Please note that the paper
distributed with the conference proceedings
is missing pages N9-2, N9-4, N9-6 and N9-8

TWENTYFIFTH EUROPEAN ROTORCRAFT FORUM

Paper n° N9

OPTIMAL DESIGN OF CRASHWORTHY STRUCTURES

BY

M. MORANDINI

SEPTEMBRE, 14-16, 1999
ROME
ITALY

ASSOCIAZIONE INDUSTRIE PER L'AEROSPAZIO, I SISTEMI E LA DIFESA
ASSOCIAZIONE ITALIANA DI AERONAUTICA ED ASTRONAUTICA

OPTIMAL DESIGN OF CRASHWORTHY STRUCTURES

Marco Morandini
Dipartimento di Ingegneria Aerospaziale, Politecnico di Milano, Italy

Abstract

This paper deals with an algorithm for optimization of crashworthy structures, which performs a multipoint approximation of the objective function and constraints. The crash response of the structure is predicted using a standard explicit finite-element code (PAM-CRASH™). Due to the expensive nature of the finite element simulation, a master-slave paradigm in an MPI environment is used to perform many different independent evaluations and to collect obtained results. The optimization of an energy-absorbing helicopter subfloor structural intersection is presented as a test case for the algorithm. The energy absorbed per unit of crushed mass proves to be a somewhat misleading performance parameter, while a more appropriate formulation of the objective function can lead to substantially better designs.

1. Introduction

Modern explicit finite-element codes are currently successfully used for the prediction of crash behaviour. The good level of confidence in the results of simulations, combined with the continuously increasing power of modern computers, allows the application of optimization methods to the design of crashworthy components. The first example of such approach can perhaps be found in the work of Mayer, Kikuki and Scott [5], where the shape optimization of an automotive rear rail is successfully performed using an optimality criteria combined with an homogenization technique. Following a completely different approach Etman, Adriaens, Slagmaat and Schoofs [5] developed an optimization tool for the commercial multi-body code MADYMO™; their code is based on a sequential linear programming algorithm and on a multipoint approximation of the objective function. Finally, the author is aware of some optimizations performed with the commercial optimization software PAM-OPT™ [5, 5 and 5], based on classical mathematic programming techniques as well as on the ability to approximate the objective function or some constraints. The use of some type of approximation is well established in the context of structural optimization. The paper by Barthelemy and Haftka [5] may be addressed for a review of the concept.

The application of mathematical programming techniques to the design of crashworthy components relies on the use of a simulation code for the prediction of crash behaviour. This means that the analyst faces an optimization problem where no derivatives information are available, and must be aware that the predicted value of the objective function or of some constraints can be inaccurate due to numerical errors or to the inadequacy of the model to predict the behaviour of the structure for a particular combination of design variables. Also, the time required for function evaluations dominates the total optimization time, while the optimization routine requires a negligible amount of computations. The optimization algorithm tries to reduce the total number of function evaluation and the influence of simulation errors. The considerable amount of cpu time required for a single function evaluation makes necessary to use a parallel environment to reduce the total optimization time.

The definition of the objective function and of constraints proves to have a big influence to the final design of the component, and should be carefully addressed before any attempt of optimization.

2. The optimization algorithm

The optimization algorithm is based on the well-established technique of sequential linear programming (SLP) [5]. This particular method of optimization was chosen because no informations are available on the regularity of the objective function – and hence on the existence of second-order derivatives – and because one has to perform a big number of function evaluations in order to estimate the hessian of the objective function by means of finite differences. Other optimization techniques, commonly known as probabilistic search methods, can be used when no derivative informations are available. Indeed, genetic algorithms, evolutionary programming or simulated annealing has been used with success for the optimization of complex industrial application problems, see for example the paper of S. Obajashi [5], but this class of optimization methods proves to require a really big number of function evaluations. Other search techniques could be used with good chances of success [5], but their application in a parallel environment seems troublesome.

