The LS-TaSC™ Tool

TOPOLOGY AND SHAPE COMPUTATIONS

THEORY MANUAL

November 2016

Version 3.2

Corporate Address

Livermore Software Technology Corporation

P. O. Box 712

Livermore, California 94551-0712

Support Addresses

Livermore Software Technology Corporation

7374 Las Positas Road

Livermore, California 94551

Tel: 925-449-2500 ♦ Fax: 925-449-2507

Email: sales@lstc.com

Website: www.lstc.com

Livermore Software Technology Corporation

1740 West Big Beaver Road

Suite 100

Troy, Michigan 48084

Tel: 248-649-4728 ♦ Fax: 248-649-6328

Disclaimer

LSTC reserves the right to modify the material contained within this manual without prior notice.

The information and examples included herein are for illustrative purposes only and are not intended to be exhaustive or all-inclusive. LSTC assumes no liability or responsibility whatsoever for any direct of indirect damages or inaccuracies of any type or nature that could be deemed to have resulted from the use of this manual.

14-Jun-17

I must say it looks a bit like science fiction to many people – Ofir Shor, June 2009, while evaluating the alpha version.

1. Topology Theory

1.1. Background

The methodology in this version is an optimality criteria method together with a multi-tensor extension for the computation of numerical sensitivities. With the distinctive properties of the method subject to change between versions, the methodology is best referred to as LS-TaSC 3.2. The traditional approach for solving topology optimization problems is based on sensitivity analysis that is inexpensive to obtain for linear-static problems. However, deriving analytical sensitivities for dynamic analysis is very difficult due to the complex interactions among material nonlinearities, geometry and mesh, and transient nature of load and boundary conditions. Numerical computation of sensitivities is also not practical due to the high computational expense. Hence, the conventional sensitivity-based approach of topology optimization is not practical for crashworthiness problems. To overcome the aforementioned difficulties in topology optimization, a different approach was proposed. This approach does not require gradients and hence there is no need to compute the sensitivities. The approach in earlier versions such as LS-TaSC 1.0 considered the work done at Notre Dame university (see e.g. [1] and [2]), which contained elements of the used fully stressed methodology (this also forms the basis for the perhaps better known optimality criterion method described in Bendsøe and Sigmund [3]). From the start the work differed from the Notre Dame project by omitted their signature use of a cellular algorithm; instead we reverted to older, proven knowledge by using a more traditionally filtering and amendment of the SIMP strategy. Additions such as the multipoint approach for constrained optimization are unique to LSTC and our vehicle crash applications. Academics doing a literature review should therefore also consult other standard views of topology optimization and our patent portfolio to understand what is currently actually implemented.

Topology optimization in structures has been studied since the 1970s resulting in many books and numerous papers. The books by Rozvany [4] and Bendsøe and Sigmund [3] provide a very comprehensive and contemporary survey of optimization techniques used in topology optimization. Most previous studies in topology optimization, e.g., [5] and [6], have focused on designing structures with static loading conditions but there is relatively little work on handling problems involving dynamic loads, like those observed in crashworthiness optimization [7]. In the context of crashworthiness, topology

optimization is a very complex problem due to non-linear interactions among material and geometry, and the transient nature of boundary conditions.

The most efficient topology optimization methods use sensitivity information e.g., optimality criterion based methods by Rozvany [4] and Bendsøe and Kikuchi [8], to drive the search for an optimum. Sensitivity calculations are computationally inexpensive for linear-static problems but not for the problems that involve non-linearities. To use the same set of topology optimization methods, one needs to explicitly calculate sensitivities which is practically infeasible due to very high computational cost involved with simulations. Thus the theory used to solve the linear static load cases, though quite mature, is not practical for the crashworthiness problems and alternative methods need to be explored.

