The LS-TaSC™ Tool

TOPOLOGY AND SHAPE COMPUTATIONS

THEORY MANUAL

November 2019

Version 4.1

Copyright © 2009-2019

LIVERMORE SOFTWARE

TECHNOLOGY CORPORATION

All Rights Reserved

Corporate Address

Livermore Software Technology Corporation

P. O. Box 712

Livermore, California 94551-0712

Support Addresses

Livermore Software Technology Corporation	Livermore Software Technology Corporation
7374 Las Positas Road	1740 West Big Beaver Road
Livermore, California 94551	Suite 100
Tel: 925-449-2500 ♦ Fax: 925-449-2507	Troy, Michigan 48084
Email: sales@lstc.com	Tel: 248-649-4728 ♦ Fax: 248-649-6328
Website: www.lstc.com	

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.

Any reproduction, in whole or in part, of this manual is prohibited without the prior written approval of LSTC. All requests to reproduce the contents hereof should be sent to sales@lstc.com.

25-Nov-19

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

Contents

1 Topology Theory

1.1 Background

LS-TaSC contains two approaches for the topology optimization: (i) an optimality criterion approach called Optimality Criteria for Dynamic Problems and (ii) a mathematical programming approach called Projected Subgradient Method. Constraints on the responses are handled using global variables together with a multi-tensor method is used – the multi-tensor method is an extension of the more familiar multi-point approaches modified for topology optimization. Geometry definitions are handled using equality constraints on the variables together with filtering.

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 optimality criteria used is that of having a uniform internal energy density, a condition which (as far as we can ascertain) probably goes back to before Prager's work in the 1960's. The full method also contains elements of the used fully stressed methodology together with SIMP and results filtering (this also forms the basis for the perhaps better known optimality criterion method described in Bendsøe and Sigmund [1]). The approach of designing for a uniform internal energy density was first applied to impact problems in work done at Notre Dame university (see e.g. [2] and [3]), From the start our work differed from the Notre Dame project by omitted their signature use of a cellular algorithm; instead we reverted to the older and more established knowledge by using a more traditionally filtering and adding a solid/void (SIMP) strategy, because the established methodologies have all of the benefits while carrying none of the risk. 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 [1] 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., mathematical programming methods [8], to drive the search for an optimum for large-scale topology optimization problems. 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 Overview of Topology Algorithm

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.
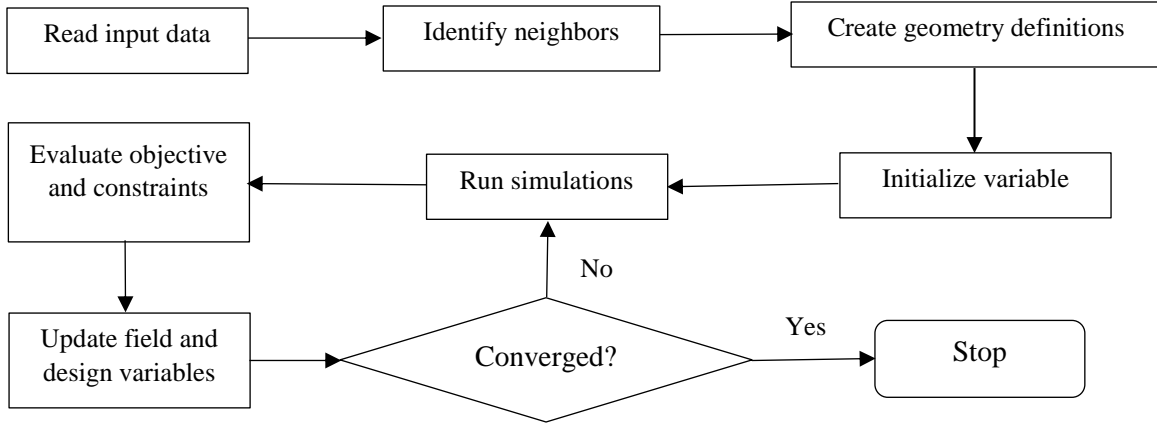
Figure 1-1: The topology optimization algorithm

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 variable 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

Filtering is a standard operation in topology optimization used to enforce a minimum feature size and to prevent an instability named checker boarding.

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

$$U_i = \frac{\sum_{j=1}^{n} w_j U_j}{\sum_{j=1}^{n} w_j},$$
(1-1)

where the subscripts i and j are, respectively, the indices of the central element and elements around the central element within the sphere. w_j are the filter weights and U_j 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-linearity 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 [1] and Patel [3]. 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].

For eigenvalue problems the method of setting the element mass to zero in sub-regions with low material density is used to prevent the localized eigenmodes. Thus, the interpolation formula for densities is parameterized as

$$\rho(x) = \begin{cases} x\rho_0 & if \ \hat{x} < x \\ x^r \rho_0 & if \ x \leq \hat{x}, \end{cases} \tag{1-6}$$

$x=0.1$, and $r=6$. Note that Equation $\rho x=x\rho 0 \quad$ *if* $x<xxr\rho 0 \quad$ *if* $x\leq x,$
($1-6$) is discontinuous at the low value $x = 0.1$ of the material density. This is not a serious problem because the discontinuity only occurs at a single point.

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 result in a target field that is less uniform over the part, because the intermediate variable values are required to obtain a uniform field.

LS-TaSC has three methods: the SIMP method, the true mechanics method, and the SIMP with continuation. In this theory manual, the SIMP and true mechanics methods are introduced. The true mechanics method only works with the internal energy density approach, while the SIMP method works well with both the internal energy density (IED) and design sensitivity analysis (DSA) approaches. The true mechanics scheme is the default in LS-TaSC, because it has proved to work well in extensive industrial testing.

1.2.5.1 Element Volume vs Material Volume

The issue is that there are two measures on energy density. The question is whether the density result 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.