The SLP method approximates the objective function and any additional constraints with a 1st order Taylor expansion. To this purpose, the jacobian of the objective function and constraints has to be evaluated. Additional move limits to design variables are added to the resulting linear programming problem in order to keep the solution in the region were the linear approximation is reasonably accurate. After solving the linear programming problem the solution is computed at the new design point; this allows to perform a solution check and to choose, according to the objective function decrease obtained and the eventual constraints violations, whether the proposed design point has to be accepted or rejected. If the proposed design point is accepted then it can be used as the starting point for a new iteration, otherwise the approximation is repeated starting from the previous point and with tighter move limits (Figure 1).

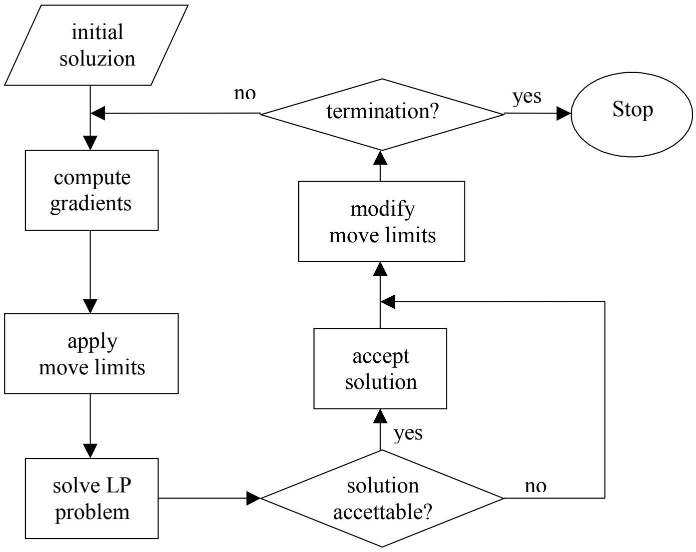

Figure 1: flowchart of the optimization program.

The jacobian of the objective function and constraints is approximated using a secant method. This approach can be found both in the commercial code PAM-OPT™ and in the work of Etman, Adriaens, Slagmaat and Schoofs [5]. The secant approximation of the derivatives is intended to reduce the influence of local "noise" introduced due to numerical errors by the finite element code and by the "chaotic" nature of crash phenomena [5, 5]. The second behaviour has been numerically investigated with the aid of Monte Carlo simulations [5], showing a somewhat chaotic behaviour of the solution for small variations of the angle of impact of a car model.

The linear programming problem is solved using the public-domain library lp-solve [5]. Key features of a successful SLP method is the choice of criteria for accepting design points and updating move limits. Details can be found in the book by Hafka and Gürdal [5] and in the paper by Etman, Thijssen, Schoofs and van Campen [5].

In order to reduce the total wall clock time needed to perform the optimization, the SLP program is based on a master-slave paradigm in an MPI environment. The master process drives the optimization, while the slave performs any function evaluation needed by the master (Figure 2). This coarse grained parallelism is effective only for the evaluation of gradients, when many independent function evaluations can be performed at the same time. On the contrarty, when a single evaluation is needed, no speed up is achieved but no additional computational time is spent due to the master-slave organization of the program. Furthermore, communication time is completely negligible with respect to the time required to perform a single function evaluation. A second level of parallelism can be achieved using a parallel version of the finite element code.

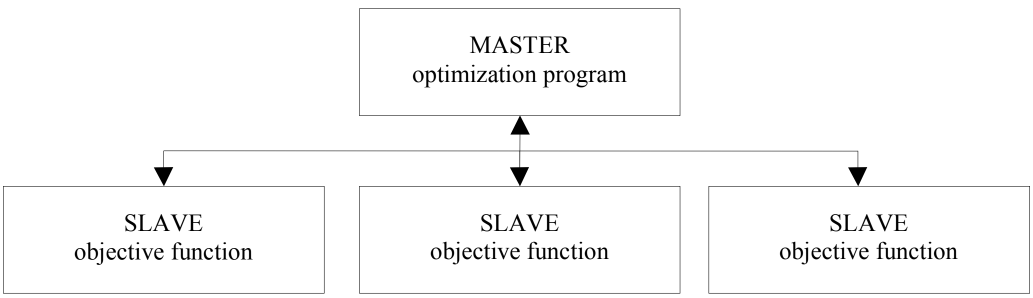

Figure 2: master/slave paradigm.

