

Two-Stage Stochastic and Deterministic Optimization

Tim Rzesnitzek, *
Dr. Heiner Müllerschön, †
Dr. Frank C. Günther,*
Michal Wozniak*

Abstract

The purpose of this paper is to explore some interesting aspects of stochastic optimization and to propose a two-stage optimization process for highly nonlinear automotive crash problems.

In the first stage of this process, a preliminary stochastic optimization is conducted with a large number of design variables. The stochastic optimization serves the dual purpose of obtaining a (nearly) optimal solution, which need not be close to the initial design, and of identifying a small set of design variables relevant to the optimization problem.

In the second stage, a deterministic optimization using only the set of relevant design variables is conducted. The result of the preceding stochastic optimization is used as the starting point for the deterministic optimization.

This procedure is demonstrated with a van-component model (previously introduced in [1]) used for crash calculations. LS-OPT [4] is used due to its ability to perform both stochastic (Latin Hypercube) and deterministic optimization.

1 Stage I: Stochastic Optimization

Stochastic optimization is often performed using the Monte Carlo method, where each design variable is individually constrained to lie within a user-specified upper and lower bound. The input values for these design variables are then determined randomly within their respective bounds.

The Latin Hypercube method provided by LS-OPT is also a random experimental design process. Figure 1 illustrates the algorithm used for generating random values for a small set of design variables (in this case, t_1 through t_4). The range of each design variable is subdivided into n equal intervals, where n is the number of experiments (in this case, the number of experiments is 5). A set of input variables is then determined for each experiment by randomly selecting one of the n sub-divisions for each variable. Each sub-division may be used within only one single experiment, thus ensuring that the entire design space is covered.

The distribution of design points for the Monte Carlo and Latin Hypercube algorithms can be influenced by adding additional constraints based on response functions. This requires a response function that depends linearly on the design variables. The responses of a new

*DaimlerChrysler, Commercial Vehicles Analysis

†DYNAmore GmbH

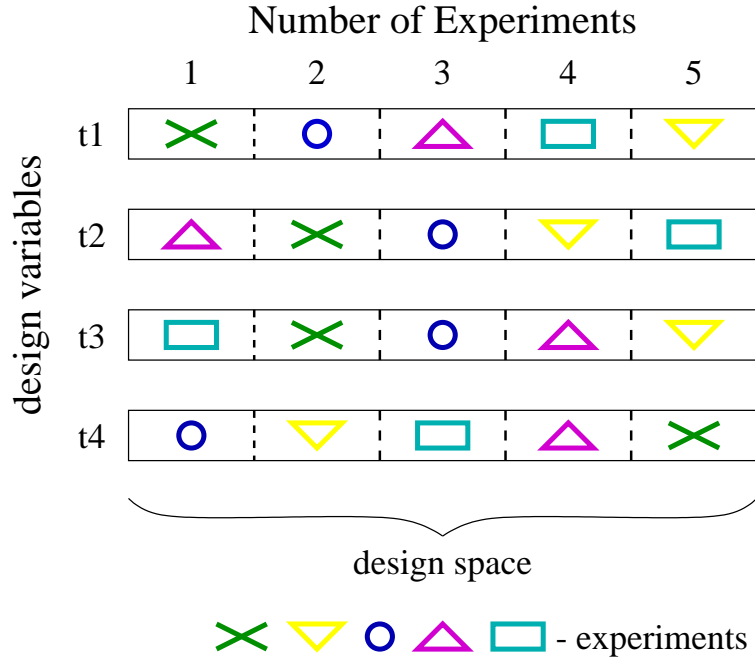


Figure 1: Latin Hypercube algorithm of LS-OPT

design can then be estimated by using the initial values of the design variables together with the corresponding initial responses.

For the optimization of a vehicle body, the mass of sheet metal parts linearly depends on sheet thickness, provided that the geometry remains unchanged. If the sheet thickness is a design variable, the mass for a new value of the sheet thickness can be estimated by multiplying the new sheet thickness by the ratio of initial sheet thickness to initial mass. Hence the total mass of all parts involved in an optimization can be estimated directly after the input values for an experiment are generated and can thus be used as a variable to define a constraint.

There are several different ways for enforcing the upper and lower bounds for a constraint variable:

1. In LS-OPT, a large number of initial design points is created. All points within this initial set exceeding the constraint are then shifted until they lie on the bound that has been exceeded. This is achieved, for example, by varying the sheet thicknesses appropriately. Depending on the number of points created for the initial set, a large number of design points may lie on the bounds of the constraint. Thus, in the second stage, the final set of design points is extracted from the initial set by the help of the D -optimality criterion [4].