1.2.5.2 SIMP Solid/Void Scheme

Solid Isotropic Material with Penalization (SIMP) forces material to 0/1 using the following equations:

$$\rho(x) = x\rho_0, \tag{1-7}$$

$$E(x) = x^p E_0, \tag{1-8}$$

$$\sigma(x) = x^q \sigma_0, \tag{1-9}$$

$$E_h(x) = x^q E_{h0}, \tag{1-10}$$

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

The use of SIMP have resulted in many problems when applied to industrial problems. For some non-linear material models the material model may not be valid for the values of *p* and *q*. 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. Also, for NVH computations, the speed of sound may be lowered so much that it interferes

with the physics under consideration. The displacements and energies of the models will also be very high during initial iterations because the scaling of the stiffness with the $E(x) = x^p E_0$ term. For a mass fraction of 0.3, this may result in a factor 10 difference; for a mass fraction of 0.01, the potential difference grows to an astonishing 10000.

Because of the above, the SIMP scheme is not recommended for industrial problems, specifically problems involving highly nonlinear mechanics and constraints.

1.2.5.3 True Mechanics Solid/Void Scheme

The standard academic research on the topic, specifically the ubiquitous SIMP scheme, is not derived to allow the robust computation of highly nonlinear behavior. The structural mechanics cannot be disregarded, because this will yield the various instabilities described in the previous section. The true mechanics scheme for solid/void scheme borrows from the mathematical field of dynamical systems [17] to both respect the structural mechanics of the problem and to force elements with small variable values to exit the equations.

$$\rho(x) = x\rho_0, \tag{1-11}$$

$$dF/dx = x^p dF_0(x)/dx, \tag{1-12}$$

$$E(x) = xE_0, \tag{1-13}$$

$$\sigma(x) = x^q \sigma_0, \tag{1-14}$$

$$E_h(x) = x^q E_{h0}, \tag{1-15}$$

This gives the desired solid/void behavior if p is taken to be 1. This scheme is the default in LS-TaSC, because it has proved to work well in extensive industrial testing.

1.2.6 Design Objectives and Constraints

The typical goal of topology optimization is to obtain the layout of a structure; the structure supporting a load with a better structure needing less material to do so. Compliance is the most commonly used objectives for linear static problems. For dynamic problems, like crashworthiness simulations, the structure needs to absorb the energy while maintaining the structural integrity and keeping the peak loads transmitted to the occupants low.

In LS-TaSC the layout of the structures is designed such as to maximize the stiffness, while other objectives and constraints such as the amount of energy absorbed and the amount of deflection is controlled by global variables such as the part mass fractions and load case weights. In theory any kind of response can be designed for, but in practice sophisticated responses requires sophisticated use of the global variables such as subdivided the part into a number of parts.

1.2.7 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.

1.2.8 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® [18]. 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 ith element at iteration t is updated by defining a weighted sum on the field variables of the current and three previous iterations as follows,

$$U_i^t = \sum_{k=0}^{3}(x_i)^{k+1}U_i^{t-k}/\sum_{k=0}^{3}(x_i)^{k+1} , \tag{1-16}$$

where x_i is the design variable associated with the ith 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.

1.3 Global Constraint Handling using Control Theory

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.

$$M^* = M^* + \Delta M ,$$
$$\Delta M = \left(\sum_{j} K_j^c \varepsilon_j \right) / J , \tag{1-17}$$

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

1.4 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_l of load case l is adjusted to change constraint C_l. The target value is computed as

$$C_{target} = \frac{\sum_{l=1}^{L}(k_l C_l + offset_l)}{L}, \tag{1-18}$$

from which we compute

$$\Delta w_i = (C_{target} - k_i C_i - offset_i)/(0.1 C_{target}). \tag{1-19}$$

and a maximum bound of $0.05 w_l$ is placed on Δw_l to ensure convergence in a reasonable number of iterations.

1.5 Optimality Criteria for Dynamic Problems

1.5.1 Design Objectives

The objective is that obtaining the stiffest structure for a given mass. These requirements yields the optimality criteria of designing for a uniform internal energy density. This approach to design dates back to at least work by Venkayya et al. [19] and is adopted by many others. The history of these developments can be found in the book by Rozvany [4]. This concept is similar to the fully-stressed design and uniform strain energy density approaches, see e.g. Haftka and Gürdal [20] or Patnaik and Hopkins [21], which are well established in literature for linear-static problems.

The optimization problem is formulated as

$$\min_{x} \sum_{i=1}^{N} \sum_{l=1}^{L}(w_l U_l(x_i) - U_l^*), \tag{1-20}$$

subject to,

$$\begin{aligned} \sum_{i=1}^{N} \rho(x_i) V_i &\leq M^* \\ x_{min} \leq x_i &\leq 1.0 \end{aligned}. \tag{1-21}$$

where $U_l(x_i)$ represents the internal energy density of the ith element in the lth load case, V_i is the volume of ith element, and U_l^* represents internal energy density set point in the lth load case. The superscripts 'low' and 'up' represent lower and upper bounds on the constraints, respectively.

The multipoint scheme can be used to introduce other objectives and constraints such as the maximization of energy absorption and limits on displacements or peak forces.

To understand how these optimality criteria were derived consider saying a situation consisting of perfectly plastic flow under the condition of constant energy (typical of impact problems). Consider a fiber A with a plastic strain larger than that at another fiber

B. The plastic work done will depend only on the strain, because the yield stress is constant. Therefore moving material from B to A will try to increase the total plastic work, because the material will absorb more energy at A than B. But the total energy is required to be constant, therefore the strains at A must be reduced in this new configuration. There may be a small corresponding increase in the strain at B, but overall the load resistance of the part increases – because more energy will be absorbed for the same displacement (strain field). Given that the total work is constant, the displacement (plastic strain level) must decrease, and the outcome is that of a structure with better load resistance. This is frequently expressed as an optimality criterion stating that the optimal part has a uniform energy density. The knowledge is due to a previous generation of engineers – research ceased long ago in favor of mathematical programming methods using derivatives.

Considering the objective functions, the optimality criteria methods and the mathematical programming methods should yield similar designs, because both essentially solve the equation describing the energy in the structure. Differences in designs will however occur due to (i) local minima found by the solvers, which are quite probable for highly nonlinear and other tough problems; and (ii) the projected subgradient method being a more powerful method.

