



Finding three-dimensional layouts for crashworthiness load cases using the graph and heuristic based topology optimization

F. Beyer¹ · D. Schneider¹ · A. Schumacher¹

Received: 26 February 2020 / Revised: 14 October 2020 / Accepted: 23 October 2020 / Published online: 14 November 2020
© The Author(s) 2020

Abstract

In this paper we present a new procedure using the graph and heuristic based topology optimization in order to find layouts for three-dimensional frame structures under crash loads. A three-dimensional graph describes the geometry and is used to derive a finite element shell model. The model of the frame structure consists of different profiles with continuous cross-sections. The ends of the profiles are currently rigidly connected. Each cross-section is defined by an individual two-dimensional graph. After performing a simulation its results are used by competing heuristics to propose new topologies for the frame structure. Most of these heuristics are derived from expert knowledge. Over several iterations, the goal is to improve the structures mechanical behavior. Typical objectives are the minimization of the structural intrusion in a crash scenario or the minimization of the maximal contact force between structural components. The presented method includes topology optimization by heuristics and shape optimization respectively sizing by mathematical optimization algorithms. The new flexible syntax for three- and two-dimensional graphs, the optimization process and the currently used heuristics are described. The performance is demonstrated for two examples, each optimized twice with opposing objectives.

Keywords Crashworthiness · Heuristic algorithms · Topology optimization · Frame structures

1 Introduction

Crashworthiness requirements play an important role in the process of vehicle development. Residual spaces have to be preserved in a crash accident, while limiting acceleration peaks to avoid injuries. This results in the need for stiff structures on the one hand and energy absorbing structures on the other hand. Most of the kinetic energy is absorbed by deformation of metal structures. In addition, there are further requirements regarding structural fatigue or noise, vibration and harshness (NVH). Furthermore, the mass of the vehicle should be decreased to increase the range and efficiency of the car while leaving more light weight capacity. This paper focuses on the crashworthiness design process in the early concept phase. Numerical simulations based on the explicit finite element methods

help to evaluate the above goals in the developing process. For layout finding, density-based topology optimization methods are often integrated in the design processes, since they can handle a vast amount of design variables through evaluating gradient information coming directly from the simulation (Bendsøe and Kikuchi 1988). For various reasons, the approach is not applicable when designing crashworthiness structures. The dynamic simulations with time dependencies, contact and nonlinear deformations and material properties prevent the usage of gradient based optimization algorithms since the sensitivities regarding an objective function cannot be analytically derived.

Current research activities focus on alternative optimization methods. Weider et al. (2018, 2019) calculate the sensitivity of, e.g., a displacement field not directly, but by an adjoint sensitivity analysis. The sensitivities are determined using a geometric substitute model consisting of a variable shaped hole. The aim is to solve an adjoint equilibrium equation and apply material dependencies and partial integrations in the time domain in order to receive inertia effects.

Ivarsson et al. (2018) use the adjoint method for calculating nonlinear sensitivities by involving a finite strain linear isotropic hardening material model. The sensitivities are determined depending on the material model properties.

Responsible Editor: Yoojeong Noh

✉ F. Beyer
fbeyer@uni-wuppertal.de

¹ Chair for Optimization of Mechanical Structures, University of Wuppertal, Gaußstraße 20, 42119 Wuppertal, Germany

Other research activities regarding topology optimization of crash structures use simplified objectives and constraint functions. Pedersen (2003, 2004) proposed a ground structure approach for transient frame structures. The topology optimization uses lattice beams with rectangular cross-section to model crashworthiness structures. The rectangular 2D beams can consider large displacements and rotations with a plastic zone formulation.

The following provides some examples of state-of-the-art topology optimization methods without mathematically calculated gradients which are addressing highly nonlinear problems.

The Hybrid Cellular Automata (HCA) (Patel et al. 2007, 2009) is a heuristic based method for crashworthiness topology optimization and does not use gradients. Its goal is to homogenize the distribution of internal energy densities inside a structure. The entire design space is discretized with solid finite elements. Each element represents an individual cell with an artificial density. A cell can consist of a group of elements. A derivative of the HCA method is commercially implemented in the software LS-TASC.

Hybrid Cellular Automata for Thin-Walled Structures (HCATWS) (Hunkeler 2013; Zeng and Duddeck 2017) is a variation of the HCA. It uses a lattice grid structure made of shell elements for the design space discretization.

The Equivalent Static Loads Method (ESL) (Choi and Park 2002; Park 2011) uses another approach by converting the nonlinear problem into multiple static problems. This enables the use of gradient based optimization algorithms. The optimization problem can contain responses that can be extracted from linear static simulations. In a first step a nonlinear finite element analysis is carried out and the displacement fields are extracted. Then, the loads are calculated, that would lead to the same deformation in a linear finite element analysis. These equivalent static loads can be created for different discrete time steps. The optimization process is then carried out on the linear static domain where the time steps are introduced as multiple loading conditions. The optimum from the linear static optimization is used as an initial design for the next iteration to generate a new deformation field. The ESL method is implemented in the commercial software solution Altair OptiStruct together with Radioss as well as GENESIS together with LS-DYNA.

The graph and heuristic based topology optimization (GHT) (Ortmann and Schumacher 2013) is developed for the optimization of profile structures with a constant cross-section. The three-dimensional structure is described by a two-dimensional mathematical graph for the cross-section and an extrusion spline. It modifies the topology of the cross-section to improve its mechanical behavior. The method is able to handle highly nonlinear crash load cases.

The optimization problem is divided into two different optimization loops. The outer loop performs topological modifications on the graph domain and the inner loop evaluates the design by a single function call, a sizing optimization (e.g., scaling the thickness) or a shape optimization. The heuristics for the topological modifications are developed for the improvement of crash loaded structures and are knowledge based rules.

Up to now only a two-dimensional graph description is used. The implementation of a new three-dimensional graph description, representing three-dimensional frame structures, is presented in this paper. The new graph still uses two-dimensional graphs to describe the cross-sections of the profiles. Both graph descriptions use the same new syntax. The finite element model of the frame is modelled with shell elements and the profiles are currently connected by rigid elements. The existing knowledge based heuristics (Ortmann and Schumacher 2013) are adapted for the new three-dimensional application.

In the following, the graph syntax of this new graph and heuristic based topology optimization for 3D-structures (GHT3D) is described. The derived shell model, the used heuristics and the optimization process are lined out. Two optimization examples are presented.

2 Optimization procedure

In the GHT multiple designs are calculated and evaluated within an iteration of the outer optimization loop. Each design proposal is generated by an individual heuristic. In each iteration the N_h competing heuristics generate up to $N_h \cdot N_{ke}$ new designs for the N_{ke} best designs from the previous iteration. This approach leads to a selective process.

In this paper, we use 6 different heuristics ($N_h = 6$) and $N_{ke} = 5$ best designs are transferred to the next iteration. These option results in 30 new designs in each iteration All proposals are evaluated by a finite element solver and sorted according to their objective.

The procedure is terminated if no further improvements towards the previous iteration can be achieved or when a maximum number of iterations is reached.

3 Three-dimensional graph description

The mathematical graph is used to describe the layout of a frame structure and the cross-sections of its profiles. It is used for the manipulation of the geometry and to check various geometrical constraints like minimum distances or connection angles. For describing a frame structure a set of vertices that defines the connection nodes and a set

of edges are needed. Edges are established between two vertices. The edges of the 3D graph represent profiles whose cross-sections are defined by assigning an additional 2D graph to them. A 2D graph only uses coordinates in the two-dimensional space and its edges represent the walls of the cross-section. The schematic relation between the graph domain and the finite element domain is shown in Fig. 1.

Each of those graph elements are internally identified by a unique identification number and a label that allows to associate all the elements with its corresponding graph. A type is used to allow different usages of the graph elements in different applications. Additional keywords define the properties of the graph element. Table 1 gives an overview of the available keywords for an edge.