3. The test case

The test case involves the optimization of a simple intersection/cruciform subfloor element (Figure 3). The component was tested using a vertical impact drop tower. The impacting mass of 110 *Kg* hits the component with a velocity of 8 *m/s*.

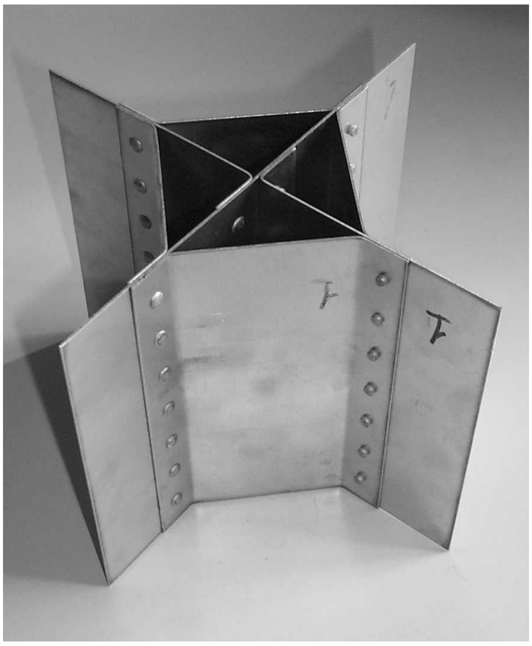

Figure 3: initial subfloor design.

The finite element model of the component was designed in order to limit the cpu time required to perform a single simulation, still keeping a good correlation between experimental and numerical behaviour (Figure 4, Figure 5). The cpu time required for a single evaluation is of about two hours on a single HP PA-RISC 8000 processor with a 180 MHz clock.

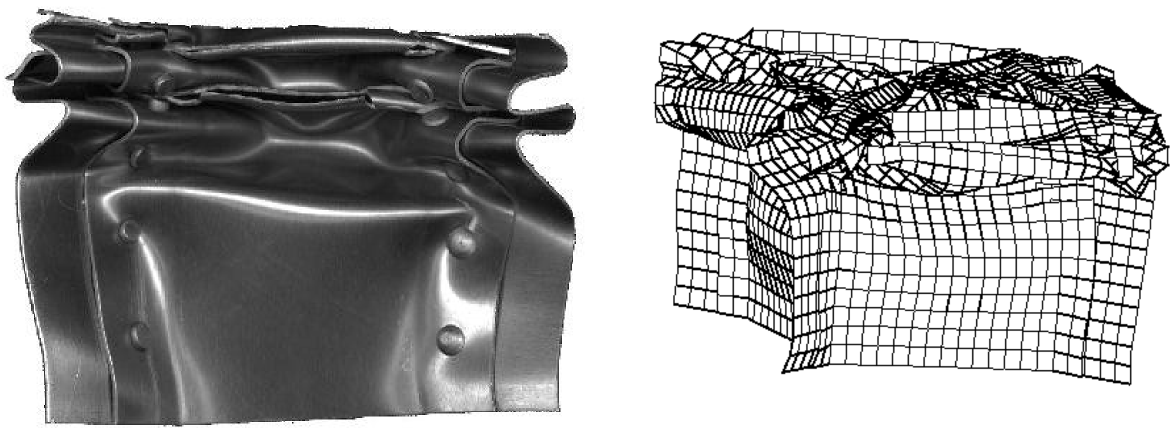

Figure 4: experimental and numerical deformations.