1.5.2 Update Rules

This is the heart of the optimality criteria 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,

$$U_i = \frac{\sum_{j=1}^n w_j U_j}{\sum_{j=1}^n w_j}, \tag{1-22}$$

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, the fully stressed method used by Patel [3] is implemented here (Figure 1-2).

The change in the design variable of ith variable (Δx_i) is computed as,

$$\Delta x_i^t = K \left(U_i^t - U^* \right) / U^*. \tag{1-23}$$

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

$$x_i^{t+1} = x_i^t + \Delta x_i^t. \tag{1-24}$$

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.
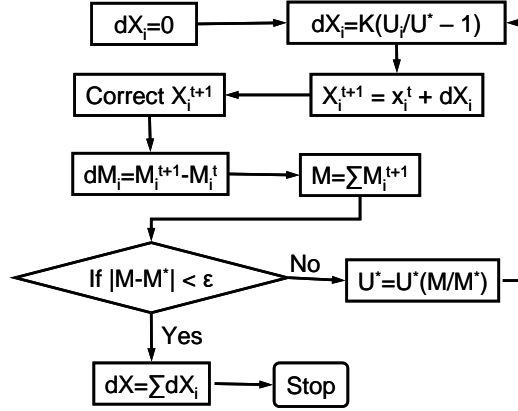
Figure 1-2: Design variable update.

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 done using bisection method.

1.6 Projected Subgradient Method

This method works for combined impact, NVH, and linear load cases. It is the only available method for the fundamental eigenvalue problems.

The method is similar to the steepest descent method which has an update written as

$$x^{t+1} = x^t - \alpha^t d^t, \tag{1-25}$$

where d^t is the descent direction that leads optimum searching, which usually refers to the derivative vector of the objective with respect to the design variables in unconstrained optimization. α^t is the desired step size. The design search vector between two iterations is represented as $\Delta x^t = -\alpha^t d^t$. The difference is that the projected subgradient decent method has a fixed step size and that the computations are projected onto a plane.

1.6.1 Design Objectives and Constraints

The optimization problem is formulated as

$$\min_{x} f(\boldsymbol{x}), \qquad (1\text{-}26)$$

subject to,

$$
\begin{aligned}
&\sum_{i=1}^{N} \rho(x_i)V_i \leq M^* \\
&C_j^{low} \leq C_j \leq C_j^{up}, j = 1, 2, \cdots, J. \\
&x_{min} \leq x_i \leq 1.0
\end{aligned}
\qquad (1\text{-}27)
$$

where $f(\boldsymbol{x})$ represents the objective, V_i is the volume of ith element, and C_j is the jth constraint. The superscripts 'low' and 'up' represent lower and upper bounds on the constraints, respectively.

The objective for the NVH load case is to maximize the fundamental frequency, while for the impact and linear statics load cases the objective is essentially to maximize the stiffness of the structure; for linear material behavior the objective is actually the minimization of the work done which is equivalent to minimizing the compliance, while for plastic flow of the material the design process will result in smaller displacements and/or higher forces but not necessarily different energy absorption. The search directions for the fundamental frequency is the derivative with respect to the element variables as described in the standard literature on design sensitivity analysis. The search directions for the impact and static load cases are computed from the internal energy densities of the elements as described in a later section.

The multipoint scheme can be used to introduce other objectives and constraints such as the maximization of energy absorption and limits on displacements or peak forces.

Considering the objective functions, the optimality criteria methods and the mathematical programming methods should yield similar designs, because both essentially solve the equation describing the energy in the structure. Differences in designs will however occur due to (i) local minima found by the solvers, which are quite probable for highly nonlinear and other tough problems; and (ii) the projected subgradient method being a more powerful method.

1.6.2 Design Variable Update

In the projected subgradient method, the design search vector is projected onto the plane of an inequality structural constraint, so that the constraint function is satisfied with the update of the design. Assume that normal vector of the plane of the constraint function is presented as $\boldsymbol{n}$. The design search vector projected onto the constraint plane can be expressed as,

$$\Delta \boldsymbol{x}^P = \Delta \boldsymbol{x} - \left(\frac{\Delta \boldsymbol{x} \cdot \boldsymbol{n}}{(|\boldsymbol{n}|)^2} \right) \boldsymbol{n}, \qquad (1\text{-}28)$$

where $\Delta \boldsymbol{x}^P$ is the design change vector after the projection onto the constraint function. Besides the constraint function, the side bounds on the design variable should be taken into consideration as well, as the side bounds on the design variables may cause the

computations to be off. In order to correct for the effect of the side bounds of the design variables, a parameter λ is introduced to compensate the side bound effect on the updated design. The design search vector is thus expressed as,

$$\Delta \tilde{x}^P = \Delta x - (1 + \lambda)\left(\frac{\Delta x \cdot n}{(|n|)^2}\right) n, \tag{1-29}$$

where $\Delta \tilde{x}^P$ is the design change vector after considering the side bound affects. In the above equation, the physical meaning of λ is to move the plane of the constraint function up and down, so that the constraint function is satisfied with updated designs within the side bounds. The value of λ can be positive, zero, or negative depending on how much the constraint plane should be moved. λ is typically found using a bisection algorithm such as to satisfy the constraint function.

Therefore, the updated design in the projected subgradient method is presented as,

$$x^{t+1} = x^t + \Delta \tilde{x}^P. \tag{1-30}$$

Due to the compensation of the side bound affects in the computation, the above updated design may exceed the range of the side bounds. Thus the updated design should be trimmed so that all the design variables have values within the range of the side bounds. The final updated design is obtained as

$$x_i^{t+1} = \begin{cases} x_{min} & if \ \ x_i^t + \Delta \tilde{x}_i^P \leq x_{min} \\ x_i^t + \Delta \tilde{x}_i^P & if \ x_{min} < x_i^t + \Delta \tilde{x}_i^P < 1.0 \ . \\ 1.0 & if \ \ x_i^t + \Delta \tilde{x}_i^P \geq 1.0 \end{cases} \tag{1-31}$$

1.6.3 Step Size and Scaling of the Gradients

For the class of problems considered here, there is a natural choice of step size: the amount of material allowed to flow during an iteration. This means the step size depends on the mechanics of the problem and not the number of variables. Numerical concerns such as the mesh size therefore do not affect the step size.

