10th World Congress on Structural and Multidisciplinary Optimization
May 19 -24, 2013, Orlando, Florida, USA

Efficient Robust Shape Optimization for Crashworthiness

Milan Rayamajhi[1], Stephan Hunkeler[1] and Fabian Duddeck[1,2]

[1]Queen Mary University of London, UK, m.rayamajhi@qmul.ac.uk
[2]Technische Universität München, Germany, duddeck@tum.de

Abstract
The work presented here focuses on the reduction of computational effort associated with Robust Design Optimization (RDO) for crashworthiness of large industrial models. For the robust design study, the computational effort is determined by the following two aspects: *(i) time for the computation of a single crash event* and *(ii) the number of evaluations required for the robustness of each design*.
For the first aspect, the possibility of using physical surrogate models (sub-structures or linear models obtained via the Equivalent Static Loads Method (ESLM)) to reduce the single simulation time is investigated. The sub-structuring approach is beneficial in situations where only a small part or sub-structure of a large complex structure has to be optimized to improve performance of the overall structure. ESLM generates the same deformation fields by linear static analysis as from the non-linear dynamic analysis and enables thereafter faster computations. Through these two approaches, the physical characteristics can still be represented sufficiently well even though the numerical effort for each design evaluation is reduced remarkably. The gain in computational time allows for the realization of a “true” robust design optimization, a so called double-loop approach, where in the optimization loop an additional loop is embedded for stochastic analysis to assess the robustness. This leads, in contrast to a single loop approach where robustness is only controlled at the end of the optimization, to optima under the robustness constraint.
For the second aspect, a Modified Double Loop RDO (MDLRDO) approach has been implemented for the first time where the robustness analysis, through non-linear dynamic analysis, is only made for designs that are in a special Target Interval (TI). Robustness analysis, through non-linear dynamic analysis, is only made on special design points, which are both *feasible* and in a certain respect *optimal*(designs closer to the boundaries of the feasible design space). For the design points outside TI, robustness analysis is approximated through linear static analysis using ESLM. Through this approach the number of expensive non-linear dynamic analysis required in a RDO can be significantly reduced and large industrial problems can be optimized rarely achieved before.
The three approaches, sub-structuring, ESLM and MDLRDO, have been successfully validated. In the next step these three approaches were combined to create an efficient robust design optimization loop. This approach is first applied to a robust design optimization problem considering uncertainties of thickness parameters only and is extended then to include variations in shape parameters and impact conditions.

Keywords: Crashworthiness, robust design optimization, shape optimization, sub-structuring, equivalent static loads method, design uncertainty, numerical effort.

1. Introduction

Recent studies for crashworthiness have been realized using more and more complex geometries and detailed finite element modeling. Such model complexity with the required high non-linearity (material, geometry and contact) leads to high numerical effort for simulation, which is in particular challenging for optimization studies for industrial-sized problems with their high number of design variables, constraints and objectives, e.g. **[1].** Standard approaches have therefore only a restricted ability to explore sufficiently well the design space, which becomes even more crucial in cases where the robustness of the derived optima has to be assured (robustness in the means of insensitivity of the design with respect to inevitable fluctuations in design and noise variables). Nevertheless an approach where robustness analysis is included into the optimization loop is necessary because the design is normally driven to the limits during an optimization. The objective of the study at-hand is therefore to investigate methods to reduce computational effort for single simulations and also the number of design evaluations required for robust design optimization such that a real robust optimization studies can be established for crashworthiness.

2. Robust Design Optimization Through Physical Surrogates

In the literature, e.g. **[2]**, robust design optimization has been based on surrogate modeling techniques using response surface approaches to replace finite element computations by evaluations on mathematical functions. In contrast to this, here it is explored how physical surrogate models can be used, i.e. models based on a concise sub-structuring and ESLM. By these approaches, the physical characteristics can be still represented sufficiently well even though the numerical effort for each design evaluation is reduced remarkably. Sub-structuring is particularly beneficial in situations where only a small part or sub-structure of a large complex system has to be optimized to improve performance of the overall structure, see **Figure 2**.

The gain in computational time allows for the realization of a true RDO, a so-called double-loop approach where in the optimization loop an additional loop is embedded for stochastic analysis to assess the robustness. This leads, in contrast to a single loop approach where robustness is only controlled at the end of the optimization, to optima under the robustness constraint. In addition to the surrogate approaches, a modified double loop RDO approach has been implemented for the first time where the explicit robustness analysis is only made for designs that are in a special Target Interval (TI), see **Figure 3**. Explicit robustness analysis is only made on special design points, which are both *feasible* and in a certain respect *optimal* (designs closer to the boundaries of the feasible design space). For the designs outside the TI, assumed robustness values can be used or, better, surrogate models (physical or mathematical) are used for these values. At present the ESLM, **[3]**, is under investigation for the latter approach. Through this approach the number of explicit design evaluations required in a RDO can be significantly reduced and large industrial problems can be optimized rarely achieved before.

