Given a helical compression spring in a spring-mass-damper system, what are optimal springs?

Alistair Bentley[1], Tim Hodges[2], Jiahua Jiang[3], Justin Krueger[4], Saideep Nannapaneni[5], Tianyu Qiu[6]

Problem Presenters: Jordan Massad[7], Sean Webb[8]; Faculty Mentors: Ilse Ipsen[9], Ralph Smith[10],

Abstract

1 Introduction

Springs have many everyday uses, and we have studied them thoroughly as a result [**?**]. We find them in cars, homes, gyms, and industry, and one common use of helical compression springs, in particular, is in mechanical switches [**?**]. Acceleration switches, such as the one seen in Figure 1, are used to close electrical circuits once the object carrying the switch reaches a certain acceleration. The main use of such switches is to provide power to sensory equipment and data recorders for very precise amounts of time, and one example of their use is collecting data from rocket sled experiments.

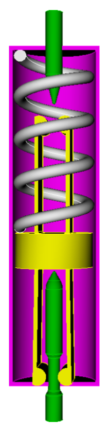


Figure 1: An example of an acceleration switch, see reference [**?**].

These rocket sled tests include very high velocities and accelerations and end in a destructive impact, and since the test is not easily replicated, we must have the ability to consistently and correctly collect data. This means we require that the spring in the acceleration switch must behave as expected to the forces exerted on it, and having a switch that is closed earlier or later than expected is undesirable [**?**]. Designing a spring to behave as desired is a complicated problem that requires satisfying numerous constraints on the spring's physical properties. That the desired spring can vary depending on the application further complicates the problem.

Designing an optimal spring is not a new problem, though. Brake et al. considered the design of an acceleration switch with enabled uncertainty [**?**]. Alternatively, Sastry et al. implemented probabilistic response

[1] Mathematics, Clemson University
[2] Mathematics, Colorado State University
[3] Mathematics, University of Massachusetts Dartmouth
[4] Mathematics, Virginia Tech
[5] Civil & Environmental Engineering,Vanderbilt University
[6] Mathematics, University of Delaware
[7] Sandia National Laboratory
[8] Sandia National Laboratory
[9] Mathematics, North Carolina State University
[10] Mathematics, North Carolina State University

surface methodology to investigate the complications of uncertainty in designing a spring[?]. Others have considered the reduction of the optimization problem to focus on a single parameter [?].

Spring design under these conditions amounts to a constrained optimization problem where certain properties of the spring may need to be maximized or minimized subject to bounds on some or all of the spring's other properties. Extending the scope of focus beyond springs' use in mechanical switches, which already cover a vast spread of specific spring designs, results in even more possibilities for design specifications. Rather than developing methods to solve these problems on an ad hoc basis, developing tools to address a wide variety of constrained optimization problem formulations for spring design would be beneficial.

Similar tools have been developed in the past [?], and in this manuscript we build on this work. Motivated by the premise of the workflow shown in Figure 5, we produce a novel tool with new capabilities.

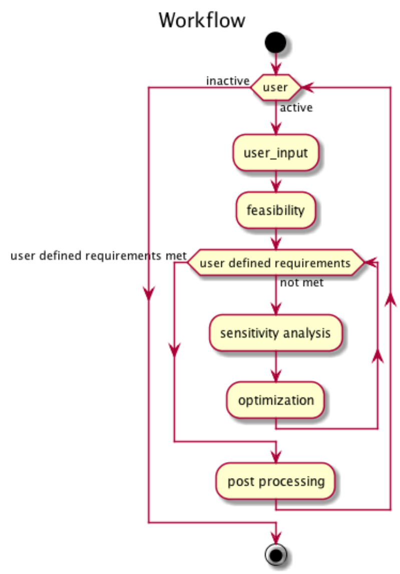


Figure 2: Illustration of the flow of our approach.