To implement material flow as controlling the allowable step size requires that the difference between two sequential designs in computation should be considered. The material flow is a scaled version of the $L1$ norm of the variable changes,

$$m_f = \frac{1}{N}\sum_{i=1}^{i=N}|\Delta x_i|, \tag{1-32}$$

while the step size is the L2 norm of the variable changes,

$$s = \sqrt{\sum_{i=1}^{i=N}(\Delta x_i)^2}, \tag{1-33}$$

where N is the total number of elements in the structure. m_f represents the $L1$-norm-based mass flow and s is the step size. The $L1$ norm and $L2$ norm of the variable changes are

equal if all the variable changes have the same absolute value. In such a case, a material flow of $\overline{m_f}$ requires an absolute change of $\overline{m_f}$ for each variable, given that the material flow is,

$$m_f = \frac{1}{N}\sum_{i=1}^{i=N}|\Delta x_i| = \frac{1}{N}\sum_{i=1}^{i=N}\overline{m_f} = \overline{m_f}, \tag{1-34}$$

For this case, the step size is therefore computed as

$$s = \sqrt{\sum_{i=1}^{i=N}(\Delta x_i)^2} = \sqrt{\sum_{i=1}^{i=N}\overline{m_f}^2} = \sqrt{N}\overline{m_f}, \tag{1-35}$$

Thus, different from the linear-search-based step sizes in the steepest decent method, a constant step size is used in the projected subgradient method.

In order to improve the computational stability, the value of N should be taken as the number of grey elements, N_g in the design space, where the grey elements means the associated design variables of elements not at the side bounds. Besides the mass flow of the grey elements, the gradients of the objective must be scaled to its norm such that it matches the step size. Thus, the step size in the computation is obtained as,

$$\alpha^t = \frac{s^t}{|d^t|} = \sqrt{N_g^{(t)}} \frac{\overline{m_f}}{|d^t|}. \tag{1-36}$$

1.6.4 Search Direction (Subgradients) for Impact and Statics Problems

The sequence of design improvements is found by understanding that a structure resists a load and that a better structure expends less energy doing so.

A design is therefore improved by a change that reduces the energy – which is implemented very efficiently by constructing a search direction from the internal energy densities. The requirement of using the internal energy density originally came from rigorous considerations of linear problems. Introducing nonlinear effects such as impact and plastic flow, being more difficult to handle rigorously, complicates matters, leaving us with the statement the design improvements actually maximizes load resistance (stiffness).

This use of the internal energy density in optimization, its relationship with the design sensitivity information for crash problems, and its usefulness for ranking variables have been studied by Öman and others [22], [23] at the Linköping research group. It is important to note that for topology optimization it is the ranking of the variables that are important, and not quite the values of the derivatives, because the addition of the solid/void scheme makes it a question of which variables are important and not a question of how much of each variable to use.

For linear problems the internal energy densities therefore comprise the best search direction considering the analytically correct solution, while for nonlinear problems the internal energy densities are known to be a good, but not necessarily the best, search direction given extensive numerical experiments, engineering knowledge, and the linear derivations. For an intuitive understanding of the engineering mechanics see section 1.2.8.

The topology design process therefore always increases load resistance, but whether the energy absorption increase or decrease depends on loading conditions such as whether it is an applied load, applied displacement, or prescribed initial velocity problem, together with impact and material yield slope effects.

The noise in the explicit dynamics problems make the use of subgradients attractive over the use of gradients information. Again the use of the internal energy densities is beneficial.

1.7 Multidisciplinary Optimization

The typical goals of topology optimization are to maximize stiffness or the fundamental frequency as described in Section 1.6.1.

These goals can be combined into a multidisciplinary design optimization problem as

$$\min_{x} \quad \sum_{l=1}^{L} w_l f_l(\boldsymbol{x}), \tag{1-37}$$

1.7.1 Auxiliary Variables for Multiple Eigenvalues

In the NVH problems, a problem exists when the eigenmodes swaps between iterations. For example, the first eigenmode in the current iteration can become the second eigenmode in the next iteration. This causes a convergence issue to the topology design optimization of the NVH problems.

This comes from the optimization problem which is

$$\max_{x}[min[\lambda_1, \lambda_2, \cdots, \lambda_n]], \tag{1-38}$$

This is problematic when one wishes to compute the derivatives of the objective. The auxiliary variables are used to resolve the derivative of the objective. This is shown below using only two frequencies. Hence the optimization problems is rewritten as,

$$\min_{\mu,x} \mu, \tag{1-39}$$

subject to

$$1/\lambda_1 - \mu \leq 0,$$
$$1/\lambda_2 - \mu \leq 0, \tag{1-40}$$
$$\mu > 0.$$

In the above formulations, one auxiliary variable μ, and n Lagrange variables are included.

However, the above scheme of using auxiliary variables cannot be used in the projected subgradient descent method. Instead we can solve for the derivatives of the objective with known step size. With the step size, we know which frequencies can be active (i.e., starting swapping). An auxiliary variable w, and variations of design variables, Δx, are introduced to maximize the lowest eigenfrequency. The frequency for the next iteration is estimated by using a Taylor series expansion around the current point and from

that a weighting of the eigenvalues can be computed to yield the best update. So we solve the below optimization problem,

$$\max_{w} \left[\min \left[\lambda_1 + \sum_{i=1}^{N} \Delta x_i \frac{\partial \lambda_1}{\partial x_i}, \lambda_2 + \sum_{i=1}^{N} \Delta x_i \frac{\partial \lambda_2}{\partial x_i} \right] \right]. \tag{1-41}$$

where $\Delta x_i = k[w \frac{\partial \lambda_1/\partial x_i}{\|\partial \lambda_1/\partial x_i\|} + (1-w) \frac{\partial \lambda_2/\partial x_i}{\|\partial \lambda_2/\partial x_i\|}]$ with k chosen such that $\|\Delta x_i\| = s$ with s the step size and $0 \le w \le 1$.

1.7.2 Design Contribution Histories and Plots

The design contribution of a load case specifies the amount of material added to design for that specific load case. For histories this is a fraction of the total material, therefore a design contribution of 0.75 indicates that 75% of all the mass were added to design for that specific load case. The fringe plots of the design contributions on the other hand show where in the structure where in the structure the material was added for the load cases.