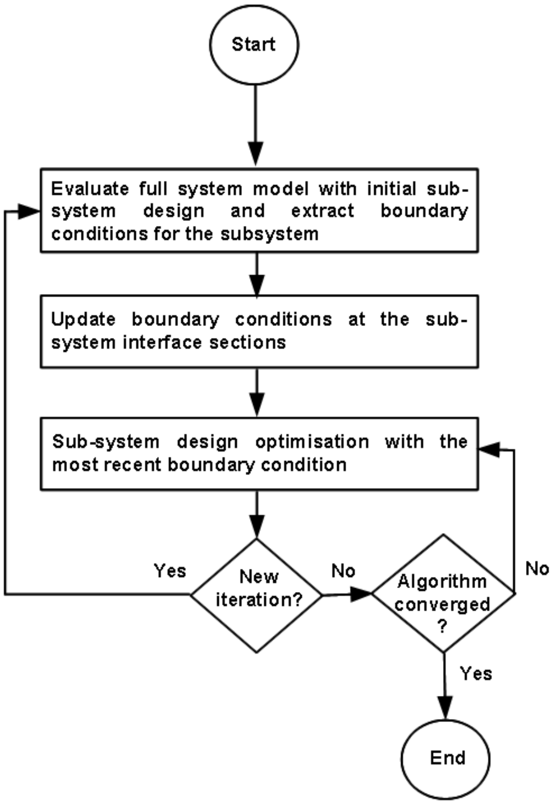

Figure 1: Sub-structure optimization loop.

2.1. Sub-structuring

A sub-structure method is useful where only a sub-region of a computationally expensive problem is of interest for the optimization study. This means that we are interested in optimization problems where the design variables are only defined in a smaller part of the total structure. The usage of sub-structures for crash optimization is rarely discussed in the literature. The only comparable work found so far in the literature was realized by people from Red Cedar Technology,**[4]**, although the publications do not discuss thoroughly careful validation or documentation.

The sub-structure optimization approach presented here is based on the idea that the sub-structure and the remaining structure are coupled to each other. Hence the interaction between the sub-structure and the remaining structure is taken into consideration, in the optimization process, by controlled updating of the boundary conditions at the interface of the sub-structure; see **Figure 1** and **Figure 2**, c.f. also to **[4]**. Here a nodal based interface boundary condition is used which is updated at each new iteration.

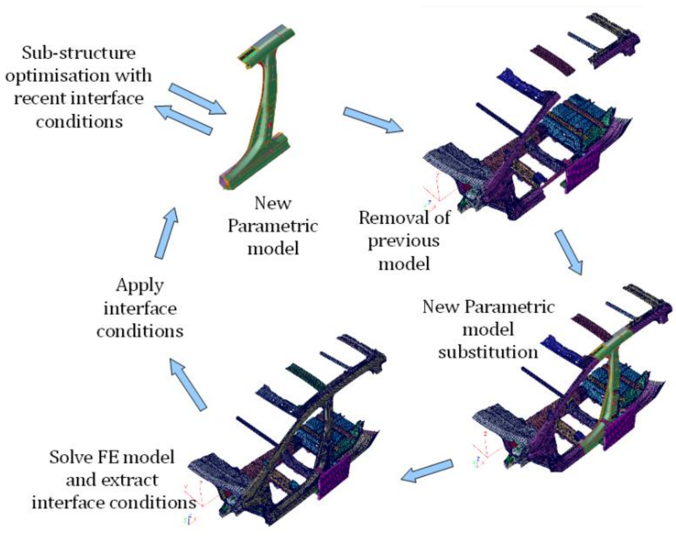

Figure 2: Sub-structure and interface condition update loop.

As an example for the sub-structuring process, a half car FE model is taken into consideration, see **Figure 2**. The goal of this study is to optimize the performance of the B-pillar in the event of lateral crash. Hence the presentation and the discussion of the sub-structure approach are based on this example.

Proposed Sub-structure Optimization Approach

Here we propose a continuous sub-structure optimization (as opposed to start/stop optimization in **[4]**) through an automatic boundary condition update criteria. The algorithm considered here is the Iterative Response Surface Method(IRSM) whose settings are explained in paragraphs below.

Definition of the Sub-structure

As an example for the industrial optimization problem, the parameterized B-pillar including the bottom and the top part are defined here as the sub-structure, **Figure 2**. We have to make sure that this sub-structure is sufficiently large such that it is reasonable to assume that the interface values between the sub-structure and the rest structure do not vary too strongly during optimization. On the other-hand, the sub-structure is small enough to increase numerical efficiency. To assure that the simulation of the sub-structure is sufficiently correct, the boundary nodes of the sub-structure are subjected to interface conditions mimicking the deformation behavior of the total structure. These interface conditions are obtained from an initial simulation of the full structure, in which the SFE CONCEPT B-pillar has replaced the original PSA[1] B-pillar. A special connection process was used here to connect the parametric B-pillar model to the remaining PSAFE model (here the option "FE CONNECT" of SFE CONCEPT was used). The interface between the parametric B-pillar model and the PSA FE model is located at the sections where the model was cut out from the main structure to generate the SFE CONCEPT model, see **Figure 2**. In our study, the interaction between the model and sub-model is taken into account using the imposed displacement on the nodes of the sub-structure at the interface sections. The interaction values are crucial for the correctness of the simulations and have to be taken into consideration in the optimization. Depending on the interaction intensity between the remaining structure and the sub-structure, slight changes made to the sub-structure, during optimization, could influence the performance characteristics of the full structure.

Sub-structure Optimization Loop

In the sub-structure optimization loop, the B-pillar reinforcement shape is changed, according to the shape parameter records in SFE CONCEPT, and the sub-structure is optimized. During the sub-structure optimization, the interface conditions will lose their correctness as long as they are not updated by a computation within the full structure. Hence, the modified sub-structure should be inserted again, after a certain optimization phase, into the rest of the structure to obtain the new interface boundary condition values. The corresponding insertion and evaluation process is shown in **Figure 2**.

Iterative Response Surface Settings

To realize the sub-structure approach for optimization, IRSM is considered here. At first the focus was set to test the sub-structure approach through a continuous optimization loop rather than the start-stop method implemented in **[4]**. The IRSM optimization was modified to support the sub-structure approach.

