11th World Congress on Computational Mechanics (WCCM XI)
5th European Conference on Computational Mechanics (ECCM V)
6th European Conference on Computational Fluid Dynamics (ECFD VI)
E. Oñate, J. Oliver and A. Huerta (Eds)

BRANCHING STRATEGIES FOR THE APPLICATION OF HEURISTICS TO THE TOPOLOGY OPTIMIZATION OF CRASH LOADED STRUCTURES

CHRISTOPHER ORTMANN AND AXEL SCHUMACHER

University of Wuppertal, Faculty D – Mechanical Engineering, Chair of Optimization of Mechanical Structures, Gaußstraße 20, 42119 Wuppertal, Germany
e-mail: ortmann@uni-wuppertal.de, schumacher@uni-wuppertal.de

Key Words: *Topology optimization, Crashworthiness, Heuristics, Expert knowledge, Graph theory.*

Abstract. The topology optimization of crashworthiness structures is an emerging field of research. These structures are subjected to nonlinearities with different kinds of sources: geometry (e.g. large displacements and rotations), boundary condition (e.g. contact) and material (e.g. plasticity, failure and strain rate dependency).

One approach to overcome these problems is the usage of heuristics derived from expert knowledge. The method of *Graph and Heuristic Based Topology Optimization (GHT)* uses a combination of heuristics and mathematical optimization algorithms for the combined topology, shape and sizing optimization of crashworthiness profile structures. The topology changes are performed by heuristics and the shape and sizing optimizations are carried out with mathematical optimization algorithms.

The major disadvantages of the *GHT* are the high risk of getting stuck in local optima and the high computational effort. This contribution introduces a branching strategy to the optimization procedure to improve these weak points. Not only a single design is tracked and modified successively during the optimization but a number of competing designs which are in competition with each other.

1 INTRODUCTION

The topology optimization of crashworthiness structures is an emerging field of research. Crashworthiness structures are subjected to nonlinearities with different kinds of sources: geometry (e.g. large displacements and rotations), boundary condition (e.g. contact) and material (e.g. plasticity, failure and strain rate dependency). Usually crash simulations are performed with finite element method codes which can handle the nonlinearities and use explicit time integration. The existence of bifurcation points, the usage of special structural responses like energy absorption and injury criteria, the costly determination of sensitivities (due to the explicit time integration) and the huge number of local optima make the optimization of crashworthiness structures even more complex.

Several extensions to already existing optimization methods or complete new optimization methods have been developed to improve the possibilities of topology optimization in nonlinear dynamic structural problems. Among others are: Mayer et al. [1], Soto [2], Pedersen

[3], Patel et al. [4] and Park [5].

Another approach to overcome these problems is the method of *Graph and Heuristic Based Topology Optimization (GHT)*, which uses a combination of heuristics (derived from expert knowledge) and mathematical optimization algorithms for the combined topology, shape and sizing optimization of crashworthiness profile structures. The method is only applicable to extrusion structures with a defined cross section like aluminum extrusion profiles.

In previous work [6], [7], [8] each iteration of this method consists of a topology change performed by heuristics and a subsequent shape and sizing optimization for the new topology class carried out with mathematical optimization algorithms. There exist different heuristics for the topology modification of the structure, which are in competition to each other because only one topology modification is allowed in each iteration. The major disadvantages of this approach are the high risk of getting stuck in local optima and the high computational effort because of the computationally expensive shape and sizing optimization in each iteration.

In this contribution the concept of branching is introduced to the *GHT*. Several different design possibilities which are the product of multiple topology modifications performed by different heuristics are tracked simultaneously. These designs are in competition to each other. The fitness of these designs is evaluated by function calls.

2 GRAPH BASED GEOMETRY DESCRIPTION

In the *GHT* the geometry of the structure to be optimized is described by a mathematical graph in order to have a flexible geometry description (Figure 1). This graph is simple, undirected, connected and planar. A special graph syntax for the description of mechanical structures has been developed based on the work of Olschinka and Schumacher [9]. A detailed description of the graph syntax can be found in [6].