The novelties we include are the inherit flexibility in our tool's framework, tools to visualize the existence of solutions for constrained optimization problems, sensitivity analysis capabilities, and design optimization under uncertainty. We also discuss the inclusion of stress relaxation constraints in the spring design, which to our knowledge has not been considered before. We provide details regarding these items as follows. Section **??** formally defines the spring design problem for helical compression springs. Section **??** expands on the workflow shown in Figure 5. Section **??** shares cases studies using our approach, and Section **??** summarizes our work and provides suggestions for future work.

2 Helical Compression Springs

Helical springs are the typical spring that comes to most peoples mind, for illustrative purposes, see figure 3[?].

The above spring has many design parameters. These range from physical parameters such as wire diameter, to purely empirical attributes, for example spring rate. The empirical attributes have to be assumed in a range of values. This is because for extreme values these attributes exhibit behavior that is not under consideration. Below is a list of a spring's key design parameters and empirical attributes. We have added a few illustrations to illustrate some of the parameters. [?]

1. Spring's inner diameter d_i, illustrated in figure 4b.
2. Spring's outer diameter d_o illustrated in figure 4a.

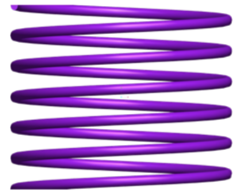
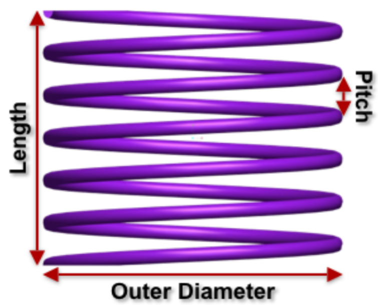
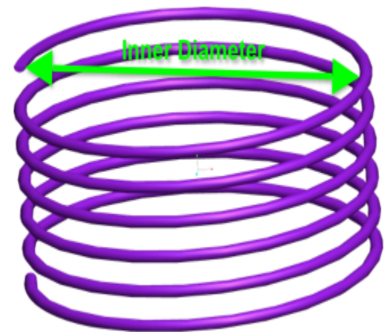


Figure 3: An example of a helical compression spring.



(a) Pitch, outer diameter, and length of a spring.



(b) Inner diameter

Figure 4: A few illustrations of parameters for a helical compression spring.

3. Spring's wire diameter d_w.
4. Total number of spring coils N_t.
5. Active number of spring coils N_a, active coils are not touching any other coils, and is subject to the spring being closed or open.
6. Pitch p illustrated in figure 4a.
7. Spring's free length L_{free}, the spring's length without any force applied.
8. Spring's solid length L_{solid}, the spring's length when all coils are compressed together.
9. Spring's open length L_{open}, the spring's length at open position, open is beginning state.
10. Spring's open length L_{close}, spring length at close position, close is the ending state.
11. Spring's open length L_{hard}, the maximum a spring can compress for the application.
12. Spring's open force F_{open}, this is the force on the spring in open position.
13. Spring's shear modulus G, this is determined by the material of the spring.
14. Spring's youngs modulus E, this is determined by the material of the spring.
15. Spring's poisson ratio ν, this is determined by the material of the spring.
16. Ultimate torsional stress, UTS.

One can see that the above parameters are physical design parameters. Each of these are subject to design tolerances. Below is a set of empirical attributes that within a range of values exhibit typical behavior for a spring. We will assume that these attributes are within that range.

Spring Rate:

$$k = \frac{G}{8N_a} \frac{d_w^4}{(d_i + d_w)^3} \tag{1}$$

Spring Index:

$$C = \frac{d_i}{d_w} + 1. \tag{2}$$

Coil Binding Gap

$$g = \frac{L_{hard} - L_{solid}(d_w, N_a; ec)}{N_t - 1} \tag{3}$$

Max Shear Stress

$$\frac{G(L_{free} - L_{hard})}{4\pi N_a(ec)} \left[\frac{d_w(4d_i^2 + 9.46 d_i d_w + 3d_w^2)}{d_i(d_i + d_w)^3} \right] < UTS \tag{4}$$

Diametral Expansion

$$d_{expand} = d_w + \sqrt{(d_i + d_w)^2 + \frac{p^2 - d_w}{\pi^2}} \tag{5}$$

3 The Problem

Add dependency graph.

