TECHNICAL TRANSACTIONS | **CZASOPISMO TECHNICZNE**
CIVIL ENGINEERING | BUDOWNICTWO

6-B/2014

JACEK BOROŃ*

SELECTED QUALITATIVE CHANGES TO THE SOLVING OF ENGINEERING OPTIMIZATION PROBLEMS

WYBRANE ZMIANY JAKOŚCIOWE W ROZWIĄZYWANIU INŻYNIERSKICH PROBLEMÓW OPTYMALIZACJI

Abstract

This article presents and describes some qualitative changes that have occurred in the engineering design of building structures over the last forty years. With widespread access to computers and the development of software tools, optimization problems, which in the nineteen-seventies were solved analytically or, when justifiable, using mathematical machinery (e.g. first Polish minicomputers type Odra 1,300) are now often settled through the use of specialized add-ins to spreadsheets. This state of affairs has created a basis for significant changes in the quality of educational opportunities in the context of construction faculties within technical universities. These changes are illustrated with a simple example of the optimization (determination of the *dimensions of the beam subjected to bending)*.

Keywords: *engineering design*, *mathematical programming*, *optimizers*

Streszczenie

W artykule przedstawiono i opisano zmiany jakościowe, jakie zaszły w projektowaniu inżynierskim konstrukcji budowlanych na przestrzeni ostatnich czterdziestu lat. Dzięki powszechnemu dostępowi do komputerów i rozwojowi oprogramowania narzędziowego problemy optymalizacji, które w latach siedemdziesiątych ubiegłego stulecia były rozwiązywane analitycznie lub tylko w uzasadnionych przypadkach za pomocą maszyn matematycznych (tak wtedy określano pierwsze minikomputery typu Odra 1300), są obecnie rozwiązywane za pomocą m.in. dodatków do arkuszy kalkulacyjnych. Taki stan rzeczy stworzył podstawy do istotnych zmian jakościowych także na polu możliwości edukacyjnych na wydziałach budowlanych uniwersytetów technicznych. Wspomniane zmiany zilustrowano przykładem optymalizacji przekroju poprzecznego belki zginanej.

Słowa kluczowe: *projektowanie inżynierskie*, *programowanie matematyczne*, *optymalizatory*

* Ph.D. Eng. Jacek Boroń, Faculty of Civil Engineering, Wroclaw University of Technology.

1. Introduction to the nature of the optimization

According to [3], the optimization of civil engineering structures involves, inter alia, choosing the geometric parameters and strength properties of both structural elements and entire structures. This choice entails searching for the extreme values in terms of specific criteria. The searching process is conducted in an objective and rational manner, therefore, without the need to draw on a designer's intuition, special skills or professional experience. In this way, optimization replaces the part of the design process which entails the selection of shapes and dimensions and then checking the strength conditions and other constraints. Due to its specific nature, certain experts label the optimization of civil engineering structures as the synthesis of structures. The question is whether the optimization can and should replace the design process in its traditional sense, i.e. is the goal of optimization to include all the variables and parameters defining the structure so that a solution to an optimization problem is tantamount to obtaining the target structural design? The author of this paper does not think it is possible to provide a clear-cut answer to this question at the present time, as an optimization problem involving all the variables and parameters of a more complex structure might turn out to be impossible to solve, or looking for such a solution might be pointless due to e.g. a lot of effort required and high computational costs. In such cases, it is advisable that a rigorously formulated optimization problem involves only part of the variables and parameters, and that a designer decides about the second part, because in the mathematical model for optimization there will always be variables and parameters which are obvious and which should not burden the optimization. The goal of synthesis is to avoid arbitrariness during, for example, the process of selecting structure shapes, as optimization aims to determine the relations between internal forces, shape parameters, and the distribution of material, which in turn should undoubtedly bring multilateral benefits. To sum up the above considerations, although the optimization of civil engineering structures should be counted among the methods of structural mechanics or of the so-called general theory of optimization, and should not be considered as an economic activity, like any other technical activity, it still concerns economic issues.