Each wall of the structure's cross section is described by a combination of the *BEAM1*-, *BEAM2*- and *BEAMG-Vertex*. While the first two are used to define the orientation of the wall, the *BEAMG-Vertex* describes the thickness and curvature of the wall. The *BEAM1*- and *BEAM2-Vertex* are connected with *LINK-Vertices* These contain Cartesian coordinates and are used to define the position of the wall within the structure's cross section as well as the kind of connection between the walls (e.g. the definition of chamfers). Finally the *PARAM-Vertex* describes the extrusion length and the density of the structure's material which is needed for a graph based calculation of the mass of the structure.

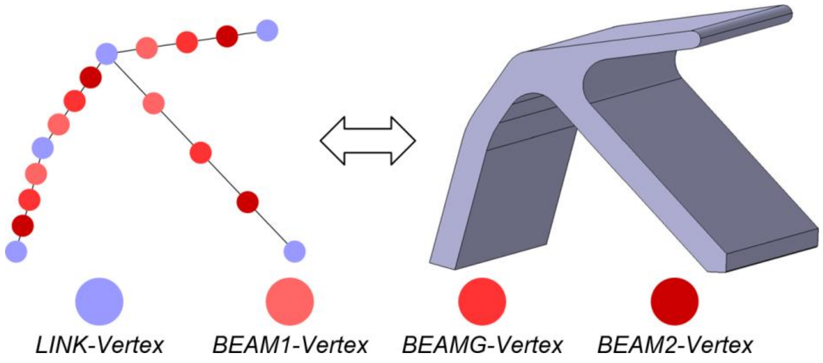
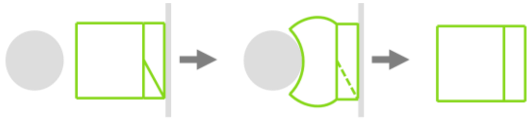


Figure 1: Geometry description with a mathematical graph

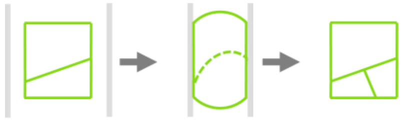
For the automatic creation of geometry and finite element models based on a graph the program *GRAMB* (*GRAph based Mechanics Builder*) has been developed. This program translates the information of the graph into program execution commands for one of the following CAE (computer aided engineering) systems: *Altair HyperMesh*®, *Dassault Systèmes CATIA*® or *SFE CONCEPT*®, which then create the geometry or finite element model. Beside the comfortable geometry description which makes even complex geometry modifications like topology changes possible, the main advantage of the graph based geometry description is the check of manufacturing constraints like: minimum and maximum wall thicknesses, minimum wall connection angels, minimum wall distances and the maximum chamber size ratio between the largest and the smallest chamber of the cross section.

3 HEURISTICS FOR THE CRASHWORTHINESS TOPOLOGY OPTIMIZATION

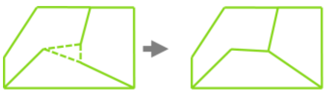
The heuristics within the *GHT* are derived from expert knowledge of automotive crash engineers and are extensively described in [6]. The basic principles of the heuristics are shown in Figure 2. The heuristics use result data of finite element simulations like finite element node velocities or finite element inner energies to make decisions about how to modify the structure.



Heuristic *Delete Unnecessary Walls*

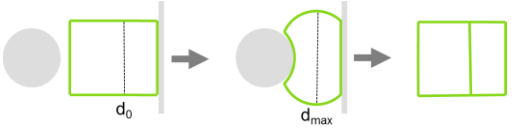


Heuristic *Support Fast Deforming Walls*

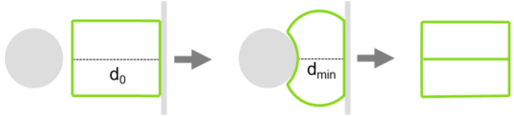


Heuristic *Remove Small Chambers*

Heuristic *Balance Energy Density*



Heuristic *Use Deformation Space Tension*



Heuristic *Use Deformation Space Compression*

Heuristic *Smooth Structure*

Heuristic *Scale Wall Thicknesses*

Figure 2: Basic principles of the heuristics

The heuristic *Delete Unnecessary Walls* (*DUW*) removes walls from the structure's cross section which have a low inner energy density compared to the other walls in all load cases. Walls which have a tendency towards buckling are supported with a new perpendicular wall by the heuristic *Support Fast Deforming Walls* (*SFDW*). The heuristic *Remove Small Chambers* (*RSC*) simplifies the structure by reducing small chambers of the structure's cross section to single walls. Walls which have a high inner energy density are connected with walls which have a low inner energy density by the heuristic *Balance Energy Density* (*BED*).