Previously different approaches have been adopted by authors to solve topology optimization with non-linearities. Pedersen used the method of moving asymptotes (MMA) for crashworthiness optimization of two-dimension structures [9]. There, a quasi-static non-linear FEA was used to account for geometric nonlinearities to handle large deformation and rotation of plastic beam elements. However, the method ignored the contact between elements arising due to non-linear structural behavior. Soto [10] and [11] presented a heuristics-based method using a prescribed plastic strain or stress criterion to vary the density to achieve the desired stress or strains with a constraint on mass. However, this method could not be generalized to solid structures. Pedersen [12] used beam elements to handle topology in crashworthiness optimization. Forsberg and Nilsson [13] proposed two algorithms to get a uniform distribution of the internal energy density in the structure. In the first method, they deleted inefficient elements and in the second method they updated the thicknesses of the shell elements. This method also was limited to a small set of optimization problems. Shin et al. [14] proposed an equivalent static load method where they calculated an equivalent static load for the dynamic problem and then used the linear-static topology optimization techniques to find the optimal topology. The main difficulty in this method is the requirement to accurately compute the equivalent loads.

1.2. Implementation

The algorithm for structural optimization is shown in Figure 1-1. After defining the problem, the topology is evolved using the simple rules defined on the variables. The constraints are accommodated during the state update procedure.

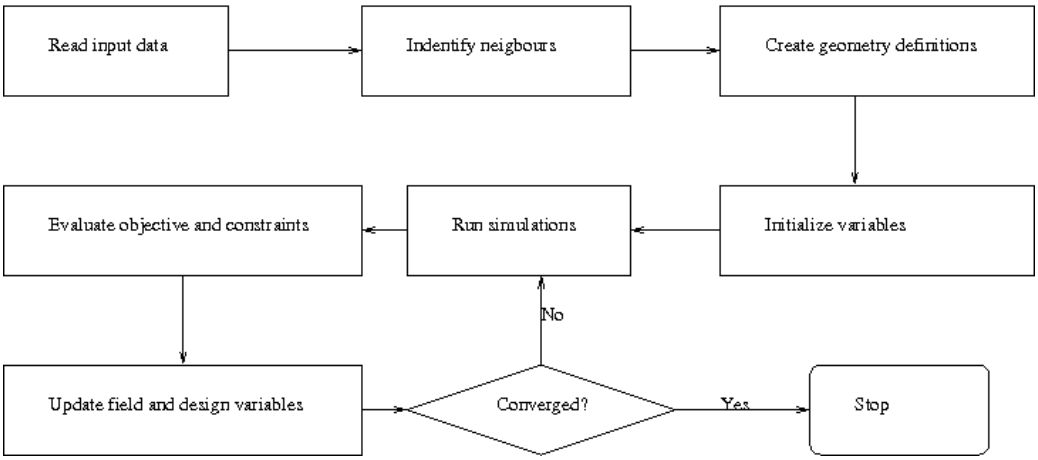

Figure 1-1: The topology optimization algorithm

1.2.1. Definition

The input data is used to identify the design domain and design material model. The input data comprises of method data e.g., number of iterations, convergence tolerance, and the problem data, e.g. load cases and design part.

1.2.2. Creating the variables

The finite element model is mapped to design variables. Each design variables is assigned to a solid element in the design domain. For extrusion and symmetry constraints, the equality constraints are defined between the variables. For casting constraints, inequality constraints are established.

1.2.3. Filtering of results

Past works were based on the structured grid arrangement of cells. This assumption would breakdown for industrial applications where structured grids are not always possible. Consequently, a radius based strategy is used to identify neighbors. In this strategy, a virtual sphere of user-defined radius is placed at the centroid of an element. All elements that are within this sphere are considered the neighbors of the corresponding element, and the results are averaged over the elements in the neighborhood

$$U_i = \frac{\sum_{j=1}^{n} w_j U_j}{\sum_{j=1}^{n} w_j}, \tag{1.1}$$

where *w* are the filter weights and *U* are initially the internal energy density values as extracted from the d3plot file. Multiple values of the internal energy density are computed for a dynamic analysis; in this case the maximum value is used.

If the user specifies a negative value then the value is assumed to be element specific and the radius used for an element is the absolute value of the specified value times twice the average distance from the center of the element to the nodes. If the value is positive then the specified value is applied to all elements. The default value is -1.0, which means the results from all elements sharing a node with an element are likely to be used.