Edges are directed, this means that they start at the first given vertex and end at the second one. For asymmetric cross-sections structural differences may result when the vertex-sequence of a 3D edge changes. The cross-section is oriented normal to the extrusion axis with its local y-axis turned in the direction of the given orientation vector around its origin. The extrusion axis is aligned along the x-axis of the local profile coordinate system. The extrusion length of a profile is limited by its bounding vertices. To avoid intersections and to leave space for the modeling of potential joints, the profile can be shortened by specifying an offset at the start and the end in the keyword *length*.

Using a graph to describe the geometry gives possibilities inside the optimization process:

- The graph can be easily manipulated with internal methods.
- The graph can be used to check manufacturing constraints like minimum distances and connection angles before simulating the designs.
- Not much work is needed to interpret the optimized structures since they are based on geometries that can be exported with an interface, e.g., to a STEP-file.
- The graph can be used for further structural optimization with cross-sections like filament winding profiles (Schneider et al. 2019), rib layouts (Schneider and Schumacher 2018) or flange structures (Link et al. 2019).

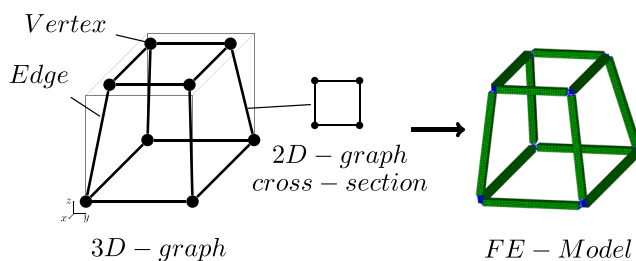


Fig. 1 The visualization of a 3D graph with a 2D graph cross-section (left) and a derived finite element model (right)

4 Derived finite element model

The finite element models are created by an internal scheme that uses the graph and additional information like the desired element length from a configuration file to automatically create a finite element model. Alternatively, a Tcl-script can be created that translates the necessary steps in commands to allow their execution in a commercially available preprocessor. In a first step, the profiles are created by extruding their cross-section with the corresponding length along the z-axis and meshing them. Components, properties and materials are created and assigned to the created elements. Through using the label of the graph element in the part name, the finite element results can later be associated with the correct graph element. All the profiles are then moved to the position defined by the 3D edge, considering the given orientation point or orientation vector. The whole model is generated without any penetrations.

After that the profiles are connected with each other and this is done by a node to node connection. Currently, only rigid bodies can be used and this allows to connect the edges at their ends without penetrations. More realistic connection nodes can be implemented in the future. In the last step the input deck is written to a file, so that it can be imported by the remaining input decks specifying the load case, material data and other components. When creating the load case, it is necessary to factor in, that new parts will be created and have to be included, e.g., in contacts or that constraints have to be defined for the newly created structure. Creating groups that exclude the fixed parts or box selections can help setting up the finite element model.

5 Developed heuristics for the new application

Heuristics are rule based procedures and generate new design proposals on the graph domain. The new three-dimensional graph framework allows topology changes while fulfilling geometrical constraints. Generally, the heuristics can create arbitrary graph structures consisting of connected straight profiles. The geometry is checked by manufacturing constraints. Every heuristic deals with certain mechanical behaviors like buckling. In a first step, the phenomena are detected by analyzing the crash simulation data. Then, a change of the geometry is proposed, that should lead to an improved mechanical behavior. Velocities, displacements and internal energies from the simulation data are used for the evaluation. The heuristics partly act in opposition to each other in a competing process. The heuristics are not directly related to a certain objective but try to improve their mechanical behavior in a general manner. The original heuristics for 2D

Table 1 Keywords for graph element “EDGE”

Keyword	Possible properties
<i>vertex</i>	Vertex IDs between which a connection should be established
<i>type</i>	<i>ExtrudedProfile</i> : represents a profile (3D), name of 2D graph is stated; <i>ExtrudedWall</i> : represents a wall in a 2D cross-section
<i>length (2D/3D)</i>	Two positions between the start and end vertex, where the extrusion of the profile/wall should begin and end
<i>orientationvector</i>	Orientation of the cross-section around the 3D edge axis; rotation axis of the cross-section is the local coordinate origin
<i>fix (2D/3D)</i>	Allows or forbids geometrical changes of the edge during the optimization
<i>outer (2D/3D)</i>	Indicates that an edge is located at the outer side
<i>material (2D)</i>	Identification number of the material that should be used

profile structures that change the cross-section are presented in Ortmann and Schumacher (2013). Based on the new requirements and the gained knowledge, the heuristics are further developed for the optimization of three-dimensional frame structures. In addition to the competing heuristics, some heuristics are defined, which, e.g., scale the thickness or straighten the graph and might be activated for all proposals. The evaluation of the simulation data for this new application is revised and enhanced to work in the three-dimensional design space. The consideration of buckling for example is no longer limited to a single wall (2D edge). Now, one of the heuristics analyzes the lateral deformation of the entire profile when a frame layout is optimized. In the following the used heuristics are described.

5.1 Support fast deforming edges

In a crash scenario it is possible that a single 3D edge buckles. It is useful to check buckling by detecting fast deforming edges and try to support the areas of instability. Figure 2 shows a schematic sequence on how edges could be supported due to fast deformations. The figure shows how a single new edge is generated. To detect buckling, the profile is divided in several sections along the extrusion axis and for each section the center point is calculated. The buckling in lateral direction is determined by high velocity differences due to fast deformations orthogonally to the 3D edge axis.

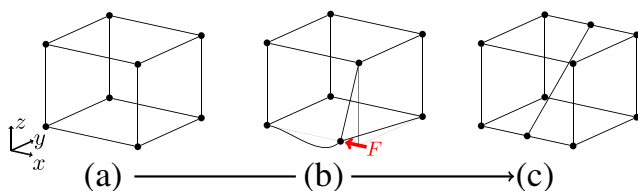


Fig. 2 Scheme for the heuristic *Support Fast Deforming Edges*. Initial design (a). Fast deformation under loads (b). New structure with supported edge (c)

Figure 3 shows the evaluation scheme between the timesteps t_0 and t_1 for an exemplary 3D edge. The global coordinates are defined with x, y, z and the local coordinates are ζ, ξ, η . The edge axis is aligned in the direction of η . Only velocities in lateral directions ($\zeta\xi$ -plane) from each time step are used to calculate a deformation index. The axial crushing velocities are not taken into account in order to avoid distorting the values. For the lateral velocities it is necessary to transform the deformation steps over all timesteps $T = \{t_0, t_1, \dots, t_{n_t}\}$ into a local coordinate system.

The set of the 3D edges is $I = \{i_1, i_2, \dots, i_{n_i}\}$ and the center points are $J = \{j_1, j_2, \dots, j_m\}$ for the edge i and N_{fe} is the number of finite element nodes. $\vec{p}_j^{(t)}$ is the location of a center point. The deformation index for buckling α_i for edge i is based on Ortmann and Schumacher (2014) and calculated by:

$$\alpha_i = \frac{1}{(N_{fe}^2 - N_{fe}) \cdot \frac{1}{2}|T|} \sum_{t \in T} \sum_{k \in J} \sum_{j \in J} \frac{|\vec{v}_j^{(t)} - \vec{v}_k^{(t)}|^2}{|\vec{p}_j^{(t)} - \vec{p}_k^{(t)}|} \quad (1)$$

whereby $k \neq j$ and the resulting lateral velocity vector $v_j^{(t)}$ of a point j at time t contains the two velocities in ζ and ξ direction:

$$\vec{v}_j^{(t)} = \begin{bmatrix} v_{\zeta j}^{(t)} \\ v_{\xi j}^{(t)} \end{bmatrix} \quad (2)$$

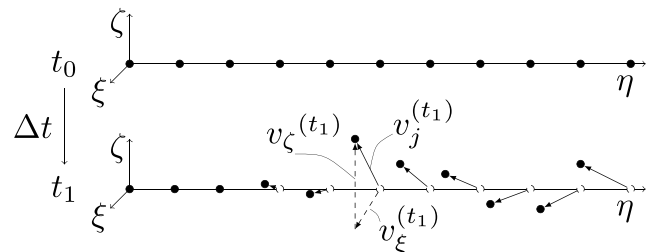


Fig. 3 Evaluation scheme for buckling in lateral direction based on weighted middle points of a 3D edge

In addition, a normalization with the average deformation index of all edges is applied:

$$\tilde{\alpha}_i = \frac{\alpha_i}{\frac{1}{|I|} \cdot \sum_{j \in I} \alpha_j} \quad (3)$$

The heuristic tries to support the edge with the largest value in the middle by connecting it orthogonally with a new edge towards another already existing edge. Furthermore, the first point that was found is used. If it is not possible due to manufacturing constraints for example, the next possible point is used.