The heuristics *Use Deformation Space Tension/Compression* (*UDST*/*UDSC*) create new walls between points of the structure's cross section which have a high positive/negative relative displacement to each other. Kinks in the shape of the structure's cross section are smoothed by the heuristic *Smooth Structure* (*SS*). The heuristic *Scale Wall Thicknesses* (*SWT*) scales all wall thicknesses of the structure to achieve a predefined mass with the objective to keep the mass of the structure constant despite the geometrical modifications performed by the other heuristics.

4 METHOD OF GRAPH AND HEURISTIC BASED TOPOLOGY OPTIMIZATION WITH A BRANCHING STRATEGY

In this contribution a branching strategy is introduced to the *Graph and Heuristic Based Topology Optimization*. Not only a single design is tracked and modified successively during the optimization but a number of competing designs which are in competition with each other. In each iteration the best designs of the last iteration are tracked further while the other are discarded. The first objective of this strategy is the improvement of the ability of the *GHT* to overcome local optima due to the better exploration of the design space. The second one is the reduction of the number of computationally expensive shape and sizing optimizations by relying more on the modifications performed by the heuristics.

Because of this extension the structure and sequence of the *GHT* differs from previous work and is described extensively in this chapter.

4.1 Basic principles

The method of *GHT* includes four basic principles:

1. The usage of mathematical graphs**.** The structure to be optimized is described by the graph syntax presented in Chapter 2. All modifications of the structure are not performed on the structure itself, but on the mathematical graph which describes the structure. Graph based algorithms are used to check manufacturing constraints and to modify the geometry of the structure. Even complex geometric modifications like a topology change can be performed in this way. At any time a finite element model of the structure can be generated based on the graph to perform simulations.

2. The usage of heuristics**.** Heuristics (rules) derived from expert knowledge are used for the geometric modification of the structure (see previous chapter). The main task of the heuristics is the topology modification of the structure based on the information about the mechanical behavior of the structure like displacement vectors of finite element nodes coming from crash simulations.

The heuristics can be divided into two groups: competing heuristics carry out topology changes of the structure such as the creation of new walls in the cross section of the structure to reinforce an instable area, non-competing heuristics only modify the shape and sizing parameters of the structure such as the scaling of all wall thicknesses of the structure to achieve a predefined mass.

3. The division of the optimization problem into an outer and an inner loop. The real optimization problem of the combined topology, shape and sizing optimization of a structure is divided into two optimization loops convoluted in each other. In the outer optimization loop the structure to be optimized is modified exclusively by the heuristics.

In the inner optimization loop a conventional shape and sizing optimization is performed with an initial design coming from the outer optimization loop. For this purpose any optimization algorithm can be used, e.g. genetic algorithms. The topology class of the structure is not changed during the inner optimization loop.

4. The parallel tracking of competing designs. During the optimization not only a single design is tracked and modified successively but a total of E designs, where E is an integer with E ≥ 1 and defined before the beginning of the optimization. The competing designs are the product of multiple topology modifications performed by different heuristics.

In order to decide in the outer optimization loop during each iteration, which designs will be tracked further and which not, a possibility of evaluation of the designs is needed. This evaluation is done via the objective function values of the different designs while only designs are considered which fulfill all constraints.

The determination of the objective function value of a design cannot be done with a single function call because the design must be given at least the opportunity to get stiffer or more compliant in order to fulfill constraints like mass, stiffness or acceleration constraints. Therefore, not only a single function call is performed, but a complete inner optimization loop, where a scaling factor for all wall thicknesses is used as the design variable. The ratio of the thicknesses of the single walls to each other will remain constant but the whole design can get stiffer or more compliant by alteration of the design variable. Since only a single design variable is used in this sizing optimization, a small number of function calls for this inner optimization loop is sufficient.

4.2 Optimization sequence

The sequence of the *GHT* which is divided into an outer and an inner optimization loop is illustrated in Figure 3 and Figure 4.

Outer optimization loop. At the beginning of each new iteration the E best designs are determined from the amount of designs of the last iteration. These designs are evaluated on the basis of the objective function value. Only designs are considered which fulfill all constraints. E = 1 applies in the first iteration because only the initial design is available.