Design an algorithm that optimizes springs with interchangeable objectives and constraints. In addition, attempt to incorporate properties stress relaxation and creep into the available objectives and constraints.

A list of possible constraints/objectives given in minimization form is below

1. $d_i < d_i^{max}$
2. $d_i < d_o$
3. $d_i + 2d_w < d_o^{max}$
4. $d_{expand} - d_0^{max} < 0$
5. $\frac{G}{8N_a} \frac{d_w^4}{(d_i + d_w)^3} - k_{max} \leq 0$
6. $\frac{d_i}{d_w} + 1 - C_{max} < 0$
7. $(L_{free} - L_{open}) \frac{G}{8N_a} \frac{d_w^4}{(d_i + d_w)^3} - F_{open} = 0$
8. $\frac{L_{hard} - L_{solid}}{N_t - 1} + g_{min} \leq 0$
9. $\frac{L_{free}}{d_i + d_w} - \pi \sqrt{\frac{2(2\nu + 1)}{\nu + 2}}$
10. $-UTS + \frac{G(L_{free} - L_{hard})}{4\pi N_a(ec)} \left[\frac{d_w(4d_i^2 + 9.46 d_i d_w + 3d_w^2)}{d_i(d_i + d_w)^3} \right] < 0$

This list is not exhaustive as we want a user to be able to define new constraints and objectives. In addition to these we should be able to minimize any parameter of the spring. The incorporation of stress relaxation and creep have been left as computational experiments. If the reader wishes to read about these please check out section 7.1.

In order to simplify the problem we have made some assumptions. First, we have assumed that constraints and objectives are all in terms of the number of total coils. This is to reduce the complexity of knowing if a constraint is dependent on the active number or total number of coils. These two numbers are dependent on the end conditions of the spring, whether it is open or closed at the end.

For this reason we also will consider all springs to be have a closed end condition. With these assumptions we simplify our constraints for pitch, p, solid length, L_{solid}, and the diametral expansion, d_{expand}.

JUSTIFY ASSUMPTIONS These assumptions have reduced the set of design parameters and have allowed us to simplify the problem. Without these assumptions we would need to deduce at run time what formulas we can use and this is a challenging problem from an implementation stand point. Also, we have a very complicated formula for diametral expansion if the end conditions are not closed. This presents issues of solving a cubic without the ability of discerning if the roots will be physically meaningful.

Describe data, how collected, their properties (removed noise, filled in missing data, applied normalizations).

4 The Approach

Present and justify approach for solving the problem Explain advantages of approach over existing ones Tell a story To be as flexible as possible we must be able to handle constraints and objectives that have an unknown number of variables a priori. To handle this uncertainty we instead of evaluating an objective or constraint with a set of values we evaluate with a "spring object" in the programming sense [**?**]. This spring has all the attributes of interest and the function merely picks what values it needs for evaluation.

This approach is also implemented in the design of a constraint and objective. An objective is an expression that we must be able to evaluate. A constraint is an expression that we must evaluate and check if this evaluation violates the constraint condition. We use the idea of inheritance and let a constraint inherit the properties of being an objective with additional framework to compare to a condition. This allows the user to interchange constraints and objectives seamlessly.

Along with the flexible software design, we have implemented the ability to run feasibility and sensitivity checks before trying to optimize. This will allow the user to know additional information about the problem. First, from feasibility, the user will know quickly if the run is unable to find a feasible point to run optimization. Second, from sensitivity, the user can know which parameters will dramatically change the optimization versus which parameters are unnecessarily adding to the dimension of the optimization space.

5 Workflow

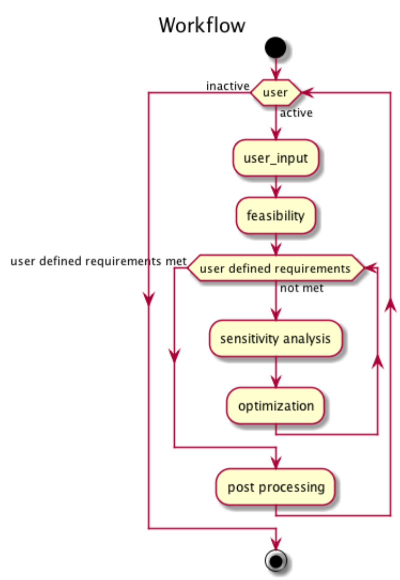


Figure 5: Illustration of the flow of our approach.