For the optimization to be continuous, the update of the boundary conditions had to be considered. The update in a continuous optimization loop will bring in various uncertainties in the performance of the optimization since the updated interface boundary condition will confuse the optimization algorithm. Therefore either an algorithm that can handle and is aware of these external influences or an algorithm that does not sustain information from iterations to iterations is required. The second option is preferable as the possibility to write a whole new algorithm is avoided. The idea here is to introduce the boundary condition update at the start of each new iteration. This way the algorithm, IRSM, takes the external influence as the effect of the new design parameters in the new design sub-region.

This is why the IRSM method was chosen since the information from one iteration to another is not sustained except for the sub-space optimum. Also the already available information on the optimum of each iteration is of importance. To get such information, additional modifications have to be made to other algorithms such as evolutionary algorithms (EA)[2]. The significance of the sub-space optimum is that the update in the boundary condition is made to this design at each iteration. The sub-structure optimization loop is given in **Figure 1**.

The interface condition update criterion at present is dependent on the new iteration. This is not the best update criterion since the chance of missing an update point within the iteration is high (depending on the amount of parameter changes). Also making too many unnecessary updates can increase the computational effort due to the

[1]PSA Peugeot and Citroën.

[2]E.g. the use of external script to find the optimum of each generation.

need of full model evaluations for interface boundary condition extraction. Therefore other measures for updates have to be investigated and implemented in the future.
Also the applicability and the generality of the method have to be analyzed. The sub-structure optimization method can be generally applied to any problems that fall in the category defined in **Section 2.1**. However the optimization algorithm at present is limited to the use of IRSM. Hence the possibility to adapt this method to other algorithms should be looked at in the near future. Due to limitation on the length of work to be presented here the reader is directed to **[5]** for the application of this approach to an industrial problem.

Conclusion
The sub-structuring technique is promising for the problem at-hand to reduce the simulation time of each design in robust design optimization. However for the half car model the simulation time was reduced from 12 hours to 3 hours. In our case the computational effort for the robust design optimization has to be reduced further to be able to investigate more design cases. Hence we present in the next section an approach to reduce the overall computation effort for RDO.

2.2. Modified Double Loop RDO (MDLRDO)

Method
Generally for RDO a double loop approach is considered where the robustness analysis is embedded within the optimization loop to analyze the robustness behavior of each optimization point. Crashworthiness optimization is already very demanding due to the expensive nature of crash simulation. Combining optimization and robustness in a single loop adds significantly to the overall computational effort. Also the computational effort is wasted when robustness analysis is made on optimization points that are infeasible solutions. Hence modifications can be made to this approach where robustness analysis is only made on the selected optimization points that are both feasible and near-optimal. In this study the double loop approach is modified such that the robustness analysis is only made on some points to reduce the computational effort.
In a constrained optimization problem the design space can be divided into feasible and infeasible regions. Promising solutions and also the near-optimal solutions lie within the feasible design space. The designs in the infeasible region are of no significant interest in-terms of robustness since they have already failed to satisfy the performance criterion. Hence the extra computational effort should not be spent on the evaluation of robustness for these infeasible designs. The performance of this approach however depends on the type of optimization algorithm used. If response surfaces are used for RDO then the robustness evaluation of all the points in the design space are necessary. This is because these robustness points are required to build the response surfaces μ and σ(mean and standard deviation) for the robustness behavior. Avoiding the robustness of infeasible designs will reduce the number of points used to generate these response surfaces (especially at the beginning of the optimization since lots of designs may be infeasible) and hence reduce the quality of the generated response surfaces. Therefore the proposed RDO approach is investigated using an Evolutionary Algorithm (EA).

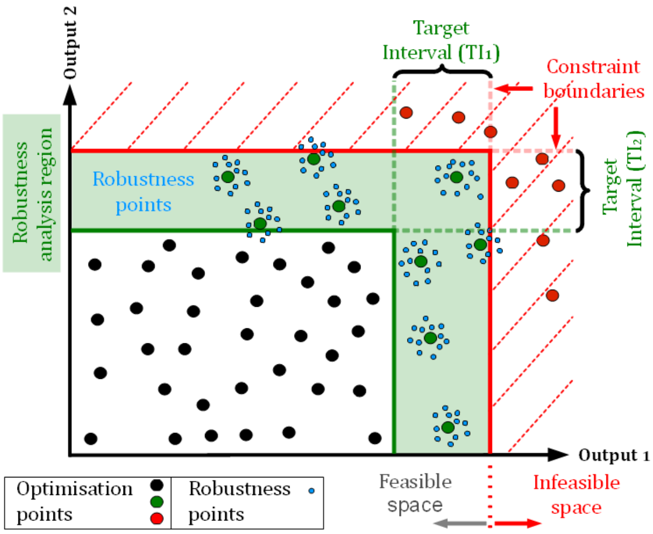

Figure 3: Target Interval (TI) definition for robustness analysis.

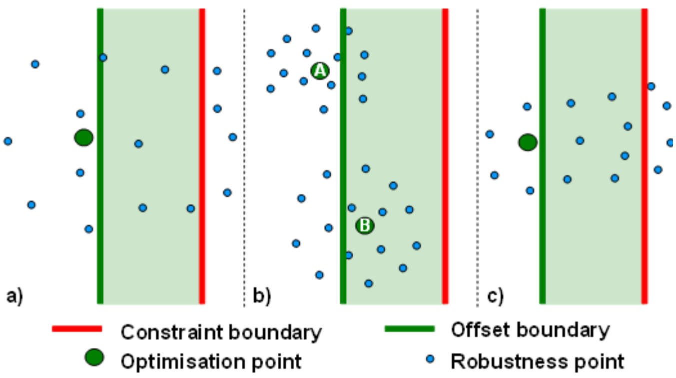

Figure 4: Some cases to consider for dummy robustness values.