For each of the E current competing designs H new designs are generated. Each of these designs is created by the application of exactly one of the H (here H = 6) competing heuristics and differs from the original design by the topology change performed with the respective competing heuristic.

The non-competing heuristics then modify the shape and the wall thicknesses of each of these designs, whereby all non-competing heuristics are always carried out successively.

After the application of the heuristics for each design an inner optimization loop is started, in which a sizing optimization with one design variable and a small number of function calls takes place to evaluate the design. The best design of this inner optimization loop is added to the amount of the designs of the current iteration. Overall, the amount of the designs of the current iteration contains now E times H designs and for all of them an objective function value is available for the evaluation.

If the objective function value of the best design of the current iteration is better than that of the best design of the last iteration, the heuristics were able to improve the structure and a new iteration is started.

If this is not the case, the heuristics alone could obtain no further improvement of the structure and the stop criterions are checked. If one of the criterions is fulfilled the optimization will be stopped. Then the final design is the best design of the last iteration which at the same time is so far the best design of the whole optimization. The stop criterions include the reaching of the maximum number of iterations and the failure of the heuristics to improve the structure in two successive iterations.

If none of the criteria is fulfilled an inner optimization loop will be started for the best design of the last iteration. Here a shape and sizing optimization with a high number of function calls is performed. The design resulting from the inner optimization loop is added to the amount of designs of the current iteration. Subsequently a new iteration starts.

Inner optimization loop. In the inner optimization loop an optimization of a design takes place which is coming from the outer optimization loop. The outer optimization loop transfers as well the information whether only a sizing optimization with a small number of function calls or a full shape and sizing optimization of the design should take place.

For the initial design of the inner optimization loop an optimization model is generated which contains all necessary information of the optimization problem. The design variables and the corresponding borders are determined automatically and this information is combined with information about the objective function, the constraints and the load cases to a complete optimization model. This process is described more in detail in [6].

In the following procedure the design variables are changed by the optimization algorithm for the shape and sizing optimization (e.g. genetic algorithms) as long as the stop criterions of the used optimization algorithm are not fulfilled. This could occur for example, when the maximum number of function calls is reached or when the improvement of the objective function from one iteration to the next falls below a certain threshold value (convergence).

The mechanical behavior of the structure and in particular the values for the objective function and the constraints are determined by function calls (here finite element simulations for all crash load cases). For this a mathematical graph is generated which describes the current design with varied design variables. Then, based on this graph, finite element models are generated before finally the finite element simulations are started.