5.2 Use deformation space tension and compression

The heuristics try to detect parts of the structure that get closer towards each other or increase their distance during the simulation. Introducing a new edge between them can lead to a higher energy absorption resulting from its deformation. It can increase the stability of the other two edges or it can transfer the deformation and energy absorption to previously inactive parts of the structure. This two cases lead to tension or compression conditions in the new edge. For evaluation, the same geometrical center points are used as in *Support Fast Deforming Edges*. The schemes are shown in Figs. 4 and 5. In essence the heuristics check the distances $d_{jk}^{(t)}$ between the center points $\vec{p}_j^{(t)}$ and $\vec{p}_k^{(t)}$, $j \in J, k \in K$ from different 3D edges over the time t by:

$$d_{jk}^{(t)} = |\vec{p}_j^{(t)} - \vec{p}_k^{(t)}| \quad (4)$$

Furthermore the criteria d_{com} for compression and d_{ten} for tension are calculated whereby $d_{jk}^{(t_0)}$ is the distance from the initial time step:

$$d_{com} = \min(d_{jk}^{(t)} - d_{jk}^{(t_0)}), t \in \{T, t \neq t_0\}, k \in K, j \in J \quad (5)$$

$$d_{ten} = \max(d_{jk}^{(t)} - d_{jk}^{(t_0)}), t \in \{T, t \neq t_0\}, k \in K, j \in J \quad (6)$$

It is necessary to check all edges with each other. At the end two edges are found for each heuristic which should be supported at the points j, k and thus the location with the biggest increase or decrease of distance.

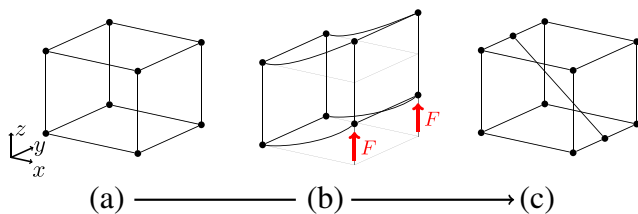


Fig. 4 Scheme for heuristic *Use Deformation Space Compression*. Initial design (a). Case of bending (b). New structure with compression loaded edge (c)

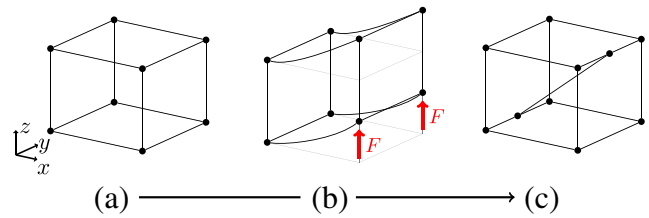


Fig. 5 Scheme for the heuristic *Use Deformation Space Tension*. Initial design (a). Case of bending (b). New structure with tension loaded edge (c)

5.3 Split long edges

The idea of this heuristic is to reduce the chance of buckling due to reducing the length of a structural element. After the edges have been sorted according to their length, the longest edge is connected through the shortest distance towards another long edge. No simulation data is used for this heuristic.

Long profiles have a high length to width ratio. The buckling tendency depends on this ratio so that shortening the lengths also reduces the buckling tendency (see Fig. 6).

5.4 Balancing energy density and delete needless edges

Large deformations lead to high internal energies. The heuristic *Balancing Energy Density* tries to connect edges with a high energy density to edges with a low energy density to homogenize the internal energy inside of the frame structure (see Fig. 7). The internal energy density ED_i is related to the initial volume V_i of a single edge i and is calculated by:

$$ED_i = \frac{\max \left(\sum_{w \in W} E_{i,w}^{(t)} \right)}{V_i^{(t_0)}}, t \in \{T, t \neq t_0\}, i \in I \quad (7)$$

whereby $E_{i,w}^{(t)}$ is the internal energy for a single cross-section wall w in time step t of the 3D edge i . Here the energies for each time step are summed up in order to consider between various events of deformation. Often in the 3D case there are more than one deformation scenarios happening over the length of an edge over time.

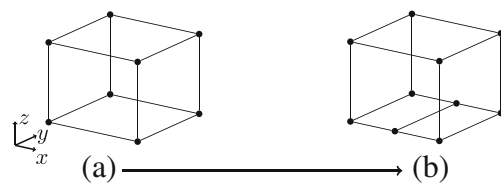


Fig. 6 Scheme for the heuristic *Split Long Edges*. Initial design (a). New edge between long edges (b)

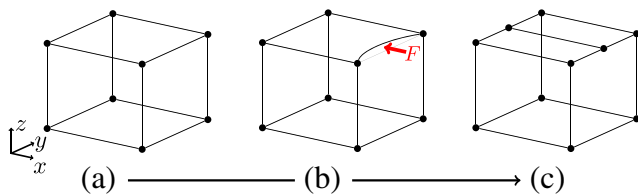


Fig. 7 Scheme for the heuristic *Balancing Energy Density*. Initial design (a). Case of high internal energy at the top. (b) Design with a new edge (c)

Furthermore, the maximum sum is divided by the volume of the whole 3D edge i .

The heuristic evaluates the energy density differences between the 3D edges $i, u \in I$ and $i \neq u$ which can be determined as:

$$\Delta ED_{iu} = |ED_i - ED_u| \quad (8)$$

In addition there is a division by the mean value of differences i, u :

$$\widetilde{ED}_{iu} = \frac{\Delta ED_{iu}}{\frac{1}{N_{\Delta ED}} \sum \Delta ED_{iu}} \quad (9)$$

where $N_{\Delta ED}$ is the number of density energy differences. As *Balancing Energy Density*, the heuristic *Delete Needless Edges* is also based on the internal energy density. The underlying idea is, that if the maximum internal energy of an edge is small compared to other edges, it can be removed from the structure without affecting it. Ideally it has no effect on the whole mechanical behavior of the structure. The energy density is normalized by the mean value of the maximum internal energy density from equation (7) with:

$$\widetilde{ED}_i = \frac{ED_i}{\frac{1}{|I|} \sum ED_i} \quad (10)$$

The edge with the lowest value is deleted from the structure.

5.5 Scale wall thicknesses and straighten graph

The heuristic *Scale Wall Thicknesses* and *Straighten Graph* are passive and non-competing in the optimization process. The heuristic *Scale Wall Thicknesses* (see Fig. 8) scales the wall thicknesses after a topological modification towards the previous mass in order to prevent an objective jump between iterations. Furthermore, the heuristic enables to create a mass equality constraint for the optimization. Due to the creation of new 3D edges inside of the structure, the mass of the frame structure would actually increase. By using the wall thickness scaling factor k with the desired target mass m_{target} and the current mass

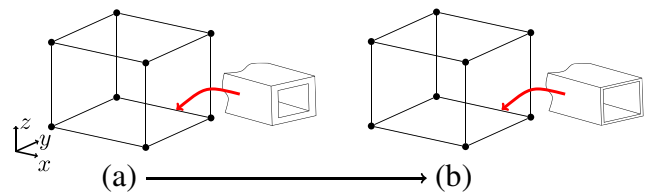


Fig. 8 Scheme for heuristic *Scale Wall Thicknesses*. Initial design (a). Scaled wall thickness (b)

$m_{current}$ the new wall thicknesses can be calculated by multiplying the old wall thickness with the factor k :

$$k = \frac{m_{target} - m_{fixed}}{m_{current} - m_{fixed}} \quad (11)$$

As a result of different thicknesses between 3D edges the primal thickness ratio should be retained. Furthermore, upper and lower bounds for the thickness can be defined. If the current thickness of an edge or wall is already at the allowed bound, its mass is fixed and the remaining walls have to be further scaled. The heuristic *Straighten Graph* (see Fig. 9) searches all edge connections with only two connected edges and a connection angle $> 160^\circ$. If such a connection is found, the heuristic substitutes both edges by a single edge and deletes the vertex at their former connection point. The process is necessary as a supplement for the heuristic *Delete Needless Edges* to avoid free edges.

