

# **Key Parameters in High-Dimensional Systems with Uncertainty**

L. Graff, J. Fender, H. Harbrecht,  
M. Zimmermann

Institute of Mathematics  
University of Basel  
Rheinsprung 21  
CH - 4051 Basel  
Switzerland

Preprint No. 2013-12  
April, 2013

[www.math.unibas.ch](http://www.math.unibas.ch)

# Key parameters in high-dimensional systems with uncertainty

**L. Graff**

BMW Group Research and Innovation Center  
Knorrstr. 147, 80937 Munich, Germany  
Email: Lavinia.Graff@bmw.de

**J. Fender**

BMW Group Research and Innovation Center  
Knorrstr. 147, 80937 Munich, Germany  
Email: Johannes.Fender@bmw.de

**H. Harbrecht**

University of Basel  
Department of Mathematics and Computer Science  
Rheinsprung 21, 4051 Basel, Switzerland  
Email: Helmut.Harbrecht@unibas.ch

**M. Zimmermann**

BMW Group Research and Innovation Center  
Knorrstr. 147, 80937 Munich, Germany  
Email: markusz@alum.mit.edu

*Key parameters may be used to turn a bad design into a good design with comparatively little effort. The proposed method identifies key parameters in high-dimensional non-linear systems that are subject to uncertainty. A numerical optimization algorithm seeks a solution space on which all designs are good, that is, they satisfy a specified design criterion. The solution space is box-shaped and provides target intervals for each parameter. A bad design may be turned into a good design by moving its key parameters into their target intervals. The solution space is computed so as to minimize the effort for design work: its shape is controlled by particular constraints such that it can be reached by changing only a small number of key parameters. Wide target intervals provide tolerance against uncertainty, which is naturally present in a design process, when design parameters are unknown or cannot be controlled exactly. In a simple two-dimensional example problem, the accuracy of the algorithm is demonstrated. In a high-dimensional vehicle crash design problem, an under-performing vehicle front structure is improved by identifying and appropriately changing a relevant key parameter.*

## 1 Introduction

Designs that fail to meet their design goals may be improved by appropriately changing relevant design parameters. When design parameters are subject to uncertainty, this can be very difficult. The deviation between desired and realized parameter settings may lead to catastrophic design failure, in particular when the design problem is non-linear and the system response abruptly changes under parameter variation. Uncertainty is present when parameters or component properties cannot be controlled exactly. As an example, the force-deformation characteristic of a structural member is difficult to adjust by detail parameters like the metal sheet thickness. This paper is concerned with, first, identifying the key parameters that can be used to improve a design with least effort, and, second, providing information on how these key parameters need to be modified in order to turn a bad into a good design in the presence of uncertainty.

Classical approaches to identify relevant parameters are sensitivity analysis, classical optimization and robust design optimization. *Sensitivity analysis* quantifies the importance of input parameters for the variability of the output [1, 2]. Local sensitivity analysis investigates the local influence of each input parameter on the output. This kind of analysis is well suited for problems that can be well approximated

by linear functions. Local sensitivity measures are obtained by computing partial derivatives of the output function with respect to the input parameters. Global sensitivity analysis takes the entire design space into account to apportion the variability of the output parameter to the variability in each input parameter. There are several measures used in global sensitivity analysis: The *regression coefficient* quantifies the slope of a linear approximation, the *Pearson correlation coefficient* measures to what degree an input parameter determines the output in a linear relationship. The *Spearman correlation coefficient* quantifies the monotony in the relationship between one input parameter and the output. *Sobol indices* are particularly tailored for multidimensional functions, see [3]. The first order Sobol index is a measure for the direct effect of an input parameter on the output. The higher order indices quantify the influence of the interactions between the input parameters. The fraction of the output variation that is related to each input parameter is measured by the total order index, see [1, 4].

Every sensitivity measure measures the importance of an input parameter in one particular sense. As all the information on how input and output parameters are related is reduced to one measure, other information is lost. Therefore, in the sensitivity measures mentioned before, no information is included on how the parameters have to be changed in order to obtain a particular result.

Contrary to sensitivity measures, *classical optimization* provides this information by seeking an optimum in the design space. Unfortunately, however, classical optimization does not take uncertainty into account. Optimal designs may be non-robust and quite sensitive to parameter variabilities. Due to the underlying uncertainty, realizing an optimum in a practical design may be impossible.

Both, sensitivity analysis and classical optimization are not concerned with uncertainty and therefore of limited use for the purpose of this paper. *Robust design optimization* does take uncertainty into account by seeking a design point in a particular neighborhood with little output variation or sufficient performance (see [5]). The size of that neighborhood is specified in advance and represents parameter variability associated with an measured or assumed underlying uncertainty. Robust design optimization prescribes the permissible variability and cannot optimize the tolerance to variations.

The approach presented in [6, 7, 8, 9] is similar to robust design optimization in that it also computes a permissible region rather than one design point. The solution space, however, is constructed to be as *large as possible* to make it easier to reach the target. Robust design optimization does not seek a large solution space, it rather looks for a neighborhood with good output performance and fixed size. For a robust design optimization problem with a performance threshold value, this implies that, either, there is no solution if the neighborhood was chosen too large, or, there is a solution with an associated neighborhood which may not be as large as possible. Maximizing the solution space, however, provides a target space which is as large as possible and therefore easier to reach.

In this paper, the work of [7, 8] is extended with a focus on reducing the effort to turn a bad design in a good design. A large solution space is sought that already includes as many parameters from the bad design as possible by formulating appropriate constraints. Parameters without constraints may lie outside of the solution space and will have to be changed to fulfill the design goal.

The paper is organized as follows. In Section 2, a simple example problem is considered, and the new approach is motivated. A mathematical problem statement for the solution space optimization with constraints is given in Section 3. Section 4 reviews the algorithm to identify solution spaces and presents the extensions for the particular purpose of this paper. The numerical accuracy of the algorithm is validated with the simple example problem. Section 5 explains the particular challenges of front crash design. In Section 6, the proposed method is applied to a non-linear and high-dimensional engineering crash problem.

## 2 A simple example problem

A simple example problem from crash analysis is considered as presented in [7]. The load case is similar to a USNCAP front crash, where the vehicle hits a rigid barrier at a speed of  $v_0 = 56$  km/h with full overlap, see Figure 1.

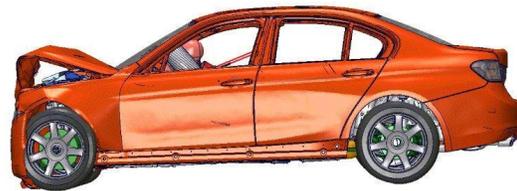


Fig. 1. USNCAP front crash.