2. Understanding how optimization software works

This paper now presents an illustration of the issue by presenting the standard add-in to the MS Excel® spreadsheet. The add-in is called Solver® and is developed by Frontline Systems. Although MS Excel® has only been equipped with the basic version of Standard Excel Solver® since 1997, the add-in's computational capabilities are able to solve problems which contain even several dozen decision variables and the same number of constraints. What else makes this add-in a useful tool? Most of all, a simple user interface which can be grasped even by people with only a general idea of how to use spreadsheets. The Excel Solve® interface does not deter designers with extensive mathematical nomenclature or complex formulas, nor does it force users to design on their own or probe the essence of the applied mathematical procedures.

The primary objective of the Excel Solver® optimization software is to find a solution to a problem – that is, to determine the values for the decision variables which satisfy all constraints (imposed on both the individual decision variables and the relationships between them) whilst at the same time, providing the extreme value (i.e. the minimum or maximum value) of a single objective function (which is a kind of yardstick of the solution quality and the function of the aforementioned decision variables). A well-formulated mathematical model of optimization is always solvable, but the solution time can be different and depends, perhaps above all, on the following three features of that model:

– The size of the model (understood as the number of decision variables and constraints, which translates into the total number of equations to be solved);
– The mathematical relationships (i.e. linear or non-linear) between the objective and constraints and the decision variables;
– The possibility of applying integer constraints (which entails a requirement that the area of feasible solutions includes only whole number values and involves the use of the methods of discrete optimization, which prolong the solution time).

Other issues, such as poor scaling, can also affect the solution time and quality, but the three above-mentioned features affect the intrinsic solvability of a problem (represented with a mathematical model). Naturally, the time needed to find a solution can also be reduced by faster algorithms and faster processors; however, solving some of the non-convex (i.e. concave) or non-smooth problems can take years, even with the fastest computers conceivable.

The total size of the mathematical model and the use of integer constraints can quite easily be assessed as early as the stage of examining the correctness and suitability of the model itself. A much harder task is to assess the mathematical relationships in the model, and it is these relationships which often have a decisive impact on the quality of the solution and on how long it takes to find a solution.

Summing up [4]:

– If the objective function and the constraints are linear functions of the decision variables (i.e. the latter are raised to the power of one), there is a fair chance that it will be possible to find a globally optimal solution (a globally conditional extreme value) relatively quickly, given the size of the mathematical model. In addition, such a linear problem is labelled as convex, this means that all the functions occurring therein are also convex. The best method for solving such problems is the Simplex LP Solving Method [7].
– If the objective function and the constraints are smooth non-linear functions of the decision variables (i.e. the latter are raised to the power other than the power of one), the search for a solution will take longer. If, in addition, the problem is convex (see Fig. 1a)), it is certain that at a global optimum will be reached. However, if the problem is non-convex (see Fig. 1b)), even finding a locally optimal solution might prove difficult. The best method for solving this type of problem is the GRG Nonlinear Solving Method.
– If the objective function and the constraints are non-linear and non-smooth, and the problem is also non-convex (in practice, if you use e.g. the spreadsheet IF, CHOOSE or LOOKUP functions that use some or all of the decision variables), the best you can obtain is a so-called good solution (which is better than the initial values, but is neither a locally nor a globally optimal solution). A whole range of Evolutionary Solving Methods has been developed in order to solve such problems.
– It may happen that within one problem, there is a desire to use integer, binary, and any other constraints and solve this problem with all the three Solving Methods mentioned above. However, this will make such a problem non-convex and all the more difficult to solve.

With the Simplex LP Solving Method, while searching for a globally optimal solution, a solution may be encountered that is close to optimal (only slightly worse in quality and still worthy of attention) and found much quicker. On the other hand, with the GRG Nonlinear and Evolutionary Solving Methods, in most cases it is expected to lead to a good, but probably not optimal, solution (and therefore not the best one).