6 Optimization workflow

The optimization process consists of two nested optimization loops. In this combination it is possible to execute the optimization strategy in a modular setup. The topological modification is performed in the outer optimization loop and the size and/or shape optimization in the inner loop (Fig. 10).

The GHT3D can start with a single function call to calculate the initial design or it can start with a size and/or shape optimization with multiple function calls to adjust the initial design to the given optimization problem. If the objective cannot be improved any further in the outer loop, a detailed size and shape optimization of the best design can be conducted to make use of a greater design freedom.

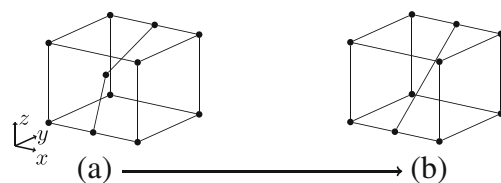


Fig. 9 Scheme for heuristic *Straighten Graph*. Initial design (a). Straightened two edges (b)

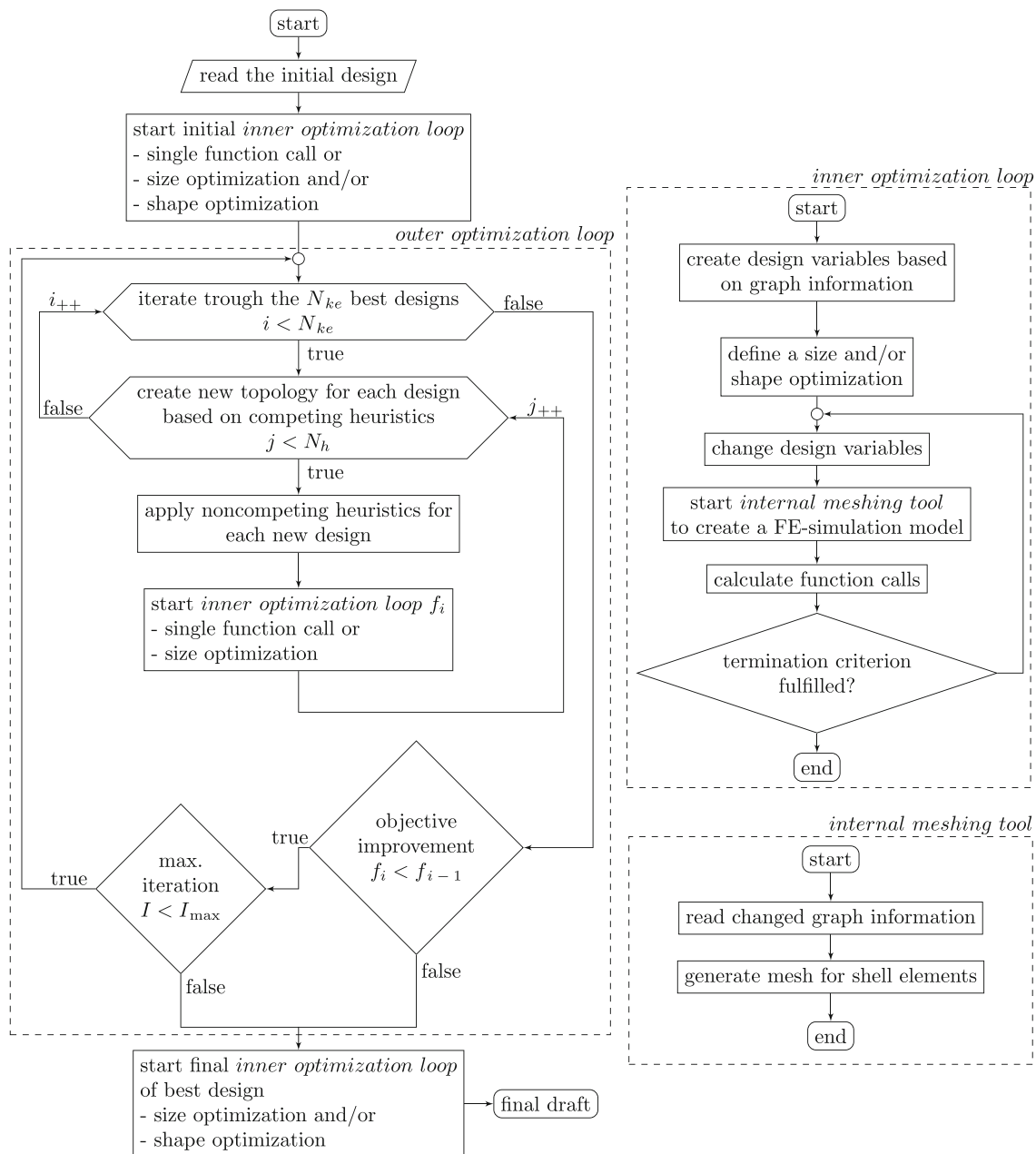


Fig. 10 GHT3D optimization flowchart. The process consists of two nested optimization loops. In the outer loop the topological modification and in the inner loop the size and/or shape optimization is performed. Internal meshing tool creates 3D shell element simulation models based on the 3D graph

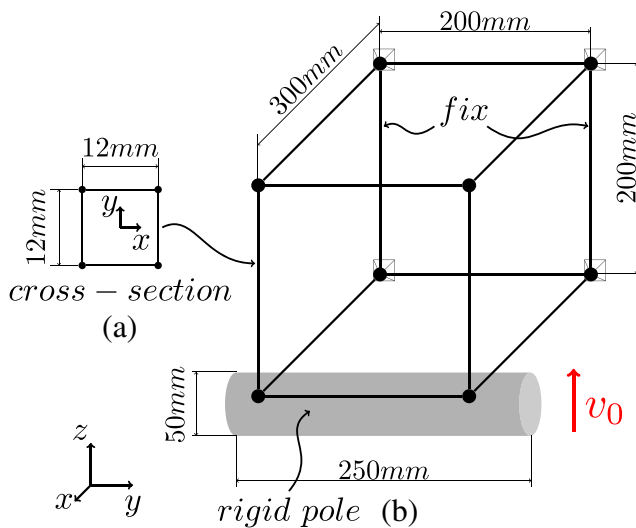
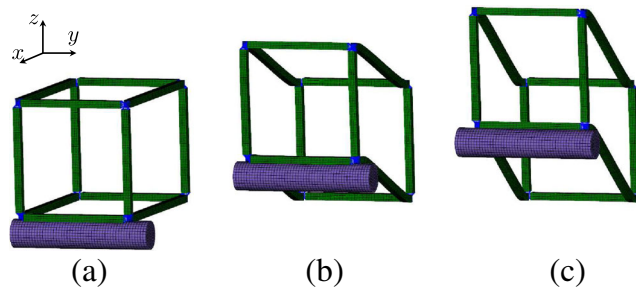
7 Additional graph functions

As mentioned earlier, it is easy to consider geometric constraints on the graph domain. For example the length of edges can be calculated as the distance between the bounding vertices. A connection angle between edges can also be easily calculated by using the corresponding vectors of the edges. It can be important to generate symmetric frame structures in order to lead the results to a better

design. With symmetry conditions it is also possible to reduce the number of shape and size variables in the inner optimization loop which is beneficial for most optimization algorithms. The number of design variables should be as small as possible. The number of size and shape variables depends on the number of structural elements, the boundary and symmetry conditions and the chosen optimization options. The implementation of the symmetry is achieved by mirroring a part of the graph by a given plane. In