A model of the vehicle structure is used that consists of two structural components, see Figure 2(a). The structural components 1 and 2 are the only deformable parts and have the deformation measures  $u_1$  and  $u_2$ , respectively. The rest of the vehicle model is rigid. The deformable components have no mass, all mass is located on the rigid part. The forces necessary to deform components 1 and 2 are  $F_1$  and  $F_2$ , respectively, see Figure 2(b).  $F_1$  and  $F_2$  are assumed to be constant while deforming. If the maximum deformations  $u_{1c}$  and  $u_{2c}$  are reached, the forces may become arbitrarily large in order to avoid further deformation. The crash performance is measured by the acceleration of the passenger cell and the order of structural deformation, that is, whether component 1 or component 2 deform first. The design goals for the simple example problem are:

- The maximum deceleration should not exceed the critical threshold value  $a_c$ , that is,  $a \leq a_c$ .
- Component 1 should deform before component 2 deforms.

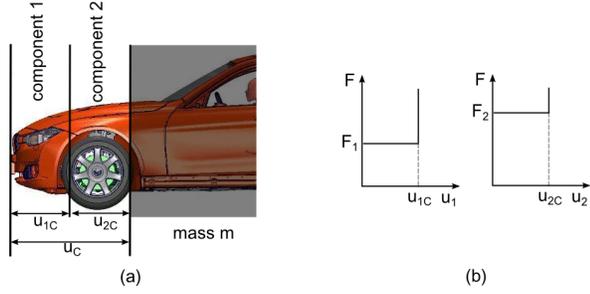


Fig. 2. (a) Vehicle structure of example problem consisting of two deformable components. (b) Force-deformation characteristics of the structural components 1 and 2.

This translates to the requirements on  $F_1$  and  $F_2$  that  $F_1 \leq F_2$ , that the deformation force of component 2 is  $F_2 \leq ma_c$  and that entire kinetic energy is absorbed, that is  $\frac{1}{2}mv_0^2 \leq F_1u_{1c} + F_2u_{2c}$ .

With the performance function

$$f(F_1, F_2) = \begin{cases} 1, & \text{if } \frac{1}{2}mv_0^2 > F_1u_{1c} + F_2u_{2c} \\ 1, & \text{if } F_1 > F_2 \\ \frac{F_2/m - a_c}{a_c}, & \text{otherwise,} \end{cases} \quad (1)$$

the design goal is met, when

$$f(F_1, F_2) \leq 0. \quad (2)$$

The solution space defined by expression (2) is shown for  $m = 2000$  kg,  $a_c = 32$  g,  $v_0 = 15.6$  m/s and  $u_{1c} = u_{2c} = 0.3$  m in Figure 3(a).

Now consider a design with  $F_1 = 275$  kN and  $F_2 = 450$  kN. It violates (2). In order to identify what parameter may be changed and by how much in order to improve the design with least effort, three scenarios are compared:

(a) A classical solution box with maximum volume is shown in Figure 3(a). Both components 1 and 2 have to be modified in order to meet the design goal.

(b) A solution box that includes  $F_2 = 450$  kN is shown in 3(b). In order to meet the design goal, only component 1, that is,  $F_1$ , will have to be changed. Note that  $F_2$  is included in the solution box with a safety margin of  $\pm 25$  kN. This is necessary, since  $F_2$  cannot be controlled exactly.

(c) Finally, a solution box that includes  $F_1 = 275$  kN is shown in Figure 3(c). In order to meet the design goal, only component 2, that is,  $F_2$  will have to be changed. The same safety margin as in scenario (b) is provided.

The solution boxes of scenarios (b) and (c) are smaller than the one of scenario (a). In this sense, designs from these boxes are less robust and more difficult to realize. However, scenario (a) requires redesigning two components, while scenarios (b) and (c) only require redesigning one component. A designer, knowing that component 1 is easier to redesign than component 2, would therefore prefer scenario (b). The

deformation force  $F_1$  would be the *key parameter* to meet the design goal.

This procedure can be generalized by seeking solution boxes under the constraint that certain parameters be included with a specified safety margin. The associated mathematical problem statement is provided in the following section.

### 3 General problem statement

Let  $\Omega$  be an axis-parallel hyperbox, which is defined as the Cartesian product of intervals  $I_i = [x_i^{low}, x_i^{up}]$

$$\Omega = I_1 \times \cdots \times I_d \subseteq \Omega_{ds}, \quad (3)$$

with  $x_i^{low}$  and  $x_i^{up}$  being the lower and the upper boundary for dimension  $i$ , respectively. The axis-parallel hyperbox  $\Omega_{ds}$  is the design space that includes all possible designs under consideration. The measure  $\mu(\Omega)$  quantifies the size of a solution space. Typically, the size is the volume given by

$$\mu(\Omega) = \prod_{i=1}^d (x_i^{up} - x_i^{low}). \quad (4)$$

Every design  $\mathbf{x} \in \Omega_{ds}$  is assigned a performance value  $z$  given by the performance function

$$z = f(\mathbf{x}). \quad (5)$$

In many engineering problems,  $z$  is computed numerically, therefore analytical properties of  $f(\mathbf{x})$  are unknown and it has to be treated like a black box. A design  $\mathbf{x}$  that satisfies the *performance criterion*

$$f(\mathbf{x}) \leq f_c \quad (6)$$

is called a *good design*, a design that does not is called a *bad design*.

The optimization problem to maximize the size of the solution space as introduced in [7] reads

$$\left. \begin{array}{l} \text{seek } \Omega \subseteq \Omega_{ds}, \\ \text{such that } \mu(\Omega) \rightarrow \max \\ \text{subject to } f(\mathbf{x}) \leq f_c \text{ for all } \mathbf{x} \in \Omega. \end{array} \right\} \quad (P1)$$

As motivated in the previous section, the classical problem statement is now enriched by constraints ensuring that parameters values are included in the resulting solution box. More specifically,

$$\left. \begin{array}{l} \text{seek } \Omega \subseteq \Omega_{ds}, \\ \text{such that } \mu(\Omega) \rightarrow \max \\ \text{subject to } f(\mathbf{x}) \leq f_c \text{ for all } \mathbf{x} \in \Omega \\ \text{and } x_k^{low} \leq x_{c,k}^{low}, x_l^{up} \geq x_{c,l}^{up} \end{array} \right\} \quad (P2)$$