1.2.4. Material Parameterization

The material model is parameterized using the *(relative) density approach*. In this approach, a design variable is directly linked to the individual material element such that each variable has its own material model. The material properties corresponding to the values of design variables are obtained using an appropriate interpolation model. The solid isotropic material with penalization (SIMP) model [15] is the most popular interpolation method. This model is a power law approach that drives the intermediate material properties towards the boundaries to obtain a 0-1 topology. According to the SIMP model, the material properties are defined as,

$$\rho(x) = x\rho_0, \tag{1.2}$$

$$E(x) = x^p E_0, \tag{1.3}$$

$$\sigma(x) = x^q \sigma_0, \tag{1.4}$$

$$E_h(x) = x^q E_{h0}, \tag{1.5}$$

where ρ denotes the density of the material, E represents the Young's modulus, σ is the yield stress, and E_h is the strain hardening modulus. The last two material properties represent material non-linearities and are required for dynamic problems like crash that involve material yielding. The subscript 0 refers to the base material properties. The design variable x, with $0 \leq x \leq 1$ is also known as relative density, varies from 0 to 1 where 0 indicates void and 1 represents full material. A more detailed description of the material model parameterization, one should refer to Bendsøe and Sigmund [3] and Patel [2]. Elements with design variable value smaller than a user-defined minimum value are deleted to improve numerical stability. To enable the use of very large FE models, this approached was implemented using a discrete material model approach [16].

1.2.5. Solid/Void behavior

Solid/void options force the elements to be either fully used or not used, instead of partially used.

Forcing elements to be either fully used or not used will results in a target field that is less uniform over the part, because the intermediate variable values are required to obtain an uniform field.

LS-TaSC has two methods: using volume based results and the SIMP method. In the original academic research the researchers found that the SIMP scheme was not really required for this implementation; this was because they used volume based results. This simple scheme is the default in LS-TaSC, seeing that the original research was reviewed by the academic sector and has proved to work well in extensive industrial testing.

1.2.6. Element volume vs material volume

The issue is that there are two measures on energy density. The question is whether the density results is reported relative to the volume of the element, or relative to the amount of material in the element. The value reported in the d3plot file for solids is relative to the volume of the element. This must be scaled with the design variable (the volume of material in the element) to obtain the actual IED for the material in that element. For example, consider an element with E=3, volume=5, and x=0.1. The EID_e (per element volume) = 3/5 = 0.6. The EID_m (per material volume) = 3 / (5*.1) = 6..

For shell elements this issues does not arise, because the thickness is the design variable. The element volume and material volume are therefore the same.

This applies only to energy density computations. The energies as reported in the glstat are always correct, because they are not scaled.

1.2.7. The SIMP solid/void scheme

Solid Isotropic Material with Penalization (SIMP) force material to 0/1:

$$\rho(x) = x\rho_0$$
$$E(x) = x^p E_0$$
$$\sigma(x) = x^q \sigma_0$$
$$E_h(x) = x^q E_{h0}$$

To use SIMP, take p=3 and q=2.666 in the above equations using element material based fields.

Using SIMP with non-linear material models may results in problems, because for some values of p and q the material model may not be valid. An additional problem is elements being driven to a very low stiffness using through the x^p term; these elements are likely to be inverted during nonlinear calculations. Like many other schemes, the SIMP scheme results in gray areas in which the elements are partially filled with material, which may be undesirable.

1.2.8. The element volume solid/void scheme

In this case the SIMP parameters p and q are both taken to be 1. and 0.66. respectively. The field results are however used on an element volume based instead of an element material basis. This results in the desired solid/void behavior.

In the original academic research the researchers found that the SIMP scheme was not really required for this implementation, this was true because they used this volume based

results. This simple scheme is the default in LS-TaSC, because it was reviewed by the academic sector and has proved to work well in extensive industrial testing.

1.2.9. Design Objectives and Constraints

The typical goal of topology optimization is to obtain a structure with the best use of the material. Compliance and the strain energy density are the most commonly used objectives for linear static problems. For dynamic problems, like crashworthiness simulations, the structure needs to absorb the maximum energy while maintaining the structural integrity and keeping the peak loads transmitted to the occupants low.