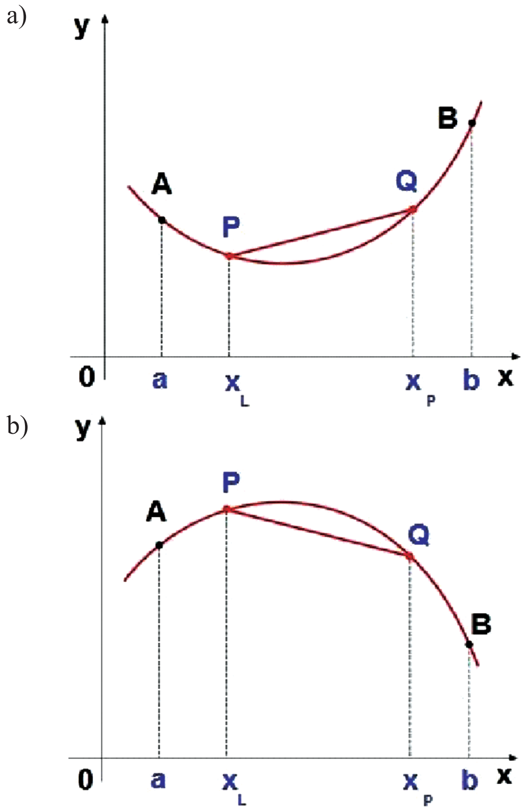

Fig. 1. Graphical representation of problems: a) convex problem and b) concave problem

3. Possible applications and computations

3.1. Rectangular wooden cross-section, graphical and analytical solution

Using the simplest example possible, the summary of which is taken from [5], we will show the difference between traditional designs (also known as intuitive or classic) and optimal designs (which is a kind of synthesis that uses mathematical programming).

The task is to design a rectangular cross-section of a wooden beam which is able to transfer a pre-set bending moment *M*.

$$M = 167\text{ kNm} = 0.167\text{ MNm}$$

The allowable stress (σ_{dop}) for the beam's material is:

$$\sigma_{dop} = 10.00\text{ MPa}$$

The beam will fulfil its function if it is able to transfer the pre-set bending moment (i.e. the stress in the beam's cross-section does not exceed the allowable values). Naturally, the designed girder must demonstrate sufficient protection against the loss of stability while bending. To simplify this task, this condition has not been explicitly formulated, but it will be satisfied once the beam's cross-sectional dimensions remain within practically reasonable limits. The ratio of cross-sectional dimensions should therefore adopt the tested values which guarantee sufficient protection against the loss of stability.

In traditional design, the course of action is to adopt some numerical values (width and height of the beam's cross-section), use these values in the beam stress formula and check whether this condition is satisfied. If the stress exceeds the allowable value σ_{dop}, a designer must adjust the dimensions accordingly and use them in the formula once again. This procedure is repeated until stress σ remains within the allowable value limits σ_{dop}.

$$\sigma = \frac{M}{W} = \frac{6 \cdot M}{b \cdot h^2} \leq \sigma_{dop}$$

where:

- b, h – beam cross-sectional dimensions: b – width and h – height,
- M – the previously mentioned bending moment; it was assumed M = 167 kNm (0.167 MNm),
- W – section modulus of rectangle; it was assumed $W = (b \cdot h^2)/6$,
- σ_{dop} – allowable stress; it was assumed σ_{dop} = 10.00 MPa (both its value and dimensioning method were adopted for illustrative purposes only).

A designer will try to 'use' the cross-section to the fullest, i.e. to choose its dimensions (b and h) so that the resulting stress values are close to the allowable limits. In further considerations, it will become whether achieving the allowable stress values is really sufficient to make the most of the material. Once the cross-section dimensions which do not exceed the beam stress limits are found, the problem is practically solved, i.e. the designed cross-section of the beam is able to transfer the pre-set bending moment (the only limitation of optimization).

However, the question of whether the selected cross-section is optimal (in other words – the best), cannot be satisfied with only one variant of the cross-section even if the allowable stress values have been reached – that is, even if the cross-section is 100% used. This is insufficient because you cannot be sure that the selected cross-sectional dimensions of the beam are the best of all possible options. In order for a girder to be optimal, it is not enough if it only fulfils its task of being able to transfer the pre-defined bending moment, as it is certainly possible to design beams with other cross-sectional dimensions which would also satisfy these requirements. Therefore, it must be clarified what is meant by saying that one design is better than the other. For this purpose, a criterion for assessing the quality of a design solution will have to be specified. The most frequently used criterion is the economic criterion, representing the generalized cost of a structure. In this example, the girder quality assessment criterion has been associated with the area of the beam's cross-section. The optimal girder will have the smallest cross-sectional area of all the girders able to transfer the pre-defined bending moment. It is obvious that a designer can design and compare only a limited number of girder variants, and that there will always be a possibility of there being another structure which has a smaller cross-sectional area and is therefore better. This possibility is always present in traditional design.