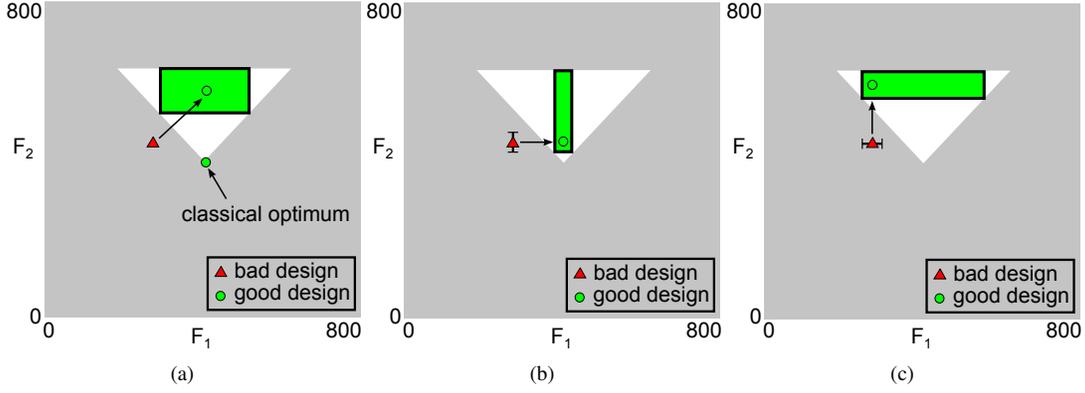


Fig. 3. Changes necessary to meet the design goal: (a)  $F_1$  and  $F_2$ , (b) only  $F_1$ , (c) only  $F_2$ .

with  $x_{c,k}^{low}$  and  $x_{c,l}^{up}$  being the constraint for the upper and lower boundary of the solution box, respectively.

## 4 Computing solution spaces with constraints

### 4.1 Review of the underlying algorithm

A solution algorithm for (P1) was introduced in [7] and analyzed in [8], and is briefly described here. Starting from a box that includes at least one good design, the algorithm computes new candidate boxes by iterative modification and evaluation, as shown in Figure 4. The iteration can be subdivided into the *exploration phase* and the *consolidation phase*.

The purpose of the exploration phase is to identify a good location of the solution box in the design space where the boundary intervals for every parameter can be extended as much as possible. The exploration phase consists of four steps: In step one, a Monte Carlo sample is computed within the candidate box (for literature on Monte Carlo Sampling see [10]). In step two, the sample is evaluated by Bayesian statistics, that is, estimating the fraction of good designs and computing a confidence level for the estimate [6]. In the third step, a new candidate box is identified that includes only good designs of the current sample. This is done by the *trimming algorithm* that removes bad space by relocating boundaries. Finally, in step four, the candidate box is extended tentatively to allow for growth into good space. Extending the boundaries that are close to bad space may decrease the fraction of good design space which will be corrected by step three. Extending the boundaries that are not close to bad space, however, will increase the size and make the candidate box evolve towards the maximum solution box.

The purpose of the consolidation phase is to ensure that the fraction of good design space in the final solution box is sufficiently large. This is accomplished by applying the trimming algorithm only.

### 4.2 Extension for constraints

The algorithm is extended to account for constraints and solve problem (P2). The extension is done by modifying the trimming algorithm, where candidate boxes without bad designs are computed in three nested loops. In the original al-

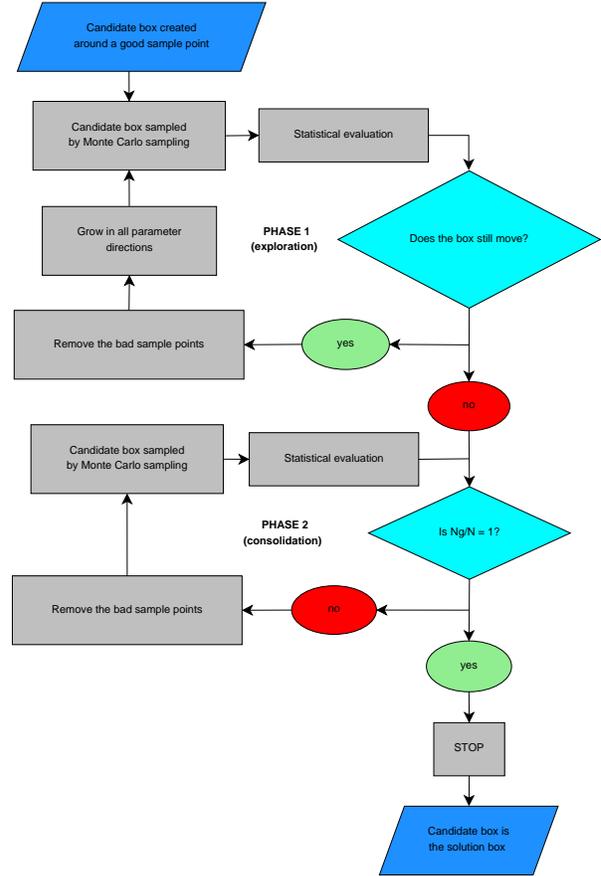


Fig. 4. Underlying algorithm to compute solution boxes.

gorithm [6], the largest box is chosen as new candidate box for the next iteration step. In the extended algorithm with constraints, an error measure is introduced that quantifies to what degree constraints are violated. The *constraint violation error*  $\varepsilon$  is defined as

$$\varepsilon^2 = \sum_k \omega_k \varepsilon_k^2 + \sum_l \omega_l \varepsilon_l^2, \quad (7)$$

with  $k$  and  $l$  being the indices of the upper and lower constrained parameter boundaries and with

$$\varepsilon_k = \begin{cases} 0, & \text{if } x_k^{low} \leq x_{c,k}^{low}, \\ x_k^{low} - x_{c,k}^{low}, & \text{otherwise} \end{cases} \quad (8)$$

and

$$\varepsilon_l = \begin{cases} 0, & \text{if } x_l^{up} \geq x_{c,l}^{up}, \\ x_l^{up} - x_{c,l}^{up}, & \text{otherwise.} \end{cases} \quad (9)$$