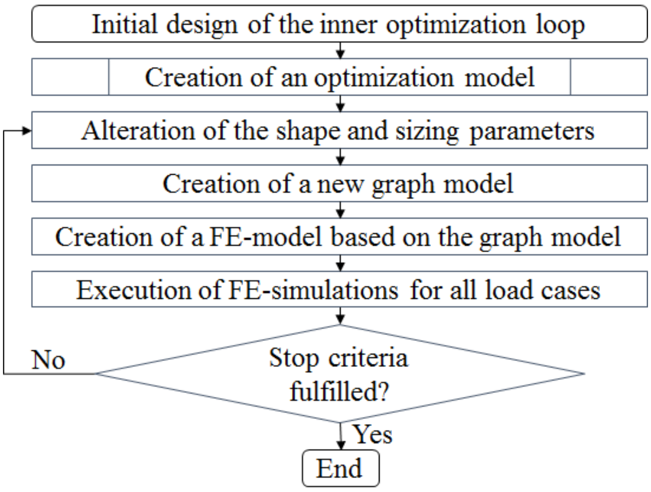


Figure 3: Inner optimization loop

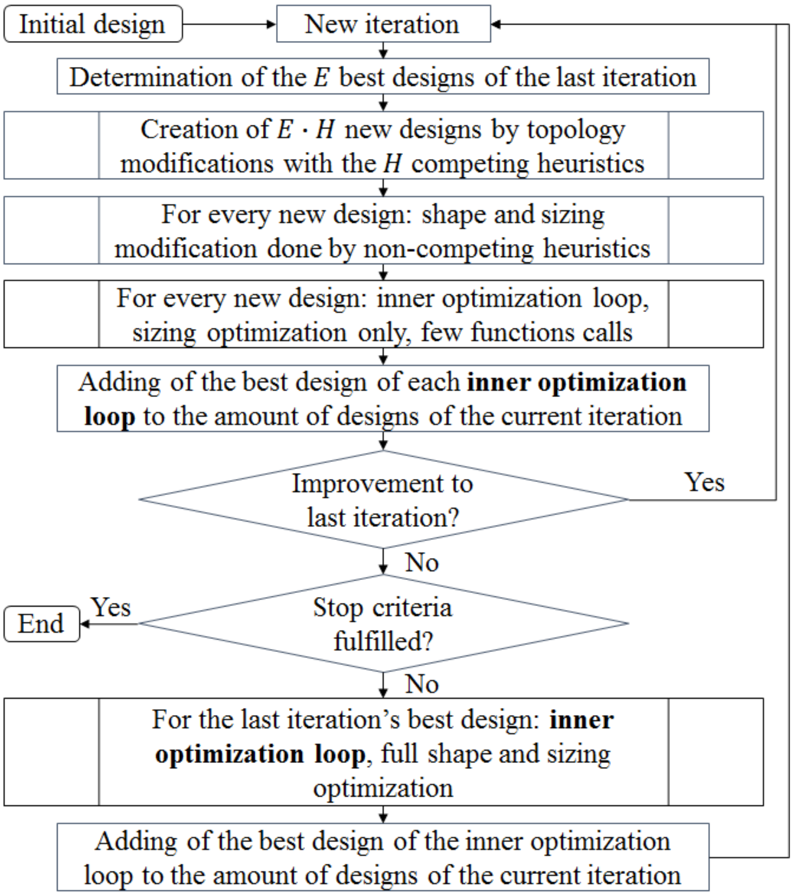


Figure 4: Outer optimization loop

For the execution of the *GHT* the program *TOC* (*Topology Optimizer for Crashworthiness structures*) has been developed. It has interfaces to *LS-DYNA*® to read simulation result data and to *LS-OPT*®, which is controlled by *TOC* during the complete procedure and is used for the shape and sizing optimizations within the inner optimization loop.

5 APPLICATION EXAMPLE

The application example is the optimization of a segment of a vehicle rocker with a length of 600 mm taking into account three load cases. This application example has first been presented in [6].

The first load case is based on the EURO-NCAP pole impact (Figure 5). In this load case the rocker is connected with a short segment of a seat crossmember. At the end of the seat crossmember segment a rigid wall with a mass of 85 kg is located. The displacement in the z-direction is constrained at the ends of the rocker segment. At the end of the seat crossmember segment all degrees of freedom except the y-direction are constrained. The rocker, the seat crossmember and the rigid wall have an initial velocity of 29 km/h in the negative y-direction and move against a rigid pole. This load case is calculated with *LS-DYNA*® explicit.

Load case two is linear static bending and load case three is linear static torsion. In both load cases one end of the rocker is clamped while the other end is loaded via a spider of rigid body elements. For the calculation of these load cases *LS-DYNA*® implicit is used.

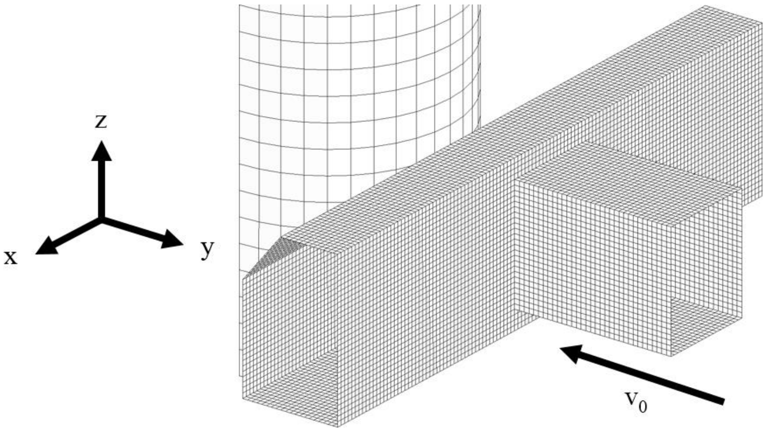


Figure 5: First load case - pole impact

The **objective** of this optimization is the minimization of:

- the maximum rigid wall force of the rigid wall which is located at the end of the seat crossmember segment in load case one.

The **functional constraints** of this optimization are:

- mass of the structure ≤ mass of the initial design (2.801 kg),
- the intrusion of the rocker in load case one ≤ 70 mm (initial design: 69.3 mm),
- stiffness in load case 2 and 3 ≥ 50 % of the initial design's stiffness.

