



HAL
open science

Advanced assistance tool for optimal compression spring design

Manuel Paredes, Marc Sartor, Alain Daidié

► **To cite this version:**

Manuel Paredes, Marc Sartor, Alain Daidié. Advanced assistance tool for optimal compression spring design. *Engineering with Computers*, 2005, 21 (2), pp.140-150. 10.1007/s00366-005-0318-6 . hal-02120259

HAL Id: hal-02120259

<https://hal.science/hal-02120259>

Submitted on 5 May 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Article in Engineering With Computers · January 2005

DOI: 10.1007/s00366-005-0318-6

CITATIONS

9

READS

299

3 authors, including:



Manuel Paredes

Institut National des Sciences Appliquées de Toulouse

47 PUBLICATIONS 164 CITATIONS

[SEE PROFILE](#)



Alain Daidié

Institut National des Sciences Appliquées de Toulouse

55 PUBLICATIONS 433 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



Application of the Design of Experiments Method to a Transversely Loaded Cylindrical Assembly [View project](#)



Numerical model for bolted T-stubs with two bolt rows [View project](#)

Advanced assistance tool for optimal compression spring design

M. PAREDES, M. SARTOR and A. DAIDIE

Laboratoire de Génie Mécanique de Toulouse, INSA, 31077 Toulouse, France

manuel.paredes@insa-toulouse.fr

Fax: +33 5 61 55 97 00

Phone: +33 5 61 55 97 15

Abstract: The main industrial programs for spring design do not make the most of the powerful capabilities for optimization currently available. Usually, such software simply provides comprehensive validation tools. This paper introduces an advanced sizing tool for compression spring design. This tool can be used at any design stage and involves a specification sheet to provide data including interval values. Interval analysis and optimization processes are then run to provide the best design as output. High-level assistance functionalities are also presented and illustrated through a case study.

Key words: *optimization, interval analysis, solution space, compression springs, industrial application.*

NOTATION (based on DIN [1] standards)

A : least compressed state of the spring

B : most compressed state of the spring

D : mean diameter [mm]

D_i : inside diameter [mm]

D_o : outside diameter [mm]

D_L : diameter of the housing [mm]

d : wire diameter [mm]

E : elastic modulus of the wire [N/mm²]

F_1 : minimum operating load [N]

F_2 : maximum operating load [N]

f_e : natural frequency of surge waves [Hz]

G : torsion modulus of the wire [N/mm²]

k : stress correction factor

L_0 : free length [mm]

L_1 : maximum operating length [mm]

L_2 : minimum operating length [mm]

L_c : solid length [mm]
 L_K : buckling length [mm]
 L_n : minimal operating length to respect the minimum space between coils [mm]
 L_r : minimal operating length to respect the maximum allowable stress [mm]
 M : spring mass [g]
 m : pitch of the spring [mm]
 N : number of cycles
 n : number of active coils
 n_i : number of inactive coils for the ends related to L_0
 n_m : number of dead coils
 n_T : total number of coils
 R : spring rate [N/mm]
 R_m : ultimate tensile strength of the wire [N/mm²]
 S_a : total minimum space between coils [mm]
 S_h : spring travel [mm]
 V_0 : overall space occupied at L_0 [cm³]
 V_2 : overall space occupied at L_2 [cm³]
 W_h : internal energy stored during operating travel [N.mm]
 w : spring index
 α_2 : percentage of R_m at L_2
 α_c : percentage of R_m at L_c
 α_F : fatigue life factor
 γ : helix angle [degrees]
 ν : end fixation factor
 ρ : wire density [kg/m³]
 τ_d : fatigue limit of the wire [N/mm²]
 τ_m : mean shear stress [N/mm²]
 τ_a : alternate shear stress [N/mm²]

Left superscripts

S: from the specifications sheet

M: from the manufacturer constraints or from standards

Right superscripts

U: Upper limit

L: Lower limit

1. Introduction

Machine design involves sizing many different mechanical components such as gears, shafts, bolted joints, cams, etc. Each component has its own particular use and requires specific manufacturing knowledge. When an elastic function is required, designers often use springs. These can be found in basic mechanisms where their sizing is not critical. Springs can also be found in high-level mechanisms such as cars [2], hand prostheses [3] and satellites [4]. In such mechanisms, spring design has to be reliable and well controlled as poor spring behavior would lead to major damage.

Even if the authoritative work on spring design was written in 1963 [5], the ever broader range of applications for springs explains the continuing need to study spring behavior today. Recent works have focused on buckling [6], large displacements [7], non-circular wire [8], natural frequency [9] and non-axial forces [10] of helical springs. Other studies are devoted to other types of springs such as conical springs [11, 12], leaf springs [13, 14, 15] or tape springs [4]. Our study deals with helical compression springs with circular wire, which are still the most commonly used today.

As with other common mechanical components, the problem of spring design is often resolved using tables and charts containing certain pre-selected specifications and objectives. Even with computer assistance, designers are often obliged to oversimplify design procedures. They assign a value to certain parameters to reduce the number of problem variables to only 2 or 3 [16, 17, 18]. However, progress in Computer Assisted Design should lead to the capability of designing components without having to resort to such simplifications. The main industrial software available to designers working on spring definition work ("Compression Spring Software" from IST[®] [19], "FED1[®]" from Hexagon[®] [20] and "Advanced Spring Design" from SMI [21]) is mainly validation software. It uses standard calculations merely to check that the proposed design matches the requirements. "Advanced Spring Design" uses an optimization algorithm from UTS [22]. It allows the designer to define the desired parameters and automatically calculates all the other parameters that can be calculated until the design is completed. However, there is no tolerance on data and objective functions cannot be defined [23]. In our view, in current practice, the powerful potential for optimization is not fully taken advantage of. In theory, it should be possible to find an appropriate resolution method to provide an optimal solution to a global design problem whatever the specifications. Such a tool would be of major interest to designers as it could be used at any stage in the design process, even at the beginning where most design parameters are as yet unknown. To answer to this need, Paredes et al have proposed several assistance tools to perform optimal design for compression [24], extension [25] and torsion [26] springs. This paper introduces an approach that can be used to make the most of optimization capabilities for compression spring design in proposing advanced assistance functionalities. In the first part of this paper, we present the window interface, the spring parameters and proposed levels of

assistance. In the second part, the optimization problem is defined and the resolution method for the highest level of assistance is presented. The third part introduces a case study that makes use of high-level assistance capabilities.

2. ASSISTANCE TOOL

2.1. User-friendly window interface

Figure 1

The assistance tool interface is based on a main window where the data can be defined and the results are shown. This window is divided into three main areas called data 1, data 2 and results (see Fig. 1).

2.2. Data 1 area: fixed data for the problem

The data 1 area is used to define the data that are to be set for a problem. The designer specifies the material, the spring ends required and the standards to be used. The number of cycles N needs to be defined to calculate the fatigue life factor α_F . Where buckling is to be avoided, the buckling length without a guide or housing L_K has to be calculated. To do so, the user selects the end fixation factor ν in an additional window. The permitted percentage for the Ultimate Tensile Strength R_m at L_c , called $^S\alpha_c^U$ also has to be defined. The elastic limit of steel is about 50% of R_m . This is the default value of $^S\alpha_c^U$ but where we are certain that the spring will not be compressed to its solid length L_c , higher R_m percentages can be accepted. Finally, to be able to calculate the best spring, the objective function F (minimum mass, maximum fatigue life, etc.) has to be provided.

2.3. Data 2 area: problem parameters

The problem parameters are defined in the data 2 area. The proposed tool is to be used at any design stage. However, in the early design stages there are always a large number of parameters that have not yet been set. It is difficult to provide set values for a problem and it is therefore more convenient to define parameters through their possible lower and upper limits. For this reason, data 2 area comprises a specifications sheet defined using interval values. Thus, all parameters can be set by giving their limits (lower and/or upper: $^SX^L$, $^SX^U$). To set the value of a parameter, it has to be placed within both the lower and upper limits defining that parameter interval in the specifications sheet (see d and S_h in Fig. 1) and, to handle most industrial problems, a large range of parameters needs to be considered. They define not only the spring geometry but also its use.