$\omega_k$  and  $\omega_l$  are weights of the constraints. If there are several boxes satisfying all constraints, that is,  $\varepsilon = 0$ , the box with the largest number of good sample points is chosen. The extended trimming algorithm is shown in Algorithm 1.

```

Data: a candidate hyperbox  $\Omega_{cand}$  and a set
 $S = \{\mathbf{x}_j \in \Omega_{cand} : f(\mathbf{x}_1) \geq \dots \geq f(\mathbf{x}_N)\}$  of sample points
Result: hyperbox  $\subseteq \Omega_{cand}$  which includes only good sample points
forall the good sample points  $\{\mathbf{x}^{good} \in S : f(\mathbf{x}^{good}) \leq f_c\}$  do
  forall the bad sample points  $\{\mathbf{x}^{bad} \in S : f(\mathbf{x}^{bad}) > f_c\}$  do
    for  $i = 1, 2, \dots, d$  do
      if  $x_i^{bad} < x_i^{good}$  then
        count the good sample points  $\mathbf{x}$  with  $x_i^{bad} \geq x_i \geq x_i^{low}$ ;
      else
        count the good sample points  $\mathbf{x}$  with  $x_i^{bad} \leq x_i \leq x_i^{up}$ ;
      end
    end
    choose the direction  $i^*$  where the fewest good sample points are removed;
    if  $x_{i^*}^{bad} < x_{i^*}^{good}$  then
      move boundary  $i^*$  to  $x_{i^*}^{low}$ ;
    else
      move boundary  $i^*$  to  $x_{i^*}^{up}$ ;
    end
    forall the directions  $i$  where a bad sample point is removed do
      if  $x_i^{bad} < x_i^{good}$  then
         $x_i^{low} := \min_j x_{i,j}$  for all remaining good sample points
         $\mathbf{x}_j$ ;
      else
         $x_i^{up} := \max_j x_{i,j}$  for all remaining good sample points
         $\mathbf{x}_j$ ;
      end
    end
    calculate  $\varepsilon^2$ ;
    remember the hyperbox  $[x^{low}, x^{up}] \subseteq \Omega_{cand}$  with smallest  $\varepsilon^2$ ;
    if  $\varepsilon^2 = 0$  then
      remember the hyperbox with most good sample points;
    end
  end
end

```

**Algorithm 1:** Extended trimming algorithm.

### 4.3 Numerical results for the simple example problem

The extended algorithm is applied to the simple example problem from Section 2. All three scenarios are computed with  $N = 100$  designs per Monte Carlo sample. The first candidate box includes the classical optimum at the lower tip of the solution space triangle. The exploration phase and the consolidation phase are run for 20 and 10 iteration steps, respectively.

Figure 5 shows how the extended algorithm drives the evolution of the candidate box for scenario (b). The constraints are chosen such that only  $F_1$  has to be modified, that is, constraints  $F_2^{low} \leq F_{c,2}^{low} = 425$  kN and  $F_2^{up} \geq F_{c,2}^{up} = 475$  kN. The first row illustrates the exploration phase: the algorithm enlarges the volume of the candidate box by extending the box boundaries. It cannot extend the box boundaries in the direction of  $F_1$ , as this would violate the constraints. In the consolidation phase, shown in the second row, bad designs are removed, until the final solution box is obtained that satisfies the constraints.

The numerical results for all three scenarios are given in Table 1 and in Figure 6. The deviations between the upper box boundaries computed numerically  $F_i^{up}$  and the analytical solutions  $F_{0,i}^{up}$  is given by  $\Delta F_i^{up} = F_i^{up} - F_{0,i}^{up}$ . The relative error is given by  $\phi_i^{up} = \Delta F_i^{up} / F_{0,i}^{up}$ . The expressions for lower box boundaries are defined in a similar way. The numerical approximation and the analytical solution agree within an error of less than 2%.

## 5 Crash design for the USNCAP front crash

### 5.1 Problem description

In the test scenario of a USNCAP front crash, a vehicle hits a quasi-rigid barrier at the speed of 56 km/h. The vehicle structure and the restraint systems are to be designed such that the loads on crash test dummy and the deformation of the passenger cell stay below critical threshold values. Design work is done on three levels, the vehicle level, the component level and the detail level <sup>1</sup>:

*Design goals on the vehicle level.* The primary focus of structural development in an early design phase lies on satisfying the following design goals [9, 12, 13]:

- The maximum deceleration of the passenger cell measured at the bottom of the B-pillar should not exceed the critical threshold value  $a_c$ , that is,  $a \leq a_c$ .
- Structural elements in the front should deform first. The deformation continues from front to rear. Structural elements attached to the firewall deform last.
- The deformation of the passenger cell should stay below a critical threshold value.

Note that these criteria are only sufficient for the preliminary design. In final tests, the vehicle performance will be evaluated with respect to dummy loads.

*Component properties on the component level.* The structural behavior in a USNCAP-type front crash depends primarily on the distributed vehicle mass and the resistance force of structural elements against deformation, expressed as *force-deformation characteristics*

$$F = \hat{F}(u), \quad (10)$$

see [12, 13].  $F$  is a longitudinal force exerted by the structural component under the relative longitudinal displacement

<sup>1</sup>This is similar to *target cascading* [11].

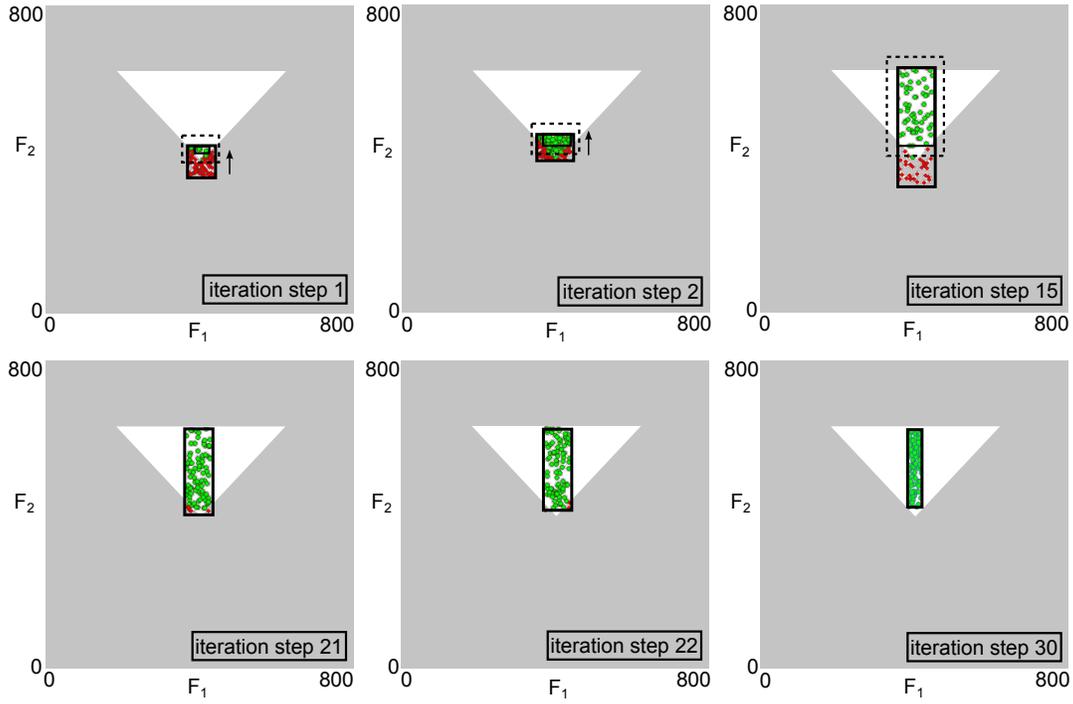


Fig. 5. Evolution of the candidate box in the exploration phase (top row) and consolidation phase (bottom row) for constraints of scenario (b) ensuring that only  $F_1$  needs to be changed, that is,  $F_2^{low} \leq F_{c,2}^{low} = 425$  kN and  $F_2^{up} \geq F_{c,2}^{up} = 475$  kN.