Following the formulation proposed by Patel [2], the goal of obtaining uniform internal energy density in the structure is defined as the objective for optimization. This concept is similar to the fully-stressed design and uniform strain energy density approaches, see e.g. Haftka and Gürdal [17] or Patnaik and Hopkins [18], that are well established in literature for linear-static problems. The use of the internal energy density in optimization, its relationship with the design sensitivity information for crash problems, and its usefulness for ranking variables has been extensively studied by Öman [19], [20].

The optimization problem is formulated as

$$\min_{x} \sum_{i=}^{N} \sum_{j=1}^{L} \left(w_j U_j(x_i) - U_j^* \right), \tag{1.6}$$

subject to

$$\begin{aligned} &\sum_{i=1}^{N} \rho(x_i) V_i \le M^* \\ &C_j^l \le C_j \le C_j^u, \quad j = 1, 2, \ldots, J \\ &x_{\min} \le x_i \le 1.0. \end{aligned} \tag{1.7}$$

where U represents the internal energy density of the i^{th} element, V_i is the volume of i^{th} element, U^* represents internal energy density set point, and C_j is the j^{th} constraint. There are L load cases with a total of J constraints. The superscripts 'l' and 'u' represent lower and upper bounds on the constraints, respectively.

1.2.10. Design Variable Initialization

The design variables are initialized to satisfy the material constraint. All elements are assigned the same design variable values. All associated field variables are also initialized to zero.

1.2.11. Simulation to Obtain Field Variables

The elements in the finite element model are modified by changing the material models, i.e. adding or deleting elements in each iteration. To his end, the input deck is re-written at each iteration. This modified input deck is analyzed using LS-DYNA® [21]. One can

take advantage of multiple processors using the MPP version of LS-DYNA. The relevant field variables for all elements are obtained from the output to completely define the state of each variable. For multiple load case conditions, the state variable is based on the output from simulations of different load cases.

For dynamic problems, it was observed that accounting for the history of evolution induces stability by reducing the element deletion rate. Hence, the internal energy density field variable of i^{th} element at iteration t is updated by defining a weighted sum on the field variable of three previous iterations as follows,

$$U_i^t = \sum_{j=0}^{3} (x_i)^{j+1} U_i^{t-j} \Bigg/ \sum_{j=0}^{3} (x_i)^{j+1}. \tag{1.8}$$

where x_i is the design variable associated with the i^{th} variable at iteration t. If the load cases are a mixture of dynamic and static problems, then this weighing is followed for all the load cases.

1.2.12. Global Constraint Handling

In presence of constraints other than the mass constraints, the target mass constraint is adjusted to satisfy the structural constraints. The mass target (M^*) is increased in proportion to the constraint violation for all constraints except force constraints for which the mass target is reduced.

$$\begin{aligned} M^* &= M^* + \Delta M, \\ \Delta M &= \left(\sum_j K_j^c \varepsilon_j \right) / J, \end{aligned} \tag{1.9}$$

where J is the total number of constraints, K_j^c is the coefficient used to scale the constraint violation of the j^{th} constraint, and ε_j is the violation of the j^{th} constraint. The total change in mass target (ΔM) is bounded to allow gradual changes in the structure.

1.2.13. Dynamic Load Case Weighing

The desired behavior is $k_1 C_1 + offset_1 = k_2 C_2 + offset_2$ with C the constraint value, k a scale factor, and an offset added as shown. The weight w_i of load case i is adjusted to change constraint C_i. The target value is computed as

$$C_{target} = \frac{\sum_{i}^{i=n} (k_i C_i + offset_i)}{n} \tag{1.10}$$

from which we compute

$$\Delta w_i = (C_{target} - k_i C_i - offset_i) / (0.1 C_{target}) \tag{1.11}$$

and a maximum bound of $0.05\, w_i$ is placed on Δw to ensure convergence in a reasonable number of iterations.

1.2.14. State Update Rules

This is the heart of topology optimization method. In this step, the state of a variable is updated based on the state of its neighbors. The state update is carried out in two steps:

1. Field variable update: The field variable (internal energy density) of a variable is updated as accounting for the field variable values of its neighbors using the filtering described in section 1.2.3 as,

$$U_i = \frac{\sum_{j=1}^{n} w_j U_j}{\sum_{j=1}^{n} w_j}, \tag{1.12}$$

2. Variable/Material Update: Once the field-variable state of each variable is defined, the design variable is updated to reflect the changes. While numerous rules are proposed in literature [15] to update design variables, a control based rule used by Patel [2] is implemented here (Figure 1-2).