For example, in a rectangular cross-section of the following dimensions:

$$b_{\mathrm{I}} = 40\ \mathrm{cm} = 0.40\ \mathrm{m}$$

$$h_{\mathrm{I}} = 50\ \mathrm{cm} = 0.50\ \mathrm{m}$$

$$A_{\mathrm{I}} = b_{\mathrm{I}} \cdot h_{\mathrm{I}} = 40 \cdot 50 = 2{,}000\ \mathrm{cm}^2 = 0.2\ \mathrm{m}^2$$

where:

A_{I} – cross-sectional area of the first, pre-established girder,

the bending moment M stated earlier yields the following stress values:

$$\sigma = \frac{M}{W} = \frac{6 \cdot M}{b_{\mathrm{I}} \cdot h_{\mathrm{I}}^2} = \frac{6 \cdot 0{,}167}{0.40 \cdot 0.50^2} = 10.00\ \mathrm{MPa}$$

$$\sigma = 10.00\ \mathrm{MPa} = \sigma_{dop} = 10.00\ \mathrm{MPa}$$

On the other hand, in a rectangular cross-section of the following dimensions:

$$b_{\mathrm{II}} = 20\ \mathrm{cm} = 0.20\ \mathrm{m}$$

$$h_{\mathrm{II}} = 71\ \mathrm{cm} = 0.71\ \mathrm{m}$$

$$A_{\mathrm{II}} = b_{\mathrm{II}} \cdot h_{\mathrm{II}} = 20 \cdot 71 = 1{,}420\ \mathrm{cm}^2 = 0.142\ \mathrm{m}^2$$

where:

A_{II} – cross-sectional area of the second, pre-established girder,

the bending moment M stated earlier yields the following stress values which do not fully utilize the stress limits:

$$\sigma = \frac{M}{W} = \frac{6 \cdot M}{b_{\mathrm{II}} \cdot h_{\mathrm{II}}^2} = \frac{6 \cdot 0.167}{0.20 \cdot 0.71^2} = 9.92\ \mathrm{MPa}$$

$$\sigma = 9.92\ \mathrm{MPa} < \sigma_{dop} = 10.00\ \mathrm{MPa}$$

However, let us consider the areas of these cross-sections:

$$A_{\mathrm{I}} = 0.2\ \mathrm{m}^2 > A_{\mathrm{II}} = 0.142\ \mathrm{m}^2$$

From the above, it follows that the beam with the cross-section A_{II} is better than the previous beam (in terms of the adopted criterion) but in this way, it is still impossible to compare all the possible projects of beam cross-sections (i.e. possible pairs of numbers for b and h). Let us now formulate the mathematical conditions of stability which were not explicitly stated earlier. To this end, it is necessary to determine the limits for the ratio of cross-sectional dimension of the girder. It must be noted that in this example, these limits are not derived from the stability conditions, but are adopted arbitrarily (now we are interested in principles and not the accuracy of the numbers). We assume that the ratio of cross-sectional dimension of the beam must satisfy the following conditions:

$$\frac{h}{b} \geq 1.00 \text{ (square or rectangular cross-section)}$$

$$\frac{h}{b} \geq 4.00 \text{ (rectangular cross-section, with the appropriate proportions)}$$

In addition, the optimality condition has been added, this defines the minimum cross-sectional area of the girder:

$$A = b \cdot h = \text{minimum}$$

The example includes only two variables which allow for a graphical representation of the task on the plane. Let us adopt a Cartesian coordinate system where the cross-section height h is on the horizontal axis, and the cross-section width b is on the vertical axis (see Fig. 2). The coordinates of each point from the first quarter of the plane may be the dimensions of the beam's cross-section. Conversely, each girder cross-section of width b and height h can be represented as a point of this plane. Let us also present on this plane an area bounded by the above specified conditions. The first condition can be represented graphically on the plane after replacing the inequality for the allowable beam stress with the equality, and after substituting M and σ_{dop} for numerical values. Next, the equality has to be transformed so that the b dimension becomes the function of the h dimension.

$$\sigma = \frac{M}{W} = \frac{6 \cdot M}{b \cdot h^2} = \sigma_{dop}$$

$$\frac{6 \cdot 167 \, \text{kNm}}{b \cdot h^2} = 10{,}000 \text{ kN/m}^2$$

$$b = \frac{1}{h^2 \cdot 10}$$

Therefore:

Condition no. 1 is shown as constraint no. 1: $b = \dfrac{1}{h^2 \cdot 10}$,

condition no. 2 is shown as constraint no. 2: $b = h$,

and condition no. 3 is shown as constraint no. 3: $b = \dfrac{h}{4}$.