The D -optimality criterion can be used with either a linear or quadratic approximation. The particular choice depends upon the equation applied to estimate the constraint variable from the design variables. If, for example, mass is defined as a constraint variable and the sheet thickness of each sheet metal part are the design variables, then linear approximation has to be used, since mass is linearly dependent from sheet thickness.

2. For the construction of a Monte Carlo design, experiments outside of the constraint are filtered out when the set of design points is created. The generation of random values for

a single experiment is repeated until a design is found for which the constraint variable meets the bounds.

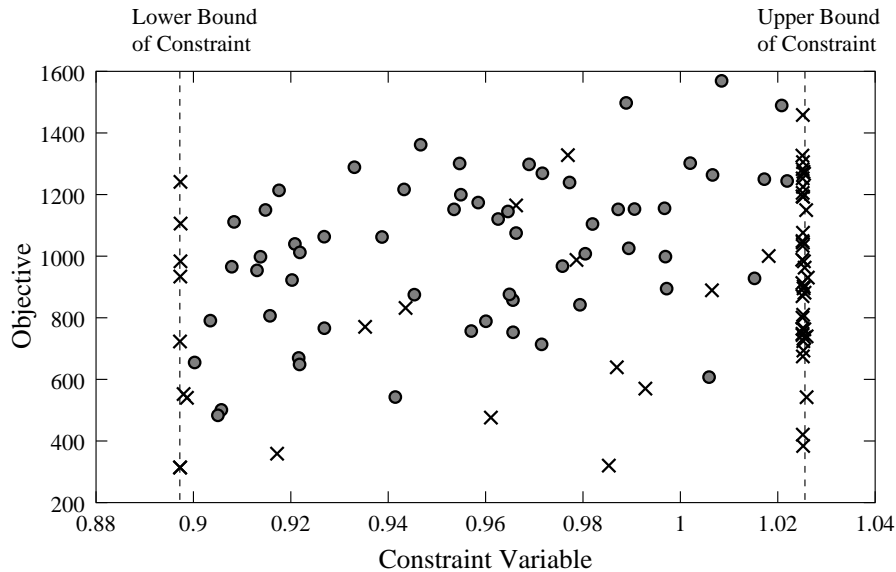


Figure 2: Distribution of design points for Latin Hypercube and the Monte Carlo algorithm

In the diagram of figure 2 the objective variable is plotted vs. the constraint variable for sets of design points generated by LS-OPT (Latin Hypercube) and the Monte Carlo algorithm. The design points produced by LS-OPT (x) often lie on the borders of the constraint, which results in an irregular distribution of the set of design points. Obviously the D -optimality criterion applied to extract the final set of design points from such an initial set does not strongly improve the distribution of the design points.

In contrast to Latin Hypercube, the set of design points created by the Monte Carlo algorithm shows the typical distribution of a random experimental design, where no design point is lying on a bound and the number of design points is decreasing towards the bounds according to a Gaussian distribution. Thus the number of design points suitable for analysis is large in comparison to Latin Hypercube.

Figure 3 illustrates the advantage of performing a constrained Monte Carlo design. Without any constraint, a simulation would be carried out for all the design points plotted on the diagram. By defining a constraint, the experiments that are uninteresting or irrelevant are effectively filtered out. Thus the number of simulations required can be reduced.

2 Stage II: Identifying Important Variables for Deterministic Optimization. Regression Analysis

2.1 General Concepts

In statistical literature, there are many different representations of the concepts involved in finding the underlying dependencies between input and output variables based on a number of experiments, e.g. [3]. Let us show one possible derivation:

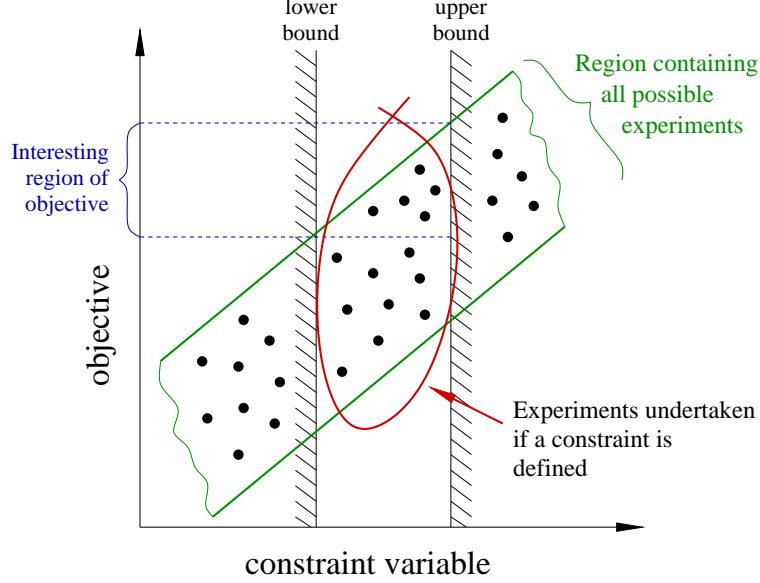


Figure 3: Definition of constraints for stochastic optimization

Let there be p experiments (simulations), each of which is characterized by an $n \times 1$ vector $\hat{\mathbf{x}}_i$ of input variables and one output variable \hat{y}_i . In crash simulations, $\hat{\mathbf{x}}_i$ could be a vector of sheet thicknesses and other parameters, and \hat{y}_i could be a measure of the crashworthiness of the resulting structure. There are several commercial software tools available that can generate p sets of input variables, and run p simulations to obtain output variables y_i . We used LS-OPT [4] as well as our own Perl [5] programs to generate the examples shown in Section 4.

Let us define a $n \times p$ matrix of all input values,

$$\hat{\mathbf{X}} = \begin{bmatrix} \hat{\mathbf{x}}_1 & \hat{\mathbf{x}}_2 & \dots & \hat{\mathbf{x}}_p \end{bmatrix}, \quad (1)$$

and a corresponding $1 \times p$ vector of output values,

$$\hat{\mathbf{y}}^T = \begin{bmatrix} \hat{y}_1 & \hat{y}_2 & \dots & \hat{y}_p \end{bmatrix}. \quad (2)$$

It is customary to center the values for $\hat{\mathbf{x}}_i$ and \hat{y}_i with their mean values over all experiments:

$$\bar{\mathbf{X}} = \hat{\mathbf{X}} - \frac{1}{p} \sum_{i=1,p} \hat{\mathbf{x}}_i = \hat{\mathbf{X}} - \frac{1}{p} \hat{\mathbf{X}} \mathbf{1}_{p \times p}, \quad (3)$$

where $\mathbf{1}_{p \times p}$ is a $p \times p$ matrix of ones. Analogously,

$$\bar{\mathbf{y}}^T = \hat{\mathbf{y}}^T - \frac{1}{p} \hat{\mathbf{y}}^T \mathbf{1}_{p \times p}. \quad (4)$$

Let us now define covariance matrices¹ for $\bar{\mathbf{X}}$ and $\bar{\mathbf{y}}^T$:

$$\mathbf{S}_x^2 = \bar{\mathbf{X}} \bar{\mathbf{X}}^T \quad (5)$$

$$\mathbf{S}_y^2 = \bar{\mathbf{y}}^T \bar{\mathbf{y}}. \quad (6)$$

¹in statistical literature, covariance matrices usually contain a factor $\frac{1}{p}$. However, this is not necessary for our purposes.

It is easily established that \mathbf{S}_x^2 is positive semidefinite. In order to obtain the standard deviation² \mathbf{S}_x with

$$\mathbf{S}_x^2 = \mathbf{S}_x \mathbf{S}_x, \quad (7)$$

we can perform an eigenvalue decomposition³ of \mathbf{S}_x^2 and take the square roots of the eigenvalues, viz.

$$\mathbf{S}_x^2 = \mathbf{T}^T \mathbf{diag} \{ \lambda_i \} \mathbf{T} \quad (8)$$

$$\mathbf{S}_x = \mathbf{T}^T \mathbf{diag} \left\{ \sqrt{\lambda_i} \right\} \mathbf{T} \quad (9)$$

$$\lambda_i \geq 0 \quad \text{due to positive semidefiniteness.} \quad (10)$$

Of course,

$$S_y = \sqrt{S_y^2} \quad (11)$$

is computed trivially.