To compute, note that the design update for the whole structure is composed from the individual contribution as

$$\boldsymbol{g} = \sum_{l=1}^{L} w_l \frac{\boldsymbol{g}_l}{\|\boldsymbol{g}_l\|}, \tag{1-42}$$

with $\boldsymbol{g}_l$ and L representing the gradient for the l-th load case and the total number of load cases, respectively.

The mass flow (mass redistribution) for the whole structure is summed from the contributions of the elements as

$$\dot{m} = \frac{\sum_{i=1}^{N} \|\Delta x_i\| v_i}{\sum_{i=1}^{N} V_i}, \tag{1-43}$$

with each element's contribution Δx_i computed from $\boldsymbol{g}$.

The mass flow for a specific load case is taken as

$$\dot{m}^{case} = \frac{\sum_{i=1}^{N} \|\Delta x_i^{case}\| v_i}{\sum_{i=1}^{N} V_i}. \tag{1-44}$$

with $\Delta x_i^{case} = \frac{w^{case} g_i^{case}}{\sum_{l=1}^{L} w^l g^l} \Delta x_i$.

The mass contribution of a load case is summed over the design iterations as

$$M^{case} = \sum_{itrations} \dot{m}^{case}, \tag{1-45}$$

The design contribution for the case can now be computed as

$$D^{case} = \frac{M^{case}}{\sum_{l=1}^{L} M^{l}},\tag{1-46}$$

The history plots available include:
1) $\dot{m}$: the mass flow (also in previous versions);
2) $\dot{m}^{case}$: the mass flow for the case;
3) D^{case}: the design distribution of each case;

The fringe plots are similar, except that it is reported per element.

For element e at the variable change for the final iteration is

$$\Delta x_i^{case,final} = \sum_{iteration=1}^{final} \Delta x_i^{case,iteration},\tag{1-47}$$

Therefore several load cases can contribute to the material in an element. Only the ones that significantly contribute are plotted in the fringe plots. These load cases are those which

$$\Delta x_i^{case,final} \geq \frac{1-massfraction}{L+1}.\tag{1-48}$$

The part fraction is included because that is the starting value for the design variables.

1.8 NVH Problems

1.8.1 DSA of Eigenvalues

The generalized eigenvalue problem can be formulated as

$$\left(K - \mu_j M\right)\psi_j = 0,\tag{1-49}$$

where ψ_j denotes the eigenmode corresponding to μ_j. Consequently, the mode shapes are orthonormalized with respect to the linear mass matrix such that

$$\psi_j M \psi_k = \delta_{jk},\tag{1-50}$$

Differentiating equation with respect to a design variable $x_i \in x$ $(i = 1, \cdots, N)$ yields

$$\left(K - \mu_j M\right)\frac{\partial \psi_j}{\partial x_i} - \frac{\partial \mu_j}{\partial x_i} M \psi_j = -\left(\frac{\partial K}{\partial x_i} - \mu_j \frac{\partial M}{\partial x_i}\right)\psi_j,\tag{1-51}$$

In what follows, it is assumed that the mass as well as the stiffness matrices are smooth functions of the design variables. If an eigenvalue is distinct or unimodal, i.e. $\mu_{j-1} < \mu_j < \mu_{j+1}$, the corresponding eigenvector ψ_j will be unique (up to a sign) and differentiable with respect to the design variables. The sensitivity of an eigenvalue μ_j with respect to an $-\mu j M \partial \psi j \partial x i - \partial \mu j \partial x i M \psi j = -\partial K \partial x i - \mu j \partial M \partial x i \psi j,$ (*1-51*) by ψj and invoking $-\mu j M \psi j = 0,$ (*1-49*) and $\psi j M \psi k = \delta jk,$

      (*1-50*) to obtain

$$\frac{\partial \mu_j}{\partial x_i} = \psi_j^T \left(\frac{\partial \mathbf{K}}{\partial x_i} - \mu_j \frac{\partial \mathbf{M}}{\partial x_i} \right) \psi_j. \tag{1-52}$$

Assuming all design variables are changed simultaneously a linear increment in the single eigenvalue can be found as

$$\Delta \mu_j = \frac{\partial \mu_j}{\partial x_i} \Delta x_i, \tag{1-53}$$

$-\mu jM\psi j=0,$ (*1-49*) yields s repeated eigenvalues, i.e. $\bar{\mu} := \mu_l = \cdots = \mu_m$ and $s = m - l + 1$, the eigenvalues are not Fr´echet differentiable and $-\mu jM\partial\psi j\partial xi-\partial\mu j\partial xiM\psi j=-\partial K\partial xi-\mu j\partial M\partial xi\psi j,$ (*1-51*) and $\mu j\partial xi=\psi jT\partial K\partial xi-\mu j\partial M\partial xi\psi j.$ (*1-52*) are no longer valid. To find sensitivities of repeated eigenvalues, Seyranian et al. [24] used directional derivatives in the design space. As a result, the increment in the multiple eigenvalue $\Delta \bar{\mu}$ is obtained as the solution of an s-dimensional sub-eigenvalue problem, i.e.,

$$det(\mathbf{F} - \Delta \bar{\mu} \mathbf{I}) = 0, \tag{1-54}$$

and the elements of the matrix $\mathbf{F}$ are defined as

$$\mathbf{F}(j - l + 1, k - m + 1) = \frac{\partial \bar{\mu}_{jk}}{\partial x_i} \Delta x_i \quad \text{with } j, k = l, \cdots, m, \tag{1-55}$$

where $\frac{\partial \bar{\mu}_{jk}}{\partial x_i}$ are the generalized gradient vectors defined in component form as

$$\frac{\partial \bar{\mu}_{jk}}{\partial x_i} = \psi_j^T \left(\frac{\partial \mathbf{K}}{\partial x_i} - \bar{\mu} \frac{\partial \mathbf{M}}{\partial x_i} \right) \psi_k. \tag{1-56}$$