Figure 2

The main parameters that define spring geometry are: $D_o, D, D_i, d, R, L_0, L_c, n_T$. Fig. 2 illustrates these parameters that characterize the intrinsic properties of the spring. Four independent parameters have to be known to calculate the four others.

A spring is conventionally described as working between two configurations, the least compressed state A and the most compressed state B . The main parameters defining the use of a spring are: F_1, F_2, L_1, L_2 and S_h (see Fig. 3). Once the design parameters are known, two independent operating parameters (to be taken from among F_1, F_2, L_1, L_2 and S_h) are needed to determine the two points A and B .

Figure 3

In addition, many other parameters can be considered by the designer as they can play a major role in some studies:

- ✓ Natural frequency of surge waves (f_e)
- ✓ Diameter of the housing (D_L)
- ✓ Internal energy stored during operating travel (W_h)
- ✓ Spring mass (M)
- ✓ Overall space occupied at free length (V_0)
- ✓ Overall space occupied at minimal operating length (V_2).

2.4. Assistance levels

Four levels of assistance are proposed at the top of the window interface. At level 1 of assistance, the designer sets requirements and asks the tool to propose a design by clicking on the Calculate button. Then, at level 2 assistance, calculation is almost instantaneous. The tool automatically proposes an optimal design for each new input (the Calculate button disappears). Thus, the designer can see at a glance what effect changes in input parameters have on the optimal design. A higher level of assistance can be obtained by activating interactive control of input (level 3). The idea here is to warn the designer where the statement of the problem lacks consistency (no acceptable solution can be found). The tool then checks the specifications introduced as input and automatically suggests corrections to those values that preclude determination of an acceptable solution domain.

The last level of assistance (level 4) shows in interactive fashion the optimal design and the acceptable bounds for each of the parameters considered in the data 2 area. This means the designer can readily see the influence of input values not only on the optimal design but also on the solution domain itself. The present paper focuses on presenting this last level of assistance.

2.5. Results area: proposed design

The results are presented on the right side of the specifications sheet. This area contains all the parameter values for the proposed solution including design and operating parameters. It also shows values for the parameters considered by standards such as the spring index or helix angle. To assist in design analysis, the objective function value is bordered and the parameter values that reach a limit of the problem (defined by the user, or imposed by standards or the spring manufacturer) are highlighted. The tool also allows a sensitivity study to be performed. An additional window shows the benefits on the objective function that can be obtained by relaxing limits. The designer can thus determine those limits that can be relaxed to derive an advantage.

Sometimes no spring can be found to fit requirements. In this case, a design closest to the specifications is proposed and those parameters that violate limits are shown with a stroke through them.

Finally, the results area also shows the allowable bounds for parameters when the highest level of assistance (level 4) is activated.

3. LEVEL 4 ASSISTANCE ALGORITHM

3.1. Main algorithm

Figure 4

The algorithm used for level 4 assistance is shown in Fig. 4. Using this activated assistance level, whenever the designer introduces new data the tool automatically shows the optimal spring characteristics and the allowable bounds for each parameter featured in the data 2 area. To achieve this, 39 optimization problems need to be automatically defined and solved. The only thing that varies between these 39 problems is the objective function. This explains why the optimization problem is prepared in one go (Step 1). The resolution process is performed using a mathematical programming process (step 4). For greater efficiency, two additional steps have been inserted (step 2 and step 3). The last step (step 5) extracts the results to show the best design and prepares data for the next user input.

3.2. Step 1: Building the optimization problem

Finding the spring design that offers the best possible value for an objective function involves an optimization problem with six continuous variables. Four design parameters (D_o , d , L_o , R) and two operating parameters (L_1 and L_2) are stored in the “vector of design variables” \mathbf{X} , so $\mathbf{X} = [D_o, d, L_o, R, L_1, L_2]^T$.

Table 1

A large range of constraints is considered allowing most real-life industrial problems to be resolved. Table 1 lists 47 selected constraints capable of expressing not just designer specifications but also standards and the spring manufacturer's technical capability limits. All the necessary formulae to encode these equations are given in the Appendix. An optimization system with 6 variables and a maximum of 47 constraints is thus obtained and has to be resolved.

Each set of data not defined in the data 2 area is automatically set to a default value to ensure that it will not influence the optimization process, with 0 for a lower limit and 10^7 for an upper limit. Even where a standardized wire range exists, a wire can be manufactured to any diameter. When using standardized wire, each wire diameter (d) considered implies setting $^S d^L = ^S d^U = d$ in the specifications sheet.

The objective function is defined on each loop between step 2 and step 3 and is expressed using the conventional form :

$$\text{Minimize or Maximize } F(\mathbf{X})$$

where $F(\mathbf{X})$ is either the fatigue life factor, or any one of the parameters that may be set in the specifications sheet: $D_o, D, D_i, D_L, d, R, n_T, L_0, L_1, L_2, L_c, S_h, F_1, F_2, M, f_e, V_0, V_2, W_h$.

Step 2 and step 3 are to increase efficiency of step 4. This explains why step 4 is presented before step 2 and step 3.

3.3. Step 4: Solving the optimisation problem

The goal here is to find a fast, reliable and complete method to solve our problem automatically whatever the specifications. As continuous variables have to be managed, we have chosen to use mathematical programming [27] and, more specifically, to implement a direct method as each step in the resolution process is based on a displacement inside the solution area. This kind of property is useful if the resolution process is interrupted before completion (where it proves difficult to obtain full convergence), with the tool being more likely to provide the designer with an adequate though non-optimal solution.

Direct methods require a starting point inside or close to the solution area. Indeed, the closer the starting point is to the final solution, the more likely the algorithm will be to converge towards the optimal solution. This is especially true here considering the high number of constraints. For this reason, we have developed a dedicated strategy to find a suitable starting point, implemented through building up a virtual catalog of springs relating to the specifications (step 2) and selecting from it the best design (step 3). One key benefit in searching for the allowable bounds of the parameters is shown in step 5.

3.4. Step 2: Building a virtual catalog

Interval arithmetic [28] is used to analyze the main parameters on the specifications sheet in order to build a spring virtual database that respects the greatest possible number of constraints. Interval arithmetic is used to full effect for the spring's main design and operating parameters (see Fig. 2 and Fig. 3). In this instance, formulae are quite simple and we take advantage of the tolerance propagation approach [29] to find springs that best respect the requirements.

3.5. Step 3: Selecting the best design from within the catalog

Selecting the best design from within the virtual catalog is not in itself an easy matter. Indeed, choosing a spring from the catalog will allow you to define the four design parameters (D_o , d , L_0 , R) of the vector of design variables, but the two operating parameters (L_1 and L_2) still remain to be determined in order to fully define X . A dedicated algorithm [30] has thus been developed to select the best design from within the virtual database. It first determines the optimal operating parameters for each spring depending on the specifications and on the current objective function. To this purpose, a method (introduced by Johnson [31]) dedicated to solving problems with a small number of variables is used. Each design obtained is then compared with the requirements and the best one is selected to initialize the mathematical programming process.

3.6. Step 5: Extracting and saving results

There are two ways to extract results, depending on whether the goal is to find the optimal design or find the allowable bound of a parameter.

When the objective function is that defined by the designer, a large number of results are extracted. The designer is offered all the design and operating parameters, and the other standard-related parameters.

When the objective is related to finding one parameter bound, the user will only be offered the final value of the objective function (that is the allowable bound related to the tested parameter).

This search for allowable bounds is not only of great interest for the designer but also extremely important for the mathematical programming process, as it involves finding the Bounds of the Solution Space [32]. Considering this, the design parameters (D_o , d , L_0 and R values) of the optimum result related to each bound are added to the next virtual catalog (this will be built at step 2 after the next designer input). This is a key issue as it guarantees our finding a starting point that respects the requirements whatever the specifications (as long as problem consistency is maintained). Indeed, at least one of these springs satisfying the current optimization problem will also satisfy the next

problem. This fact is illustrated in Fig. 5 for a simplified problem involving only two variables (being represented in a 2D space) d and D_0 .