Let us now introduce the transformed variables

$$\mathbf{X} = \mathbf{S}_x^{-1} \bar{\mathbf{X}} \quad (12)$$

$$\mathbf{y}^T = S_y^{-1} \bar{\mathbf{y}}^T, \quad (13)$$

assuming \mathbf{S}_x is invertible and $S_y \neq 0$. These variables are normalized in the sense that

$$\mathbf{X} \mathbf{X}^T = \mathbf{S}_x^{-1} \bar{\mathbf{X}} \bar{\mathbf{X}}^T \mathbf{S}_x^{-1} = \mathbf{S}_x^{-1} \mathbf{S}_x^2 \mathbf{S}_x^{-1} = \mathbf{I} \quad (14)$$

$$\mathbf{y}^T \mathbf{y} = S_y^{-1} \bar{\mathbf{y}}^T \bar{\mathbf{y}} S_y^{-1} = S_y^{-1} S_y^2 S_y^{-1} = 1. \quad (15)$$

Our goal is to find a linear approximation with a $n \times 1$ vector \mathbf{w} such that the residual

$$\mathbf{r} = \mathbf{y} - \mathbf{X}^T \mathbf{w} \quad (16)$$

is minimized. Minimizing the quadratic norm of \mathbf{r} is equivalent to minimizing $\mathbf{r}^T \mathbf{r}$, hence the name Least Squares Approximation.

With the orthogonality conditions 14 and 15, we obtain

$$\mathbf{r}^T \mathbf{r} = (\mathbf{y}^T - \mathbf{w}^T \mathbf{X}) (\mathbf{y} - \mathbf{X}^T \mathbf{w}) \quad (17)$$

$$= \mathbf{y}^T \mathbf{y} - \mathbf{y}^T \mathbf{X}^T \mathbf{w} - \mathbf{w}^T \mathbf{X} \mathbf{y} + \mathbf{w}^T \mathbf{X} \mathbf{X}^T \mathbf{w} \quad (18)$$

$$= 1 - 2\mathbf{w}^T \mathbf{X} \mathbf{y} + \mathbf{w}^T \mathbf{w} \quad (19)$$

In index notation, this becomes

$$r_i r_i = 1 - 2w_i X_{ij} y_j + w_i w_i, \quad (20)$$

where summation over double indices is implied.

The minimum of equation 20 can be found by taking the derivative with respect to \mathbf{w} and setting it to zero. In index notation:

$$\frac{\partial (r_i r_i)}{\partial w_k} = -2\delta_{ik} X_{ij} y_j + \delta_{ik} w_i + w_i \delta_{ik} \quad (21)$$

$$= -2X_{kj} y_j + 2w_k = 2(w_k - X_{kj} y_j). \quad (22)$$

²Give or take a factor $\frac{1}{\sqrt{p}}$.

³This process is significantly simplified if the input variables \mathbf{x} are completely independent of each other. In such a case, the covariance matrix becomes diagonal: $\mathbf{S}_x^2 = \mathbf{diag} \{ \lambda_i \}$ and $\mathbf{T} = \mathbf{I}$, hence $\mathbf{S}_x = \mathbf{diag} \{ \sqrt{\lambda_i} \}$

The Kronecker delta δ_{ij} is defined as

$$\delta_{ij} = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{otherwise.} \end{cases} \quad (23)$$

Back in matrix notation, we find that

$$\mathbf{w} = \mathbf{X}\mathbf{y}. \quad (24)$$

Our linear approximation becomes

$$y = \mathbf{w}^T \mathbf{x} = \mathbf{y}^T \mathbf{X}^T \mathbf{x}. \quad (25)$$

This gives rise to interpreting the elements of \mathbf{w} as the significance of each normalized input variable in the vector

$$\mathbf{x} = \mathbf{S}_x^{-1} \bar{\mathbf{x}} \quad (26)$$

for the result of the simulation. Generally, the elements of \mathbf{x} are a linear combination of the elements of $\bar{\mathbf{x}}$.

In terms of centered variables 3 and 4 we obtain

$$\bar{y} = \underbrace{\bar{\mathbf{y}}^T \bar{\mathbf{X}}^T (\bar{\mathbf{X}} \bar{\mathbf{X}}^T)^{-1}}_{\bar{\mathbf{w}}} \bar{\mathbf{x}}. \quad (27)$$

The weight vector $\bar{\mathbf{w}}$ contains the non-normalized regression coefficients. Hence, its values cannot be compared to each other directly.

