed Load Paths," in *IDETC/CIE 2014*, no. August, (Buffalo, New York, USA), 2014. DOI: 10.1115/DETC2014-35373
- [21] X. Qian, "Full analytical sensitivities in NURBS based isogeometric shape optimization," *Computer Methods in Applied Mechanics and Engineering*, vol. 199, no. 29-32, pp. 2059–2071, 2010. DOI: 10.1016/j.cma.2010.03.005
- [22] MathWorks. (2016). "Least-Squares (Model Fitting) Algorithms," [Online].
- [23] K.-T. Fang, R. Li, and A. Susjanto, *Design and Modeling for Computer Experiments*. Boca Raton: Taylor & Francis Group, LLC, 2006.
- [24] S. R. Deepak and G. K. Ananthasuresh, "Perfect Static Balance of Linkages by Addition of Springs But Not Auxiliary Bodies," *Journal of Mechanisms and Robotics*, vol. 4, no. 2, p. 021014, 2012. DOI: 10.1115/1.4006521

[25] J. L. Herder, R. Barents, B. M. Wisse, and W. D. V. Dorsser, “Efficiently variable zero stiffness mechanisms,” in 4th International Workshop on Human-Friendly Robotics, pp. 2–3, 2011.