Figure 5

Fig. 5a shows the solution domain in the $[d, D_0]$ space. It is bounded by the manufacturing constraints and previous inputs from the designer (only ${}^S D_0^U$ in this case):

- ✓ on d (${}^M d^L, {}^M d^U$) as represented by two vertical lines: $d = {}^M d^L, d = {}^M d^U$
- ✓ on the spring index w represented by two lines: $D_0 = d ({}^M w^U + 1)$ and $D_0 = d ({}^M w^L + 1)$
- ✓ on D_0 represented by a line: $D_0 = {}^S D_0^U$

Note that the constraint on D , which is represented by a line $D_0 = {}^M D^U + d$, is not activated.

When the designer sets another value in the specifications sheet, the solution domain is reduced. Fig. 5b illustrates the consequence of a ${}^S D_i^L$ input. The solution space includes the new constraint:

- ✓ on D_i represented by a line: $D_0 = {}^S D_i^L - 2 d$.

The solutions found when searching for the allowable parameter bounds are located on the Bounds of the Solution Space. At least one of these solutions can be retained for the next optimization problem (constructed when there is new designer input). This will be the solution relating to the search for the opposite bound compared with the input. In the example shown, the next user input relates to ${}^S D_i^L$. Thus, the solution found when searching for the greatest value of D_i (represented by a circle) provides a common solution for both optimization problems.

To conclude, storing the results of the allowable bound search ensures that we have an acceptable starting point for each optimization problem. But this initial design can be remote from the optimal one and we know that direct methods are more likely to converge when the starting point is close to the optimal solution. This explains why the algorithm used to build a virtual catalog of springs (step 2) is still of interest when it comes to finding a better initial design for each optimization problem.

4. Case Study: spring for a hydraulic pump

Figure 6

This case study deals with a spring for a hydraulic pump and involves a dual presentation. The first of these quickly introduces a design leading to an acceptable solution. The second simulates a similar study where a value on the specifications sheet is changed, leading to no acceptable solution. We then go on to demonstrate the main advantages of the advanced assistance level.

Figure 7

The initial result is shown in Fig. 7. The proposed design fits the requirements and its associated fatigue life factor is equal to 1.73. It can be seen that the limits defined on the specifications sheet on F_2 , L_2 , f_e , D_L , S_h and L_n are reached and that the limits related to R are inactive. Obtaining such inactive limits often occurs when using this kind of tool as the designer has no need to pre-define the main parameters for the problem beforehand.

4.1. First study

Requirements were defined as follows. The spring was of shot peened steel with closed and ground ends. Spring travel S_h was constant (14 mm) and the system study defined the maximum spring load F_2 (200 N) together with the limits of the spring rate R ($4 \leq R \leq 10$ N/mm) and the minimum natural frequency of the spring f_e (200 Hz). To avoid any problem of wear due to friction, the spring was not to buckle, the supports were parallel and guided to give an end fixation factor [1] of 0.5. Finally the compressed spring had to be included within the piston, with its housing diameter D_L lower than DM (22 mm) and L_2 lower than LM (64 mm). As we were able to control the conditions for assembly and carriage strictly, the theoretical breakage of the spring at its solid length could be admitted. In this design stage there was no other geometrical constraint and the final design of the contiguous parts can be adjusted to match optimal spring design. The goal here was to obtain the greatest fatigue life factor for 10^7 cycles.

Figure 8

4.2. Second study

The same study was simulated considering the minimum natural frequency of the spring f_e to be now twice the initial value ($f_e^L = 400$ Hz). At assistance level 1, problem resolution using continuous variables provided the result shown in Fig. 8. Here, no spring appeared to fit the requirements. For this reason, the tool automatically proposes a design closest to the specifications. We can see that the proposed spring has a natural frequency of 395 Hz, that is less than the 400 Hz required by the specifications. At this step the designer can either accept the spring or modify the requirements in order to find a better one by trial and error.

Figure 9

The same study was performed activating level 4 assistance. Data were set in the order they appear in the specifications sheet: $F_2^L = 200$ N; $F_2^U = 200$ N; $L_2^U = 64$ mm; $f_e^L = 400$ Hz; $R^L = 4$ N/mm; $R^U = 10$ N/mm; $D_L^U = 22$ mm. At this step, the tool shows that the maximum allowable value for S_h is 13.625 mm. Thus, when the next data item, $S_h^L = 14$ mm, is entered in the specifications sheet, the

assistance tool automatically corrects the value to maintain consistency and proposes 13.625 mm instead of 14 mm (see Fig. 9). The designer knows that there is no solution for the initially sought specifications and the last data item, ${}^S S_h^U = 14$ mm, is entered without leading to any change.

Figure 10

For the system under consideration, spring travel must be 14 mm and something therefore has to be changed in the specifications. Here, sensitivity analysis can help in determining what could be done to improve allowable spring travel with the objective being to attain ‘greatest spring travel’. This analysis shows (see Fig. 10) that maximum gain can be obtained by decreasing ${}^S F_2^L$, decreasing ${}^S f_e^L$ or increasing ${}^S R^U$. It also shows that no gain would be obtained in modifying either the geometrical limits or the manufacturing limits.

The sensitivity study tests each limit set on the specifications sheet, as also the manufacturing limits and material limits. One optimization problem is thus defined and solved for each tested limit. Each problem is in fact similar to the initial optimization problem (as defined in the main window) but the constraint relating to the tested limit is relaxed by 10%. Thus, by comparison with the initial problem, we enlarge the solution space and, as a result, the optimal design shown in the Results area represents an acceptable point. Moreover, as the objective function remains unchanged (being the one defined by the designer in the data 1 area), this design solution comes very close to the new optimal solution. In this way, the design is defined as the unique starting off point for each optimization problem. The sensitivity study window shows the relative gain on the objective function for each tested limit. Having considered the overall system, a value of ${}^S F_2^L = 180$ N proves to be acceptable if certain modifications are introduced on other parts.

The assistance tool can also be used to define optimal springs made from standardized wires.

Considering the current problem, the tool shows that $2.22 \leq d \leq 2.58$ mm. The only standardized wire diameter available is $d = 2.5$ mm. A last calculation is thus performed with ${}^S d^L = 2.5$ mm and ${}^S d^U = 2.5$ mm. This leads to the solution shown in Fig. 1, which is retained for the final design.

5. CONCLUSION

In the present paper, we introduced a comprehensive dimensional synthesis tool for compression spring design. Its main advantage is in making full use of optimization processes to propose a design working directly from the specifications. Furthermore, the specifications sheet can incorporate data tolerances to be used in the early design stages. The advantages of using optimization algorithms are also used to the full in proposing advanced assistance functionalities. The tool can automatically correct data to maintain problem consistency and also allows a sensitivity study to be performed to see what effects are induced by modifying the input data. Finally, the assistance tool can be used interactively to outline the allowable parameter bounds, providing the designer with an overview.

This approach can be extended to the other types of springs such as extension, torsion or conical springs. More generally, as this approach can be time consuming, it could be of major interest to help designers when analytical formulae can be exploited.

Acknowledgements

We would like to thank Schneider Electric and the Institute of Spring Technology for their technical support.

Appendix: Details of all formulae used to define the problem constraints.

Constraints related to variables

- ◇ Specifications of the spring and wire manufacturer on L_0 and d (${}^M L_0^U$, ${}^M d^U$, ${}^M d^L$) have to be taken into account. Here are the values used in our study for steel: ${}^M L_0^U = 630$ mm, ${}^M d^U = 12$ mm, ${}^M d^L = 0.3$ mm.

Constraints related to design parameters

- ◇ Mean diameter: $D = D_o - d$

Specification of the spring manufacturer on D (${}^M D^U$) is to be taken into account. The value used in our study is ${}^M D^U = 200$ mm.

- ◇ Inside diameter: $D_i = D_o - 2 d$

- ◇ Number of active coils: $n = G d^4 / (8 R D^3)$

DIN standards explain that all the formulae used are available if the number of coils is greater than ${}^M n^L = 2$.

- ◇ Total number of coils: $n_T = n + n_m + 2$ (for closed ends)

- ◇ Solid length: $L_c = d (n + n_i + n_m)$

- ◇ Helix angle: $\gamma = 180 \operatorname{atan}(m / \pi / D) / \pi$ with $m = (L_0 - n_i d - n_m d) / n$

The helix angle has to be less than ${}^M \gamma^U = 7,5^\circ$ for the calculations to remain valid.