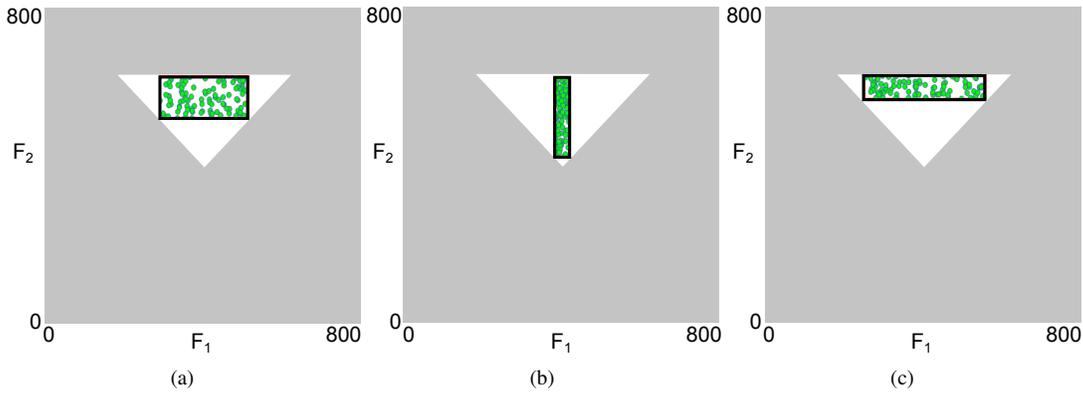


Fig. 6. Computed solution boxes (a) without constraints, (b) constraints ensuring that only  $F_1$  needs to be changed ( $F_2^{low} \leq F_{c,2}^{low} = 425$  kN,  $F_2^{up} \geq F_{c,2}^{up} = 475$  kN), and (c) constraints ensuring that only  $F_2$  needs to be changed ( $F_1^{low} \leq F_{c,1}^{low} = 250$  kN and  $F_1^{up} \geq F_{c,1}^{up} = 300$  kN).

	(a) $F_1$ and $F_2$ to be changed				(b) only $F_1$ to be changed $F_2^{low} \leq F_{c,2}^{low} = 425$ kN $F_2^{up} \geq F_{c,2}^{up} = 475$ kN				(c) only $F_2$ to be changed $F_1^{low} \leq F_{c,1}^{low} = 250$ kN $F_1^{up} \geq F_{c,1}^{up} = 300$ kN			
	analytical $F_{0,i}$ in kN	numerical $F_i$ in kN	error $\Delta F_i$ in kN	rel. error $\phi_i$ in %	analytical $F_{0,i}$ in kN	numerical $F_i$ in kN	error $\Delta F_i$ in kN	rel. error $\phi_i$ in %	analytical $F_{0,i}$ in kN	numerical $F_i$ in kN	error $\Delta F_i$ in kN	rel. error $\phi_i$ in %
$F_1^{low}$	290.96	290.74	0.22	0.08	381.50	384.30	2.80	0.73	250.00	249.14	0.86	0.34
$F_1^{up}$	515.55	516.31	0.76	0.15	425.00	422.47	2.53	0.59	556.50	556.32	0.18	0.03
$F_2^{low}$	515.55	515.81	0.26	0.05	425.00	422.06	2.94	0.69	556.50	563.67	7.17	1.29
$F_2^{up}$	627.84	622.59	5.25	0.84	627.84	623.69	4.15	0.66	627.84	627.48	0.36	0.06

Table 1. Analytical solutions and numerical results for the simple example problem, scenarios (a), (b) and (c).

$u = u_B - u_A$ , with  $u_A$  and  $u_B$  being the x-displacements of the component boundaries, see Figure 7. Expression (10) is correct only for loading, that is, when  $\dot{u} \geq 0$ . For a fixed vehicle mass, the maximum deceleration is given by

$$a = f(\hat{F}_1(u_1), \hat{F}_2(u_2), \dots, \hat{F}_n(u_n)) \quad (11)$$

where  $\hat{F}_k(u)$  denotes the force-deformation characteristic of the  $k$ -th out of  $n$  components. Other quantities from the vehicle level, like an indicator variable of the order of deformation, or the deformation of the passenger cell can be expressed in a similar way.

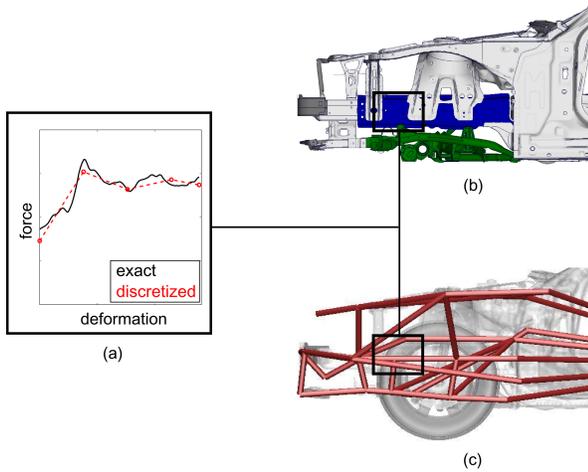


Fig. 7. (a) Force-deformation characteristics of a component of the vehicle structure: exact and discretized. (b) Detail vehicle model. (c) Simplified model.

*Detail level.* The force-deformation characteristics depend again on geometrical and material detail parameters  $p_j$ , that is

$$\hat{F}_k(u) = \bar{F}_k(u; p_1, p_2, \dots, p_m), \quad (12)$$

with  $j$  being a parameter index, and  $m$  being the number of detail parameters. Detail parameters may be sheet metal thicknesses, profile geometries, yield strengths or hardening curves. They will determine the force-deformation characteristics of each structural member and, thus, also determine the overall structural behavior.

In classical vehicle design, detail parameters are varied, until the design goals on the vehicle level are satisfied. In a new design approach, the component level was introduced to enable the design of component properties such as force-deformation characteristics without specifying the underlying detail parameters [13]. This is useful, for example in an early design phase, when detail parameters are difficult to be specified or simply unknown. For flexible and robust design, requirements on component properties are to be identified as permissible intervals [7]. In a subsequent development step,

detail parameters are then specified such that all component requirements are satisfied.