2.2 Variable Screening in LS-OPT (ANOVA)

LS-OPT provides the capability of performing a significance test for each variable in order to remove those coefficients or variables which have a small contribution to the design model [4]. For each response, the variance of each variable with respect to that response is tracked and tested for significance using the partial F-test [2]. The importance of a variable is penalized by both a small absolute coefficient value (weight) computed from the regression, as well as by a large uncertainty in the value of the coefficient pertaining to the variable. A 90% confidence level is used to quantify the uncertainty and the variables are ranked based on this level.

The $100(1 - \alpha)\%$ confidence interval for the regression coefficients b_j , $j = 1, \dots, n$ is determined by

$$b_j - \frac{\Delta b_j}{2} \leq \beta_j \leq b_j + \frac{\Delta b_j}{2} \quad (28)$$

where

$$\Delta b_j = 2 t_{\alpha/2, p-(n+1)} \sqrt{\hat{\sigma}^2 C_{jj}} \quad (29)$$

and $\hat{\sigma}^2$ is an unbiased estimator of the variance σ^2 given by

$$\hat{\sigma}^2 = \frac{\varepsilon^2}{p - (n + 1)} = \frac{\sum_{i=1}^p (y_i - \hat{y}_i)^2}{p - (n + 1)} \quad (30)$$

C_{jj} is the diagonal element of $(\hat{\mathbf{X}}^T \hat{\mathbf{X}})^{-1}$ corresponding to b_j and $t_{\alpha/2, p-(n+1)}$ is the t or Student's Distribution. ε is the error between the response y_i and the estimated response \hat{y}_i

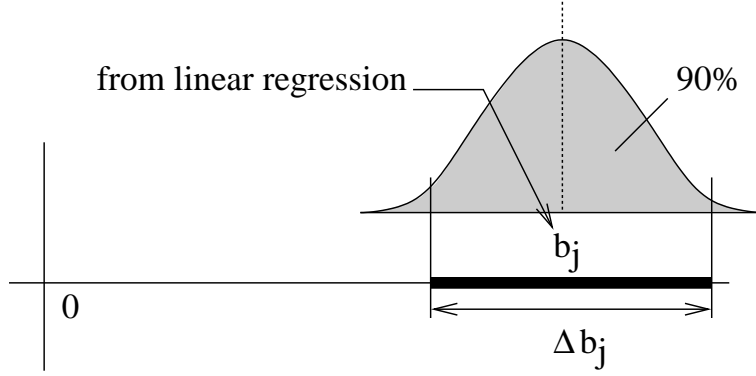


Figure 4: Confidence interval for the regression coefficient for variable j

evaluated by linear regression. $100(1 - \alpha)\%$ therefore represents the level of confidence that b_j will lie within the computed interval, (4).

Figure 5 demonstrates graphically the criterion for the ranking of the variables. A 90% confidence level defines the lower bound and the variables are ranked according to this lower bound. If the lower bound is less than zero, the variable is declared as insignificant.

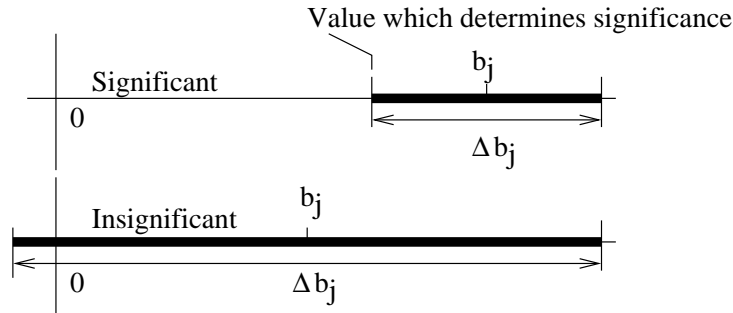


Figure 5: Criterion for the significance of a variable

Each variable is normalized with respect to the size of the design space. Thus all variables range within the interval $[0,1]$ and a reasonable comparison can be made for variables of different kinds.

3 Description of Van Component Model for Crash

The scope of the optimization is an assembly of a vehicle body for a commercial van as shown in Figure 6.

The main components of that assembly are the first cross member and the front part of the longitudinal member of the frame. Furthermore, the absorbing box between first cross member and longitudinal member, parts of the wheelhouse and the closing panels of the longitudinal member are also included in the assembly.

In Figure 7, boundary conditions and part numbers of important components are illustrated.