For the proposed RDO approach, the feasible design space is further divided into two regions; the designs that are close to the feasible region boundary and the designs that are far away from the feasible region boundary. Since the optimization tries to push the designs to the boundary of the feasible design space, the sub-optimum solutions lie within this region. Because we want designs that are optimal and robust, we only evaluate the robustness of these near-optimal designs. For this an offset distance has to be created for the constraint boundary. The space between the constraint boundary and its offset defines the TI for the robustness evaluation, see **Figure 3**. In The **Figure 3**, the blue dots represent the values used for the robustness analysis of the green center points. For the designs that are within the TI (**Figure 3**, green dots), true robustness analysis is made whereas for designs that are outside the TI (**Figure 3**, black and red dots), dummy robustness values are created. To generate the dummy robustness values, the output values of the optimization points are used. Robustness outputs are created around the optimization output using uniform distribution.

Although the designs outside the TI are not evaluated for robustness analysis, these design should however give a direction to the algorithm for the better regions of the design space. Also the designs evaluated for robustness (inside TI) should provide enough information for the algorithm search direction for robust optimal region. The proposed approach should reduce the computational effort for RDO significantly especially at the beginning of the optimization where lots of designs may be infeasible or non-optimal. Hence most of the robustness analysis will be performed only at the end of the optimization when the algorithm has nearly converged and is based on a population of mostly sub-optimal solutions. There are certain specifics to this approach which are discussed in the following paragraphs.

Implementation

Dummy robustness values

Dummy robustness values are created for designs that are outside the TI. These dummy robustness values are generated using the nominal value of the optimization point and a constant sigma value. The generated dummy robustness values are uniformly distributed around the optimization point. In a RDO the designs are ranked in-terms of both the mean and the standard deviation(sigma) values. Here since the sigma value is the same for all designs that are outside the TI, these designs are ranked depending only on the nominal values. This assists the algorithm to make decisions for design selections for future generations depending on real outputs rather than dummy outputs.

The size of the dummy sigma values plays a vital role for the robustness characterization of the designs outside the TI. If the dummy sigma value is too big then in a situation where a design which is in the feasible space but outside the TI may have some of the robustness points ending up in the infeasible space making the design non-robust, see **Figure 4a**. This is not true since dummy values are used. Also the dummy sigma value must be bigger than the true sigma values to avoid exchange of the designs at the offset boundary making the non-robust design robust. For example if at the boundary of the offset there are two designs; design A: outside TI and design B: inside TI whose objective values are similar. If design A has a smaller sigma than design B, then design A may be taken as the robust optimum in comparison to design B. This is however not correct due to the use of dummy robustness values, see **Figure 4b**. Hence a preliminary robustness study on some designs or experience can be used to assign dummy sigma value.

Also the distribution of dummy robustness outputs is important. They have to be well distributed around the optimization point for correct characterization of

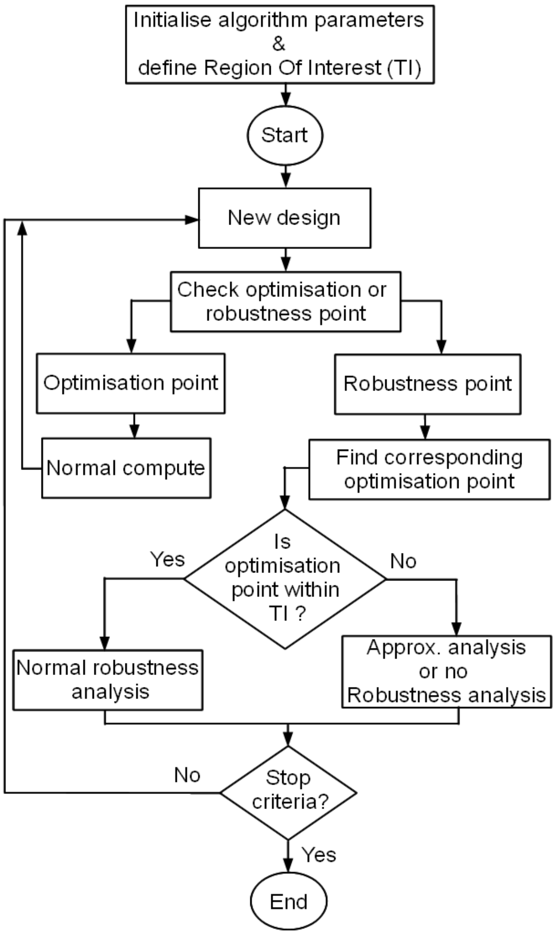

Figure 5: Flowchart for MDLRDO approach.

robustness, see **Figure 4c**. Uniform distribution or normal distributions of dummy outputs around the design can be made.

Constraints

The offset distance of the constraint has to be made depending on how active the constraint is in the optimization. The more active the constraint, the smaller the offset distance should be made so the number of robustness analysis made is reduced. Also multiple constraints have to be considered. This is useful in cases where one output is within the TI and the other output is outside the TI. In such cases, the constraints can be scaled in-terms of their importance. And the decision can be made for the robustness analysis depending on the driving constraint. Another approach would be to combine the constraints using their importance factor.