In the example considered here, a bad vehicle design is given, and all detail parameters are known. Rather than varying detail parameters according to the classical design approach, however, the relevant component to be modified is identified by computing appropriate solution spaces on the component level.

## 5.2 Crash simulation models

In a *detailed Finite Element model*, all detail parameters are specified. For crash simulations, this is typically a model of the entire vehicle. From this, all quantities on the vehicle level, such as the vehicle deceleration, and on the component level, such as force-deformation characteristics, can be computed. Force-deformation characteristics are derived from section forces  $F(t)$  and deformations  $u(t)$  for  $\dot{u}(t) \geq 0$ . A detailed Finite Element model maps the detail level onto the vehicle and the component level.

By contrast, a *simplified model* as described in [12, 13], computes the vehicle behavior directly from force-deformation characteristics, as in expression (11). It maps the component level onto the vehicle level.

Force-deformation characteristics contain more information than necessary. The mechanical behavior of a structural member that is relevant for a USNCAP-type front crash can be sufficiently well approximated by 4-10 discrete force values at specified support points. Therefore,  $\hat{F}_k(u)$  is discretized as shown in Figure 7(a). The maximum vehicle deceleration is then given by

$$a = f(F_1, F_2, \dots, F_d) \quad (13)$$

with  $F_i$  being the force values at specified support points of the force-displacement characteristics, and  $d$  being the total number of discrete force values. Expression (13) may be computed with the simplified crash model.

## 5.3 Why vehicle crash design is difficult

Improving bad designs is difficult because of nonlinearity. All mappings between the detail, component and vehicle level are typically highly nonlinear. Nonlinearity between the component and the vehicle level can be observed in Figure 8. For a bad design with maximum deceleration  $a > a_c$ , the force-deformation characteristics of the crash box and the front rail are modified. When making only the crash box stronger,  $a$  becomes worse. Strengthening the front rail, improves  $a$ , however it remains supercritical. Combining the modification with worsening effect with the modification with insufficient effect, produces a good design with  $a < a_c$ . The influence of the design parameter "force-deformation characteristic of crash box" is changed by modifying the other parameter "force-deformation characteristic of front rail". This effect will be called *parameter interaction*.

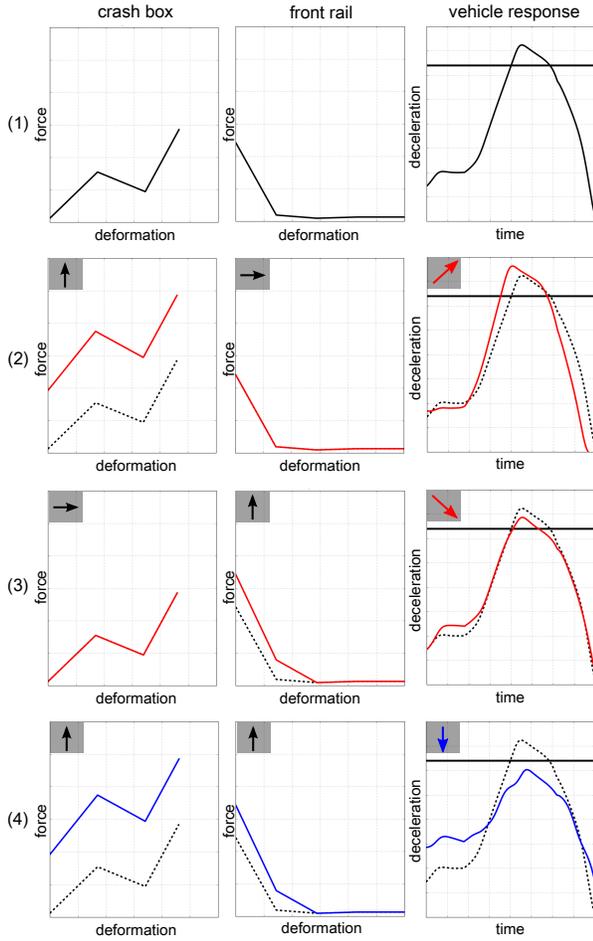


Fig. 8. Force-deformation characteristics and their nonlinear influence on the maximum deceleration. (1) Original design. (2)&(3) 2 modifications yielding bad designs each. (4) Combined modification yielding a good design.

The parameter interaction between the force-deformation characteristics of the crash box and the front rail can be explained physically: while deforming, structural members exert a decelerating force on the vehicle that reduces the kinetic energy. When all structural members completed their deformation, the remaining kinetic energy is absorbed in an abrupt collision of the passenger cell with the engine block that is already in contact with the barrier wall. This final collision is associated with a deceleration signal, that increases with increasing remaining kinetic energy. One may assume that increasing the force of the crash box should increase the initial force that decelerates the vehicle, and therefore reduce the remaining kinetic energy and the maximum deceleration. However, this is not the case for variant (2) in Figure 8: the front rail behind the crash is not strong enough to support the load of the crash box, resulting in a premature collapse of the front rail. This leads to an even lower force to decelerate the vehicle in the beginning, and, consequently, to a higher maximum deceleration. In variant (4), the front rail is sufficiently strong, the influence of the parameter "force-deformation characteristic of crash

box" is reversed, and the system exhibits the desired overall behavior.

In addition to parameter interaction, other nonlinear phenomena are present in crash design, such as abrupt changes of vehicle responses or non-monotonous dependencies on design variables. Nonlinearities make it difficult to assess the influence of each parameter, and therefore obstruct the identification of the key parameters and their setting necessary to turn a bad into a good design.

## 6 Application in crash design

### 6.1 Identification of a key component

A vehicle structure as shown in Figure 7(b) is considered. It consists of nine structural members with force-deformation characteristics shown in Figures 9(a). The force-deformation characteristics are measured in a detail model. The performance in the USNCAP front crash is insufficient, because  $a = a_1 > a_c$ . In order to identify the relevant components and the necessary modifications, solution spaces for the force-deformation characteristics are computed.

The solution space for a force-deformation characteristic is represented by an upper and a lower boundary line in a force-deformation diagram, see solid bold lines in Figure 9(a). The region bounded by these two lines is called a *corridor*. Corridors are approximated by linear interpolation between two support points.