This may not be accurate anymore, need Justin's input. Above is an illustration of the workflow that is our approach. This section will go in depth into the feasibility, sensitivity analysis and optimization of our approach.

Should we add a workflow of feasibility, sensitivity, and optimization?

5.1 Feasibility

The feasibility of a solution depends on if a solution satisfies all the constraints in the problem, as well as, constraints that are not included. We also need a feasible solution to start the optimization process, otherwise we may never find an solution that is feasible and optimal. Implementing this requirement is simply sampling the available space and checking each sample against all constraints.

Refer to section 6 to see plots that are showing feasibility given two or three state variables.

5.2 Sensitivity Analysis

As the dimension of design variable space increases, the computational expense of the optimization procedure increases. To reduce the computational expense, it is often desirable to reduce the design variable space by removing the variables that have very little influence on the objective function. Thus, a dimension reduction strategy is required to reduce the design variable space. Dimension reduction approaches have been divided into two categories - (1)filter approach, and (2)wrapper approach. In the filter approach, the input variables are ranked according to a ranking criterion and the most dominant variables can be selected by assuming a threshold influence value. In the wrapper approach, a subset of variables is selected from the list of all possible subsets of the input variables that best estimate the output variable. Sensitivity analysis, a filter approach, is used in this work for dimension reduction. Two types of sensitivity analysis have been developed in the literature - local sensitivity analysis and global sensitivity analysis. The local sensitivity index of a variable measures the sensitivity of the model output when the variable is fixed at a single value whereas the global sensitivity (GSA) index measures the variation of model output when the variable is varied over its range. Therefore, GSA is used as it considers the entire range of a variable in computing the sensitivity to the output. Note that the input variables represent the design variables and model represents the objective function. Consider a objective function, G, with n design variables given by x_1, x_2, ... x_n, given by

$$Y = G(x_1, x_2, ...x_n)$$

In GSA, two types of indices can be calculated for each variable - first order index and total effects index. The first-order index (S_i^I) quantifies the uncertainty contribution of an input variable, without considering its interactions with other variables, to the output variable uncertainty. Similarly, the total effects index (S_i^T) quantifies the uncertainty contribution of an input variable by considering its interactions with all variables, to the output uncertainty. The expressions for the two sensitivity indices are given below as

$$S_i^I = \frac{V_{X_i}(E_{X_{-i}}(Y|X_i))}{V(Y)}$$

$$S_i^T = \frac{E_{X_{-i}}(V_{X_i}(Y|X_{-i}))}{V(Y)}$$

Given a design range (lower and upper bounds), a variable can be assumed to be uniformly distributed in the design range. For each variable, the first-order index is calculated and if it is less than an assumed threshold value, then that variable is assumed insensitive and removed from the optimization procedure. Thus, dimension reduction is implemented for a faster design optimization.[**?**] [**?**] [**?**] [**?**]

5.3 Optimization

A general constrained design optimization problem can be formulated as follows

$$Min \;\; f(X)$$

such that

$$g(X) \leq 0$$

$$lb_X \leq X \leq ub_X$$

Several optimization algorithms (both local and global) are available to solve the above optimization problem. Two algorithms - BFGS (local) and DIRECT (global) have been tried for design optimization. The key differences between the two algorithms are described below. BFGS algorithm requires the gradient and Hessian of the objective function, and also an initial point for optimization whereas DIRECT does not require the objective function to be differentiable. Since DIRECT is a global algorithm, it does not require an initial point. A downside of DIRECT is that it requires more computations compared to BFGS. The 'fmincon' function in MATLAB, which implements the BFGS algorithm, requires the linear and non-linear constraints to be provided separately. Separation of linear and non-linear constraints is hard to implements in an automated software framework because they are problem-dependent whereas DIRECT does not require such segregation between constraints. Therefore, DIRECT global algorithm is used for optimization.