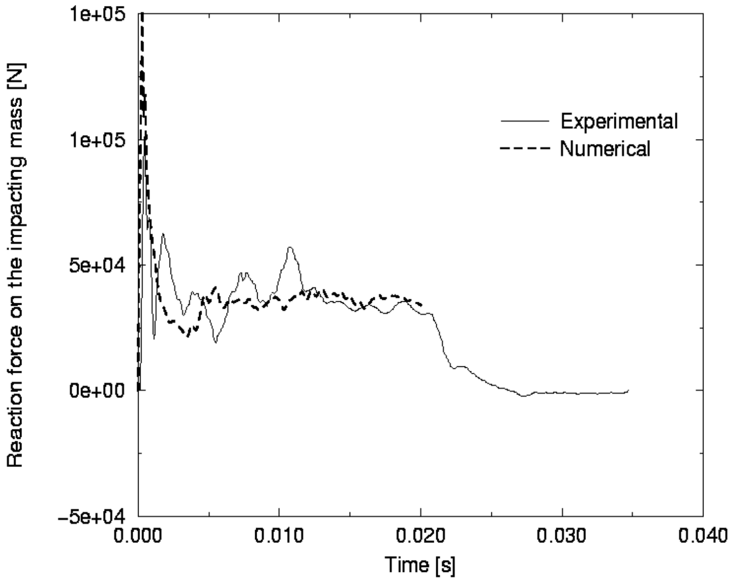

Figure 5: reaction forces on the impacting mass, initial design.

The optimization was carried out varying the thickness of diagonal panels, the thickness of crossing panels, the angle between angular and crossing panels and the distance between the centre of the component and the attachment point of angular panels (Figure 6), leading to a total of four optimization variables. The mesh has to be automatically generated for every possible combination of variable values.

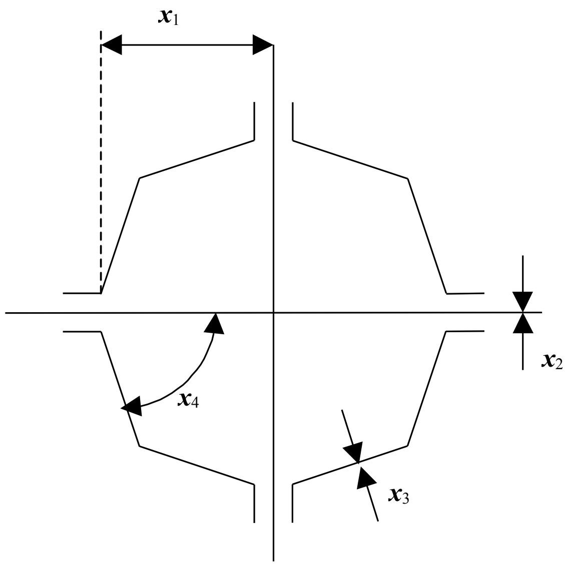

Figure 6: optimization variables.

4. Results

A first optimization run was carried out in order to maximize the absorbed energy per unit of crushed mass (SE). Being the density of the material constant, results are presented as a function of absorbed energy per unit of crushed volume. The range of possible values for each variable is constrained between reasonable limits (Table 1). Furthermore, the ratio between the maximum and the average deceleration of the impacting mass (LUR) is limited to a maximum value of 1.5. The energy absorbed as well as the LUR are computed after filtering the deceleration curve with a SAE J211b compliant 60 *Hz* filter. The optimization procedure was able to improve the value of the objective function within few finite element evaluations (Table 2), and led to a design were the angular panels carry the major part of the load (Figure 7). The optimization was carried out with four slave process (one processor for each variable), reducing the wall clock time from 38 hours, that would have been required with a single processor, to 20 hours. The negative influence of the single solution check to the total wall clock time is evident, as a perfectly scaling algorithm should require about 10 hours for the same problem.. This influence will be mitigated in problems with more variables provided that the number of available processors is allowed to grow accordingly.

Inf. limit	Variable	Sup. limit
30 *mm*	$\leq \boldsymbol{x}_1 \leq$	70 *mm*
0.5 *mm*	$\leq \boldsymbol{x}_2 \leq$	3 *mm*
0.5 *mm*	$\leq \boldsymbol{x}_3 \leq$	3 *mm*
0.174 *rad*	$\leq \boldsymbol{x}_4 \leq$	1.57 *rad*

Table 1: variable bound constraints.