The change in the design variable of i^{th} variable (Δx_i) is computed as,

$$\Delta x_i^t = K \left(U_i^t - U^* \right) / U^*. \tag{1.13}$$

where K is a scaling factor and U^* denotes the internal energy density set point. The design variable is updated as,

$$x_i^{t+1} = x_i^t + \Delta x_i^t. \tag{1.14}$$

The change in the variable is constrained by the bounds on the value of the design variable i.e.,

I. if $x_i^{t+1} < LB$, then $x_i^{t+1} = LB$,

II. if $x_i^{t+1} > UB$, then $x_i^{t+1} = UB$,

and only certain discrete values are allowed.

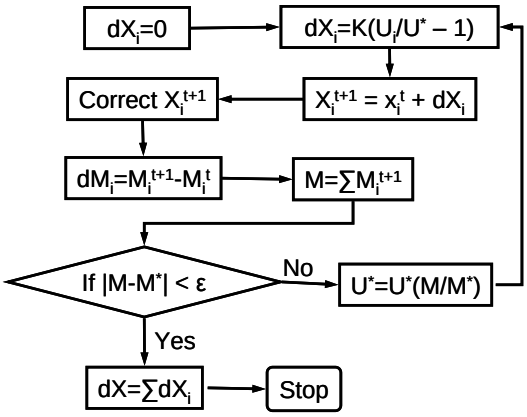

Figure 1-2: Design variable update.

The mass of each element is then calculated by using the appropriate material model associated with the design variables. If the total mass of the structure meets the constraint, the total change in design variables in this iteration is calculated, and the design variable update is considered completed. If the mass constraint is not satisfied, the IED set point is updated iteratively to accommodate the material constraint as,

$$U^* \equiv U^* = U^* M / M^*. \tag{1.15}$$

where *M* is the mass of the structure.

1.2.15. Stopping Criteria

Two termination conditions are used to stop the optimization process.

1. The number of iterations has exceeded the maximum number of iterations, or
2. The change in the topology is smaller than the tolerance, i.e.,

$$dX^t = \sum_{i=1}^{N} \Delta x_i^t \le \varepsilon. \tag{1.16}$$

The numerical oscillations in convergence are limited by averaging the total change in topology over two iterations.

1.2.16. Constrained optimization

Optimization using global variables (mass fractions and load case weights) is:

$$\min_{\boldsymbol{\xi}} f(\boldsymbol{\xi}) \text{ with } \boldsymbol{\xi} = (M_1, \dots, M_p, w_1, \dots, w_L),$$

subject to

$$g_i(\boldsymbol{\xi}) < 0 \; with \; i = 1, \dots, m$$

$$\xi_i^L \le \xi_i \le \xi_i^U$$

The derivatives with respect to the global variables can be estimated using finite differences using a multipoint scheme [22]. The Taylor expansion for a function g around a point $\boldsymbol{\xi}_0$ is simply:

$$G(\boldsymbol{\xi}) = g(\boldsymbol{\xi}_0) + \sum_{i=1}^{n} (\xi_i - \xi_{0i}) \left(\frac{\partial g}{\partial \xi_i}\right)_{\xi_0}$$

Using $F(\boldsymbol{\xi})$ and $G_i(\boldsymbol{\xi})$ as the Taylor expansion to $f(\boldsymbol{\xi})$ and $g_i(\boldsymbol{\xi})$, and the move limits $\xi_i^{L\prime}$ and $\xi_i^{U\prime}$, the optimization problem becomes:

$$\min_{\boldsymbol{\xi}} \mathrm{F}(\boldsymbol{\xi}) \text{ with } \boldsymbol{\xi} = (M_1, \dots, M_p, w_1, \dots, w_L) ,$$

subject to

$$G_i(\boldsymbol{\xi}) < 0 \; with \; i = 1, \dots, m$$

$$\xi_i^{L\prime} \leq \xi_i \leq \xi_i^{U\prime}$$

The global variable move limits $\xi_i^{L\prime}$ and $\xi_i^{U\prime}$ are centered around the optimum of the previous iteration and are chosen here as

$$\xi_i^{L\prime} = \xi_i k\left(1 - e^{-iteration/10}\right)$$

$$\xi_i^{U\prime} = \xi_i k\left(1 + e^{-\frac{iteration}{10}}\right)$$

The weight for the first load case need not be a variable, and the load case weights $(w_1, w_2, \cdots, w_L)$ variables are therefore rewritten as $(1, w_2, \cdots, w_L)$.