The **manufacturing constraints** of this optimization are:

- 1.6 mm ≤ wall thickness ≤ 3.5 mm,
- wall distance ≥ 10 mm,
- wall connection angel ≥ 15°,
- chamber size ratio between the largest and the smallest chamber of the cross section ≤ 20.

Within the optimization E = 5, so five competing designs are tracked simultaneously during the optimization. In the inner optimization loop for the evaluation of the heuristics a metamodel based optimization approach with domain reduction is used. In the inner optimization loop for the full shape and sizing optimization genetic algorithms are used.

The optimization history of the objective function and the constraints over the iterations is summarized in Table 1. The final design is found in the third iteration with an objective function value of 42.27 kN, which is an improvement of 24.3 % compared to the initial design. The best competing design of iteration 3a has a higher objective function value than the best design of iteration 2. Therefore for the best design of iteration 2 a full shape and sizing optimization with a high number of function calls (3,000) is performed in iteration 3b. In iteration 4 the heuristics fail again to improve the structure further. This fact fulfills the stop criterion and the optimization ends.

Table 1: Optimization history

	Max. RW-Force [kN]	Bending stiffness [%]	Torsion stiffness [%]	Mass [kg]	Intrusion [mm]	Thick-ness [mm]	Function calls so far
Initial design	55.82	100	100	2.801	69.03	3.50	0
Iteration 1	52.04	74.31	70.66	2.269	69.85	2.36	52
Iteration 2	44.90	61.99	60.44	2.176	69.44	2.10	273
Iteration 3a	47.01	61.38	55.65	2.032	69.84	1.83	546
Iteration 3b (final design)	42.27	66.28	64.52	2.353	69.93	2.24	3546

An overview of the graphs of the competing designs the first three iterations can be found in Figure 6. All in all there have been 43 competing designs in the first three iterations. The best design of each iteration is highlighted by a rectangular frame.