$det\mathbf{F}-\Delta\mu\mathbf{I}=0,$ (*1-54*) is also symmetric. Considering $\mu j=\partial\mu j\partial xi\Delta xi,$ (*1-53*), two special cases are $\mu j=\partial\mu j\partial xi\Delta xi,$ (*1-53*) reduces to Equations $-\mu jM\partial\psi j\partial xi-\partial\mu j\partial xiM\psi j=-\partial K\partial xi-\mu j\partial M\partial xi\psi j,$ (*1-51*) in the unimodal case, i.e. $l = m$ and $s = 1$. Second, if all off-diagonal elements of the matrix $\mathbf{F}$ are zero, the sub-eigenvalue problem becomes determined as

$$\Delta \mu_j = \frac{\partial \bar{\mu}_{jj}}{\partial x_i} \Delta x \quad \forall j = l, \cdots, m. \tag{1-57}$$

and hence $\frac{\partial \bar{\mu}_{jj}}{\partial x_i} \equiv \frac{\partial \mu_j}{\partial x_i}.$

$-\mu jM\psi j=0,$ (*1-49*) is performed, computing the $\mu j\partial xi=\psi jT\partial K\partial xi-\mu j\partial M\partial xi\psi j.$ (*1-52*) and $det\mathbf{F}-\Delta\mu\mathbf{I}=0,$ (*1-54*) hold on the element level. Furthermore, the element stiffness and mass matrices are linear function of the (interpolated) elastic modulus and material density, respectively. Consequently, their

Ke∂xi=pxip−1Ke(E0),   (*1-58*) and *∂Me∂xi=Meρ0if xi<xirxir−1Me(ρ0)if xi≤xi,*   (*1-59*) as

$$\frac{\partial K^e}{\partial x_i} = p x_i^{p-1} K^e(E_0),$$
(1-58)

$$\frac{\partial M^e}{\partial x_i} = \begin{cases} M^e(\rho_0) & if\ \hat{x}_i < x_i \\ r x_i^{r-1} M^e(\rho_0) & if\ x_i \le \hat{x}_i' \end{cases}$$
(1-59)

where the superscript *e* designates the standard element matrices. Since the element matrices are computed using the penalized material properties, i.e. *E* and *ρ* in LS-DYNA, *Ke∂xi=pxip−1Ke(E0),*   (*1-58*) and *∂Me∂xi=Meρ0if xi<xirxir−1Me(ρ0)if xi≤xi,*   (*1-59*) in terms of these *x=xρ0 if x<xxrρ0 if x≤x,*   (*1-6*) and equations in Section 1.2.4 yields

$$\frac{\partial K^e}{\partial x_i} = p x^{-1} K^e(E),$$
(1-60)

$$\frac{\partial M^e}{\partial x_i} = \begin{cases} x_i^{-1} M^e(\rho) & if\ \hat{x}_i < x_i \\ r x_i^{-1} M^e(\rho) & if\ x_i \le \hat{x}_i \end{cases}.$$
(1-61)

Consequently, the information exchange between LS-DYNA and LS-TaSC reduces to the two scaling terms being input to LS-DYNA and the $s(s+1)/2$ gradients being input to LS-TaSC.

1.8.2 Mode Tracking

Design optimization of eigenfrequencies of a structure is to improve the eigenfrequencies of interest by changing the structural design at every iteration. However, due to changes of the structural design, all other eigenfrequencies are also affected. Some eigenmodes will be inevitably switched order in terms of optimizing eigenfrequency. For the case where eigenmode of the structure matters, it is better to keep tracking the eigenmode corresponding to the eigenfrequency of interest, such as bending mode and torsion mode, during the entire optimization process.

Model Assurance Criterion (MAC) has been used in LS-TaSC for mode tracking, and it compares the reference mode shapes to the new mode shapes from the iterative design by computing the below MAC value,

$$MAC(\varnothing_r, \varnothing_i) = \frac{|\varnothing_r^T \varnothing_i|^2}{(\varnothing_r^T \varnothing_r)(\varnothing_i^T \varnothing_i)}.$$
(1-62)

where $\varnothing_r$ is the eigenmode of interest, also reference mode, and $\varnothing_i$ is the eigenmode of the new design. MAC relies on comparing the shape of two eigenmodes, and the MAC-value represents the projection of the new mode shape on the reference mode shape. This value equals to 1 for aligned vectors and 0 for orthogonal vectors.

1.8.3　Frequency Constraints

A subgradient update scheme is described in below sections to solve NVH problems of a structure with frequency constraints while using Projected Subgradient Method.

1.8.3.1　Searching Direction Update Scheme

The main idea of the search direction update scheme is to integrate gradients of frequency constraints into the subgradients with weighing multipliers, so that the new subgradients ensure the descendancy of the objective value as well as the satisfactory of the constraints at the same time.

Assume that a total of m frequency constraints are considered, and all the frequency constraints are written into a standard form of

$$g_j \leq 0, with \, j = 1, \cdots, m, \tag{1-63}$$

∇g_j represents the derivative vector of the j-th frequency constraint. g_{max} represents the maximum constraint value among all constraints, which means that this particular constraint is either the most unsatisfied (with a positive value) or the least inactive (with a negative value). ∇g_{max} refers to the corresponding gradients of g_{max}.

With ∇g_{max}, an approximation of the constraint related to g_{max} is conducted by assuming the search direction is led only by the objective gradients in the next iteration, and the approximated value of g_{max} is expressed as,

$$g_{max,app} = g(x(\nabla f)), \tag{1-64}$$

where ∇f is the derivative vector of the objective with respect to the design variables. Thus, a possible movement of the constraint related to g_{max} between the feasible and infeasible domains can be predicted according to the signs of g_{max} and $g_{max,app}$. Based on different sceneries of this movement prediction, the searching direction d^t for the NVH problems with frequency constraints can be updated based on below schemes.

