Optimization of a car seat under impact load

F. J. Szabó
Department of Machine Elements, University of Miskolc, Hungary

Abstract

The behaviour of the metal structure of a car seat subjected to impact loading has been investigated computationally by the finite element program system COSMOS/M, where a macro has been developed for the calculation of energy transformations during impact with the body of the driver in an accident. The finite element results and the calculated impact behaviour have been verified by an experiment. Having good agreement between the computational and experimental results, the macro calculating the behaviour data of the structure has been implemented to the optimization process for minimum mass. For the optimization the Random Virus Algorithm (RVA) is used, which is a new algorithm developed by the author for use in multidisciplinary optimization problems. The new concept of the algorithm is to model the very fast reproduction of biological (or computer) viruses, which give extremely good efficiency to the optimization algorithm. The optimization algorithm, the finite element model and the energy calculation are stored and running as one macro inside the finite element system. The results of the finite element calculations, experiments and of the optimization are presented in tables, finite element contour pictures and photos of the experiments. The method is able to take into account higher eigenmodes and eigenshapes therefore it can be applicable for higher speed impacts, too, after strong experimental verification. The macro makes it possible to use all the multidisciplinary or multiphysics features of the original finite element program system, therefore makes it possible to combine several phenomena during the multidisciplinary optimization, e.g. heat transfer or temperature dependent material characteristics with impact load, which can open new horizons for the optimization of structures for extreme loads during accidents, terror-attacks, explosions or tornados. A numerical example of the metal structure of a car seat is presented.

Keywords: optimization, impact load, metal structures, programming FEM.

WIT Transactions on the Built Environment, Vol 98, © 2008 WIT Press
www.witpress.com, ISSN 1743-3509 (on-line)
doi:10.2495/SU080231

1 Introduction

By using the built-in programming language of the COSMOS/M finite element program system, a macro has been developed for the modelling, analysis and optimisation of structures loaded by impact load. The impact load is supposed as the dynamic load during the impact of a body having its mass (m) and dropped from a given height (h). All the necessary data are calculated by the program and the results of displacements and stresses caused by the impact load can be post-processed as conventional finite element results. The main steps of the calculation process:

- Build-up the finite element model of the structure;
- Calculation of static displacement caused by the weight of the dropped body at the contact area of the target body;
- Determine the mass reduction factor by using numerical integration of the displacement field determined in the previous step;
- Calculation of the equivalent mass and the mass ratio;
- Determine the dynamic factor.

Multiplying the static stress and displacement results by the dynamic factor, one can get the results for impact load. The macro contains an optimum searching algorithm developed for multidisciplinary optimisation. The algorithm is the Random Virus Algorithm (RVA), which has been developed specifically for finite element programming usage. The theoretical basis of the algorithm is the simulation and modelling of the behaviour and quick reproduction of biological (and computer) viruses. The computer code of the algorithm is very simple, easy to program in any programming language, the working and thinking of the algorithm are very efficient regarding the number of objective function evaluations and the number of constraint checks until reaching the optimum result. By combining the method for the analysis of the impact load and the RVA optimisation algorithm, it is possible to solve several special multidisciplinary optimisation problems, because the macro can use all the facilities and possibilities of the original finite element program. For example, it will be possible to combine the multiphysics possibilities with optimisation and impact load, which could lead to the solution of special complex problems, e.g. thermal effects (fire) with impact load and optimisation, as well as magnetic structure or fluids combined with impact load and optimisation. The solution to these kinds of problems could be very difficult or impossible in conventional finite element program packages.

The results of the method and of the macro are presented through the numerical example of the metal structure of a car seat back loaded by the car passenger in the case of an impact at 30 kph. (It is supposed that a person sits in the car and another car hits the car from the backside and the seat will deform because of the inertial force exerted by the person sitting in the car). Before the optimisation, the results for the structural behaviour are compared with the experimental ones in order to verify the accuracy of the numerical calculations.