FEM evaluations	Objective function [KN/mm^2]
1	0.064
6	0.136
11	0.154
16	0.154
19	0.164

Variable	Initial value	Finale value
$\boldsymbol{x}_1$	45 *mm*	64 *mm*
$\boldsymbol{x}_2$	1 *mm*	0.5 *mm*
$\boldsymbol{x}_3$	1 *mm*	1.55 *mm*
$\boldsymbol{x}_4$	0.785 *rad*	1.22 *rad*

Table 2: optimization convergence history

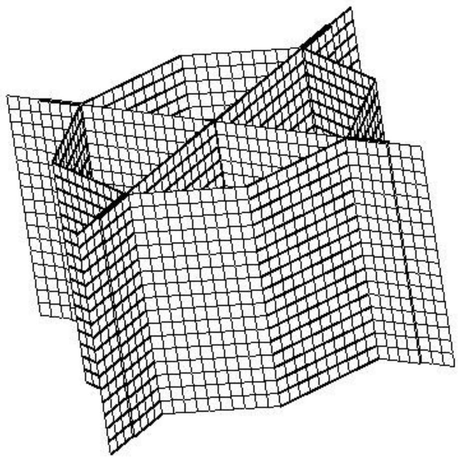

Figure 7: optimal shape for maximum specific energy absobtion.

To test the final result of the optimization, the force on the impacting mass predicted by the finite element was compared with the value obtained with an experimental crash test, showing good agreement between the two time histories (Figure 8). Unfortunately, the initial peak of force in the experimental diagram is cut due to the saturation of the accelerometer channel.

The final solution leads to a noticeable improvement in the SE value at the expense of a very stiff response. We can suppose that the problem to be solved is to absorb a given amount of energy with a given maximum deceleration. As we are not performing the optimization of a complete subfloor, the number of cruciform elements needed to absorb the required energy is not taken in account. The solution of the optimization seems to bring to a design of the subfloor with few stiff energy absorbing elements. Clearly, resulting high reaction forces have to be distributed to the rest of the structure, leading to a difficult and heavier design of components surrounding the energy absorbing element.

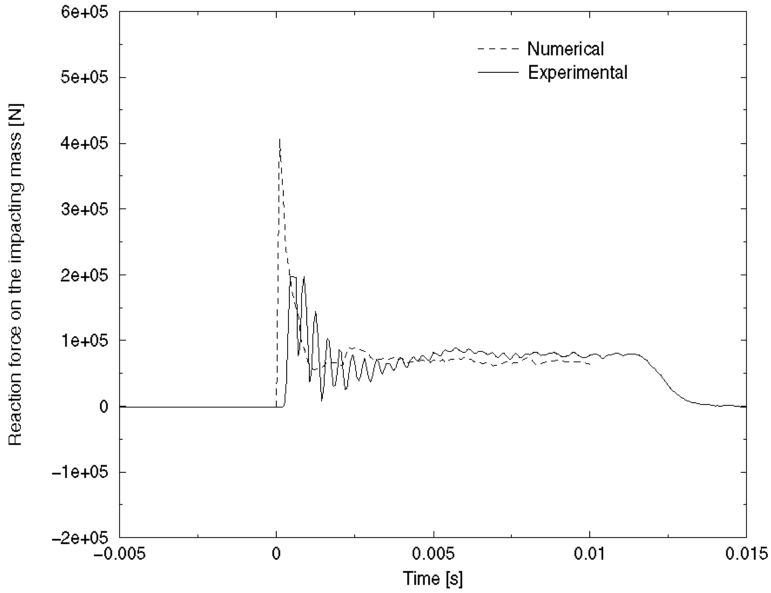

Figure 8: reaction forces on the impacting mass, design optimized for specific energy absorbtion.

Different approaches can be used to solve this difficulty. For example, the engineer can impose an upper bound to the reaction force provided by a single cruciform element, still keeping as objective function the SE index. To investigate the problem, four runs were made with same bound constraints and maximum LUR as before, but limiting the deceleration of the hitting mass respectively to 50, 40, 30 and 20 m/s^2. Optimization runs led to designs quite different from the above presented solution (Figure 9, Figure 10, Table 3). The constraint on the maximum deceleration proved to have a big influence on the capacity of the algorithm to increment the SE index from the initial design value. As the average deceleration of the initial design lies between the 40 and 30 m/s^2 limits, the optimization algorithm behaves in two different ways: for the 50 and 40 m/s^2 cases the optimization algorithm was able to increment the objective function increasing the average deceleration of the impacting mass while, for the 30 and 20 m/s^2 cases, it was only able to modify the design in such a way to keep a constant value of the objective function while decreasing the deceleration value down into the feasible region (Figure 11, Figure 12). The constraint on the LUR parameter didn't become active during optimizations.