Different offsets can be formulated for different constraints depending on their activeness in the optimization. Hence a constraint with active behavior would have a small offset whereas a constraint with less activity will have bigger offset. If this approach is implemented, a criterion can be set such that all the constraint offsets have to be satisfied for a design to qualify for the robustness analysis. To make a decision on how active the constraint is and what offset to take, a preliminary study can be made on few designs[3]. A general rule for the offset distance is given in Eq. (1), where *F* is dummy value and *R* is real value.

$$\text{Offset distance} > 2\sigma_F > 2\sigma_R \qquad (1)$$

Sample recycling

Sample recycling cannot be used in this approach due to the generated dummy values for robustness. If these dummy values are chosen as the recycled samples for the next optimization then the evaluated robustness will become incorrect.

Do not solve duplicate designs

The "do not solve duplicate designs" (or similar options in any other optimization package) has to be disabled. This is because the generated dummy robustness point output values may be used as an actual optimization point or robustness point output or vice versa making the robustness evaluations incorrect.

Adaptive TI

As the optimization algorithm converges, most of the designs fall within the TI, see **Figure 6**. This means that the number of evaluations required at the end of the RDO increases with this approach. To address this issue, an adaptive approach to the TI can also be implemented, which changes the TI as the optimization proceeds. Such adaptation of the TI helps to reduce the number of robustness evaluations at the end of the optimization.

To implement the adaptation of the TI all designs have to be monitored in-terms of their location (within or outside the TI), their robustness values and also the information on the optimum so-far achieved. Once this information is available, the number of robust designs that are within the TI, for each generation, is known. If the number of robust designs within the TI is less than 20 %- 50 % of the population then the TI can be increased by taking a bigger offset distance. Also the optimum so far has to be taken into account while adjusting the TI to make sure that the TI is not too small.

Hence the size of the TI should be maintained at 2 or 3 times the size of the real sigma value at all times. Such size of the TI accommodates the robust designs that are close to the constraint boundaries.

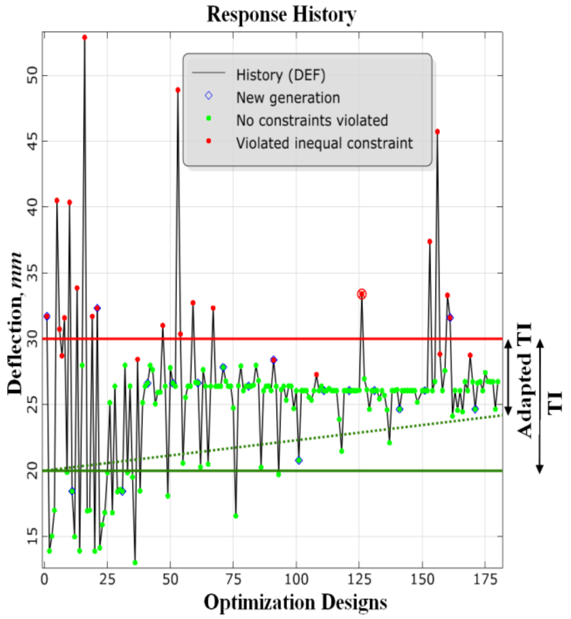

Figure 6: Deflection output history and the imposed offset distance (TI).

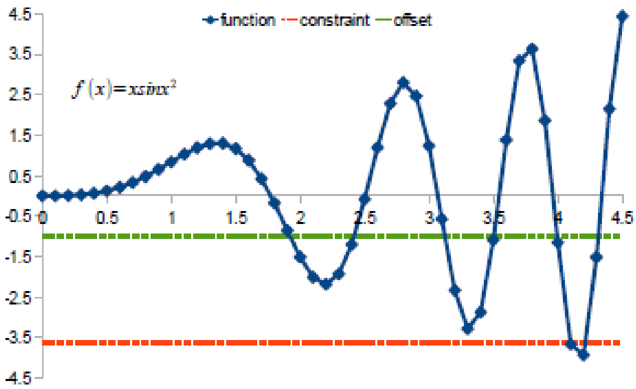

Figure 7: Analytical function to test the MDLRDO.

[3]The offset distance is related to the sensitivity of the designs to the uncertainties.

Validation

To validate the proposed robust design approach, different analytical functions and simple impact cases were used. Some results are presented in the following paragraphs.

Analytical function tests

Several analytical test functions were used to validate the MDLRDO approach. One of the analytical functions is shown in **Figure 7**. The function has 3 local and 1 global minima. A constraint on the function is set at $f(x) = -3.5$ which eliminates the global minimum at $x = 4.2$. Hence now the global optima is at $x = 3.3$.However this peak is relatively narrow and hence we expect this minimum to be non-robust. This leaves a robust optimum at $x = 2.2$. The TI is defined by creating an offset boundary at $f(x) = -1$.

Different RDO runs for the function were made using a different starting point. The algorithm managed to find the robust optimum at $x = 2.2$ in most of the cases. This was also the case for other test functions used.

Small impact case

For test and validation purpose, a very simple impact case is used which is computationally cheap, simulation time approx. 2 minutes, **Figure 8**. The beam is formed of two similar profiles which are laser welded together with reinforcement inside the beam. The optimization parameters are the thickness of the individual profiles and the thickness and position of the reinforcement within the beam. The objective is to reduce the overall mass of the beam while respecting the deflection constraint, $d \leq 30$ *mm*. An offset is created for the deflection at $d = 20$ *mm*, creating a TI with size 10 *mm*, see **Figure 7**. The design variables and the uncertainty parameters are listed in **Table 1**.