The assembly is connected to the bottom sheet of the vehicle body at the flanges of the front and back closing panels. In the simulations, the assembly is fixed in y - and z -directions

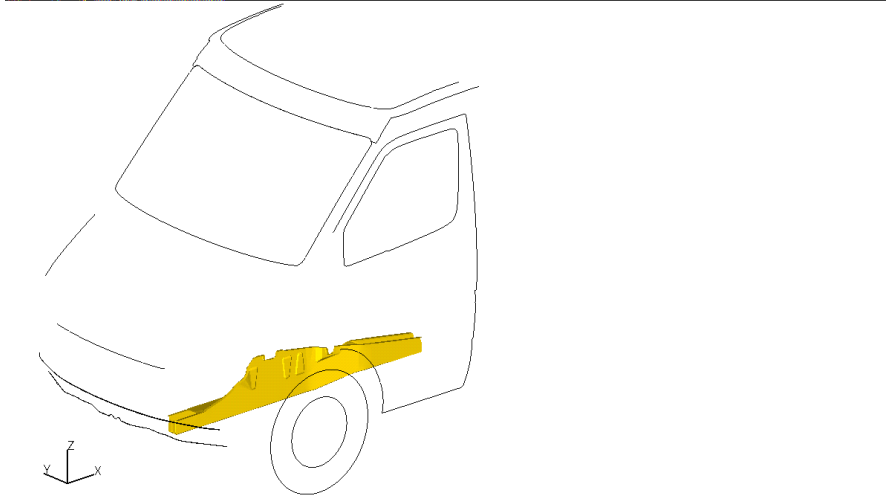


Figure 6: Position of the assembly within the vehicle

at these flanges. The assembly is displaced in x-direction at a constant velocity and deforms at the stonewall.

For simulation purposes, only half of the frame is represented (since geometry and loads are symmetric to the yz-plane) and the first cross member is cut off at the yz-plane. Therefore, the translation in the y-direction and the rotation about the x- and z-axis have to be fixed along the cutting edge.

4 Application of Concepts to the Example Model

A two-stage optimization has been performed for the component model described above. In the first stage, two different kinds of random distributions are applied in order to generate samples of design variables. The two methods are a *Monte Carlo* distribution provided by a perl programm and a *Latin Hypercube* design combined with D-Optimal strategy provided by LS-OPT. The two approaches are compared to each other.

In the second stage, a deterministic optimization using the successive response surface methodology of LS-OPT is performed. The start design for this is the best set of variables resulting out of the first stage.

4.1 Description of the optimization problem

The design variables consists of the sheet thicknesses of 15 parts and an additional bead located on the longitudinal member on the outer and inner side. The geometry and the location of the bead is identical on both sides (symmetry). The bead is defined by 5 design variables, the depth t , the x-location x_{min}, x_{max} and the z-location z_{min}, z_{max} . Preprocessing of the LS-DYNA input file for the bead is done by a Perl programm (for additional information, see [1]). In total there are 20 design variables.

The objective of the optimization is to maximize the ratio of the maximum value of the internal energy E_{max} and the mass M of the considered components:

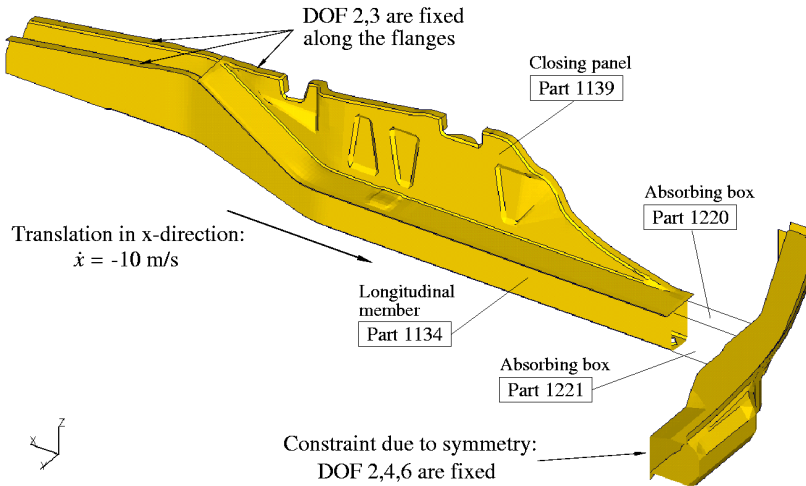


Figure 7: Boundary Conditions

$$E_M = \frac{E_{\max}}{1000 M} \longrightarrow \max E_M. \quad (31)$$

Two constraints are required to be fulfilled:

- Bounds for the Mass: $0.895 > M > 1.022$
- Upper bound for the stonewall force: $\max\text{RWFORC} = 1.25$

For the evaluation of the maximum rigid wall force, $\max\text{RWFORC}$, the history curve (ASCII-file `RWFORC`) is filtered using SAE 180.