Given the total energy to be absorbed, one still has to choose between few "stiff" and many "soft" elements. A different approach could be useful if the total amount of energy to be absorbed by a single element could be known, being the number of cruciform elements fixed; in this case, a proper objective function would be the weight of a structural element being able to absorb the required energy, with limited dimensions and decelerations.

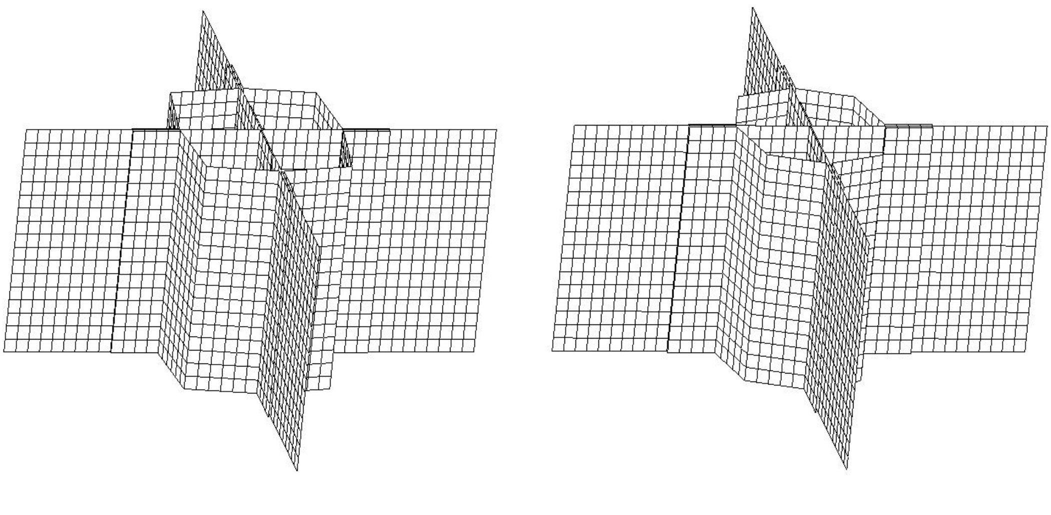

Figure 9: optimal shapes, 50 and 40 m/s^2 maximum deceleration.

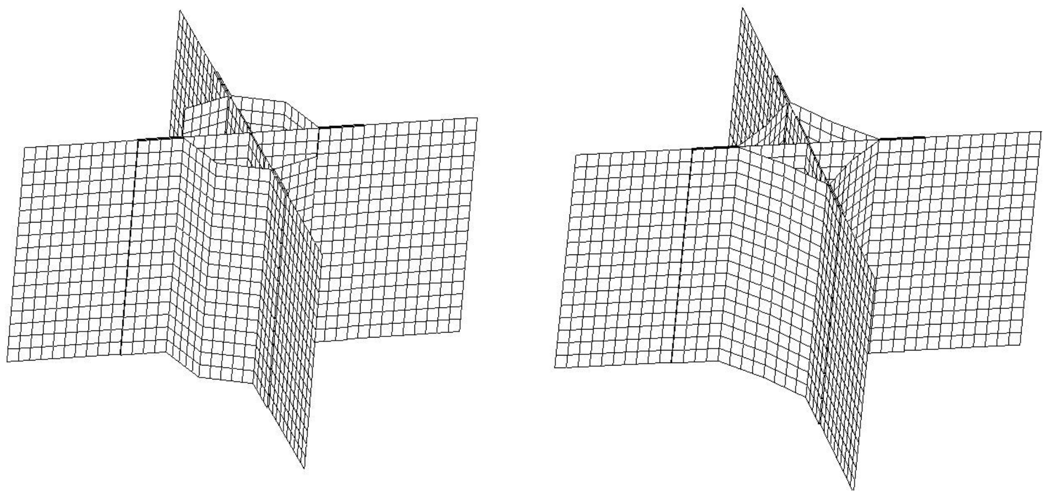

Figure 10: optimal shapes, 30 and 20 m/s^2 maximum deceleration.