Table 2 compares the parameters and the outputs of the initial and the optimum designs. The mass and the deflection have improved by 21.7% and 28.6% respectively. The optimum was found after 2249 evaluations at the 14th iteration (149th design). **Figure 6** shows the deflection history and the TI. It can be seen from **Figure 6** that at the beginning of the optimization many designs are outside the TI. This is because the optimization algorithm is exploring the design space. However after about the 4th iteration most of the designs fall within the TI, hence we entered the exploitation stage of the algorithm. In this case the computational efficiency was roughly 28% since out of 2249 designs, 645 designs were not evaluated but instead dummy values were generated. This was because 43 designs out of the 178 designs were outside the TI. Hence, as mentioned in Paragraph (*adaptive TI*), an adaptive TI can be implemented to further reduce the computational effort.

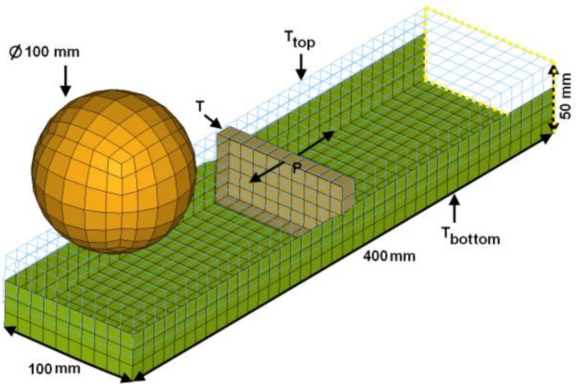

Figure 8: A simple beam deflection case.

Table 1: Design and noise variables overview.

	Variable	Lower bound	Mean	Upper bound
Thickness variables	T_{top}(*mm*)	0.5	1.0	2.0
	T_{bottom}(*mm*)	0.5	1.0	2.0
	T (*mm*)	0.5	1.0	2.0
Position variable	P (*mm*)	-20	0	20
Noise variables	Vo (ms^{-1})	2.12	2.22	2.32
	Mo (*kg*)	7.9	8.0	8.1

Table 2: Comparison of the initial design and the robust design using MDLRDO.

Initial Design		Robust Design					
Outputs		Parameters				Outputs	
Mass (*kg*)	Deflection (*mm*)	T (*mm*)	TB (*mm*)	TT (*mm*)	P (*mm*)	Mass (*kg*)	Deflection (*mm*)
1.2	37.4	0.5	1.16	0.5	-15	0.94	26.7

Conclusions

The modified double loop RDO approach was successfully implemented. This was then evaluated with some analytical functions and a simple impact case. Through these test cases, the approach has managed to reduce the overall computational effort associated with robust design optimization. As discussed earlier the efficiency of the approach can be enhanced by implementing an adaptive TI approach. One of the major difficulties faced with this approach was to approximate the TI which is directly related to the sensitivity of the designs to the uncertainties and also the activeness of the constraints. Although this can be judged through some design runs before the optimization however to exactly formulate the rule for each optimization case is difficult. Hence, rather than using the dummy robustness values for the robustness analysis of the designs outside the TI, the approach could benefit from the use of other approximation methods such as ESLM, **[3]**.The substitution of this approximation method adds the physical relevance to the overall approach, which could not be provided by the dummy robustness values.

2.3. Equivalent Static Loads Method

Crash simulations through Finite Element Analysis (FEA) are expensive due to the requirement to capture the true behavior (dynamic and non-linear) of crash events. Considering dynamic loads and non-linearity is expensive because the responses are evaluated in the time domain. However during optimization the requirement to precisely evaluate designs may not always be required. For example, approximate evaluations during the start of the optimization and accurate evaluations during the end of the optimization. Since at the start of the optimization the optimization algorithm is still exploring the design space, the design analysis accuracy is not required. At this stage enough information to guide the optimization algorithm towards the promising regions of the design space should be of interest. This can be achieved through some approximation approach such as ESLM. In ESLM the characteristics of crash, such as deformation and contact, which are time dependent are ignored. Hence the solution from such approach cannot be used as a precise analysis but as an approximation to the dynamic non-linear analysis case. In this study the ESLM is proposed for the analysis of the design outside the TI. Through this approach, the optimization algorithm will have enough information for the direction of the robust optimal regions in the design space.

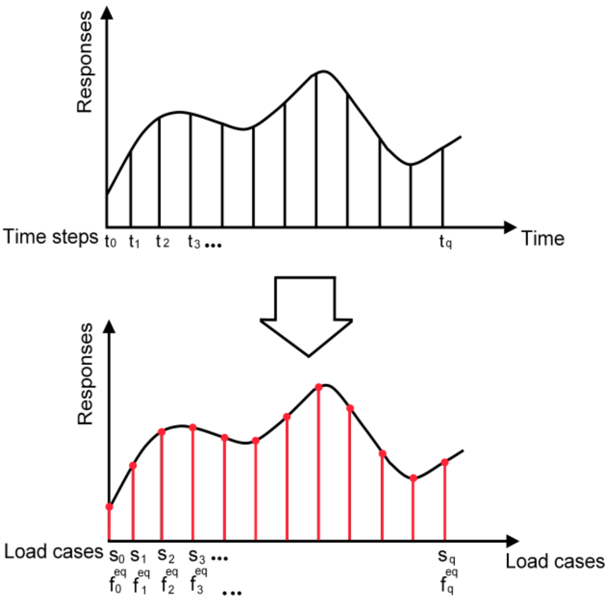

Figure 9: Equivalent static loads at every time interval.