It is also essential to account for the variability in the manufacturing process(tolerance) in the design of springs. The tolerance for each of the spring parameters is assumed to be equal to one percent of the value of the variable. Thus, each variable follows a uniform distribution with unknown mean and known variance. The optimization formulation after accounting for tolerances can now be written as

$$Min \;\; \mu_f(X, d)$$

such that

$$Pr(g_i(X, d) \leq 0) \geq p_t^i$$
$$Pr(X \geq lb_X) \geq p_{lb}$$
$$Pr(X \leq ub_X) \geq p_{ub}$$
$$lb_d \leq d \leq ub_d$$

where X, d represent the design variables with tolerances and non-design variables with tolerances respectively. The first constraint represents the probabilistic inequality constraint and the other constraints represent the bounds for the design variables. Optimization with tolerance is a nested double loop process where optimization is carried out in the outer loop (using DIRECT)and in each iteration of optimization, reliability analysis is carried out in the inner loop to check the probabilistic constraints. In this work, Monte Carlo simulations are used to carry out reliability analysis.

[?] [?] [?] [?]

6 Computational Experiments

6.1 Case 1:

Objectives: Minimize spring rate and spring index.
Constraints: Inner diameter and outer diameter relation, elation on inner, wire, and outer diameter, coil binding gap, buckling slenderness, and maximum shear stress.
State Variables: d_i, d_w, and N_t

6.2 Case 2:

Objectives: Minimize spring rate and spring index.
Constraints: Relation on inner, wire, and outer diameter, diametral expansion, coil binding gap, buckling slenderness, and maximum shear stress, stress relaxation.
State Variables: d_i, d_w, and N_t

6.3 Case 3:

Objectives: Minimize force at open position.
Constraints: Relation on inner, wire, and outer diameter, diametral expansion, coil binding gap, buckling slenderness, and maximum shear stress.
State Variables: d_i, d_w, and N_t

6.4 Case 4:

Objectives: Maximize stress relaxation
Constraints: Relation on inner, wire, and outer diameter, diametral expansion, and coil binding gap.
State Variables: d_i, d_w, and N_t

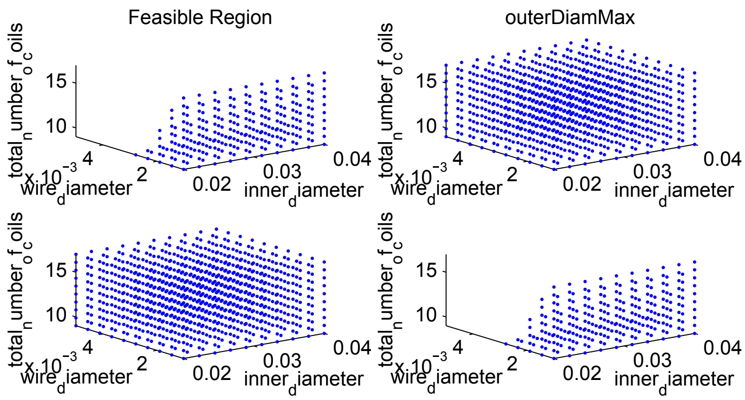


Figure 6: Feasible region for case 4.

Axes need to be worked on.

6.5 Case 5:

Objectives: Minimize spring rate and maximize stress relaxation.
Constraints: Relation on inner, wire, and outer diameter, diametral expansion, coil binding gap, and maximum shear stress.
State Variables: d_i, d_w, and N_t