- ◇ Spring index: $w = D / d$. The standard requires the spring index to lie between ${}^M w^L = 4$ and ${}^M w^U = 20$

- ◇ Diameter of the housing: $D_L = D_o + 0.1 (m^2 - 0.8 m d - 0.2 d^2) / D$

Constraints related to operating parameters

◇ Minimum spring load: $F_1 = R (L_0 - L_1)$

◇ Maximum spring load: $F_2 = R (L_0 - L_2)$

◇ Spring travel: $S_h = L_1 - L_2$

◇ The minimum operating length must respect the minimum space between coils:

$$S_a = n (0.0015 D^2 / d + 0.1d).$$

S_a is multiplied by 1.5 when dynamic use is required ($N > 10^7$).

Thus, according to DIN [1] standards: $L_n = L_c + S_a$

◇ If required in the specifications sheet we also need to take into account the constraint related to buckling according to the end fixation factor ν [1]:

If $\nu L_0 / D < \pi \sqrt{\frac{2\mu + 1}{\mu + 2}}$ then $L_K = 0$ (no risk of buckling)

$$\text{else } L_K = L_0 \left\{ 1 - \frac{\mu + 1}{2\mu + 1} \left[1 - \sqrt{1 - \frac{2\mu + 1}{\mu + 2} \left(\frac{\pi D}{\nu L_0} \right)^2} \right] \right\}$$

with $\mu = \frac{E}{2G} - 1$

◇ Natural frequency of surge waves: $f_e = \sqrt{\frac{R}{\pi^2 \rho n D d^2}}$

◇ Internal energy during operating travel: $W_h = 0.5 R S_h^2$

◇ Total mass: $M = \pi^2 \rho D d^2 (n + n_m + 1.5) / 4$ (for closed and ground ends)

◇ Free overall space occupied: $V_0 = \pi L_0 D_o^2 / 4$

◇ Operating overall space occupied: $V_2 = \pi L_2 D_o^2 / 4$

◇ Minimum static safety under operating conditions. The spring is automatically designed to ensure operating length L_2 achieves a maximum α_2^U % of R_m .

$$L_r = L_0 - (\pi \alpha_2^U R_m d^3) / (800 D R k)$$

$$\alpha_2 = L_2 / L_r$$

with $k = (w + 0.5) / (w - 0.75)$; stress correction factor (many other formulations of this factor exist, all with equivalent results [5,19])

In the calculation $\alpha_2^U = 50$ for steel.

$$\diamond \text{ Allowable percentage of } R_m \text{ at } L_c: \alpha_c = 800 D R k (L_0 - L_c) / (\pi R_m d^3)$$

Where R_m is a function of d depending on the material

\diamond Fatigue life factor [24]

$$\alpha_F = \text{Min} \left\{ \frac{\tau_{d(N)} (2 \alpha_2^U R_m - \tau_{d(N)})}{\tau_a (2 \alpha_2^U R_m - \tau_{d(N)}) + \tau_m \tau_{d(N)}}, \frac{\alpha_2^U R_m}{\tau_a + \tau_m} \right\} \text{ with:}$$

$$\tau_m = 4 D R (2 L_0 - L_1 - L_2) / (\pi d^3)$$

$$\tau_a = 4 D R (L_1 - L_2) / (\pi d^3)$$

$$\tau_{d(N)} = \begin{cases} \bullet \tau_d & \text{if } N \geq 10^7 \text{ (fatigue limit)} \\ \bullet \left[(\tau_d - \alpha_2^U R_m) \ln N / \ln 10 + 7 \alpha_2^U R_m - 4 \tau_d \right] / 3 & \text{if } 10^4 < N < 10^7 \\ \bullet \alpha_2^U R_m & \text{if } N \leq 10^4 \text{ (static limit)} \end{cases}$$

Where R_m and τ_d are functions of d depending on the material.

References

- [1] DIN, Burggrafenstra e 6, postfach 11 07, 10787 Berlin, Germany.
- [2] M. Langa, A. Ouakka (2001) Optimisation du ressort de suspension dans l'environnement v hicule: apports de l'outil de simulation num rique. M canique et Industrie 2 181-188.
- [3] J. Yang, E. Pe a Pitarch, K. Abdel-Malek, A. Patrick, L. Lindkvist (2004) A multi-fingered hand-prosthesis. Mechanism and Machine Theory 555-581.
- [4] K.A. Seffen, Z. You, S. Pellegrino (2000) Folding and deployment of curved tape springs. International Journal of Mechanical Sciences 42:2055-2073.
- [5] A.M. Wahl (1963) Mechanical Springs. McGraw-Hill Book Company, New York.
- [6] J.E. Mottershead (1987) The large displacements and dynamic stability of springs using helical finite element. International Journal of Mechanical Sciences, 24:547-558.
- [7] L.E. Becker, W.L. Cleghorn (1992) On the buckling of helical compression springs. International Journal of Mechanical Sciences, 34:275-282.
- [8] L. Le, G. Ji, Y. Lin (1994) Stress Analysis and Optimal Cross-Sections of Noncircular Spring Wire. Springs, Official publication from the Spring Manufacturer Institute, October:30-45.
- [9] L.E. Becker, G.G. Chassie, W.L. Cleghorn (2002) On the natural frequency of helical compression springs. International Journal of Mechanical Sciences, 44:825-841.

- [10] M. Southward (2003) Non-axial forces in compression springs. *Wire*, 53:26-27.
- [11] E. Wolansky (1996) Conical Spring Buckling Deflexion. *Springs*, Winter:62.
- [12] E. Wolansky (2001) Lateral Loading of Conical Springs. *Springs*, April:95.
- [13] E. Sancaktar, M. Gratton (1999) Design, analysis and optimization of a composite leaf spring for light vehicle applications. *Composite Structures* 44: 195-204.
- [14] I. Rajendran, S. Vijayarangan (2001) Optimal design of a composite leaf spring using genetic algorithms. *Computers and structures* 79:1121-1129.
- [15] M.M. Shokrieh, D. Rezaei (2003) Analysis and optimisation of a composite leaf spring. *Composite Structures*, 60:317-325.
- [16] S. Metwalli, A. Radwan, A.A. Elmeligy (1994) CAD and Optimization of helical torsion springs. *Computers in Engineering*, 767-773.
- [17] R. Deford (2003) Iterative logic for nested compression spring design. *Springs*, June:27-28.
- [18] PRODMOR System Inc., 3-2133 Royal Windsor Drive, Mississauga, ON, L5J 1K5, Canada, www.promodor.com.
- [19] IST[®], Institute of Spring Technology, Henry Street Sheffield S3 TEQ, United Kingdom, www.ist.org.uk.
- [20] EXAGON[®], Industriesoftware GmbH, Stiegelstrasse 8, 73230 Kirchheim/Teck, Germany, www.hexagon.de.
- [21] SMI, Spring Manufacturers Institute, 2001 Midwest Road, Suite106 Oak Brook, Illinois 60521, USA.
- [22] UTS, Universal Technical Systems (E), Ltd, 1 Brittens Court, Clifton Reynes, Olney, Buckinghamshire MK46 5LG, United Kingdom, www.uts.com.
- [23] R. Deford (2004) Exploring ASD6. *Springs*, January:38-40.
- [24] M. Paredes, M. Sartor, C. Masclet (2002) Obtaining an optimal compression spring design directly from a user specification, *Journal of Engineering Manufacture Part B, Proceedings of the institution of mechanical engineers*, 216:419-428.
- [25] M. Paredes, M. Sartor, C. Masclet (2001) An optimization process for extension spring design. *Computer Methods in Applied Mechanics and Engineering*, 191:783-797.
- [26] M. Paredes, M. Sartor, J.C. Wahl (2002), Optimization techniques for industrial spring design. *Optimization in Industry*, Springer, pp. 315-326.
- [27] G. N. Vanderplatts (1984) *Numerical Optimization Techniques for Engineering Design*. McGraw-Hill Book Company, New York.
- [28] R.E. Moore (1979) *Methods and Applications of Interval Analysis*. SIAM, Philadelphia.
- [29] E. Hyvonen (1992) Constraint reasoning based on interval arithmetic: the tolerance propagation approach. *Artificial Intelligence*, 58:71-112.
- [30] M. Paredes, M. Sartor, J.C. Fauroux (2000) Stock spring selection tool. *Springs*, Winter:53-67.
- [31] R. C. Johnson (1980) *Optimum design of mechanical elements*. Wiley-interscience Publication.
- [32] Z. Yao, A.L. Johnson (1997) On estimating the feasible solution space of design. *Computer-Aided Design*, 29:649-655.