If $g_{max} > 0$ and $g_{max,app} > 0$, the searching direction is pointing from the INFEASIBLE domain to the INFEASIBLE domain. The searching direction of the optimization process will be updated by

$$d^t = \nabla f^t + \lambda(\nabla g_{max})^t, \tag{1-65}$$

If $g_{max} > 0$ and $g_{max,app} \leq 0$, the searching direction is pointing from the INFEASIBLE domain to the FEASIBLE domain. The searching direction of the optimization process will be updated by

$$\boldsymbol{d}^t = \nabla f^t + \sum_{j=1}^{m} \lambda_j (\nabla g_j)^t, \qquad\qquad (1\text{-}66)$$

If $\mathrm{g}_{max} \leq 0$ and $g_{max,app} > 0$, the searching direction is pointing from the FEASIBLE domain to the INFEASIBLE domain. The searching direction of the optimization process will be updated by

$$\boldsymbol{d}^t = \nabla f^t + \sum_{j=1}^{m} \lambda_j (\nabla g_j)^t, \qquad\qquad (1\text{-}67)$$

If $\mathrm{g}_{max} \leq 0$ and $g_{max,app} \leq 0$, the searching direction is pointing from the FEASIBLE domain to the FEASIBLE domain. The searching direction of the optimization process will be updated by

$$\boldsymbol{d}^t = \nabla f^t. \qquad\qquad (1\text{-}68)$$

where $\boldsymbol{\lambda_j}$ denotes the weighing multiplier of the *j*-th constraint. The solution of weighing multipliers of frequency constraints is described in the following section.

1.8.3.2 Solution of Weighing Multipliers

The updated searching direction of the optimization process should satisfy the feasible condition as follows,

$$\begin{aligned}
&\sum_{j=1}^{m} \mu_j \rightarrow min, \\
s.t.\ &\mu_j = -\left(\nabla g_j\right)^T \boldsymbol{x}(\boldsymbol{d}(\boldsymbol{\lambda}), \alpha) - g_j \geq 0, \qquad (1\text{-}69)\\
&\lambda_j \geq 0.
\end{aligned}$$

The
j=1mµj→min,s.t. µj=−∇gjTx(dλ,α)−gj≥0,λj≥0.   (*1-69*) can be solved by using a linear programming method, e.g. Big-M Simplex algorithm.

1.8.3.3 Step Size Scaling Scheme

Following with the searching direction update, the step size will be updated accordingly by following the step size estimation described in Equation $\alpha^t = \dfrac{s^t}{|\boldsymbol{d}^t|} = \sqrt{N_g^{(t)}} \dfrac{\overline{m_f}}{|\boldsymbol{d}^t|}.$

(*1-36*). However, this step size should be scaled, otherwise it may cause severe oscillations in eigenfrequencies due to overshooting. Thus, a scheme of iteratively driving a scale of 1.0 gradually approaching to a proper scale value of step *j=1mµj→min,s.t. µj=−∇gjTx(dλ,α)−gj≥0,λj≥0.*   (*1-69*).

The step size scaling scheme is based on the rate of the mass flow drawn from the original subgradient and step size to the updated subgradient and step size. Assume that the original subgradient and step size before considering frequency constraints are represented as $\boldsymbol{d}_{orig}$ and α_{orig}, respectively, where $\boldsymbol{d}_{orig}$ and α_{orig} are computed with the objective gradients as described in Section 1.6. The tempera subgradient and step size after

considering frequency constraints are denoted as $\boldsymbol{d}_{temp}$ and α_{temp}, respectively. Two mass flow values can be evaluated as below

$$mf_1 = mf(\boldsymbol{d}_{temp}, \alpha_{temp}),\qquad(1\text{-}70)$$

$$mf_2 = mf(\boldsymbol{d}_{orig}, \alpha_{orig}).\qquad(1\text{-}71)$$

where "mf" denote the mass flow evaluation function. mf_1 is the mass flow evaluated from the tempera subgradient and step size that are meant to be updated. mf_2 is the mass flow evaluated with the original subgradient and step size. The ratio of the above two mass flow evaluations is expressed as

$$\eta = \frac{mf_2}{mf_1},\qquad(1\text{-}72)$$

Then, update the tempera step size as below

$$\alpha_{temp} = \eta * \alpha_{temp}.\qquad(1\text{-}73)$$

j=1mμj→min,s.t. μj=−∇gjT$\boldsymbol{x}$(dλ,α)−gj≥0,λj≥0.   (*1-69*),   the constraint weighing multipliers will be updated, so will the subgradient $\boldsymbol{d}_{temp}$. With the updated $\boldsymbol{d}_{temp}$ and α_{temp}, the mass flow will be evaluated again. This procedure will repeat until the mass flow ratio equals to 1.0, and the final $\boldsymbol{d}_{temp}^*$ and α_{temp}^* will be used in the following optimization.

1.9 Constrained optimization using the Multipoint Scheme

1.9.1 Problem Formulation

This formulation allows any response to be used as an objective or constraint. The mass fractions and load case weights are used as global variables and their values are selected to minimize the objective and satisfy the constraints. The topology of the parts are however still computed using the mass fraction and load case weights to create the stiffest part. The optimization problem using the global variables is:

$$\min_{\xi} f(\xi) \; with \; \xi = (M_1, \dots, M_p, \; w_1, \dots, w_L),\qquad(1\text{-}74)$$

subject to

$$g_i(\xi) < 0 \; with \; i = 1, \dots, m,\qquad(1\text{-}75)$$

$$\xi_i^L \leq \xi_i \leq \xi_i^U$$

1.9.2 Numerical Derivatives

The derivatives with respect to the global variables can be estimated using finite differences using a multipoint scheme [25]. The Taylor expansion for a function g around a point $\boldsymbol{\xi}_0$ is simply:

$$G(\boldsymbol{\xi}) = g(\boldsymbol{\xi}_0) + \sum_{i=1}^{n} (\xi_i - \xi_{0i}) \left(\frac{\partial g}{\partial \xi_i} \right)_{\xi_0}. \tag{1-76}$$

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 $\xi_i^{L'}$ and $\xi_i^{U'}$, the optimization problem becomes:

$$\min_{\boldsymbol{\xi}} F(\boldsymbol{\xi}) \; with \; \boldsymbol{\xi} = (M_1, \dots, M_p, \; w_1, \dots, w_L), \tag{1-77}$$

subject to

$$G_i(\boldsymbol{\xi}) < 0 \; with \; i = 1, \dots, m, \tag{1-78}$$

$$\xi_i^{L'} \le \xi_i \le \xi_i^{U'}.$$

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

$$\xi_i^{L'} = \xi_i k \left(1 - e^{-\frac{iteration}{10}} \right), \tag{1-79}$$

$$\xi_i^{U'} = \xi_i k \left(1 + e^{-\frac{iteration}{10}} \right). \tag{1-80}$$

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 $(1, w_2, \cdots, w_L)$.