A simplified crash model provides the mapping (13) for a total of  $d = 55$  parameters  $F_i$ . For each component  $k$ , a solution space  $\Omega_k$  is computed under the constraint, that all force-deformation characteristics are included, except the one of component  $k$ . This procedure is similar to the one that was applied in Section 2. Unfortunately, no corridor is obtained that strictly satisfied the constraints.  $\Omega_5$  however, see Figure 9, violates the constraints only to a negligible degree<sup>2</sup> for components 4, 6 and 9. Noting that the force deformation-characteristic of component 5 lies below the associated corridor, it can be concluded that

- component 5 is a key component, that is, its force-deformation characteristic is a key parameter, and
- it needs to be reinforced to lie within its corridor and to turn the bad into a good design.

### 6.2 Design improvement

A straightforward reinforcement, for example by increasing the sheet metal thickness, is unfortunately not a good design measure: the force-deformation characteristic of component 5 is already at the upper limit for deformations close to 0. By analyzing the deformation of component 5 during the crash, however, an appropriate design measure can be identified. The detailed finite element simulation

<sup>2</sup>For component 4, the force-deformation characteristic lies outside the corridor at very large deformations where the force measurement is assumed to be inaccurate. For components 6 and 9, the force-deformation characteristic lies outside the corridor for deformation intervals that are much smaller than the total deformation.

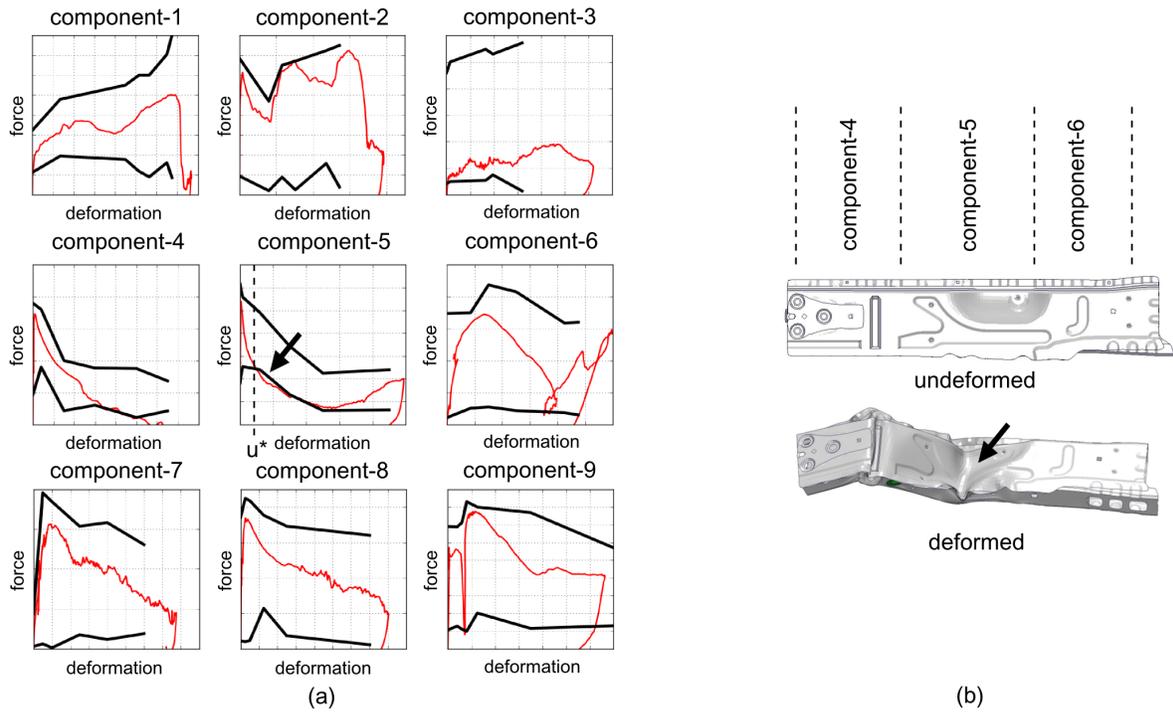


Fig. 9. (a) Measured force-deformation characteristics of design 1 with  $a > a_c$  and corridors  $\Omega_5$ . (b) The front rail undeformed and deformed.

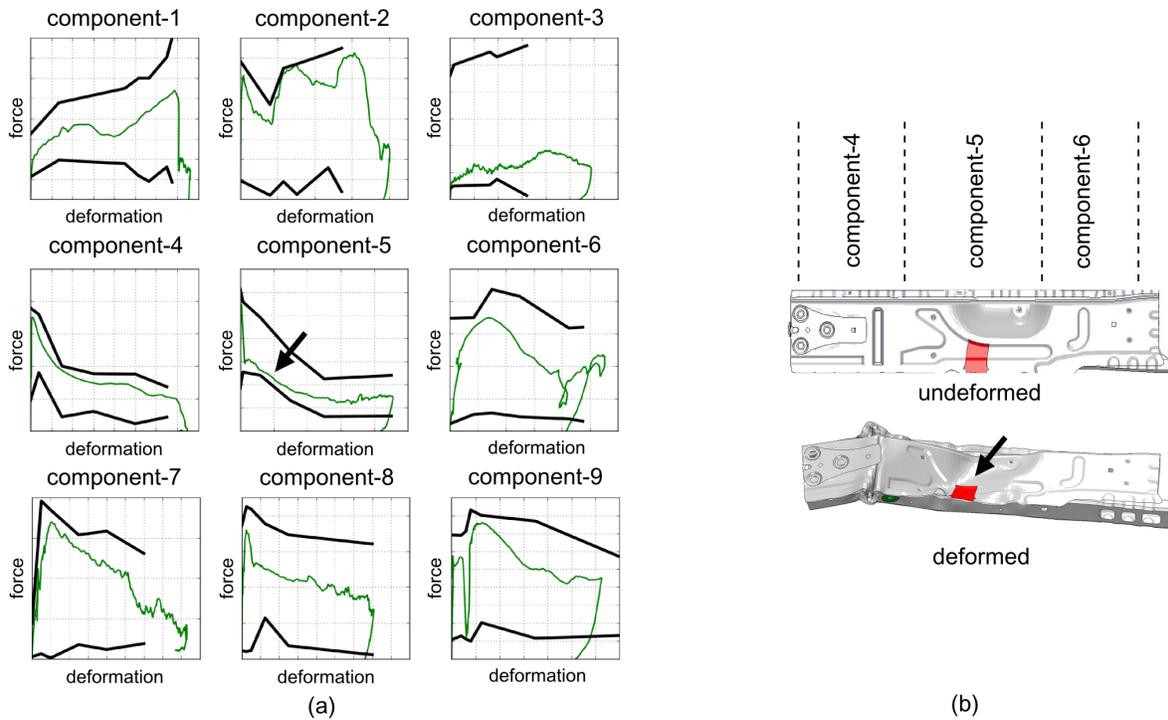


Fig. 10. (a) Measured force-deformation characteristics of design 2 with  $a < a_c$  and corridors  $\Omega_5$ . (b) The reinforced front rail undeformed and deformed.

shows that at the deformation  $u^*$  the deformation force drops below the corridor. This happens exactly when the profile of component 5 collapses by forming a distinct fold as shown in Figure 9(b).