Advanced Optimal Compression Spring Design

Erase Data DOT Txt Edit manufacturing limits Assistance : Level 4 : show allowable bounds for parameters PRINT QUIT

Data 1 Material : Steel DH shot panned Number of cycles : 10 E7 Fatigue data for d inside [1mm, 10mm].
 Ends type : Closed and ground Calculus of Ln : DIN Buckling, End fixation factor of 0.5
 Dead coils : At solid length : break Check for helix angle
 Use stress correction
 Propose the spring with The greatest Fatigue life factor

	Specification limits		Allowable bounds		Proposed design characteristics			
	Lower	Upper	Lower	Upper				
Outside diameter (Do) in mm			20.05	21.86	20.05	Do (mm)	Ln (mm)	28.20
Mean diameter (D) in mm			17.55	19.36	17.55	D (mm)	Spring index	7.02
Inside diameter (Di) in mm			15.05	16.86	15.05	Di (mm)	Helix angle (°)	6.70
Wire diameter (d) in mm	2.5	2.5	2.50	2.50	<u>2.50</u>	d (mm)	Fatigue life factor	<u>1.03</u>
Free length (L0) in mm			43.46	58.43	52.72	L0 (mm)	Stress at solid (%Rm)	53
Total number of coils (nt)			7.92	9.36	9.36	nt	Buckling length (mm)	0.00
Operating load 1 (F1) in N			40.000	64.013	43.642	F1 (N)	<input type="button" value="Sensitivity study"/>	
Operating load 2 (F2) in N	180	200	179.979	196.624	<u>180.000</u>	F2 (N)		
Operating length 1 (L1) in mm			37.66	54.43	48.36	L1 (mm)		
Operating length2 (L2) in mm		64	24.03	40.43	34.72	L2 (mm)		
Solid length (Lc) in mm			19.81	23.40	23.40	Lc (mm)		
Volume at L0 (V0) in cm3			14.5895	19.9677	16.6489	V0 (cm3)		
Mass (M) in g			18.5682	19.8873	19.8873	M (g)	Inactive parameters: normal Active parameters: <u>underlined</u> Violated parameters: Striked Thru	
Natural frequency (fe) in Hz	400		399.97	414.11	<u>399.97</u>	fe (Hz)		
Spring rate (R) in N/mm	4	10	9.06820	10.0010	<u>10.00000</u>	R (N/mm)		
Housing diameter (DL) in mm		22	20.16	22.00	20.21	DL (mm)		
Spring travel (Sh) in mm	13.6358	14	13.64	14.00	<u>13.64</u>	Sh (mm)		
Volume at L2 (V2) in cm3			8.7390	12.7643	10.9647	V2 (cm3)		
Energy (Wh) in N.mm			843.052	980.000	929.680	Wh (N.mm)		