Method[3]

At any time during dynamic analysis, ESL is the load set that generates the same displacement field as from the dynamic analysis. The total number of load sets in ESL analysis is equal to the total number of time intervals at which the displacement fields are taken from dynamic analysis to calculate the ESLs, see **Figure 9**. These load sets are used as the external loads for linear static analysis. The equilibrium equation of a structure considering time domain in finite element method is given as

$$\mathbf{M}(\mathbf{b})\ddot{\mathbf{z}}_N(t) + \mathbf{K}_N\big(\mathbf{b}, \mathbf{z}_N(t)\big)\mathbf{z}_N(t) = \mathbf{f}(t)$$

$$(t = t_0,\ t_1,\ t_2,\ \ldots,\ t_n) \tag{2}$$

In Eqn. (2), **M** is the mass matrix, **K** is the stiffness matrix, $\ddot{\mathbf{z}}_N(t)$ is the acceleration, $\mathbf{z}_N(t)$ is the displacement, **b** is the design variable, subscript *N* represents non-linear analysis, *t* is the time and *n* is the total number of time intervals. From Eqn. (2) the non-linear displacement $\mathbf{z}_N(t)$ can be calculated at each time interval *t*. These displacement fields and the linear stiffness matrix $\mathbf{K}_L$ are used to calculate the ESLs at each time step, see Eqn. (3).

$$\mathbf{f}_{eq}^{\mathbf{z}}(s) = \mathbf{K}_L\mathbf{z}_N(t)$$

$$(s = s_0,\ s_1,\ s_2,\ \ldots,\ s_n) \tag{3}$$

In Eqn. (3), subscript L represents static analysis and $\mathbf{f}_{eq}^{\mathbf{z}}(s)$ is the s^{th} equivalent static load calculated at i^{th} time interval. The evaluated ESL sets can now be used as external loads for static analysis, Eqn. (4).

$$\mathbf{K}_L(\mathbf{b})\mathbf{z}_L(s) = \mathbf{f}_{\mathbf{eq}}^{\mathbf{z}}(s) \tag{4}$$

From Eqn. (4), the linear displacement $\mathbf{z}_L$ can be calculated. The displacement, $\mathbf{z}_L(s)$, from Eqn. (4) is the same as the non-linear displacement, $\mathbf{z}_N(t)$, from Eqn. (2).

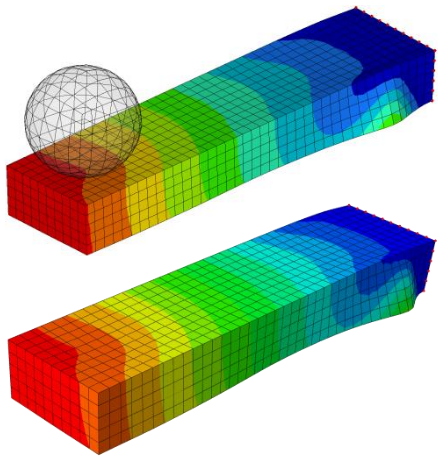

Figure 10: Displacement comparison between dynamic non-linear and ESL analysis.

Implementation

The purpose to use ESL in this study is to approximate the robustness of designs outside the TI. Hence for the designs outside the TI, the optimization points are still evaluated using non-linear dynamic analysis but the robustness points are evaluated using ESL. The idea here is to calculate the ESL sets from the optimization point and apply these ESL sets to each robustness points of this optimization point. Since the ESLs are only used for the robustness points, the limitations of ESLM, especially for large shape modifications for shape optimization, is avoided. This is because for the robustness, if the uncertainties in shape are considered, the shape modifications are very small and therefore the ESLs are still correct after these small modifications. However one of the main issues to use ESL in here is to consider the impactor uncertainties, this is not discussed here.

Validation

To validate the use of ESL for the robustness points, some simple test cases were used to compare the results between the dynamic non-linear analysis and analysis using ESL. Some of the validation cases are presented here.

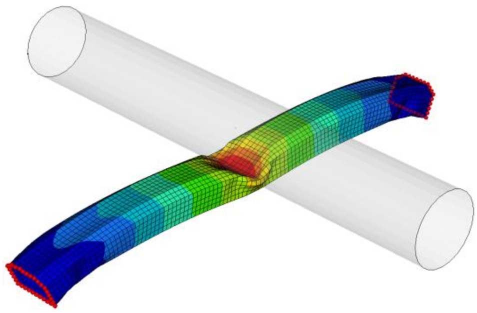

Figure 11: Full structure dynamic non-linear analysis.

1. Cantilever beam impact

One of the simple tests cases used is a cantilever beam with a ball impactor. **Figure 10** (top) shows the deflection contour from the dynamic non-linear analysis and **Figure 10** (bottom) shows the deflection contour from the ESL analysis at the last time interval. The ESL analysis produces a good approximate of the non-linear dynamic analysis in this case.

2. Beam deflection with sub-structuring

In this test case, we take both the sub-structuring and the ESL approach for a simple beam bending case shown in **Figure 11**. A sub-structure is created at the center of the full beam shown in **Figure 12**(top).For the sub-structure, the displacement fields are imposed at the interface nodes (shown in red) which are obtained from the full structure analysis, **Figure 11**. From **Figure 11** and **Figure 12**(top) it can be seen that the displacement contours of the full structure and the sub-structure are at a good agreement. For the analysis with sub-structuring and ESL, the displacement fields are imposed at the interface nodes and the ESLs are applied to the remaining nodes of the sub-structure. Again the comparison of the displacement contour plots in **Figure 12**(top)and **Figure 12**(bottom)are in good agreement in this case.