1.9.3 Metamodels

Linear metamodels is one of the option for estimating the numerical derivatives. Metamodels (multi-point approximations to the structural behavior) is a mature field with some highly cited papers [26]. The methodology followed here has its origin in the work of Schoofs [27] and Roozen-Kroon [28] with the added refinement of intermediate variables and responses as described by Barthelemy and Haftka [29] as described by Roux *et al* [30]. Earlier work using global variables together with local variables and responses surfaces is that of Venkataraman [31].

1.10 Stopping Criteria

Three 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.,

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

<div align="right">(1-81)</div>

3. The numerical oscillations in convergence are limited by averaging the total change in topology over two iterations.
4. Solidification is defined to measure the discreteness of the optimized designs. A higher Solidification value indicates better topological designs. Assuming the total number of design variables as $N = N_{void} + N_{grey} + N_{solid}$, we define Solidification

$$M = min(M_1, \, M_2).$$

<div align="right">(1-82)</div>

where $M_1 = \frac{N_{void} + N_{solid}}{N}$, and $M_2 = 1 - \frac{\sum_{i=1}^{N} 4x_i(1-x_i)}{N}$. A default Solidification value of 0.95 is used in LS-TaSC.

Normally, both of maximum number of iterations and topology change are used as termination conditions in Optimality Criteria algorithm; both of maximum number of iterations and solidification are used as termination conditions in Projected Subgradient algorithm.

2 Surface Design Theory

2.1 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.

2.2 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.
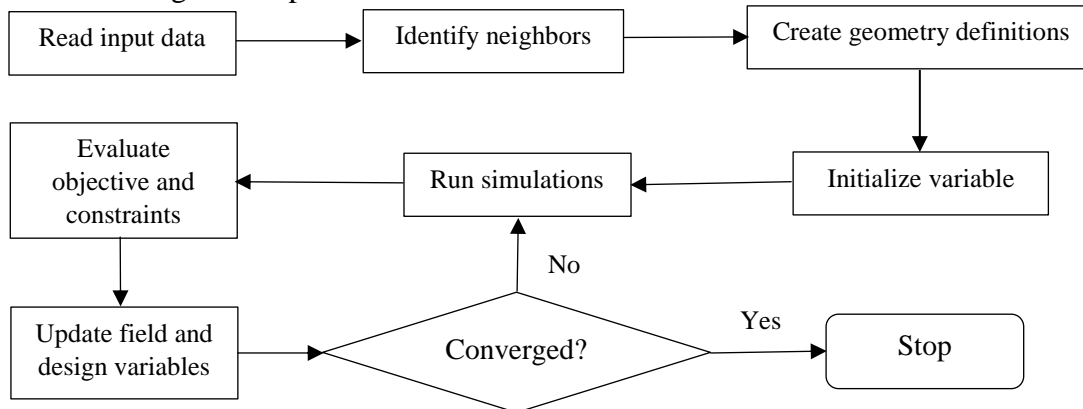

Figure 2-1: The surface design algorithm

2.2.1 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.

2.2.2 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.

2.2.3 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

$$U_i = \sum_{j=1}^{n} w_j U_j \Big/ \sum_{j=1}^{n} w_j.$$
(2-1)

2.3 Design Objective

The goal of shape design is to obtain surface with a uniform stress. The optimization problem is formulated as,

$$\min_{x} \sum_{i=1}^{i=n} [U(x_i) - U_{t\arg et}].$$
(2-2)

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

2.4 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.

2.5 Design Variable Initialization

All design variables are also initialized to zero.

2.6 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 ith variable at iteration t is updated by defining a weighted sum on the field variable of three previous iterations as follows,

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

(2-3)

where x_i is the design variable associated with the ith 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.

2.7 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,

$$U_i = \sum_{j=0}^{n} U_j \bigg/ \sum_{j=0}^{n} 1.$$

(2-4)

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 $\delta_i = U_i - U_{t\,\arg et}$. Now compute $\Delta x_i = \delta_i / \dfrac{\partial U}{\partial x}$ with x_i the required movement of node i normal to the surface.

2.8 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.,

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

(2-5)

References

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

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

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

[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. S. a. O. S. S. Rojas Labanda, „Mathematical programming methods for large-scale topology optimization problems," *DTU Wind Energy,* 2015.

[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] H. M. a. S. S, Differential rquations,dynamical systems, and linear algebra, San Diego: Academic press, inc, 1974.

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

[19] V.B. Venkayya,N. S. Khot and V. S. Reddy, „Energy distribution in an optimal structural design," *AFFDL-TR-68-156,* 1968.

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

[21] 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.

[22] M. Öman, Optimization and robustness of structural product families, Linköping University: PhD Thesis, 2011.

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

[24] E. L. a. N. O. A.P. Seyranian, „Multiple eigenvalues in structural optimization problems," *Structural Optimization,,* Bd. 8, pp. 207-227, 1994.

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

[26] 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.

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

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

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

[30] 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.

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

[32] F. J. a. N. L., „Topology optimization in crashworthiness design," *Structural Multidisciplinary Optimization,* Bd. 33, pp. 1-12, 2007.

[33] X. Y. M. H. A. a. S. G. P. Chu D. N., „Evolutionary structural optimization for problems with stiffness constraints," *Finite Elements in Analysis and Design,* Bd. 21, p. 239–251, 1996.

[34] B. P. P. N. M. a. R. J. E. Mozumder C. K., „Thickness based topology optimization for crashworthiness design using hybrid cellular automata," in *12th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, Victoria, 2008.

[35] R. W. a. S. N. Goel T., „A topology optimization tool for LS-DYNA users: LS-OPT/Topology," in *7th European LS-DYNA Conference*, Salzburg, 2009.

[36] 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.

[37] 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.

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