1.2.17. Metamodels and Numerical derivatives

Metamodels (multi-point approximations to the structural behavior) is a mature field with some highly cited papers [23]. The methodology followed here has its origin in the work of Schoofs [24] and Roozen-Kroon [25] with the added refinement of intermediate variables and responses as described by Barthelemy and Haftka [26] as described by Roux *et al* [27]. Earlier work using global variables together with local variables and responses surfaces is that of Venkataraman [28].

2. Surface Design Theory

1.3. Background

The traditional approach for solving shape design problems is based on sensitivity analysis that is inexpensive to obtain for linear static problems. However, deriving analytical sensitivities for dynamic analysis is very difficult due to the complex interactions among material nonlinearities, geometry and mesh, and transient nature of load and boundary conditions. Numerical computation of sensitivities is also not practical due to the high computational expense. Hence this approach is not practical for crashworthiness problems. To overcome the aforementioned difficulties, a different approach was proposed. This approach does not require gradients and hence there is no need to compute the sensitivities. The methodology is best referred to as LS-TaSC 3.2.

1.4. Implementation

The algorithm is shown in Figure 2-1. After defining the problem, the surface shape is evolved using the simple rules defined on the variables.

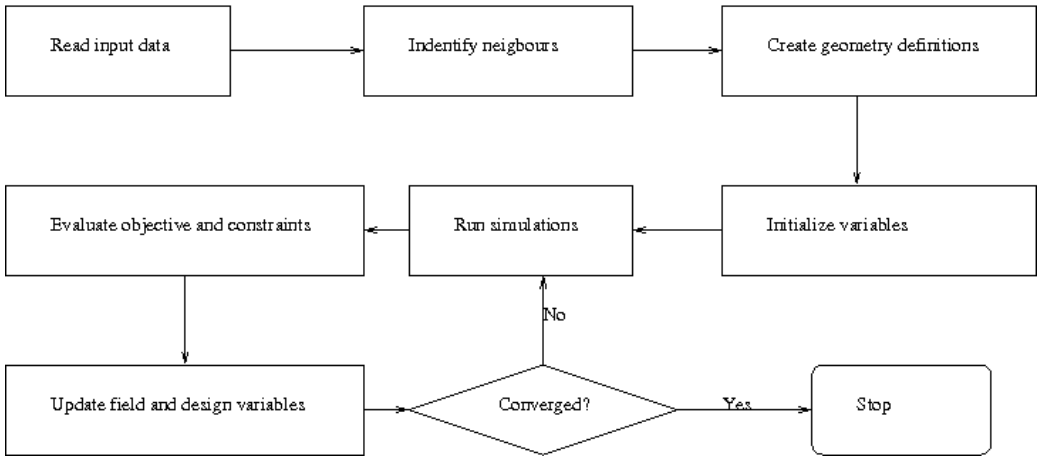

Figure 2-1: The surface design algorithm

1.4.1. Definition

The input data is used to identify the design problem. The input data comprises of method data e.g., number of iterations, convergence tolerance, and the problem data, e.g. load cases and design surface.

1.4.2. Creating the variables

The discrete surface is mapped to design variables. The normal displacement to each node in the design surface assigned to a design variable. For extrusion and symmetry constraints, the equality constraints are defined between the variables.

1.4.3. Filtering of results

A radius based strategy is used to identify neighbors. In this strategy, a virtual sphere of user-defined radius is placed at the centroids of an element. All elements that are within this sphere are considered the neighbors of the corresponding element, and the results are averaged over the elements in the neighborhood

$$U_i = \sum_{j=1}^{n} w_j U_j \Big/ \sum_{j=1}^{n} w_j . \tag{1.17}$$

1.4.4. Design Objective

The goal of shape design is to obtain surface with a uniform stress.