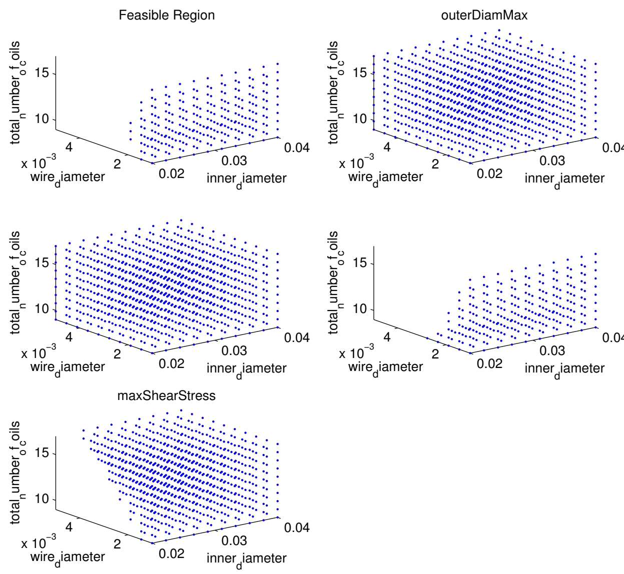


Figure 7: Feasible region for case 5

Axes need to be worked on.

6.6 Case 6:

Saideep added a 7a, need to deal with it.
Objectives: Minimize spring rate and coil binding gap.
Constraints: Relation on inner, wire, and outer diameter, diametral expansion, coil binding gap, buckling slenderness, and maximum shear stress.
State Variables:d_i, d_w, L_{hard}, and N_t

7 Summary and Future Work

The ability to interchange constraints and objective functions with any number of design variables allows the user the utmost flexibility. With the addition of feasibility and sensitivity analysis it is possible for any configuration of objective function, constraints, and design variables to be analyzed for refinement. The quantification of stress relaxation and creep allow the user a chance to incorporate these properties into any configuration, especially those that have never been tested.

Some limitations of the approach outlined are as follows. The choice of optimization and sensitivity analysis are fixed, however, they are modularized to allow a different optimization routine and sensitivity analysis to be ported in. Given the amount of flexibility that is enabled, a user will have to be able to decide if a infeasible solution is due to user error.

Future work is bountiful for this approach. More analysis of the stress relaxation and creep could result in better performance. An in depth analysis of different models of stress relaxation and their performance in our model would be beneficial. The flexibility allows the inclusion of many different models of stress relaxation and creep to be added

Senstivity for optimization

Uncertainty Quantification incorporation into the model.

7.1 Basics about uni-axial creep and stress relaxation

There are many models for stress relaxation and un-axial creep. We choose a model outlined in **CITE!**. For this model the spring is heated to $0.3 - 0.5T_m$ (T_m is the melting temperature of the material) and loaded by a tensile force. The induced normal stress σ is much less than the yield limit of the material σ_y. The load and temperature are constant throughout the test. The strain ϵ^{cr} will slowly increase.

Conventionally, creep can be divded into 3 stages. In the first stage(primary/reduced/trasient) ,the creep strain rate decreases to a certain value(minimum creep rate). The second stage(secondary/steady/stationary creep) is characterized by a nearly constant creep rate(minimum creep rate). In the third stage(tertiary creep), the creep strain rates increases rapidly and leads to rupture. The first stage is usually reversible with time after unloading, while the second/third ones are not. Since the primary creep occurs in a short duration and the tertiary one leads quickly to rupture, the secondary creep is under most serious consideration in many design in many engineering applications.

Stress relaxation is the phenomenon that the stress decreases when the strain is held constant in time. During the test the load is continuously decreased in such a way that the initial strain remains constant.

7.2 Creep rate law of material

The starting point is the assumption that the creep rate may be described as a product of two separate functions of stress, temperature and time

$$\epsilon^{cr} = f_\sigma(\sigma) f_T(T) f_t(t)$$

The widely used functions of stress $f_\sigma(\sigma)$ are:

$a\sigma^n$	Norton, 1929, Bailey, 1929
$b\left(\exp\frac{\sigma}{\sigma_0} - 1\right)$	Soderberg, 1936
$a\,sinh\,\frac{\sigma}{\sigma_0}$	Prandtl, 1928, Nadai, 1938, McVetty, 1943
$a_1\sigma^{n_1} + a_2\sigma^{n_2}$	Johnson et al., 1963
$a\,sinh\left(\frac{\sigma}{\sigma_0}\right)^n$	Garofalo, 1965

where $a, b, a_1, a_2, \sigma_0, n_1, n_2$ are material constants that could depend on time t. The dependence on the temperature $f_T(T)$ is usually expressed by the Arrhenius law

$$f_T(T) = \exp\left(\frac{-Q}{RT}\right)$$

where Q and R denote the activation energy and the Boltzmann's constant.