2 Analysis of structures subjected to impact load by programming FEM

Let us suppose that the investigated structure is subjected to a load caused by an object of mass m_o dropped from height h. After the collision of the object into the structure, the structure starts a vibrating motion. Supposing uniform mass distribution of the structure, this motion can take place through many possible eigenfrequencies (infinite number of possible eigenfrequencies). According to the Carnot rule, if we investigate only the first eigenfrequency for this motion, we lose an amount of energy compared to the situation if we suppose the structure to move with the impact speed at every point, according to Ponomarjov [1]. This rule says that the error made in stress results is four times higher than in case of the results for displacements. For this reason, it is always recommended to verify the computational results by experiments. These experiments will show the limits of applicability of the method for higher speeds and nonlinear deformations, too. Taking into account the higher eigenfrequencies will increase the accuracy of the method.

Although the method is able to handle general three dimensional structures, for simplicity reasons let us investigate a plate-like structure. These kinds of structures are commonly used in many fields of engineering science (cars, sidewalls of buildings, roof structures, covers of machines, etc.). Many more complicated problems can be modelled by or can be originated in plate structures. The following equations are derived for non-elastic impact (when the structure and the falling object can move together after the impact). In the case of elastic impact, when the falling object will rebound from the structure, before the rebounce they move together for a short time and during this period the equations will be usable. In these equations, the structure is substituted by a one-mass vibrating system, with a spring constant calculated by the static displacement of the structure caused by the object's mass:

$$c = \frac{w_{s\max}}{m_o g} \tag{1}$$

where w_{smax} is the static displacement of the structure at the point of impact and g is the gravity. For the calculation of the kinetic energy of the structure it is supposed that the velocity of the different points of the structure are proportional to the displacement:

$$\frac{v_p}{w_p} = \frac{v_{\max}}{w_{\max}}, \tag{2}$$

$v_p(x,y)$ is the velocity of a point of the structure and $w_p(x,y)$ is the displacement of the same point and v_{max} is the velocity of the impact point on the structure. The kinetic energy can be calculated:

$$E_k = \int_{x=0}^{a} \int_{y=0}^{b} \frac{q v_p^2}{2} dxdy = \frac{v_{\max}^2}{2 w_{\max}^2} \int_{x=0}^{a} \int_{y=0}^{b} q w_p^2 dxdy, \tag{3}$$

q is the specific mass of the structure:

$$\int_{x=0}^{a}\int_{y=0}^{b} q\,dxdy = m \tag{4}$$

The kinetic energy of the substituting one-mass vibrating system should be equal to the energy calculated in eqn (3). Supposing uniform mass distribution:

$$m_e \frac{v_{\max}^2}{2} = \frac{v_{\max}^2}{2w_{\max}^2} m \int_{x=0}^{a}\int_{y=0}^{b} w_p^2\,dxdy \tag{5}$$

and

$$\frac{m_e}{m} = k_m = \int_{x=0}^{a}\int_{y=0}^{b} \left(\frac{w_p}{w_{\max}}\right)^2 dxdy \quad ; \quad m_e = m\,k_m \tag{6}$$

where k_m is the coefficient of mass reduction. The equivalent mass is denoted by m_e.

Supposing non-elastic impact, the point of impact on the structure and the falling object are moving together after the impact by v_{max} velocity. This means that the velocity of the structure will change from the original 0 [m/s] to v_{max}, and the velocity of the falling object will change from the maximum v_1 speed reached during the falling down from the height h [m] to v_{max}. The equilibrium equation of momentum leads to the following:

$$m_o v_1 = (m_0 + m_e) v_{\max} \;\; ; \;\; v_{\max} = \frac{m_o}{m_0 + m_e} v_1 \;\; ; \;\; v_1^2 = 2gh \tag{7}$$

The total kinetic energy of the structure:

$$E_{kr} = \frac{m_0 + m_e}{2} v_{\max}^2 = \frac{m_0 v_1^2}{2} \frac{m_0}{m_0 + m_e}. \tag{8}$$