The fold forms at a location that does not deform before the profile collapses. Therefore, a local reinforcement of this

location has no effect on the force-deformation characteristic for  $u < u^*$ , and the force-deformation characteristic does not cross the upper boundary line for deformations close to 0. It does nevertheless increase the deformation force at  $u \approx u^*$ , as intended.

The corridor provides a target region for the required

force-deformation characteristic. A target region rather than a target point is necessary, as the component properties cannot be controlled exactly, that is, force deformation-characteristics can assume only particular shapes that are not known in advance. In this sense, the component properties are *uncertain*. Wide corridors are necessary for successful design work under uncertainty.

The reinforcement is realized by increasing the sheet thickness locally. The resulting deformation and the force-deformation characteristics are shown in Figure 10. All force-deformation characteristics lie within their corridors<sup>3</sup>, and the maximum deceleration dropped below the critical value, see Figure 11.

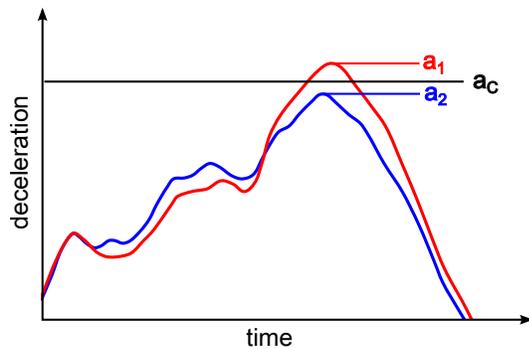


Fig. 11. Deceleration of the good and bad design.

Note that the force-deformation characteristic of component 4 also changed, although this was not intended. The local reinforcement in component 5 has a stiffening effect on component 4, because they both share parts of the same structural member. This may also be regarded as *uncertainty*. As the corridor for component 4 is wide enough, there is enough tolerance for the unintended variation: the force-deformation characteristic still lies within the corridor, and the design remains a good design.

The physical explanation for the improvement is similar to the one in Section 5.3: The reinforcement of the front rail, that is active at the deformation level  $u^*$ , decelerates the vehicle more in the beginning of the crash. The remaining kinetic energy and, thus, the maximum deceleration become smaller. If component 5 were reinforced such that the force is increased for deformation levels close to 0 (which would result in a force-deformation characteristic outside the corridor), a different component may collapse, causing the initial deceleration to decrease and the maximum deceleration to increase.

Note that the local thickness at the reinforcement was identified as relevant detail parameter by the physical information that was extracted from the force-deformation characteristics and the associated target corridors. Identifying relevant detail parameters by variation instead could be prohibitively expensive, because the local thicknesses of many

possible locations would have to be considered.

Using corridors for force-deformation characteristics as design goals helped identifying a key component, a key parameter and an appropriate design measure. In the example considered here, the design measure is small and modifies accurately the mechanical behavior of the nonlinearly interacting structural members.

## 7 Conclusion

A method was presented that provides precise information on how to turn a bad into a good design. The proposed method is applicable to arbitrary non-linear and high-dimensional design problems with uncertainty. It is based on an algorithm that seeks box-shaped solution spaces with a sufficiently large fraction of good designs. A solution space provides target intervals for parameters. The target regions are independent of each other and therefore decoupled. A solution space provides the information on how parameters have to be changed in order to reach the design goal. The algorithm was extended so as to optimize solution spaces under constraints. In order to reduce the number of key parameters that need to be changed, solution spaces are computed under the constraint that some parameter values of the bad design are already included.

An example problem with two input parameters was considered to validate the accuracy of the algorithm. The applicability to engineering problems was demonstrated by considering a front crash design problem. Starting from a bad design, the corresponding target regions were calculated and the key component was identified. By appropriate modification of the structural member, the design was changed to reach the design goal.

## References

- [1] Saltelli, A., Chan, K., and Scott, E., 2000. *Sensitivity analysis*. Wiley, New York.
- [2] Sudret, B., 2007. "Global sensitivity analysis using polynomial chaos expansions". *Reliability Engineering and System Safety*, doi:10.1016/j.ress.2007.04.002.
- [3] Wagner, S., 2007. "Global sensitivity analysis of predictor models in software engineering". *Proceedings of the Third International Workshop on Predictor Models in Software Engineering (PROMISE '07)*.
- [4] Sobol, I., 1993. "Sensitivity analysis for nonlinear mathematical models". *Mathematical Modelling and Computational Experiment*, **1(4)**, pp. 407–414.
- [5] Beyer, H.-G., and Sendhoff, B., 2007. "Robust optimization. a comprehensive survey". *Computer Methods in Applied Mechanics and Engineering*, **196**, pp. 3190–3218.
- [6] Lehar, M., and Zimmermann, M., 2012. "An inexpensive estimate of failure probability for high-dimensional systems with uncertainty". *Structural Safety*, **36–37**, pp. 32–38.
- [7] Zimmermann, M., and von Hoessle, J., 2013. "Computing solution spaces for robust design". *Interna-*

<sup>3</sup>The forces measured in components 5,6,8 and 9 cross the corridor lines only when unloading, that is, when  $\dot{u} < 0$ .

*tional Journal for Numerical Methods in Engineering*, **94**, pp. 290–307.

- [8] Graff, L., Harbrecht, H., and Zimmermann, M., 2012. “On the computation of solution spaces in high dimensions”. *Preprint SPP1253-138, DFG Priority Program 1253*.
- [9] Zimmermann, M., 2013. “Vehicle front crash design accounting for uncertainties”. *Proceedings of the FISITA 2012 World Automotive Congress*, **197**, pp. 83–89.
- [10] Siebertz, K., van Bebbber, D., and Hochkirchen, T., 2010. *Statistische Versuchsplanung: Design of Experiments (DoE)*. Springer, Berlin Heidelberg.
- [11] Kim, H., Rideout, D., Papalambros, P., and Stein, J., 2003. “Analytical target cascading in automotive vehicle design”. *Journal of Mechanical Design*, **125**, pp. 474–480.
- [12] Fender, J., Duddeck, F., and Zimmermann, M., 2012. “On the calibration of simplified crash models”. *Under review*.
- [13] Kerstan, H., and Bartelheimer, W., 2009. “Innovative prozesse und methoden in der funktionsauslegung - auslegung fuer den frontcrash”. *Fahrzeugsicherheit, VDI-Berichte*, **2078**.