	50 m/s^2	**40 m/s^2**	**30 m/s^2**	**20 m/s^2**
x^1	34.67 *mm*	31.5 *mm*	30 *mm*	30 *mm*
x^2	0.5 *mm*	0.54 *mm*	0.5 *mm*	0.5 *mm*
x^3	1.39 *mm*	1.95 *mm*	0.88 *mm*	0.684 *mm*
x^4	1.45 *rad*	1.295 *rad*	1.235 *rad*	0.733 *rad*

Table 3: final values of optimization variables.

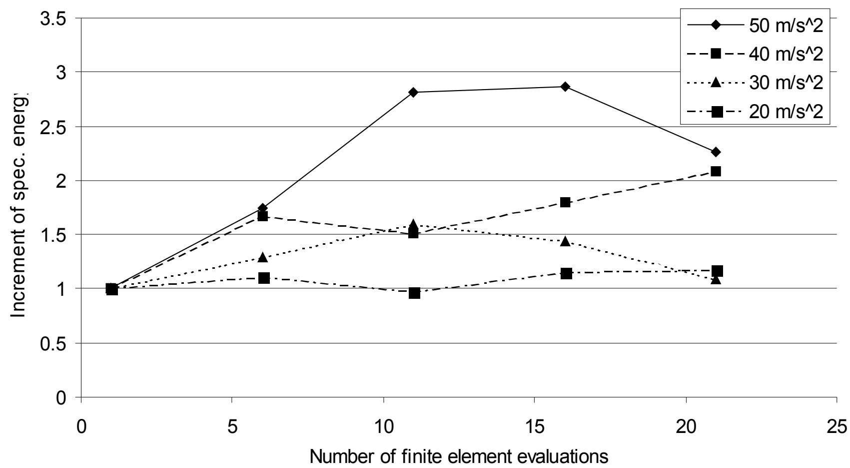

Figure 11: increment of specific absorbed energy with respect to the initial design.

5. Conclusions

This paper shows how, using standard optimization techniques, the optimization of simple energy absorbing elements can be performed with success. This way, finite element codes can be used not only for the analysis of crash responses, but also for the automatic design of such structures. To carry out this type of approach to the design of crashworthy structures it is necessary to

- have a way to automatically generate the mesh;
- have a sufficient confidence in the results of the simulation, even for configurations which are quite different from the initial one;

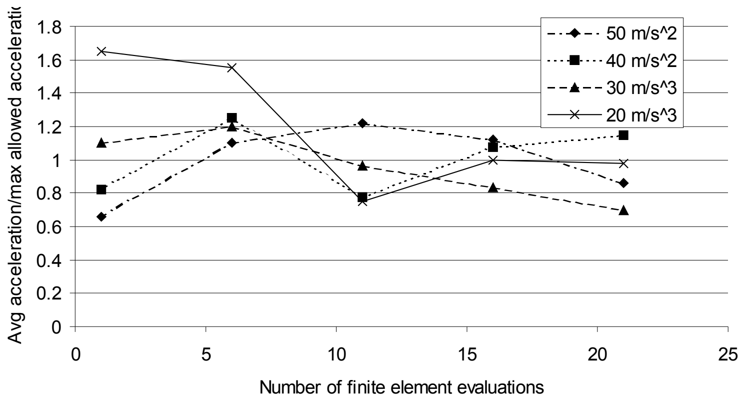

Figure 12: relative average deceleration values.

Provided that these conditions are satisfied, the optimization can be carried out and give useful information for the design of the structure. However, it is worth noting that the second condition is nothing but a trivial requirement, and can be in contrast with the need to simplify the mesh in order to reduce the cpu time required for a single evaluation. To achieve a good balance between these two requirements – confidence in simulation results and reduced simulation time – a good understanding of the dynamic of the system can be very useful. Also, the analyst has to choose a proper set of objective function and constraints, thus precisely identifying every requirement of the structure.