The time dependence part $f_t(t)$ is

t	secondary creep
bt^m	Bailey
$(1+bt^{1/3})\exp kt$	Andrade
$\sum_j a_j t^{m_j}$	Graham and Walles

For simplicity, we are going to only use the Norton-Bailey law.

7.3 Stress relaxation of helical spring

Due to conservation of the total shear strain, the sum of the creep strain ϵ^{cr}'s rate of change and that of elastic shear strain ϵ_{el} is zero:

$$\dot{\epsilon}_{cr} + \dot{\epsilon}_{el} = 0$$

The elastic shear strain is related to the shear stress by shear modulus G:

$$\epsilon_{el} = \sigma/G$$

and therefore

$$\dot{\epsilon}_{el} = \dot{\sigma}/G$$

According to Norton-Bailey law(also known as time hardening law),

$$\dot{\epsilon}_{cr}(t) = c\sigma^{n+1}t^{k-1} \tag{6}$$

where c is the shear strain rate, n, k are temperature dependent material constants.

Substituting the above two equations to the conservation law,

$$\dot{\sigma}(t)/G + c\sigma(t)^{n+1}t^{k-1} = 0 \tag{7}$$

The initial condition is

$$\sigma(0) = G\theta r$$

where θ is the initial twist angle per unit length, r the radius of wire.

$$\sigma = \left((G\theta r)^{-n} + \frac{c}{k}Gnt^k\right)^{-\frac{1}{n}}$$

The torque can be written as

$$M(t) = 2\pi\int_0^{d_w} r^2\sigma(r,t)\,\mathrm{d}r = {}_2F_1\left(\frac{4}{n}, \frac{1}{n}; \frac{4+n}{n}; \frac{c\theta^n G^{n+1} n t^k}{k}\frac{d_w^n}{2^n}\right)M(0)$$

where d_w is the wire diameter.

Since the spring load is linearly related the torque $P_z(t) \propto M(t)$, given the constant deflection s,

$$\frac{P_z(t)}{P_z(0)} = {}_2F_1\left(\frac{4}{n}, \frac{1}{n}; \frac{4+n}{n}; \frac{c\theta^n G^{n+1} n t^k}{k}\frac{d_w^n}{2^n}\right)$$

where

$$\theta = \frac{2s}{\pi N_a((d_i+d_o)/2)^2}$$

The closer this quantity is to 1, the better the spring quality is.

7.4 Creep of helical spring

The starting point is still the Norton-Bailey law (). There is naturally another way to write the shear strain rate:

$$\dot{\epsilon}^{cr} = \dot{\theta} r = \frac{8\dot{s}r}{\pi N_a (d_i + d_o)^2}$$

We obtain σ from substiting the above equation into Norton-Bailey law. Given the constant spring force P_z^0,

$$P_z^0 \frac{d_i + d_o}{4} = M(0) = 2\pi \int_0^{d_w} r^2 \sigma(r,t)\,\mathrm{d}r = \frac{\pi}{4} \frac{n+1}{4+3n} \left(\frac{8 d_w^{4+3n} \dot{s}}{t^{k-1}(d_i+d_o)^2 \pi N_a c (d_i + d_o)^2} \right)^{\frac{1}{n+1}}$$

It can be deduced that the spring length s follows

$$s(t) = s(0) + \left(\frac{(d_i + d_o) P_z^0}{\pi} \frac{4+3n}{n+1} \right)^{n+1} \frac{\pi (d_i + d_o)^2 N_a c}{8 k d_w^{4+3n}} t^k$$

where $s(0)$ is the initial spring length. The less the difference of the spring length $s(t) - s(0)$, the better the spring quality is.

[?] [?] [?] [?] [?]

7.5 Analysis of stress relaxation and creep models