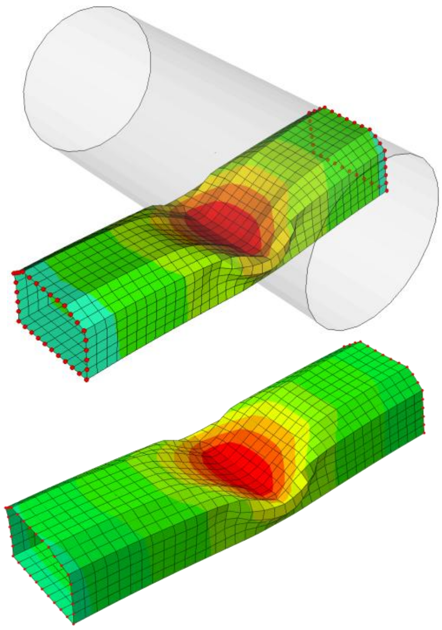

Figure 12: Sub-structuring with dynamic non-linear analysis and ESL analysis.

Conclusions

The ESL analysis was also tested for slight thickness variation (± 0.07 *mm*) from one design to another design (not presented here due to paper length limitation). The ESL analysis gives a good approximation for the tested cases. Hence the use of ESLs seems suitable for robustness design analysis outside the TI.

MDLRDO with ESL - Validation

The coupling of the MDLRDO and ESL was tested with the simple test case shown in **Figure 1**. First this approach was tested with only thickness parameters and thickness uncertainty. The optimization loop used is shown in **Figure 13.** The objective was to reduce the overall mass of the beam while respecting the deflection constraint, $d \leq 20\ mm$. An offset is created for the deflection at $d = 10\ mm$, creating a TI with size 10 *mm*. The computational time for the dynamic non-linear analysis of the beam was 4 minutes and the analysis using ESLs took 1 minute (the whole process of calculating ESL sets and using them for linear static analysis). The computational efficiency was roughly 32% due to the use of ESLM analysis instead of dynamic non-linear analysis.

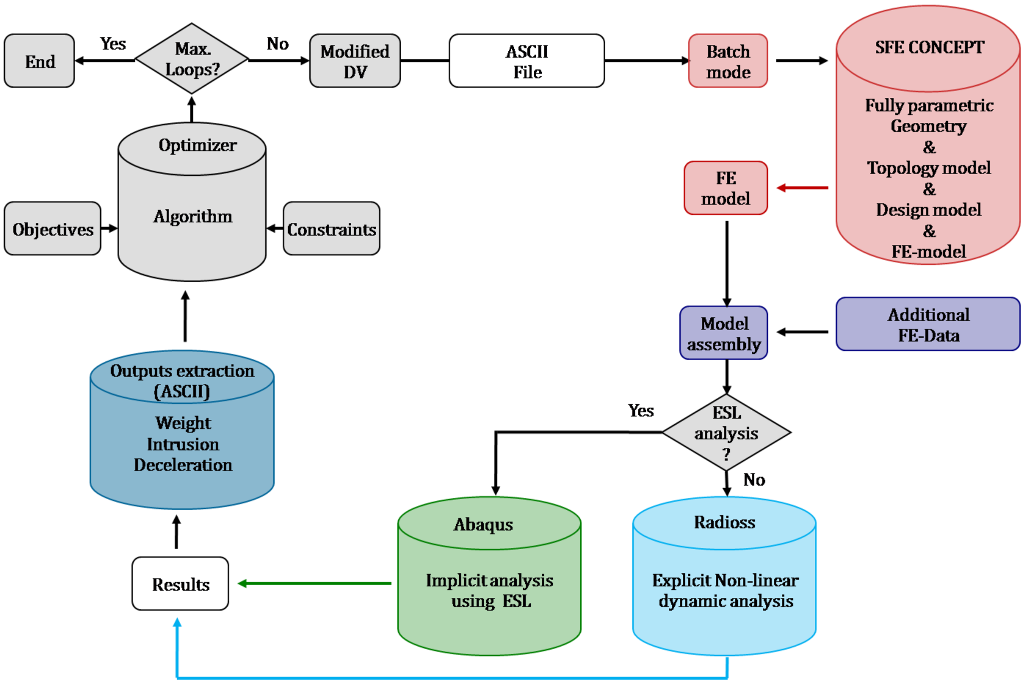

Figure 13: Optimization loop for MDLRDO and ESL.

3. Overall Conclusions

In this study an approach for efficient RDO via the use of approximation methods; sub-structuring and ESLM is presented. The approximation methods are used for design analysis to reduce the simulation time of design for optimization. Also a modified RDO approach was presented to reduce the number of design evaluations for a RDO study. The three approaches were first validated separately with some simple cases. The sub-structuring and ESLM showed good approximation characteristics while reducing the simulation time. Finally the MDLRDO and ESLM were coupled together to perform a robust design optimization considering thickness only.
For the future work, the three approaches will be coupled together and applied to an industrial sized problems such as the B-pillar. Also the shape parameters and impactor uncertainty will be considered.

References

[1] Duddeck F, Multidisciplinary optimization of car bodies. *Structural and Multidisciplinary Optimization*, 35(4),375–389, 2008.

[2] Sun G, Li G, Zhou S, Li G, Hou S and Li Q, Crashworthiness design of vehicle by using multi-objective robust optimization. *Structural and Multidisciplinary Optimization*,44, 99–110, 2011.

[3] Park GJ, Equivalent static loads method for non-linear static response structural optimization. *In German LS-DYNA Forum*, Bamberg, Germany, 2010.

[4] Averill R, Efficient shape optimization of crashworthy structures using a new substructuring method. *In 3rd German LS-DYNA Forum*, Bamberg, Germany, 2004.

[5] Rayamajhi M, A sub-structure approach for shape optimisation for crash. *Pres 4th GACM Colloquium*, Technische Universität Dresden, Dresden, Germany, 2011.