The following equation states that the total kinetic energy of the structure plus the work of the gravity must be equal to the work of deformation:

$$\frac{m_0 v_1^2}{2} \frac{m_0}{m_0 + m_e} + (m_0 + m_e) g w_{\max} = \frac{m_0 g}{2} \frac{w_{\max}^2}{w_{s\max}}. \tag{9}$$

Introducing Ψ dynamic factor and μ mass ratio:

$$\Psi = \frac{w_{\max}}{w_{s\max}} \quad , \quad \mu = \frac{m_e}{m_0}. \tag{10}$$

Using these factors in eqn (9), we get:

$$A\Psi^2 - B\Psi - C = 0, \tag{11}$$

where:

$$A = \frac{m_0 g}{2} w_{s\max} \;\; ; \;\; B = (m_0 + m_e) g w_{s\max} \;\; ; \;\; C = \frac{m_0^2 v_1^2}{2(m_0 + m_e)}.$$

The solution of eqn (11):

$$\Psi = (1+\mu) \pm \sqrt{(1+\mu)^2 + \frac{2h}{(1+\mu)w_{s\max}}} \quad . \tag{12}$$

By using this method, one can determine the dynamic deformation caused by the falling object and it is possible to calculate the stresses, too.
Summarizing the steps of the method:

1. Calculation of the mass of the structure.
2. Finite element calculation of the static deformations caused by the weight of the falling object, w_{smax}.
3. Calculation of the mass reduction factor.
4. Determining of the equivalent mass.
5. Calculation of the mass ratio.
6. Solving the dynamic factor.
7. Result for the dynamic deformation: $w_{max} = \Psi\ w_{smax}$.

These steps could be steps of a program code, because the method is easy to program in any programming language. Writing this code in a built-in programming language of a finite element program system (e.g. COSMOS/M, or ANSYS APDL), makes it possible to perform the necessary static calculation of static deformations inside the same finite element program system. By using numerical integration, this program can be developed into a complete program for the analysis of any three dimensional structure for impact load. The code of this program was written in the built-in programming language of the COSMOS/M finite elements program system of SRAC [2] by applying a numerical integration technique and performing steps 1–7 (Szabó [3]). An important advantage of this method is that we can always use all the possibilities of the original finite element program system and one can combine these new features of impact with all the multidisciplinary and multiphysics features of the original program system. In this way it will be possible to solve complex and special problems, which could be very difficult or impossible to solve in any other program system. For example, one can combine the impact load and optimisation with nonlinear structural behaviour, or with thermal effects and temperature dependent material characteristics (e.g. in the case of fire), magnetic effects, fluids, etc. The solution of these kinds of special and complex problems could be very useful during the design and optimisation of structures subjected to extreme loads in the case of disasters (fire, tornado, etc.), accidents or explosions due to natural catastrophes or terrorist attacks. The results and solutions of these problems could lead to the design, optimisation and fabrication of newer and safer products, building elements, car parts, human protective clothing or helmets, safety elements, walls, covers, etc.

3 The RVA algorithm for multidisciplinary optimisation

Evolutionary algorithms are very efficient and robust algorithms for optimum searching and they are capable of handling a large number of design variables and/or computer capacity consuming calculations of the objective functions and design constraints. These characteristics make it possible to apply these algorithms in multidisciplinary optimization problems. The computation time necessary to reach the final optimum depends on the thinking and strategy of the algorithm and also on the computation time necessary to evaluate the objective function and to check the fulfilment of the design constraints. Therefore it is very important that the algorithm should be very efficient from the point of view of the necessary objective function evaluations and design constraint checks until reaching the final optimum. This fact makes it possible and necessary to develop newer algorithms having a higher and higher efficiency in this context.