Table 2 Material card — LS-DYNA: MAT_PIECEWISE_LINEAR_PLASTICITY (mm, t , s, N)

Mid	rho	E	PR	sigy			
1	$2.7e-9$	70,000	0.33	311.0			
eps1	eps2	eps3	eps4	eps5	eps6	eps7	eps8
0.0	0.0034373	0.0085932	0.016327	0.027928	0.045329	0.071431	0.11058
es1	es2	es3	es4	es5	es6	es7	es8
311.0	316.64	324.52	335.2	348.96	365.52	383.7	401.65

**Fig. 11** Initial frame structure with quadratic cross-sections (12 × 12 mm) (a). Impacted by a rigid pole (b)**Fig. 12** Initial frame impacted by rigid pole: Initial state at $t_0 = 0.000$ s (a) Deformed state at $t = 0.015$ s (b) Deformed state at $t = 0.030$ s (c)**Table 3** Optimization config — objective: Minimize the intrusion of the pole

Constraints	bounds
Wall thickness	$0.8 \text{ mm} \leq x \leq 3.5 \text{ mm}$
Edge length	$\geq 35.0 \text{ mm}$
Mass constraint	$= 500.0 \text{ g}$
Connection angle	$\geq 15.0^\circ$
Edge distances	$\geq 35.0 \text{ mm}$
Symmetric line up	xy -plane

the three-dimensional application symmetry can be defined along the XY , YZ and ZX plane, which leads to seven possible combinations for the symmetry condition. The size variables are symmetric as well.

The thickness of thin-walled structures has a large influence on the structural behavior. Three options are available to define the thickness variables. The whole graph can have the same thickness or each 3D edge has an individual thickness or even each wall of each edge has its own thickness.

Another kind of design variables exists for the shape of the frame. The shape variables are automatically generated by iteratively growing boundary boxes in Cartesian space to check the explicit shape constraints. It is based on the idea of the growing two-dimensional squared areas as implemented in the primal GHT2D (Ortmann and Schumacher 2013). If a 3D vertex is not limited by outer geometric constraints it generates three shape variables.

8 Application example

In this section we demonstrate the performance of the new three-dimensional layout optimization scheme. A cubic frame structure with a dimension of 300 mm × 200 mm × 200 mm between the vertices and two different load cases is modelled with shell elements. For each load case two optimizations with different objectives are carried out. Geometric constraints are added on the graph domain. An additional displacement constraint is active in the last iteration to allow starting from a non-feasible design. The optimization strategy starts with a shape optimization followed by automatic topological graph modifications through heuristics and ends with a detailed size optimization of the final draft. The shape design space is constrained by the outer dimension of the 3D graph. The mass of the cubic structure remains constant at 500 g in all iterations. The finite element simulations are calculated with the explicit solver LS-DYNA. For the size and shape optimization a standard SRSM algorithm from LS-OPT in the inner optimization loop with 600 function calls is used. Further constraints are described in the following application examples. The tables with the iterations show

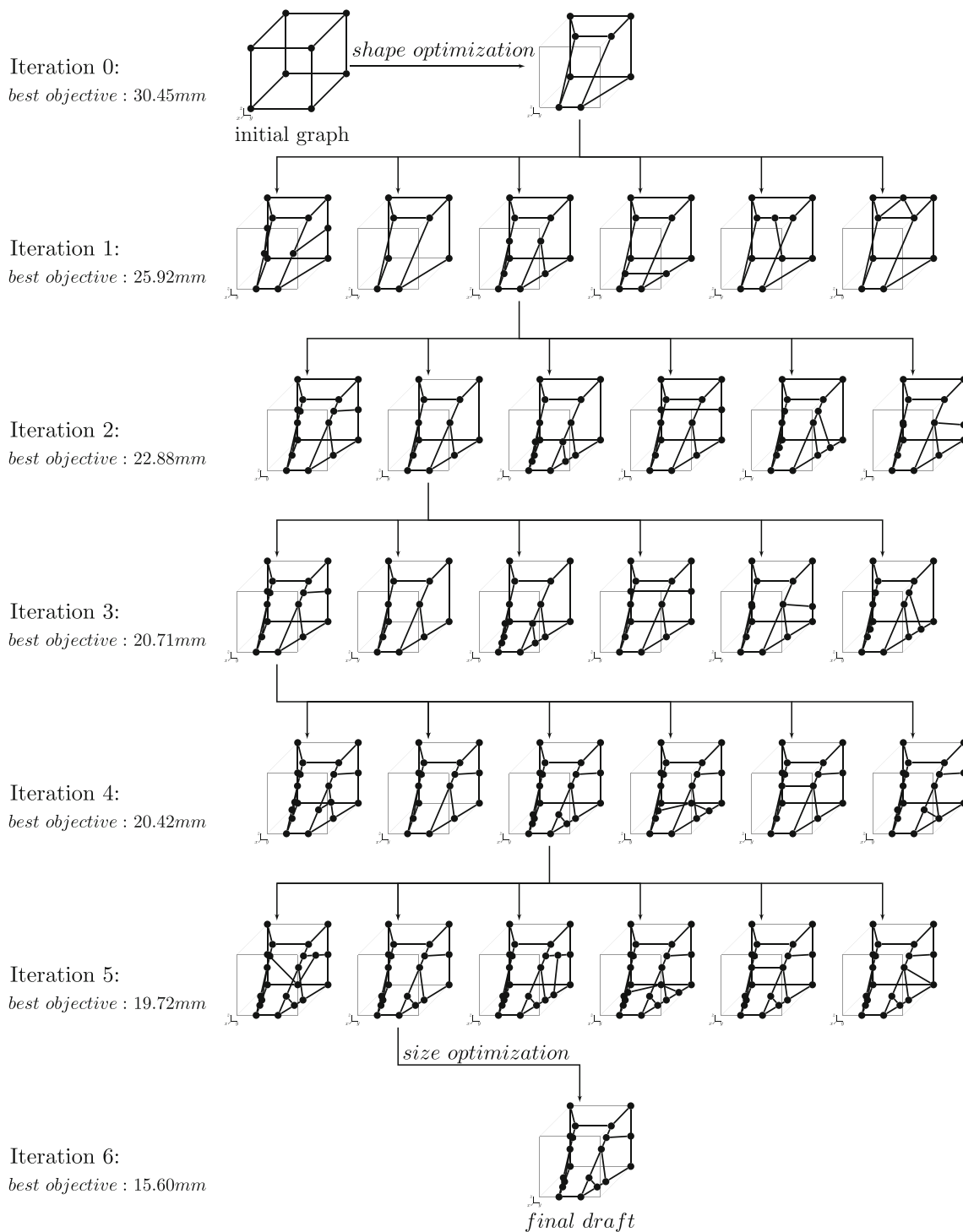


Fig. 13 Example 8.1.1, minimize the intrusion of the rigid pole, path towards the optimized design between iteration 0 and iteration 6 on graph domain

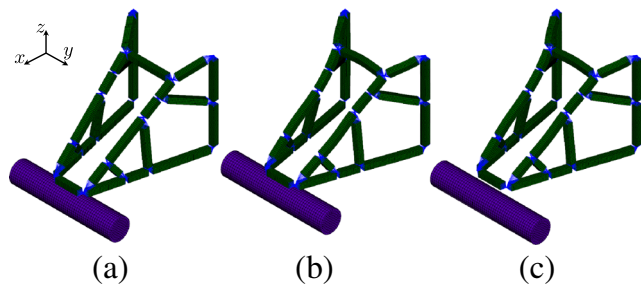
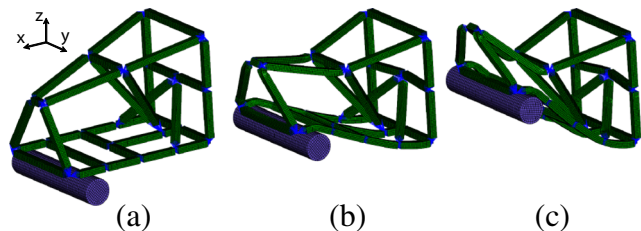
the designs that result in the best design. The used material card for aluminum behavior is described in Table 2. The required computing time per function call is approximately 10–15 min on 16 CPUs, depending on the complexity of the model.