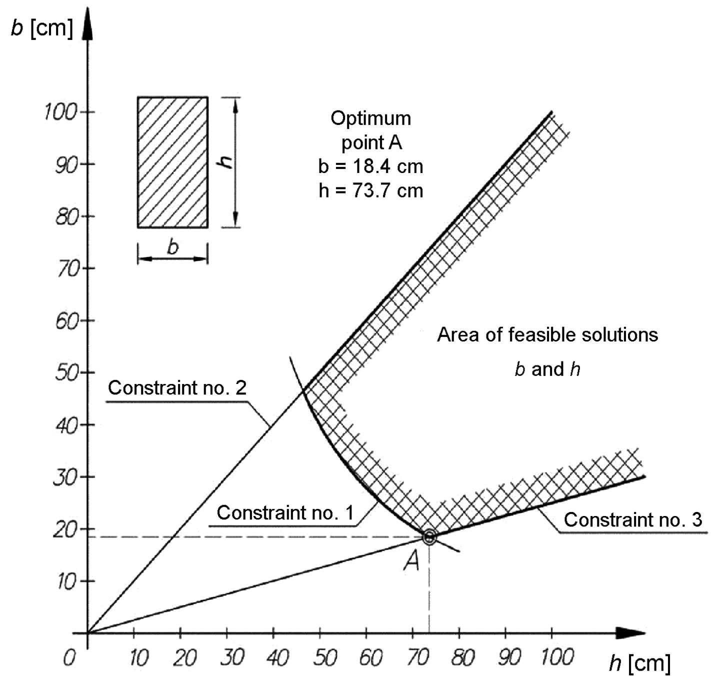

Fig. 2. Optimization of a rectangular cross-section of a bending beam

As shown by the figure above, the point A (the optimum) lies at the intersection of the lines drawn according to the following equations:

$$\begin{cases} b = \dfrac{1}{h^2 \cdot 10} \\ b = \dfrac{h}{4} \end{cases}$$

Solving the above system of equations has yielded the optimum cross-sectional dimensions of the beam:

$$b = 18.4 \text{ cm} = 0.184 \text{ m}$$

$$h = 73.7 \text{ cm} = 0.737 \text{ m}$$

3.2. Rectangular wooden cross-section, numerical solution using Standard Excel Solver®

Let us now solve the same problem with the popular optimization software. First, it is necessary to place the mathematical model in the spreadsheet: the decision variables b and h (cells H59:H60); the optimization criterion (formula $z = b \cdot h$ in cell H62); the constraints (definitions in cells B62:B64; see Fig. 3).

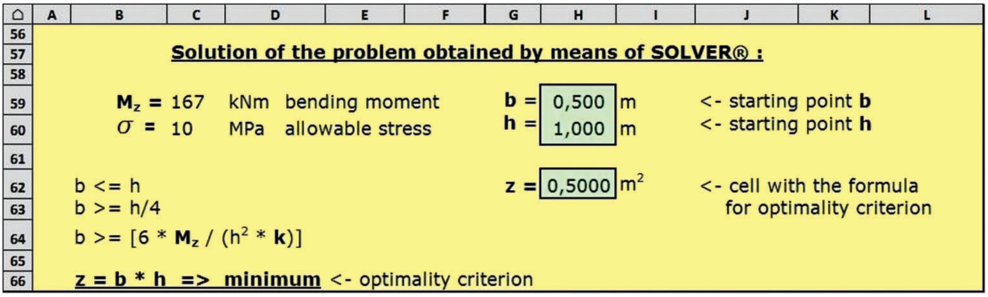

Fig. 3. Fragment of the spreadsheet with the essential elements of the optimization model

Building of the mathematical model continues by opening the Excel Solver® dialogue box and entering the address of the objective function, the type of the extreme value (i.e. the minimum value), the location of the decision variables and the constraint formula (see Fig. 4). Due to the nonlinearity of the problem, the GRG Nonlinear Solving Method is the right method here. The next step is to run SOLVE, keeping the other default settings of the add-in intact.