The optimization problem is formulated as,

$$\min_x \sum_{i=1}^{i=n} [U(x_i) - U_{target}] \tag{1.18}$$

where U represents the design field at the node associated with design variable x_i, and U_{target} represents the target value of the design field. The design field is typically the von Mises stress field.

1.4.5. Target Stress

The goal of shape design is to obtain surface with a uniform stress. In order to complete this task we need to define a target stress. There are the following possibilities of selecting a target stress:

- Average over the surface
- The maximum value on the surface
- The minimum value on the surface
- A user-defined value

Using the above target stresses it should be noted that the goal becomes more subtle than obtaining the a surface with a uniform stress: if selecting the maximum is as the target stress, then the weight of the structure will be reduced; while if the minimum is selected, then the average stress is reduced.

1.4.6. Design Variable Initialization

All design variables are also initialized to zero.

1.4.7. Simulation to Obtain Field Variables

The elements in the finite element model are modified the nodal locations for all iterations. So the input deck is re-written for all iterations. The relevant field variables for all nodes are obtained from the output to completely define the state of each variable. For multiple load case conditions, the state variable is based on the output from simulations of different load cases.

For dynamic problems, it was observed that accounting for the history of evolution induces stability by reducing the element deletion rate. Hence, the field variable (internal energy density) of i^{th} variable at iteration t is updated by defining a weighted sum on the field variable of three previous iterations as follows,

$$U_i^t = \sum_{j=0}^{3} (x_i)^{j+1} U_i^{t-j} \Bigg/ \sum_{j=0}^{3} (x_i)^{j+1}. \tag{1.19}$$

where x_i is the design variable associated with the i^{th} variable at iteration t. If the load cases are a mixture of dynamic and static problems, then this weighing is followed for all the load cases.

1.4.8. Variable Update

This is the heart of shape design method. In this step, the state of a variable is updated based on the state of its neighbors. The state update is carried out in two steps:

1. Field variable update: The field variable (internal energy density) of a variable is updated as accounting for the field variable values of its n neighbors as,

$$U_i = \sum_{j=0}^{n} U_j \Bigg/ \sum_{j=0}^{n} 1. \tag{1.20}$$

2. Variable update: Once the field-variable state of each variable is defined, the design variable is updated to reflect the changes..

The change in field value required is $\delta_i = U_i - U_{target}$. Now compute $\Delta x_i = \delta_i / \frac{\partial U}{\partial x}$ with x_i the required movement of node i normal to the surface.

1.4.9. Stopping Criteria

Two termination conditions are used to stop the optimization process.

1. The number of iterations has exceeded the maximum number of iterations, or
2. The change in the topology is smaller than the tolerance, i.e.,

$$dX^t = \sum_{i=1}^{N} \Delta x_i^t \leq \varepsilon . \tag{1.21}$$

1.5. References

[1] A. Tovar, Bone remodeling as a hybrid cellular automaton optimization process, University of Notre Dame: PhD Thesis, 2004.

[2] N. M. Patel, Crashworthiness design using topology optimization, University of Notre Dame: PhD Thesis, 2007.

[3] M. P. Bendsøe and O. Sigmund, Topology optimization: Theory, methods and applications, Heidelberg: Springer Verlag, 2003.

[4] G. I. N. Rozvany, Structural design via optimality criteria, Dordrecht: Kluwer Academic Publishers, 1989.

[5] H. A. Eschenauer und N. Olhoff, „Topology optimization of continuum structures: A review," *Applied Mechanics Reviews,* Bd. 54, Nr. 4, pp. 331-390, 2001.

[6] G. I. N. Rozvany (ed.), Topology optimization in structural mecshanics, Vienna: Springer Verlag, 1997.

[7] C. A. Soto, „Applications of structural topology optimization in the automotive industry: Past, present and future," in *V World Congress of Computational Mechanics* , Vienna, 2002.

[8] M. P. Bendsøe und K. N, „Generating optimal topologies in structural design using a homogenization method," *Computer Methods in Applied Mechanics and Engineering*, Bd. 71, pp. 197-224, 1988.