8.1 Rigid pole loadcase

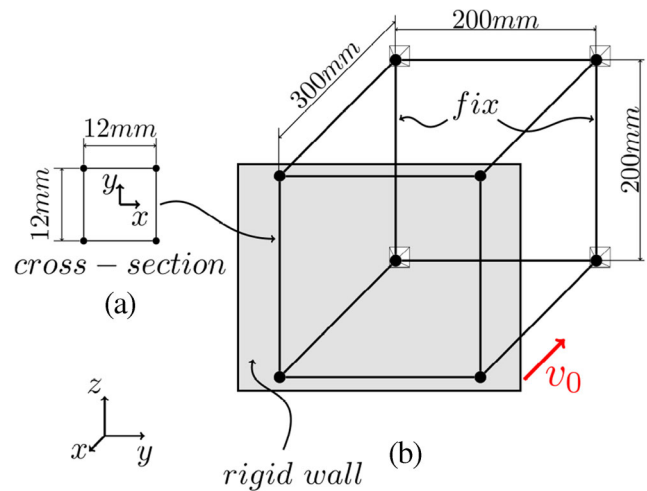
In the first load case, an aluminum frame is impacted by a rigid pole (Fig. 11b). The rigid pole has an initial velocity of 10 m/s in z-direction and a mass of 9 kg. Two edges near

Table 4 Optimization history — rigid pole model: Minimize the intrusion

Iteration (-)	Objective (mm)	Action (-)	Function calls (-)
<i>initial design</i>	174.34	–	1
0	30.45	6 shape variables	600
1	25.29	Heuristic graph manipulation	6
2	22.88	Heuristic graph manipulation	28
3	20.71	Heuristic graph manipulation	26
4	20.42	Heuristic graph manipulation	19
5	19.72	Heuristic graph manipulation	24
6	15.60	15 size variables	618

**Fig. 14** Optimum of the rigid pole model: Minimize the intrusion. Initial state at $t_0 = 0.000$ s (a). Deformed state at $t = 0.003$ s (b). Deformed state at $t = 0.010$ s (c)**Fig. 15** Optimum of rigid pole model: Minimize the contact force. Initial state at $t_0 = 0.000$ s (a). Deformed state at $t = 0.010$ s (b). Deformed state at $t = 0.030$ s (c)**Table 5** Optimization history — rigid pole model: Minimize the contact force

Iteration (-)	Objective (N)	Displacement (mm)	Action (-)	Function calls (-)
<i>initial design</i>	10,015	174.34	–	1
0	8420	128.34	6 shape variables	600
1	8270	129.84	Heuristic graph manipulation	6
2	8169	98.88	Heuristic graph manipulation	28
3	7746	137.79	Heuristic graph manipulation	26
4	7447	137.79	Heuristic graph manipulation	19
5	6751	128.03	Heuristic graph manipulation	24
6	6733	128.48	Heuristic graph manipulation	24
7	6732	129.23	Heuristic graph manipulation	24
8	5387	149.99	19 size variables	627

**Fig. 16** Initial frame structure with quadratic cross-sections (12×12 mm) (a). Impacted by a rigid wall (b)

the SPCs are declared as “fix” (Fig. 11) to prevent them from being deleted. Furthermore, they are excluded from shape optimization. The resulting mechanical loads consist mostly of bending with plastic deformation. Each edge has the same cross-section definition (Fig. 11a) with an initial wall thickness of 1.7 mm. The frame has fixed boundary conditions at the marked corners (Fig. 11b) in all degrees of freedom. The deformation of the initial design is shown in Fig. 12.

8.1.1 Objective: Minimize the intrusion of the pole

The first optimization is carried out with the objective of minimizing the intrusion of the rigid pole whereby the gravity point is used. The impact takes place in 0.040 s. The initial structure has an intrusion of 174 mm. The optimization objective and the constraints are shown in Table 3. The best designs through the optimization are shown in Fig. 13. It shows the path to the best design between iteration 0 and iteration 6 on the graph domain.

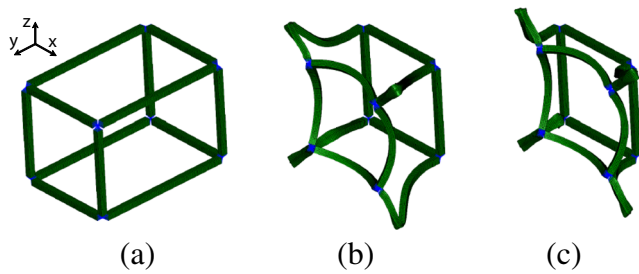


Fig. 17 Initial frame impacted by rigid wall: Initial state at $t_0 = 0.000$ s a Deformed state at $t = 0.020$ s b Deformed state at $t = 0.050$ s c

Table 6 Optimization history — rigid wall model: Minimize the intrusion

Iteration (-)	Objective (mm)	Action (-)	Function calls (-)
<i>initial design</i>	209.84	–	1
0	116.53	3 shape variables	600
1	104.36	Heuristic graph manipulation	6
2	82.79	Heuristic graph manipulation	23
3	70.08	Heuristic graph manipulation	19
4	70.50	Heuristic graph manipulation	19
5	64.68	Heuristic graph manipulation	18
6	49.25	Heuristic graph manipulation	15
7	49.61	Heuristic graph manipulation	15
8	45.87	Heuristic graph manipulation	18
9	45.74	Heuristic graph manipulation	12
10	44.18	15 size variables	615

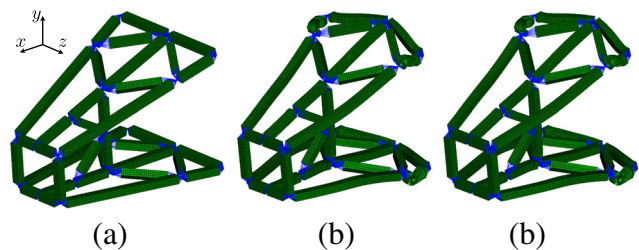


Fig. 18 Optimum of rigid wall model: Minimize the intrusion. Initial state at $t_0 = 0.000$ s (a). Deformed state at $t = 0.010$ s (b). Deformed state at $t = 0.020$ s (c)

Table 7 Optimization history — rigid wall model: Minimize the contact force

Iteration (-)	Objective (N)	Displacement (mm)	Action (-)	Function calls (-)
<i>initial design</i>	89,856	209.84	–	1
0	89,856	209.84	3 shape variables	120
1	76,460	156.56	Heuristic graph manipulation	6
2	66,871	152.68	Heuristic graph manipulation	27
3	54,945	134.16	Heuristic graph manipulation	30
4	53,015	131.22	Heuristic graph manipulation	30
5	53,015	127.31	Heuristic graph manipulation	16
6	52,634	129.25	8 size variables	29

This picture only shows six from the up to 30 possible designs generated in each iteration. The procedure is started with a shape optimization to adjust the initial geometry to the load situation. The initial layout has a huge influence on the outcome of the optimization. Only six shape variables are defined for the initial design due to a symmetric line up and single point constraints in the finite element model on one side. As Table 4 shows, the initial shape optimization has a significant influence on the objective function. In the following iterations the heuristics continue to lower the intrusion while retaining a constant mass.

In the first iterations of heuristic manipulations there are completely new different drafts with an improvement of the objective. The heuristic modification stops after iteration 5 because no further improvement can be achieved. The final draft as a finite element shell model with deformation is shown in Fig. 14 and the pole is already moving back at 0.010 s.