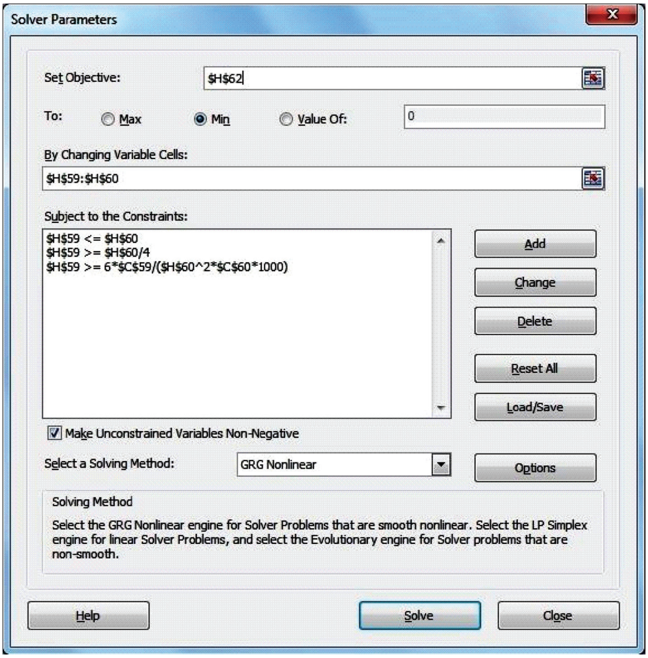

Fig. 4. Dialogue box: Solver Parameters

In this case, the solution time reduces down to a fraction of a second, and Solver® informs the user about the effects of its actions in the next dialogue box (see Fig. 5).

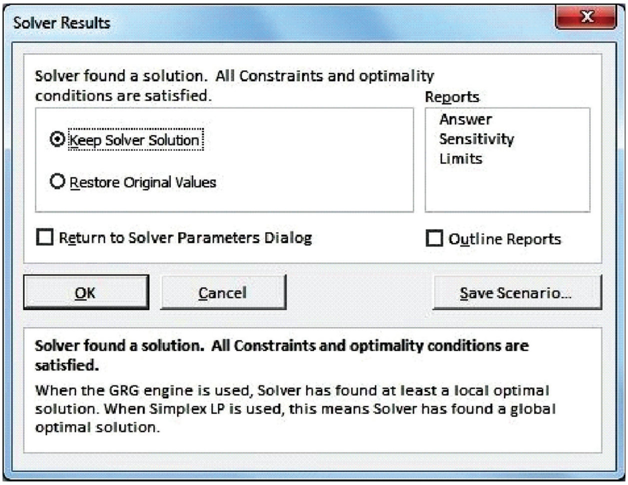

Fig. 5. Dialogue box: Solver Results

The dialogue box shows that Excel Solver® has found an exact solution, which can be stored and seen in the spreadsheet by clicking OK (see Fig. 6).

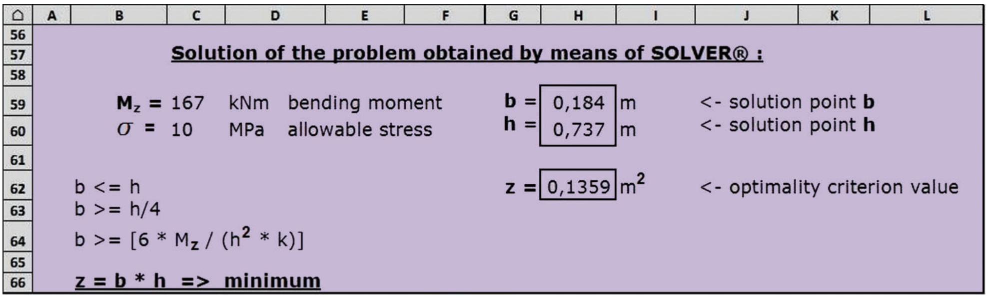

Fig. 6. Fragment of the spreadsheet with the solution to the optimization problem

While still in the Solver Results dialogue box (see Fig. 5), there is the opportunity to additionally produce and save (in a separate worksheet of the current workbook) any of the available reports (e.g. the answer report, the sensitivity report, or the limits report). The sample answer report (see Fig. 7) includes all the elements of the mathematical optimization model and also additional information on which constraints are 'binding' and which are not.