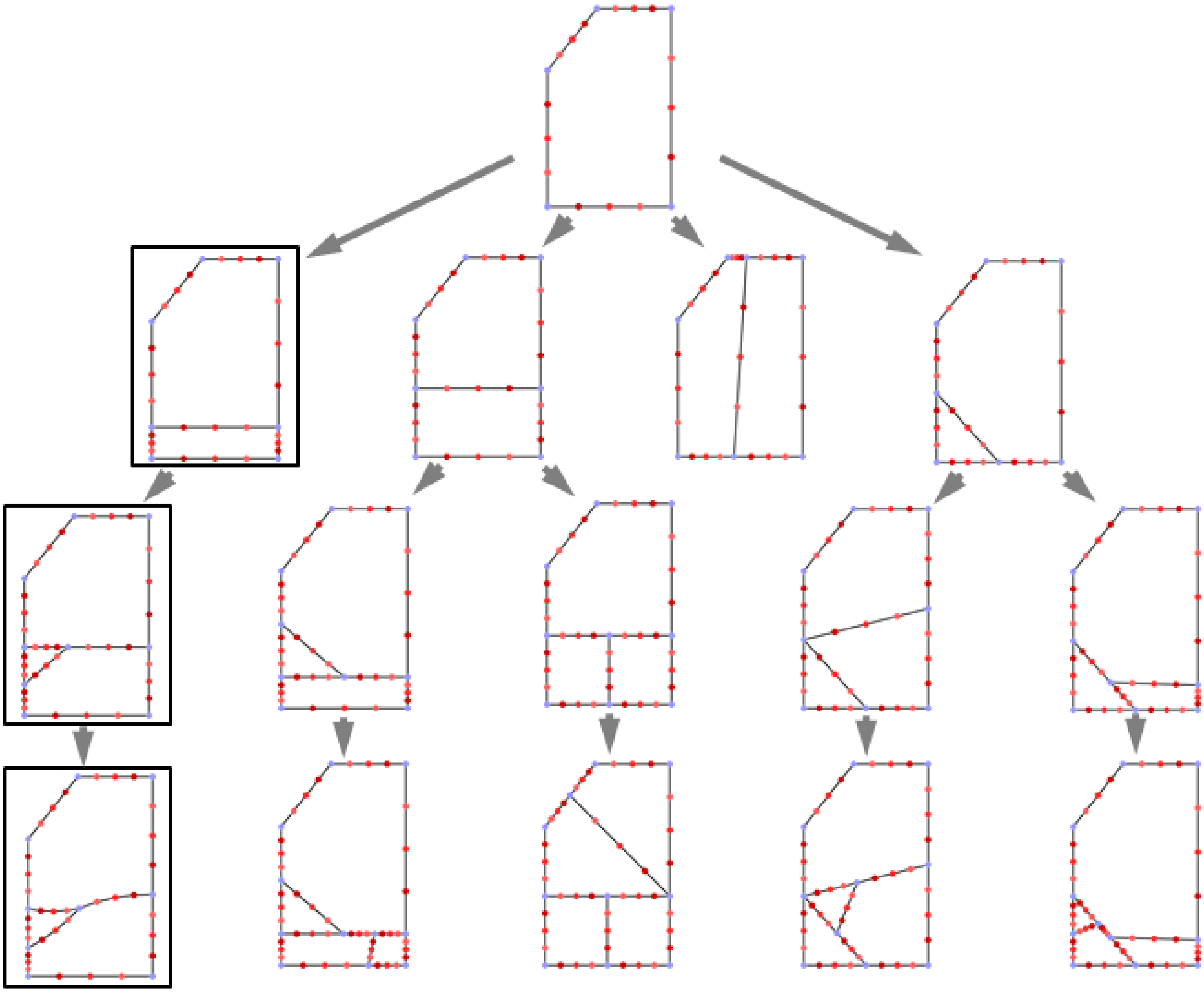


Figure 6: Overview of the competing designs of iteration 1 – 3

The deformation behavior of the initial design and the final design of the optimization in load case 1 can be found in Figure 7. The final design consists of three slightly curved walls and shows an efficient energy absorbing mechanism through controlled buckling. The curvature of the walls reduce the force peaks which are necessary to initiate the buckling.

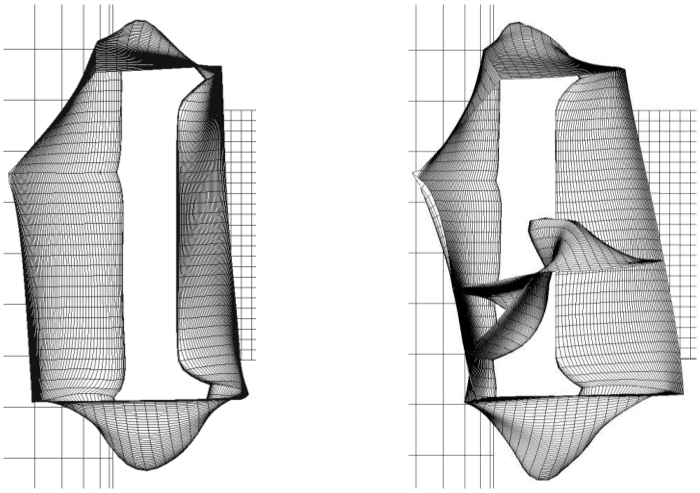


Figure 7: Deformation behavior of the initial design and the final design

The force-time curves of the initial design and the final design of the rigid wall which is located at the end of the segment of the seat crossmember can be found in Figure 8.

The quick reaching of a certain force level and the maintaining of this force level until the elastic rebound occurs, is very desirable for a crashworthiness structure. In this way force or acceleration peaks can be reduced while the structure still absorbs the necessary amount of kinetic energy. The final structure shows such behavior. It reaches a force level of about 42 kN quickly and maintains this force level constantly.