The most obvious change in the first iteration is the reduction to a triangular structure. In addition, the structure is shortened so that the point of impact is at the outermost border of the structure. The structure also becomes more slender on this end. Afterwards, the heuristics create new edges in a symmetric line up to support the profiles and lower the intrusion. Furthermore, two edges are removed near the single point constraints in iteration two and five. This allows to redistribute the mass to other profiles. Each profile has its own wall thickness. In example 8.1.1, due to the symmetric line up the amount results in 15 design variables. The objective is lowered by 20% from 19.72 to 15.60 mm during the sizing optimization, but only accounts for 2.36% in the entire process.

8.1.2 Objective: Minimize the contact force of the pole

In the second optimization the same constraints are used as in the previous optimization. An opposing objective is set to lower the maximum occurring contact force between the pole and the frame structure. Figure 15 shows the deformation over time of the optimized geometry.

In order to guarantee, that the pole is not flying through the structure without resistance, its displacement is constrained to 150 mm in the optimization problem, so that the frame structure has to stop it. Table 5 shows the optimization history with the contact force during the optimization. In the last iteration, 19 variables are used for the sizing of the wall thicknesses. After 601 function calls, the allowed displacement of 150 mm is completely utilized. All in all, the maximum contact force is lowered by 46.2% during the optimization. The results show that the outer front structure is narrowed. In seven iterations the heuristics introduced inner structures to the side and the bottom. Near the point of impact the structure is rather soft. In addition, the created edges near the single point constraints generate enough stiffness to meet the displacement constraint. The force-displacement curve with a comparison of the initial design and the final draft is presented in Fig. 21a.

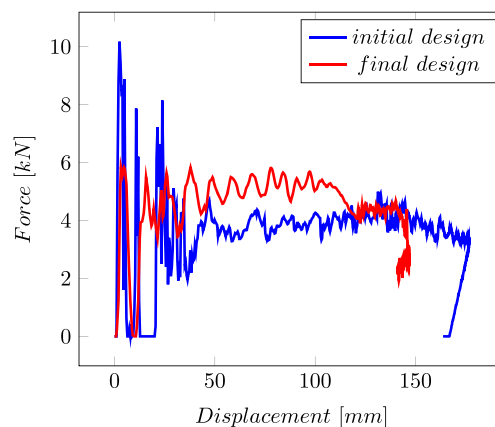
8.2 Rigid wall loadcase

In the second load case the pole is replaced by a rigid wall normal to the x -axis moving in negative x -direction. The frame structure undergoes large deformations during the compression. The cross-sections are equal to those in the previous application. The initial velocity of the rigid wall is 10 m/s and it has a mass of 65 kg. Two optimizations with separate objectives are started. Figure 16 shows the principle load case. Figure 17 shows the deformation over time of the initial geometry. The rigid wall hits the front of the frame structure in the negative x -axis direction. Due to the mass and initial velocity, the rigid wall has a particular momentum that has to be absorbed by the frame. The simulation time is limited to 0.050 s. The initial design cannot stop the rigid wall before the simulation ends. The goal is to change the structure to minimize the displacement or the contact force of the rigid wall. The rigid body elements lead to deformations with tangent conditions at the connection corners.

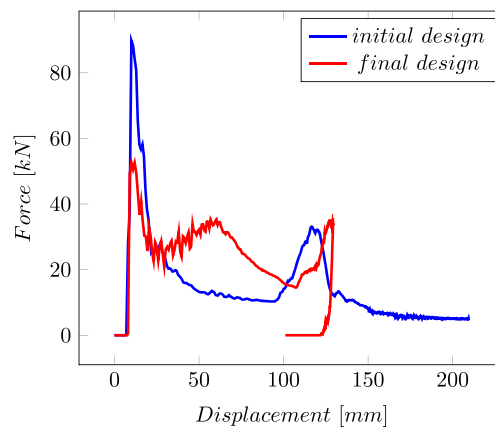
8.2.1 Objective: Minimize the intrusion of rigid wall

A double symmetry condition for this objective is defined for the geometry along the xy and zx plane. The remaining constraints are the same as in the previous optimization examples. The optimization runs over 10 iterations and detailed information are outlined in Table 6. Only 3 design variables are defined in the first iteration due to the double symmetry condition. As a result, it is possible to reduce the intrusion by 44% in the first iteration.

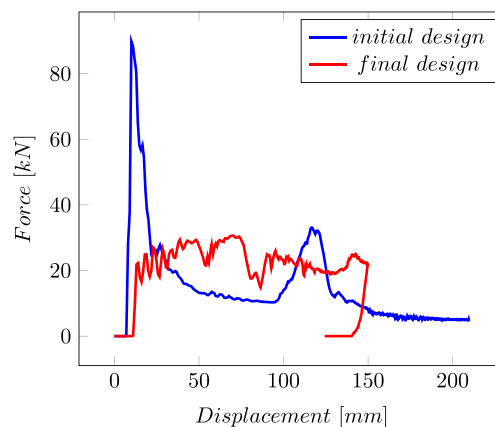
In iteration 7 there is a slight increase of the objective value. The optimization does not stop because of other parallel drafts in the same iteration that still improved. The heuristic modification stops because no further



(a)



(b)



(c)

Fig. 19 Force-displacement curve comparison of the initial design and final draft of application 8.1.2 (a), application example 8.2.2 (b), and application example 8.2.3 (c)

improvement can be achieved after 9 iterations. The number of function calls during the heuristic graph manipulation shows the amount of simultaneously calculated drafts. Figure 18 shows the final draft for the minimization of the

Table 8 Optimization history rigid wall model: Minimize the contact force, changed strategy

Iteration (-)	Objective (N)	Displacement (mm)	Action (-)	Function calls (-)
<i>initial design</i>	89 856.21	209.84	-	1
1	76 460.10	156.56	Heuristic graph manipulation	6
2	66 871.04	152.68	Heuristic graph manipulation	27
3	54 945.14	134.16	Heuristic graph manipulation	30
4	53 885.71	131.22	Heuristic graph manipulation	30
5	53 015.13	127.31	Heuristic graph manipulation	16
6	30 869.14	146.40	15 shape variables	601
7	30 607.32	149.72	19 size variables	608

intrusion of the rigid wall. The huge reduction of intrusion in the first iteration is caused by the change of the outer shape towards a trapezoidal structure. It increases the stiffness and lowers the intrusion. New edges are introduced by the heuristics while the wall thickness decreases to ensure a constant mass. Most edges are created at the bottom and at the top side with a centered connection through the middle plane to protect against buckling. Two edges between the single point constraints are removed, since they have no influence on the structural behavior. The mass of the removed edges is distributed among the other edges. At time 0.020 s (Fig. 18c) the four short edges near the fixed corners undergo severe deformation and absorb much of the energy. These short edges cannot be supported any further because it would violate the minimum edge length constraint.

8.2.2 Objective: Minimize the contact force of rigid wall

The process for the force minimization runs through six iterations and is shown in Table 7 in detail. In the first iteration it is not possible to improve the objective value with the given shape variables. If the outer shape changes to a trapezoid it would get stiffer and increase the contact force.

The optimization runs further with the initial design and improves the objective by activating the heuristics. After five iterations the heuristic manipulation stops due to all walls reaching the lower wall thicknesses bound.

The entire frame is with 0.804 mm very close to the lower limit of the wall thickness. As a result, the thickness optimization has nearly no room for improvements.

The shape variables are located at the connection nodes and, as a result, there are more variables in higher iterations with a higher number of edges. With this knowledge in mind, another approach could be to move the shape optimization to the end of the GHT3D process. Figure 19 shows the deformation of the final draft.

In the following, a modified variant of the last iteration is shown. The heuristics create the same structure as in the previous example (Table 8). With an additional subsequent shape optimization with 15 variables it is possible to reduce the objective further between iteration five and six. The deformation of the final design is shown in Fig. 20. In the last iteration the thickness is sized by using 10 design variables. The shape and size optimizations are separated in order to keep the amount of design variables in each optimization small. This draft does not violate any constraint. The force-displacement curve is presented in Fig. 21c and shows a steady energy absorption within 150 mm displacement. With this set up it is possible to use the allowed displacement of the rigid wall and reduce the objective by 66%. In the final draft there are less straight parts which create less stiffness and smaller force peaks. The rigid wall stops before 150 mm. Most of the structural elements are involved in the deformation.