Genetic Algorithm, Particle Swarm Algorithm and Ant Algorithm are based on investigations of biological systems and the thinking of these algorithms simulates the behaviour of these systems. The efficiency of these algorithms is because of the efficiency of the given biological system behind the theory of the algorithm. The given biological system is a result of several thousands of years of evolution, the existence and development of these systems is the proof of their success. This success is transferable into the optimization process and the given algorithm will be successive too. Let us investigate a very efficient biological construction: a virus. The efficiency of this biological system is in its very fast reproduction capacity. If the circumstances and conditions (temperature, light, oxygen, food) are good, viruses can reproduce themselves at a very high speed and they will cover almost all the possible places in the area of investigation. If the life conditions are not good, or the conditions show a non-uniform distribution, higher numbers of viruses can be found in an area of better conditions than in some areas giving poor conditions for virus life. Therefore these biological structures are very efficient at finding the best conditions for life, the highest number of virus entities will be found in the area having the best life conditions. Another very important thing is that a virus is always a very simple construction, containing only the most important information necessary for life and reproduction. This simplicity gives a very high flexibility to a virus in changing and mutation, therefore they can accommodate several conditions very easily. The efficiency of a virus is therefore very high from the point of view of behaviour, construction, life reproduction and changing. This efficiency is applied during the development of computer viruses, which show several similarities to real biological viruses (simple structures, very fast reproduction, changing easily). It is mainly these characteristics that make a computer virus also a very efficient system. Applying this multi-form efficiency in development of an optimization algorithm could result in high efficiency in the optimum searching process too (Szabó [4]).

During the build-up of the algorithm, the first step is to find the starting points fulfilling all the explicit and implicit constraints. It is possible to generate coordinates using the explicit constraints and check the generated points against the implicit constraints. In order to keep the simplicity of the virus algorithm, the number of starting points is proposed to be very low. The coordinates of the starting points can be denoted by x_i , $i = 1,2,\ldots,n$ where n is the number of design variables. In this case the points can be denoted as P_j , $j= 1,2,\ldots,m$, where m is the number of the starting points. The explicit constraints of the design variables:

$$l_i \le x_i \le h_i \qquad i=1,2,\ldots,n, \tag{13}$$

l_i are the lower limits, h_i are upper limits of the constraints. The implicit constraints can be written in the following form:

$$u_k \le f_k(x_i) \le v_k \quad , \quad i=1,2,\ldots,n \quad , \quad k=1,2,\ldots,p, \tag{14}$$

where p is the number of implicit constraints. The starting points are vectors in the design space:

$$P_j = \{x_i\}_j . \tag{15}$$

They can be found in the feasible region of explicit constraints by using random numbers and after they are checked against the implicit constraints. If a point is unfeasible, a new one should be generated. The goal of the optimization process is to find the extremum value (maximum or minimum) of the objective function:

$$\Omega = extr[F(x_i)], \tag{16}$$

where F is an arbitrary nonlinear function of the design variables.

Once the starting points are generated, the reproduction procedure is started:

$$y_i^{\alpha} = x_i + R_i q(h_i - l_i). \tag{17}$$

Here y_i are the coordinates of the new point generated, R_i are random numbers between the values of 0 and 1, α is the number of new entities generated in the reproduction procedure and q is the spreading parameter. The proposed value of the spreading parameter is between 0.5 and 0.8 in the case of the first three generations and between 0.2 and 0.4 afterwards. The reproduction step is executed for each starting point. The new generation created in this step can be denoted as generation α. The next generation (we can call it generation β, after γ and so on) can be created using the reproduction formula of eqn (17) for each point of the previous generation. In order to prevent an overwhelming number of points being controlled at the same time, it is necessary to select the points that have the best objective function value and destroy the points that have the worst objective function value. This procedure can be continued until a given number of generations is reached or the procedure can be ended if the maximum difference in objective function values regarding a generation will be under a given small value.

4 Numerical example – optimisation of the metal structure of a car set back for impact loading

Supposing a car accident, where a car hits another car from the back, the person sitting in the “target car” will be pressed into his seat due to his inertial forces. This load is an impact-like, sudden load exerted to the back of the seat. Before the optimisation, numerical and experimental investigations were made in order to observe the behaviour of the car seat structure subjected to impact loading and to verify the calculation method introduced in section 2 of this paper. The three dimensional model of the seat was made in the SolidEdge program and the finite element model was built in the COSMOS/M program system. Figs. 1 and 2 show the 3D and FEM model used for the investigations.