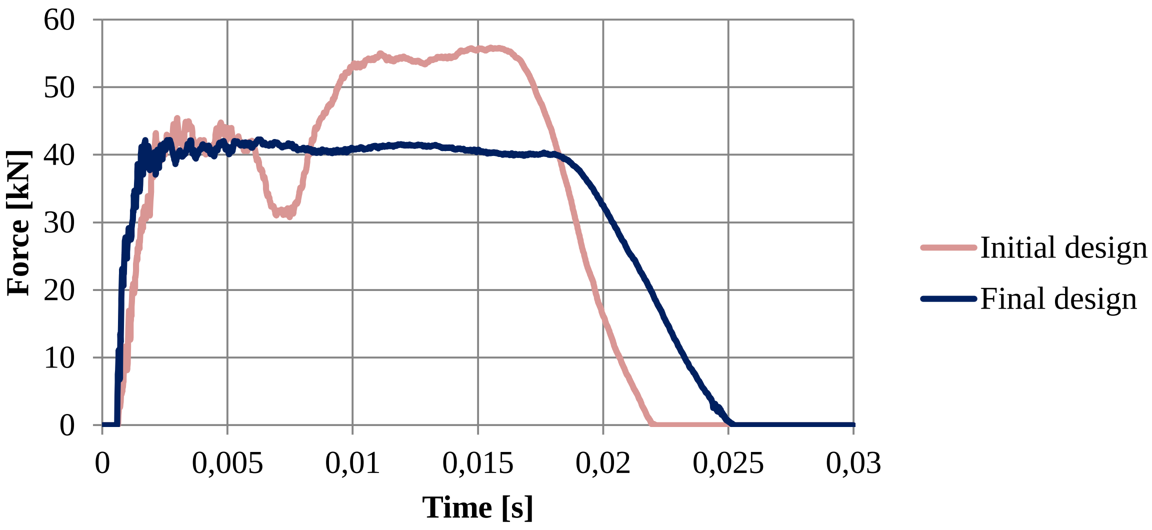


Figure 8: Force-time curves of the initial design and the final design

6 COMPARISON AND CONCLUSION

The optimization result of the application example presented in [6], which has been generated without the usage of competing designs and the optimization result, which has been generated with the usage of competing designs, are compared in Figure 9. Both optimization procedures used about 3,500 function calls. The result which was generated by using competing designs has a lower maximum rigid wall force and a higher stiffness in the linear static load cases. This comparison shows that the usage of the branching strategy has improved the ability of the *GHT* to overcome local optima. A better design has been found within the design space. The rigid wall force of 42.27 kN is close to the theoretical optimum of 40.72 kN which can be calculated by using the initial kinetic energy of the structure and the maximum allowed displacement of 70 mm, assuming a perfectly inelastic collision [6].

43.42 kN, 53.76 % torsion stiffness

42.27 kN, 64.52 % torsion stiffness

Figure 9: Graphs of the final designs generated without (left) / with (right) the usage of competing designs

A high number of function calls has been used in this application example to investigate how close the *GHT* can come to the theoretical optimum. The optimization problem of a force minimization with displacement constraints is a sensible one, which requires a high number of function calls. Only designs which use the completely available deformation space and therefore are close to the boundary of the displacement constraint, can archive low force values.

The execution of the *GHT* without the computationally expensive full shape and sizing optimizations could be a way to reduce the computational effort. In the application example presented in this contribution a good design with a maximum rigid wall force of 44.9 kN is found in the second iteration with only 273 executed function calls so far.

REFERENCES

[1] Mayer R.R., Kikuchi N. and Scott R.A. Application of topological optimization techniques to structural crashworthiness. *Int J Numer Meth Eng* (1996) **39**:1383-1403.

[2] Soto C.A. Structural topology optimization for crashworthiness. *Int J Crash* (2004) **9(3)**:277-283.

[3] Pedersen C.B.W. Crashworthiness Design of transient frame structures using topology optimization. *Comput Methods Appl Mech Eng* (2004) **193**:653-678.

[4] Patel, N.M., Kang, B.S., Renaud, J.E. and Tovar, A. Crashworthiness Design Using Topology Optimization. *J Mech Des* (2009) **131**:061013.1-061013.12.

[5] Park G.J. Technical overview of the equivalent static loads method for non-linear static response structural optimization. *Struct Multidisc Optim* (2011) 43:319-337.
[6] Ortmann C., Schumacher A. Graph and heuristic based topology optimization of crash loaded structures. *Struct Multidisc Optim* (2013) **47(6)**:839-854
[7] Ortmann C., Schumacher A. Hierarchical topology and shape optimization of crash-loaded profile structures. *Proceeding of the 10th World Congress on Structural and Multidisciplinary Optimization*, ISSMO, Orlando, Florida, USA, 2013
[8] Schumacher A., Ortmann C. Rule generation for optimal topology changes of crash-loaded structures, Proceeding of the 10th World Congress on Structural and Multidisciplinary Optimization, ISSMO, Orlando, Florida, USA, 2013
[9] Olschinka C., Schumacher A. Graph based topology optimization of crashworthiness structures. *PAMM Proc Applied Math Mech* (2008) **8(1)**: 10029-10032