References

1. J. F. M. Barthelemy (NASA Langley Research Center – USA) – R. T. Haftka (Virginia Polytechnic Institute and State University – USA), Approximation Concepts for Optimum Structural Design – a Review; Structural Optimization, vol. 5, pp. 129-144, 1993, Springer-Verlag.
2. Pierre Culière – Jan Clinckemaillie – Xiaomin Ni – Adré Petitjean (ESI Group – FRANCE), A review of PAM-CRASH™ Stability Issues; Proceedings of Pam `97 Conference, Oct. 1997, ESI Group, Prague.
3. Pierre Ducrocq – Eric Markiewicz – Pascal Drazetic (Univeristy of Valeciennes – FRANCE) – Pierre Guyon (ESI Group – FRANCE), An Inverse Approach to Determine the Constitutive Model Parameters Using Drop Tests and Numerical Tools; Proceedings of Pam `97 Conference, Oct. 1997, ESI Group, Prague.
4. L. F. P. Etman (Eindhoven Univeristy of Technology – THE NETHERLANDS) – J. M. T. A. Adriaens – M. T. P. van Slagmaat (TNO Road Vehicles Research Institute – THE NETHERLANDS) – A. J. G. Schoofs (Eindhoven Univeristy of Technology – THE NETHERLANDS), Crash Worthiness Design Optimization Using Multipoint Sequential Linear Programming; Structural Optimization, vol. 12, pp. 222-228, 1996, Springer-Verlag.
5. L. F. P. Etman – E. J. R. W. Thijssen – A. J. G. Schoofs - D. H. van Campen, Optimization of Multibody Systems using Sequential Linear Programming; Proceedings of ASME Advances in Design Automation, DE-69-2, pp. 525-530, Minneapolis, B. J. Gilmore – D. A. Hoeltzel – D. Dutta – H. A. Eschenauer (eds.).
6. Raphael T. Haftka – Zafer Gürdal (Virginia Polytechnic Institute and State University – USA), Elements of Structural Optimization, 3rd edition, 1992, Kluwer Academic Publisher.
7. Tatsuo Kuboshima (Nissan Shatai Co. – JAPAN) – Seiji Ishizuka (Nissan Shatai Eng. – JAPAN), The Application of PAM-OPT for the Design of Car-Body; Proceedings of Pam `97 Conference, Oct. 1997, ESI Group, Prague.
8. J. Marczyk (CASA Space Division - SPAIN) – M. Holzner – H. Mader (BMW – GERMANY) – J. Clinkemaillie – S. Meliciani (ESI - FRANCE) – M. Noak – J.Sejbold (Univeristy of Stuttgart – GERMANY) – C.Tanasescu (SGI GmbH - GERMANY), Stochastic Automotive Crash Simulation: a New Frontier in Virtual Prototyping; Proceedings of Pam `97 Conference, Oct. 1997, ESI Group, Prague.
9. R. R. Mayer (General Motors Co. – U.S.A) – N. Kikuchi – R. A. Scott (Univeristy of Michigan – USA), Application of Topological Optimization Techniques to Structural Crashworthiness; Int. J. Numerical Methods in Egineering, vol 39, pp. 1383-1403, 1996, J. Wiley & Sons.

10. Naoki Niii – Ken Nakagawa (Isuzu Motors Limited – JAPAN), Collision of Heavy-duty Vehicles in Isuzu; Proceedings of Pam `97 Conference, Oct. 1997, ESI Group, Prague.
11. Shigeru Obajashi (Tohoku University – JAPAN), Pareto Genetic Algorithm for Aerodynamic Design Using the Navier-Stokes Equations; Genetic Algorithms and Evolution Strategies in Engineering and Computer Science, pp.245-266, 1997, John Wiley & Sons, D. Quagliarella, J. Périaux, C. Poloni, G. Winter Eds.
12. M. J. D. Powell (University of Cambridge – ENGLAND), Direct Search Algorithm for Optimization Calculations; Acta Numerica, vol 7, Cambridge University Press, 1998.
13. Helmut Schwab, Documentation for lp-solve, available at ftp.es.ele.tue.nl/pub/lp_solve

Acknowledgements

This research was sponsored by Politecnico di Milano and CIRA S.p.A.
Cpu time on an HP Exemplar 2000/XA32 was offered by CILEA.