What is most important is that the results of the structural optimization of a simple wooden element, obtained with the use of two completely different methods (with 40 years between them), are exactly the same. Problems which once required designers to have access to company computers are now successfully solved in the privacy of one's home, in university laboratories or while travelling thanks to the wide availability of desktop PCs and laptops.

Microsoft Excel 14.0 Answer Report
Worksheet: [solver-belki_raporty.xls]Belka drewniana
Report Created: 2014-03-08 17:45:36
Result: Solver found a solution. All Constraints and optimality conditions are satisfied.
Solver Engine
Engine: GRG Nonlinear
Solution Time: 0,016 Seconds.
Iterations: 3 Subproblems: 0
Solver Options
Max Time 100 sec, Iterations 1000, Precision 0,0000001
Convergence 0,000001, Population Size 100, Random Seed 0, Derivatives Forward, Require Bounds
Max Subproblems Unlimited, Max Integer Sols Unlimited, Integer Tolerance 0,001%, Solve Without Integer Constraints, Assume NonNegative

Objective Cell (Min)

Cell	Name	Original Value	Final Value
$H$62	z = from local extremes, which can be several	0,5000	0,1359

Variable Cells

Cell	Name	Original Value	Final Value	Integer
$H$59	b = from local extremes, which can be several	0,500	0,184	Continous
$H$60	h = from local extremes, which can be several	1,000	0,737	Continous

Constraints

Cell	Name	Cell Value	Formula	Status	Slack
$H$59	b = from local extremes, which can be several	0,184	$H$59<=$H$60	Not Binding	0,552972825
$H$59	b = from local extremes, which can be several	0,184	$H$59>=$H$60/4	Binding	0,000
$H$59	b = from local extremes, which can be several	0,184	$H$59>=6*$C$59/($H$60^2*$C$60*1000)	Binding	0,000

Fig. 7. Sample Answer Report to the optimization problem

4. Conclusions

The Excel Solver® optimization add-in, developed and constantly streamlined by Frontline Systems®, is a universal and user-friendly tool to solve various problems, including civil engineering optimization problems. It must be remembered that optimization problems may be both simple and 'cheap' and complex and 'expensive' to solve. This is due to the mathematical relationships within the optimization model itself (linear and non-linear/convex and non-convex problems) which determine the difficulty of obtaining a solution and the level of certainty as to whether the solution is a real conditional extreme value or merely a value somewhere near the extreme. Additionally, these relationships have a substantial impact on the size of mathematical models, and thus, on the possibility of solving the problems which such models describe. Some highly advanced optimization tools are able to divide the main problem into sub-problems (linear, smooth non-linear, and non-smooth) and then to match each subproblem to the most appropriate solving method. What we, the users of tool software (both engineers and managers), care most about [2] is that the mathematical relationships in models are as simple as possible.

References

[1] Baker K.R., *Optimization Modeling with Spreadsheets*, 2nd ed., Wiley, 2011.

[2] Boroń J., *Optimizers in Solving Managerial and Structural Problems*, [in:] *Information Systems Architecture and Technology: Model Based Decisions*, eds. J. Świątek et al., Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław 2008, 15–25.

[3] Brandt A.M., *Podstawy optymalizacji elementów konstrukcji budowlanych* (*Basics of Optimization the Components of Building Structures, in Polish*), Państwowe Wydawnictwo Naukowe, Warszawa 1978.

[4] Frontline Solvers, http://www.solver.com/.

[5] Leśniak Z.K., *Metody optymalizacji konstrukcji, przy zastosowaniu maszyn matematycznych* (*Methods for Structural Optimization Using Mathematical Machines, in Polish*), Arkady, Warszawa 1970.

[6] Powell S.G., Baker K.R., *Management Science: The Art of Modeling with Spreadsheets*, 4th ed., Wiley, 2013.

[7] Vanderbei R.J., *Linear Programming: Foundations and Extensions*, 3rd ed., International Series in Operations Research & Management Science, 2008, Vol. 114, Springer Verlag.

[8] Winston W.L., Venkataramanan M., *Introduction to Mathematical Programming: Applications and Algorithms*, Vol. 1, 4th ed., Thomson Learning, 2002.