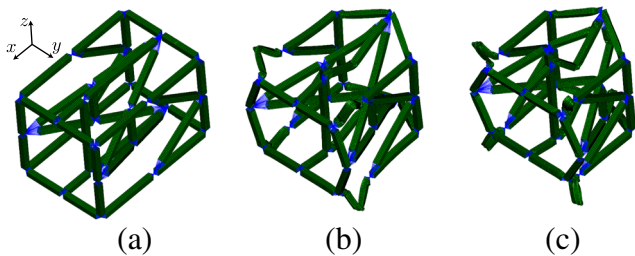


Fig. 20 Optimum of rigid wall model, Table 7: Minimize the contact force. Initial state at $t_0 = 0.000$ s (a). Deformed state at $t = 0.010$ s (b). Deformed state at $t = 0.040$ s (c)

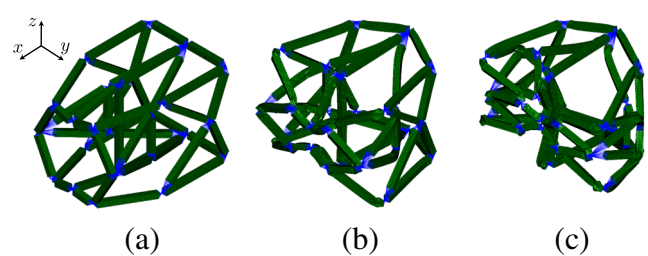


Fig. 21 Optimum of rigid wall model, Table 8: Minimize the contact force. Initial state at $t_0 = 0.000$ s (a). Deformed state at $t = 0.010$ s (b). Deformed state at $t = 0.040$ s (c)

9 Conclusion

A new graph and heuristic based optimization approach is proposed for finding three-dimensional layouts for individual crashworthiness load cases. The procedure is explained and demonstrated with two examples, each with two different objectives. In addition, the method can be started with different strategies. The presented heuristics are derived from the primal GHT2D, which uses a two-dimensional graph to optimize cross-sections. Further research on existing and new heuristics for this three-dimensional case will be needed in the future. Furthermore, different problem formulations might need their own strategy for finding good designs. In general, it could be a strategy to come from a simple structure which becomes more detailed in every iteration. It should be clear that the topology, shape and thickness affect the structural behavior and should be considered. Until now the heuristics cannot perform shape changes like the shortening of the frame in the first example, but in combination with the size- and shape optimization it is possible to achieve enormous improvements for the given load cases.

One of the objectives of the optimizations is absorbing energy for a given displacement without force peaks, but in the finite element models the connection nodes are represented by rigid body elements that generate an artificial stiffness. This leads to a non-physical deformation behavior. One of the next development steps of the GHT3D should be the realization of more adequate connection elements in the finite element models.

At this time, we have not performed any investigations on robustness yet. The consideration of robustness is particularly important when optimizing the behavior of nonlinear crash models. It is possible to carry out the confidences of robustness. However, the determination of the confidences is very time consuming. Robust design processes adapted to our task must be developed here.

Due to the dynamic changing loads and a lot of phenomena for absorbing kinetic energies like buckling, it is important to find not only straight connections between loads and boundary conditions. New heuristics might be developed to address this aspects.

Funding Open Access funding enabled and organized by Projekt DEAL. This work was funded by the Research Association of Automotive Technology (FAT), part of the German Association of the Automotive Industry (VDA).

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

Replication of results The data that support the findings of this study are available from the corresponding author upon reasonable request.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Bendsøe MP, Kikuchi N (1988) Generating optimal topologies in structural design using a homogenization method. *Comput Methods Appl Mech Eng* 71(2):197–224, [https://doi.org/10.1016/0045-7825\(88\)90086-2](https://doi.org/10.1016/0045-7825(88)90086-2)
- Choi W, Park G (2002) Structural optimization using equivalent static loads at all the time intervals. *Comput Methods Appl Math* 191(19):2077–2094, [https://doi.org/10.1016/S0045-7825\(01\)00373-5](https://doi.org/10.1016/S0045-7825(01)00373-5)
- Hunkeler S (2013) Topology optimization in crashworthiness design via hybrid cellular automata for thin-walled structures. PhD thesis, Queen Mary University of London, UK
- Ivarsson N, Wallin M, Tortorelli D (2018) Topology optimization of finite strain viscoplastic system under transient loads. *Int J Numer Methods Eng* 114:1351–1367, <https://doi.org/10.1002/nme.5789>
- Link S, Schneider D, Schumacher A, Ortmann C (2019) Integration of flange connections in the graph and heuristic based topology optimization of crashworthiness structures. In: Rodrigues H et al (eds) *EngOpt 2018 Proc. of the 6th International Conference on Engineering Optimization*. Springer Nature, Switzerland, pp 619–631. https://doi.org/10.1007/978-3-319-97773-7_55
- Ortmann C, Schumacher A (2013) Graph and heuristic based topology optimization of crash loaded structures. *Struct Multidiscip Optim* 47:839–854, <https://doi.org/10.1007/s00158-012-0872-7>
- Ortmann C, Schumacher A (2014) Mathematical description and algorithmization of expert knowledge for the support of the topology optimization of crashworthiness structures. *PAMM Proc Appl Math Mech* 14:1023–1026, <https://doi.org/10.1002/pamm.201410488>
- Park G (2011) Technical overview of the equivalent static loads method for non-linear static response structural optimization. *Struct Multidiscip Optim* 43(3):319–337, <https://doi.org/10.1007/s00158-010-0530-x>
- Patel N (2007) Crashworthiness design using topology optimization. PhD thesis, University of Notre Dame, USA
- Patel N, Renaud JAT (2009) Crashworthiness design using topology optimization. *ASME J of Mech Des* 132(6)
- Pedersen C (2003) Topology optimization design of crushed 2d-frames for desired energy absorption history. *Struct Multidiscip Optim* 25:368–382, <https://doi.org/10.1007/s00158-003-0282-y>
- Pedersen C (2004) Crashworthiness design of transient frame structures using topology optimization. *Comput Methods Appl Mech Eng* 193:653–678, <https://doi.org/10.1016/j.cma.2003.11.001>
- Schneider D, Schumacher A (2018) Finding best layouts for rips on surfaces for crash loads using the graph and heuristic based topology optimization. In: Schumacher A, Vietor T, Fiebig S, Bletzinger K-U, Maute K (eds) *Advances in Structural and Multidisciplinary Optimization*. Springer, Cham, pp 1615–1628. https://doi.org/10.1007/978-3-319-67988-4_121

- Schneider D, Schumacher A, Donhauser T, Huf A, Schmeer S (2019) Flexible graph syntax for the topology optimization of crashworthiness profile structures made from thermoplastic composites. In: Hausmann JM (ed) Selected, peer reviewed papers from the 22 nd Symposium on Composites, vol 809. Trans Tech Publications Ltd, Kaiserslautern, pp 493–499. <https://doi.org/10.4028/www.scientific.net/KEM.809.493>
- Weider K, Schumacher A (2018) A topology optimization scheme for crash loaded structures using topological derivatives. In: Schumacher A, Vietor T, Fiebig S, Bletzinger K-U, Maute K (eds) Advances in Structural and Multidisciplinary Optimization. Springer, Cham, pp 1601–1614. https://doi.org/10.1007/978-3-319-67988-4_120
- Weider K, Schumacher A et al (2019) Adjoint method for topological derivatives for optimization tasks with material and geometrical nonlinearities. In: Rodrigues H (ed) EngOpt 2018 Proc. of the 6th International Conference on Engineering Optimization. Springer Nature, Switzerland, pp 867–878. https://doi.org/10.1007/978-3-319-97773-7_75
- Zeng D, Duddeck F (2017) Improved hybrid cellular automata for crashworthiness optimization of thin-walled structures. Struct Multidiscip Optim 56:101–115, <https://doi.org/10.1007/s00158-017-1650-3>

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.