[9] C. B. W. Pedersen, „Topology optimization design of crushed 2D-frames for desired energy absorbtion history," *Structural and Multidisciplinary Optimization,* Bd. 25, Nr. 5-6, pp. 368-382, 2003.

[10] C. A. Soto, „Structural topology optimization: From minimizig the compliance to maximizing energy absorbtion," *International Journal for Vehicle Design,* Bd. 25, Nr. 1-2, pp. 142-163, 2001.

[11] C. A. Soto, „Structural topology optimization for crashworthiness," *International Journal of Crashworthiness,* Bd. 9, Nr. 3, pp. 277-283, 2004.

[12] C. B. W. Pedersen, „Crashworthiness design of transient frame structures using topology optimization," *Computer Methods in Applied Mechanics and Engineering,* Bd. 193, Nr. 6-8, pp. 653-678, 2004.

[13] J. Forsberg and L. Nilsson, "Topology optimization in crashworthiness design," *Structural and Multidisciplinary Optimization*, vol. 33, pp. 1-12, 2007.

[14] M.-K. Shin, K.-J. Park und G.-J. Park, „Optimization of structures with nonlinear behavior using equivalent loads," *Computer Methods in Applied Mechanics and Engineering*, Bd. 196, Nr. 4-6, pp. 1154-1167, 2007.

[15] M. P. Bendsøe und O. Sigmund, „Material interpolation schemes in topology optimization," *Archives of Applied Mechanics*, Bd. 69, pp. 635-654, 1999.

[16] T. Goel and W. J. Roux, "Topology optimization for designing engineering product". US Patent 8,126,684, filed April 10, 2009, and issued February 8, 2012.

[17] R. T. Haftka and Z. Gürdal, Elements of structural optimization, 3rd ed., Dodrecht, Netherlands: Kluwer Academic Publishers, 1992.

[18] S. N. Patnaik and D. A. Hopkins, "Optimality of fully-stressed design," *Computer Methods in Applied Mechanica and Engineering*, vol. 165, pp. 215-221, 1998.

[19] M. Öman, Optimization and robustness of structural product families, Linköping University: PhD Thesis, 2009.

[20] M. Öman and L. Nilsson, "Structural optimization based on internal energy distribution," *Engineering Optimization*, vol. 45, no. 4, pp. 397-414, 2013.

[21] J. O. Hallquist, LS-DYNA Manual version 971, Livermore Software Technology Corporation, October 2007.

[22] W. Roux, „Structural topology optimization using numerical derivatives". United States of America Patent 20160078161 A1 (pending), 2014.

[23] T. Simpson, J. Poplinksi, P. Koch und J. Allen, „Metamodels for computer-based engineering design: survey and recommendations," *Engineering with computers*, Bd. 17, Nr. 2, pp. 129-150, 2001.

[24] A. Schoofs, Experimental Design and Structural Optimization, Eindhoven, 1987.

[25] P. Roozen-Kroon, Structural optmization of bells, Eindhoven, 1992.

[26] J.-F. Barthelemy und H. RT, „Function Approximation," in *Structural Optimization*, Washington, AIAA, 1993, pp. 51-70.

[27] W. Roux, N. Stander und R. Haftka, „Response Surface Approximations for Structural Optimization," *International Journal for Numerical Methods in Engineering*, Bd. 42, pp. 517-534, 1998.

[28] S. Venkataraman, Modeling, analysis and optimization of cylindrical stiffened panels for reusable launch vehicle structures, University of Florida, 1999.

[29] [Online]. Available: http://mathworld.wolfram.com. [Accessed 23 March 2008].

[30] P. Hajela and B. Kim, "On the use of energy minimization of CA based analysis in elasticity," *Structural and Multidisciplinary Optimization*, vol. 23, pp. 24-33, 2001.

[31] N. M. Patel, B.-S. Kang and J. E. Renaud, "Crashworthiness design using a hybrid cellular automata algorithm," in *Proceedings of the 2006 International Design Engineering Technical Conference*, Philadelphia PA, 2006.

[32] T. G. a. W. Roux, „Topology optimization for designing engineering product". United States of America Patent 8126684 B2, 2009.