4.2 Stage I: *Monte Carlo* and *Latin Hypercube* Distributions

4.2.1 *Monte Carlo* Distribution

In the first attempt, 60 simulations with 60 different sets of input variables are performed. Variable sets are generated by a Perl program using a modul for generating random values. The total mass of the component model is checked directly and sets that are outside the specified bounds are rejected. This is done until 60 feasible variable sets are available.

The three runs with the maximum ratio E_M and with a stonewall force less than 1.5 are chosen to be the starting values for 15 additional runs each. Thus, three sets of 15 variables vary their values with these starting values as centerpoints. This is repeated for the three best points of the previous 45 runs with $3 \times 15 = 45$ more runs.

Thus, in total $1 \times 60 + 3 \times 45 + 3 \times 45 = 150$ runs are put into execution.

4.2.2 *Latin Hypercube* Distribution

The *Latin Hypercube* design is provided by LS-OPT. LS-OPT generates an initial 1000 *Latin Hypercube* points (sets of variables, see Section 1). Points that violate the mass constraint are moved automatically into the feasible mass region. Out of these 1000 feasible points, a sub-set

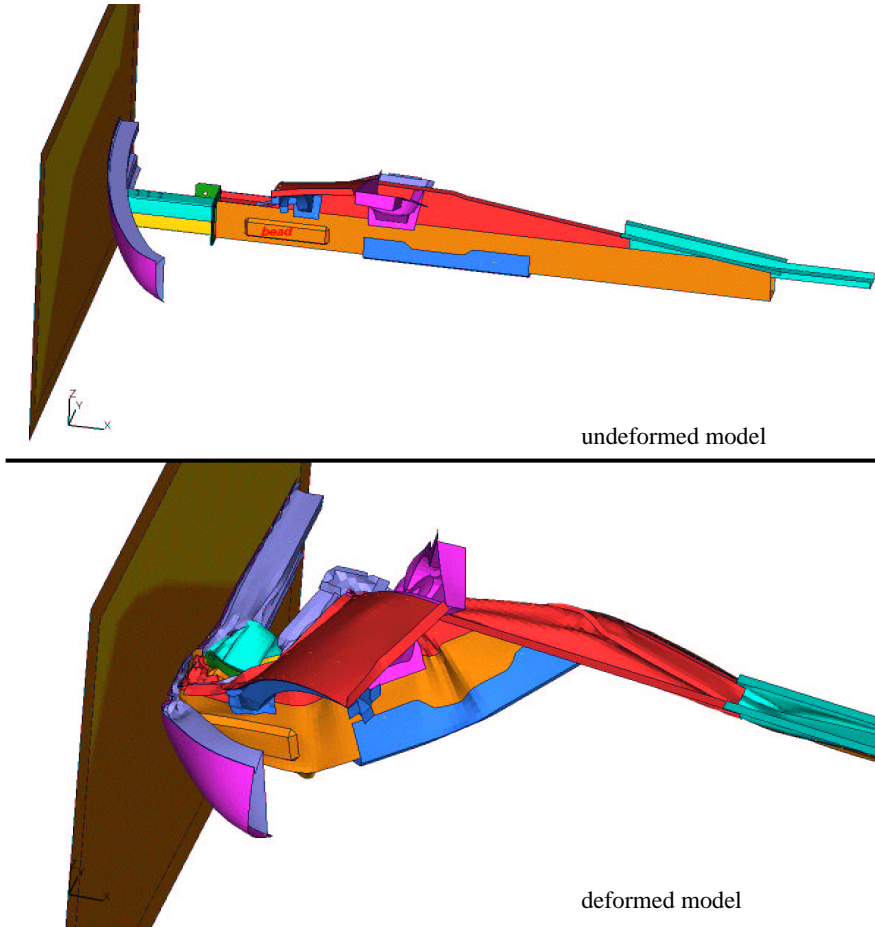


Figure 8: Simulation model - initial and deformed configuration

of 60 points is selected by applying the D-Optimal criterion (see [4]). Points chosen by the D-Optimal criterion generally display good capabilities with respect to linear regression.

In the same manner, 30 more points are generated using as the centre point the best run out of the first 60 runs.