Fig. 1 User friendly window interface

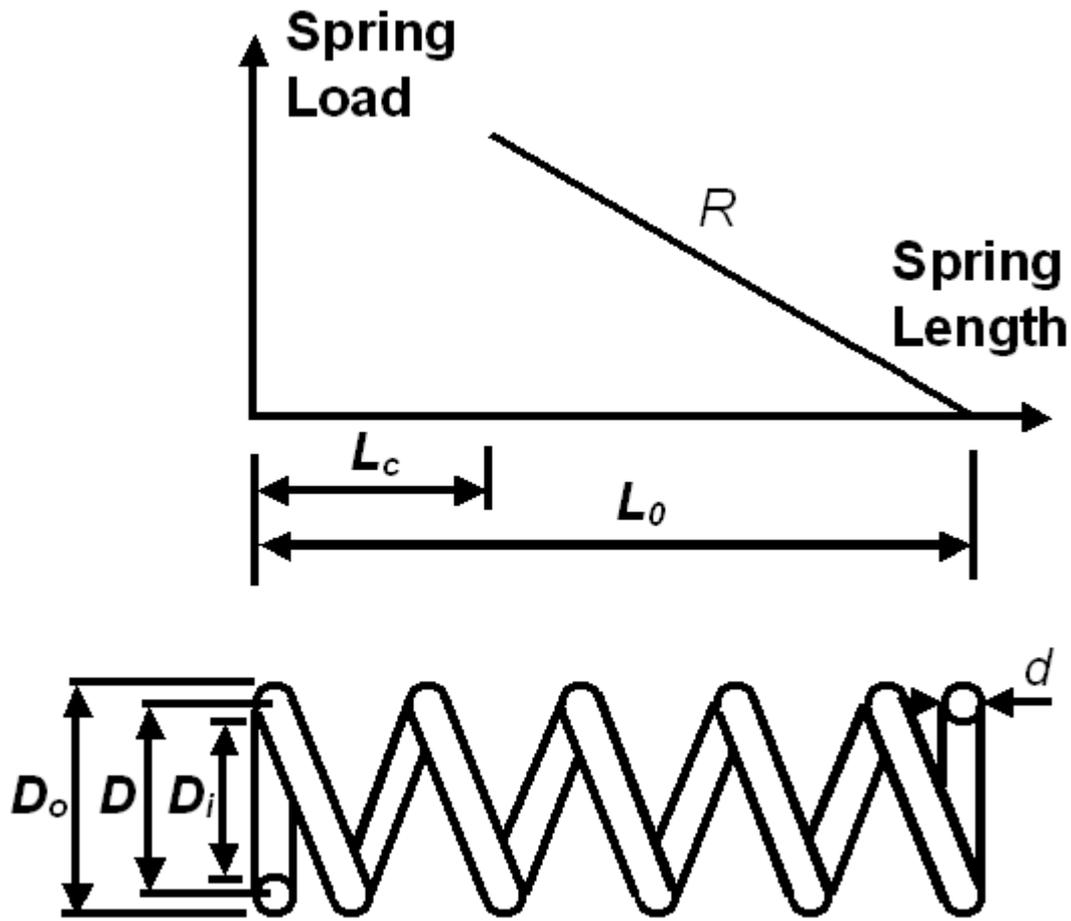


Fig. 2 Main design parameters

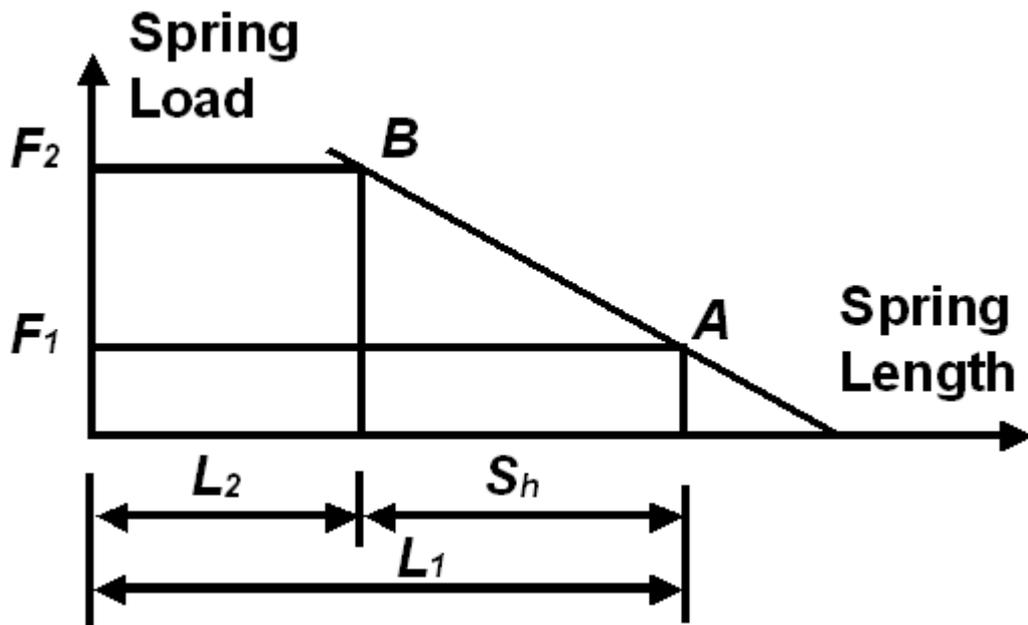


Fig. 3 Main operating parameters

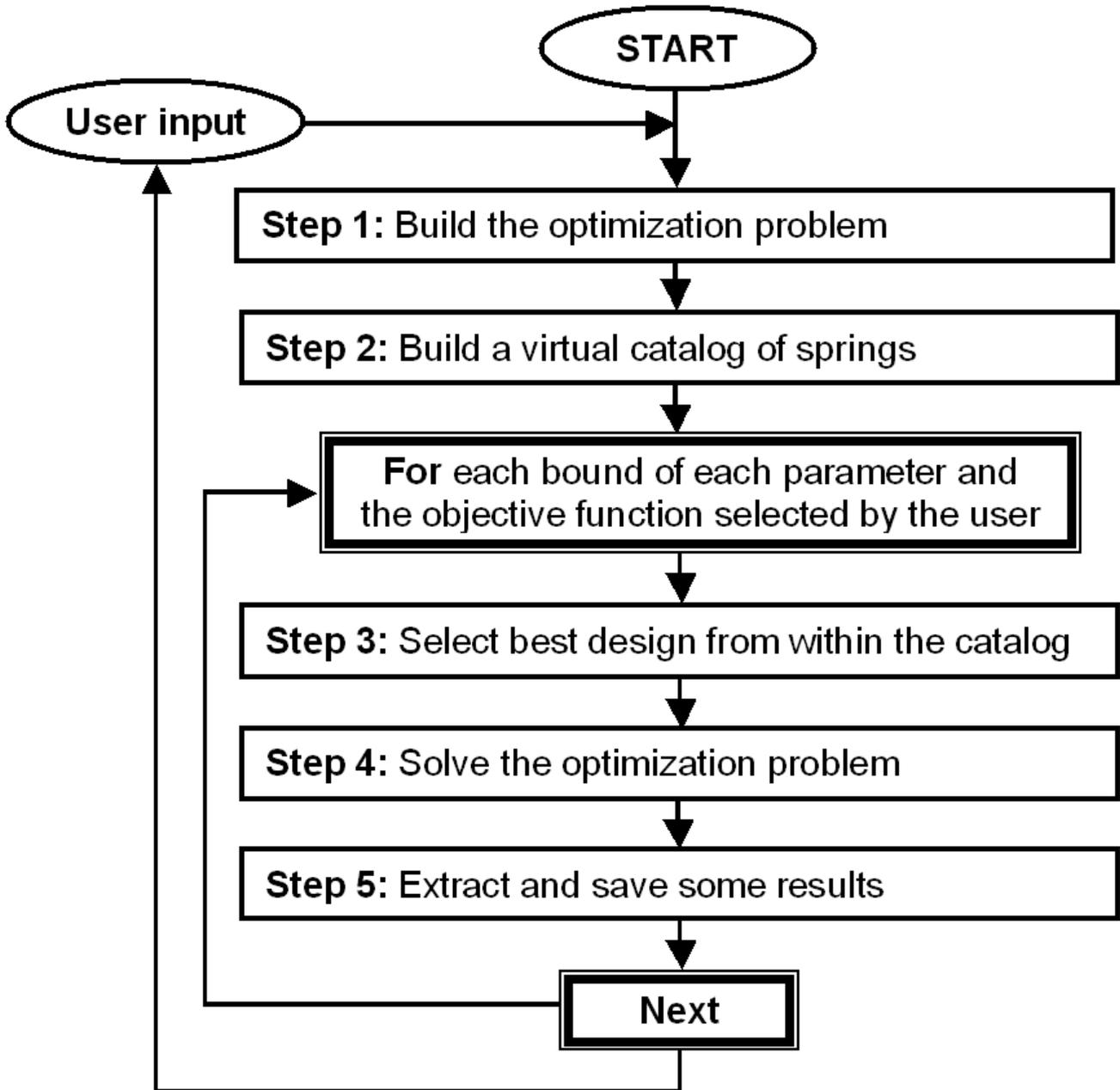


Fig 4. Level 4 assistance algorithm

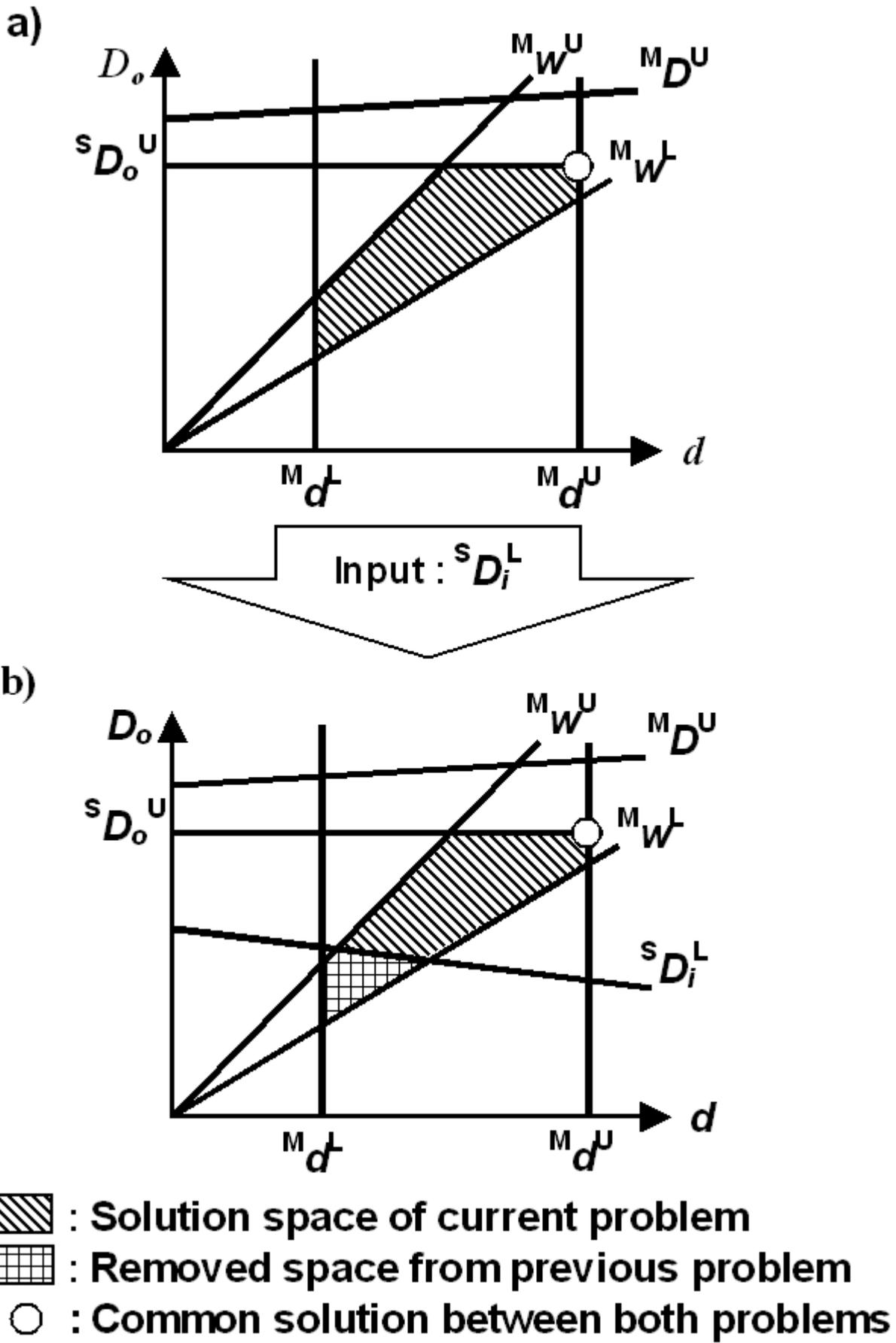


Fig 5. Evolution of the solution space depending on inputs

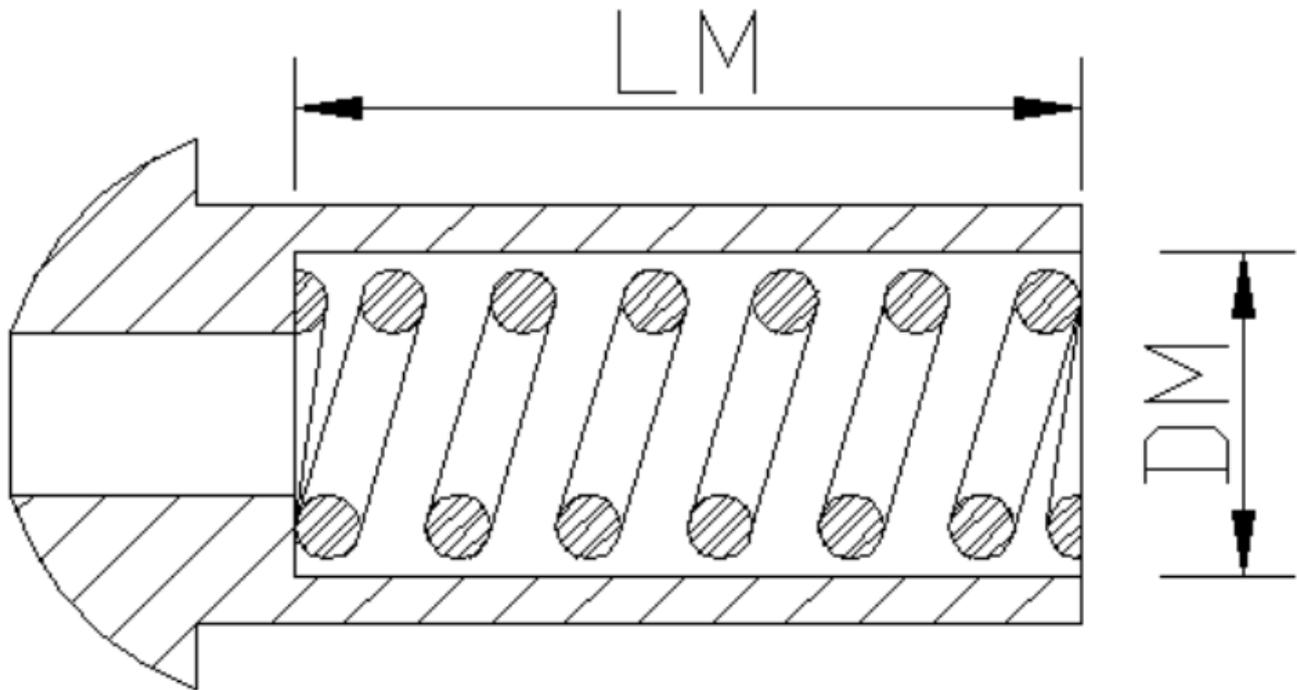


Fig 6. Piston design

Advanced Optimal Compression Spring Design

Erase Data | DOT Txt | Edit manufacturing limits | Assistance: Level 1 : calculus on order | PRINT | QUIT

Data 1

Material: Steel DH shot panned | Number of cycles: 10 E7 | Fatigue data for d inside [1mm, 10mm].

Ends type: Closed and ground | Calculus of Ln: DIN | Buckling, End fixation factor of 0.5

Dead coils: | At solid length: break | Check for helix angle

Use stress correction

Propose the spring with: The greatest | Fatigue life factor | Calculate

Data 2

	Specification limits	
	Lower	Upper
Outside diameter (Do) in mm		
Mean diameter (D) in mm		
Inside diameter (Di) in mm		
Wire diameter (d) in mm		
Free length (L0) in mm		
Total number of coils (nt)		
Operating load 1 (F1) in N		
Operating load 2 (F2) in N	200	200
Operating length 1 (L1) in mm		
Operating length2 (L2) in mm		64
Solid length (Lc) in mm		
Volume at L0 (V0) in cm3		
Mass (M) in g		
Natural frequency (fe) in Hz	200	
Spring rate (R) in N/mm	4	10
Housing diameter (DL) in mm		22
Spring travel (Sh) in mm	14	14
Volume at L2 (V2) in cm3		
Energy (Wh) in N.mm		

Results

Allowable bounds		Proposed design characteristics	
Lower	Upper		
21.93		Do (mm)	<u>64.00</u>
18.89		D (mm)	6.22
15.86		Di (mm)	5.11
3.04		d (mm)	<u>1.73</u>
88.05		L0 (mm)	34
17.43		nt	0.00
83.573		F1 (N)	
<u>200.013</u>		F2 (N)	
78.00		L1 (mm)	
<u>64.00</u>		L2 (mm)	
52.90		Lc (mm)	
33.2536		V0 (cm3)	
58.7617		M (g)	
<u>200.00</u>		fe (Hz)	
8.317166		R (N/mm)	
<u>22.00</u>		DL (mm)	
<u>14.00</u>		Sh (mm)	