Figure 1: Three dimensional model of the car seat.

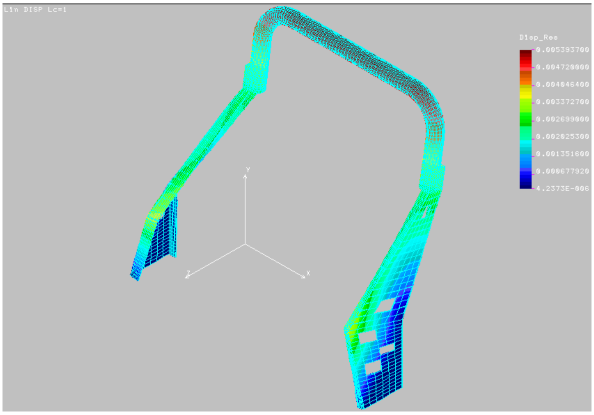

Figure 2: Finite element results of the optimised structure (displacements).

During the collision it is supposed that a 50 kg part of the person's body is hitting the car seat back at 30 km/h velocity. This is the impact loading to the seat back. As an experimental investigation, a mass of 50 kg was dropped from 3,5 m height onto the seat back, while the seat was positioned so that the back was horizontal (fig. 3). The experimental result for the maximum deformation of the seat back due to this loading was 250 mm, the finite element result using the method described in section 2 was 220 mm. An important part of the difference was caused by the deformation of the axial element placed at the bottom of the seat back, which was not as rigid as supposed in the finite element model. It can be said that the experimental and finite element results are in good agreement, therefore the proposed method can be used for the optimisation. During the optimisation the objective function was the mass of the structure, because it is placed in a car. The constraints of maximum deformation and maximum stress were applied. The design constraints were the diameter of the (upper) pipe element (d) of the seat, the thickness of the pipe element (t), the width (b) and the thickness of the lower element (c). For the optimisation the Random Virus Algorithm was used. The code of the algorithm as well as the code of the method described in section 2 were written in COSMOS/M built-in parametric language. The results of the optimisation are shown in table 1.

Figure 3: Position of the car seat for the experiment.

Table 1: Final results of the optimisation.

	d [mm]	t [mm]	b [mm]	C [mm]	mass [kg]
original	20	1,5	88	1,5	0,85
optimal	18	1	85	1,5	0,69
diff. [%]	-	-	-	-	18

It can be seen from the table that the final optimum structure, having the same rigidity characteristics as the original structure taken from a car selected from the marketplace, has approximately 18% smaller mass.

5 Conclusions

The metal structure of a car seat back has been optimised by using the RVA algorithm for minimum mass. The loading is the impact load when an accident is supposed. The analysis of structural behaviour for impact load has been performed in the COSMOS/M program system and the entire process is programmed as a macro. The results of the numerical analysis are in good agreement with the experimental ones. As a result of the optimisation process, the final optimum structure, having the same rigidity characteristics as the original structure taken from a car selected from the marketplace, has approximately 18% smaller mass. The results of the optimisation procedure and of the macro presented in this paper can be useful for the design, verification and optimisation of special security elements, covers, walls, car elements designed to be secure against crashes, blasts, terrorist attacks, tornados, etc.

References

[1] Sz. D. Ponomarjov (1965): Szilárdsági számítások a gépészetben 6. , (Strength calculations in machine design, in Hungarian), Műszaki Könyvkiadó, Budapest.
[2] Structural Research and Analysis Corporation (1997): COSMOS/M User's Guide, SRAC, Santa Barbara CA, USA.
[3] F. J. Szabó (2007): Analysis of structural behaviour for impact load by programming FEM, MicroCAD International Scientific Conference, Session K., 22–23 March, 2007, Miskolc, Hungary.
[4] F. J. Szabó (2007): Multidisciplinary optimization of gears using the Random Virus Algorithm. IFToMM 2007 World Congress, 18–22 June, 2007, Besançon, France.