24.1713		V2 (cm3)	
815.082		Wh (N mm)	

Sensitivity study

Inactive parameters: normal
Active parameters: underlined
Violated parameters: ~~Stroked Thru~~

Fig. 7 Result of the first study with level 1 assistance

Advanced Optimal Compression Spring Design

Erase Data | DOT Txt | Edit manufacturing limits | Assistance: Level 1 : calculus on order | PRINT | QUIT

Data 1 Material: Steel DH shot panned | Number of cycles: 10 E7 | Fatigue data for d inside [1mm, 10mm].
 Ends type: Closed and ground | Calculus of Ln: DIN | Buckling, End fixation factor of 0.5
 Dead coils: | At solid length: break | Check for helix angle
 Use stress correction
 Propose the spring with: The greatest | Fatigue life factor | Calculate

	Specification limits		Allowable bounds		Proposed design characteristics	
	Lower	Upper	Lower	Upper		
Outside diameter (Do) in mm			21.38	Do (mm)	Ln (mm)	26.99
Mean diameter (D) in mm			18.81	D (mm)	Spring index	7.30
Inside diameter (Di) in mm			16.23	Di (mm)	Helix angle (°)	<u>7.46</u>
Wire diameter (d) in mm			2.58	d (mm)	Fatigue life factor	<u>0.99</u>
Free length (L0) in mm			56.86	L0 (mm)	Stress at solid (%Rm)	61
Total number of coils (nt)			8.68	nt	Buckling length (mm)	0.00
Operating load 1 (F1) in N			57.698	F1 (N)	Sensitivity study	
Operating load 2 (F2) in N	200	200	198.892	F2 (N)		
Operating length 1 (L1) in mm			51.14	L1 (mm)		
Operating length2 (L2) in mm		64	37.14	L2 (mm)		
Solid length (Lc) in mm			22.35	Lc (mm)		
Volume at L0 (V0) in cm3			20.4204	V0 (cm3)		
Mass (M) in g			20.9720	M (g)		
Natural frequency (fe) in Hz	400		395.45	fe (Hz)		
Spring rate (R) in N/mm	4	10	<u>10.08858</u>	R (N/mm)	Inactive parameters: normal	
Housing diameter (DL) in mm		22	21.61	DL (mm)	Active parameters: <u>underlined</u>	
Spring travel (Sh) in mm	14	14	<u>14.00</u>	Sh (mm)	Violated parameters: Striked Thru	
Volume at L2 (V2) in cm3			13.3399	V2 (cm3)		
Energy (Wh) in N.mm			988.042	Wh (N.mm)		

Fig. 8 Result of the second study with level 1 assistance

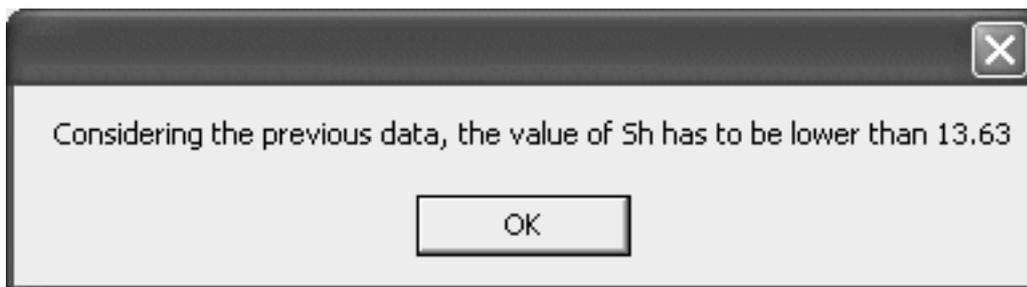


Fig. 9 Message providing corrected data to maintain consistency

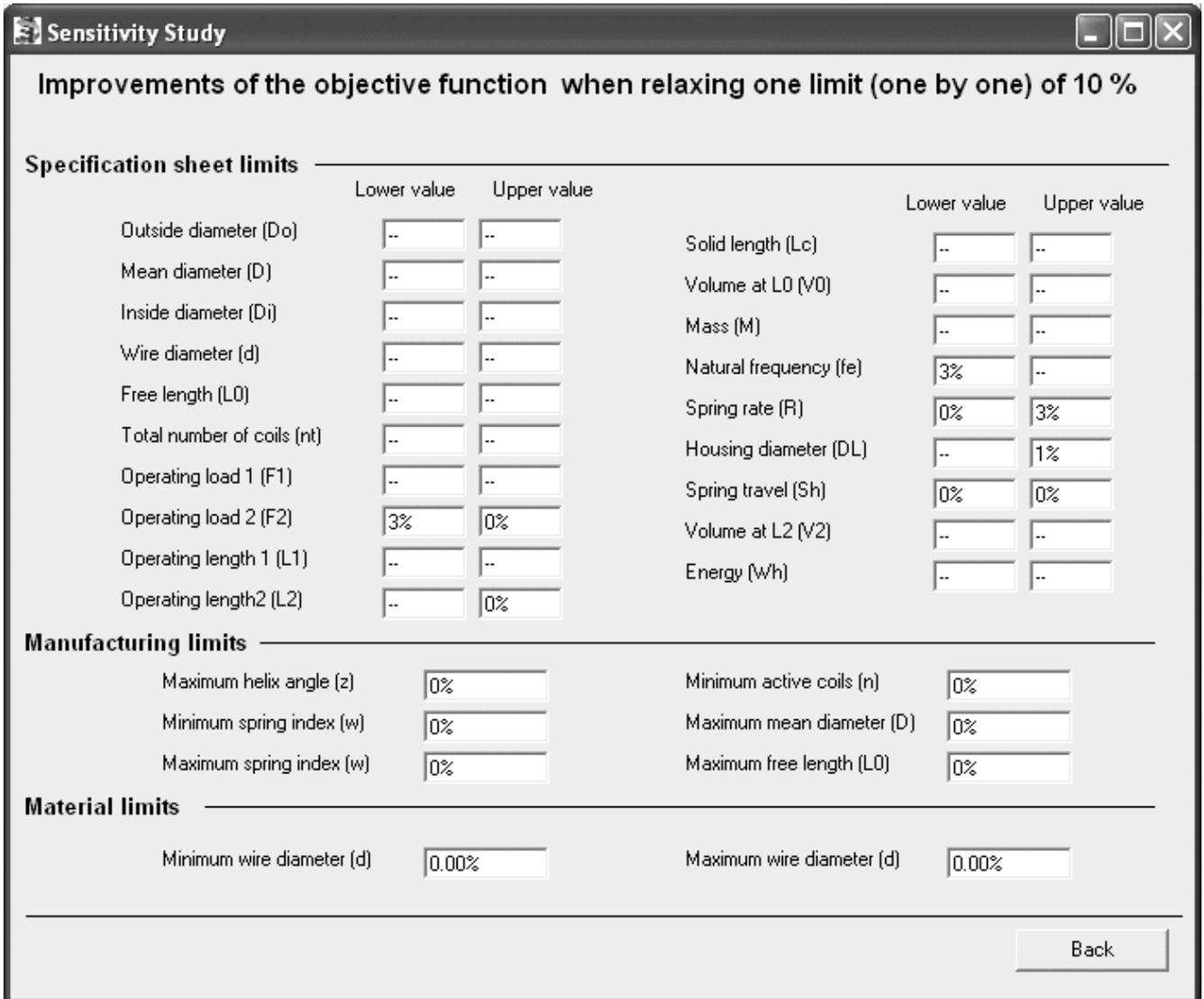


Fig. 10 Result of the sensitivity study.

Table 1 Constraints

Constraint on:	Upper limit:	Lower limit:
D_o	$g1(X): D_o - {}^S D_o^U \leq 0$	$g2(X): {}^S D_o^L, - D_o \leq 0$
d	$g3(X): d - \text{Min}({}^S d^U, {}^M d^U) \leq 0$	$g4(X): \text{Max}({}^S d^L, {}^M d^L) - d \leq 0$
L_0	$g5(X): L_0 - \text{Min}({}^S L_0^U, {}^M L_0^U) \leq 0$	$g6(X): {}^S L_0^L - L_0 \leq 0$
R	$g7(X): R - {}^S R^U \leq 0$	$g8(X): {}^S R^L - R \leq 0$
L_1	$g9(X): L_1 - {}^S L_1^U \leq 0$	$g10(X): {}^S L_1^L - L_1 \leq 0$
L_2	$g11(X): L_2 - {}^S L_2^U \leq 0$	$g12(X): {}^S L_2^L - L_2 \leq 0$
D	$g13(X): D - \text{Min}({}^S D^U, {}^M D^U) \leq 0$	$g14(X): {}^S D^L - D \leq 0$
D_i	$g15(X): D_i - {}^S D_i^U \leq 0$	$g16(X): {}^S D_i^L - D_i \leq 0$
D_L	$g17(X): D_L - {}^S D_L^U \leq 0$	$g18(X): {}^S D_L^L - D_L \leq 0$
n	*****	$g19(X): {}^M n^L - n \leq 0$
n_T	$g20(X): n_T - {}^S n_T^U \leq 0$	$g21(X): {}^S n_T^L - n_T \leq 0$
γ	$g22(X): \gamma - {}^M \gamma^U \leq 0$	*****
F_1	$g23(X): F_1 - {}^S F_1^U \leq 0$	$g24(X): {}^S F_1^L - F_1 \leq 0$
F_2	$g25(X): F_2 - {}^S F_2^U \leq 0$	$g26(X): {}^S F_2^L - F_2 \leq 0$
S_h	$g27(X): S_h - {}^S S_h^U \leq 0$	$g28(X): {}^S S_h^L - S_h \leq 0$
w	$g29(X): w - {}^M w^U \leq 0$	$g30(X): {}^M w^L - w \leq 0$
L_n	*****	$g31(X): L_n - L_2 \leq 0$
L_K	*****	$g32(X): L_K - L_2 \leq 0$
L_c	$g33(X): L_c - {}^S L_c^U \leq 0$	$g34(X): {}^S L_c^L - L_c \leq 0$
f_e	$g35(X): f_e - {}^S f_e^U \leq 0$	$g36(X): {}^S f_e^L - f_e \leq 0$
W_h	$g37(X): W_h - {}^S W_h^U \leq 0$	$g38(X): {}^S W_h^L - W_h \leq 0$
M	$g39(X): M - {}^S M^U \leq 0$	$g40(X): {}^S M^L - M \leq 0$
V_0	$g41(X): V_0 - {}^S V_0^U \leq 0$	$g42(X): {}^S V_0^L - V_0 \leq 0$
V_2	$g43(X): V_2 - {}^S V_2^U \leq 0$	$g44(X): {}^S V_2^L - V_2 \leq 0$
α_2	$g45(X): \alpha_2 - \alpha_2^U \leq 0$	*****
α_c	$g46(X): \alpha_c - {}^S \alpha_c^U \leq 0$	*****
α_F	*****	$g47(X): 1 - \alpha_F \leq 0$