Thus, in total $1 \times 60 + 1 \times 30 = 90$ runs are put into execution.

4.3 Stage II: Deterministic Optimization using LS-OPT

In order to reduce the amount of variables for the deterministic optimization in Stage II, an ANOVA analysis (Section 2.2) is performed. Therefore, the first 60 runs of the *Latin Hypercube* Distribution are used. Figure 9 shows the results of the variable screening. LS-OPT ranks the variables by using the bound of the 90% confidence interval that is closest to zero as the ranking number.

The four most significant variables $t1134$, $t1139$, $t1210$ and $t1221$ are chosen for the deterministic optimization using the response surface method in LS-OPT. For the starting variables the values are taken out of the best run of the 150 *Monte Carlo* simulations. The values of the remaining 16 variables are also taken out of the optimal *Monte Carlo* run and are kept constant. It is slightly surprising, that in the third iteration no improvement of E_M is observed (Figure 10). Thus, the starting value for the deterministic optimization comes from

Approximating Response 'E_M' using 60 points						

Linear Function Approximation:						

Mean response value	=	0.8890				
RMS error	=	0.2238	(12.33%)			
Maximum Residual	=	0.4611	(25.40%)			
Average Error	=	0.1824	(10.05%)			
Square Root PRESS Residual	=	0.3687	(20.31%)			
Variance	=	0.0751				
R^2	=	0.8524				
R^2 (adjusted)	=	0.8524				
R^2 (prediction)	=	0.5995				

Individual regression coefficients: confidence intervals						

Coeff	Coeff. Value	Confidence Int.(90%)		Confidence Int.(95%)		% Confidence not zero
		Lower	Upper	Lower	Upper	

t1134	1.58	1.255	1.906	1.19	1.971	100
t1139	1.015	0.7089	1.321	0.6476	1.382	100
t1140	-0.6216	-0.9263	-0.3169	-0.9874	-0.2559	100
t1144	-0.2674	-0.6063	0.07145	-0.6743	0.1394	80
t1210	-1.055	-1.416	-0.6931	-1.489	-0.6206	100
t1211	-0.4619	-0.8002	-0.1236	-0.868	-0.05573	97
t1220	-0.5574	-0.8209	-0.294	-0.8737	-0.2411	100
t1221	-0.9985	-1.337	-0.6604	-1.404	-0.5926	100
t1222	0.3766	0.01415	0.739	-0.05852	0.8117	91
t1223	-0.2613	-0.6392	0.1165	-0.7149	0.1923	74
t1224	-0.1445	-0.4433	0.1543	-0.5032	0.2142	57
t1410	-0.0268	-0.3932	0.3396	-0.4666	0.413	10
t1411	-0.05109	-0.3883	0.2861	-0.4559	0.3537	20
t1412	-0.544	-0.8206	-0.2674	-0.876	-0.212	100
t1413	-0.3308	-0.7064	0.04484	-0.7818	0.1202	85
xmin	0.3381	-0.06837	0.7446	-0.1499	0.8261	83
xmax	-0.3318	-0.5868	-0.07684	-0.638	-0.02572	96
zmin	0.2477	-0.02525	0.5206	-0.07996	0.5753	86
zmax	-0.1905	-0.5596	0.1786	-0.6336	0.2526	60
t	0.1564	-0.1283	0.4412	-0.1854	0.4983	63

Ranking of terms based on bound of confidence interval						

Coeff	Absolute Value (90%)	10-Scale				

t1134	1.255	10.0				
t1139	0.7089	5.6				
t1210	0.6931	5.5				
t1221	0.6604	5.3				
t1140	0.3169	2.5				
t1220	0.294	2.3				
t1412	0.2674	2.1				
t1211	0.1236	1.0				
xmax	0.07684	0.6				
t1222	0.01415	0.1				
zmin	Insignificant	0.0				
t1413	Insignificant	0.0				
xmin	Insignificant	0.0				
t1144	Insignificant	0.0				
t1223	Insignificant	0.0				
t	Insignificant	0.0				
t1224	Insignificant	0.0				
zmax	Insignificant	0.0				
t1411	Insignificant	0.0				
t1410	Insignificant	0.0				

Figure 9: Output of LS-OPT for variable screening with respect to the response E_M

the sample distribution of the second iteration. Within the *Latin Hypercube* samples, the best result with respect to the maximum Energy-Mass ratio E_M is significantly worse than the best result for the *Monte Carlo* samples (compare Figure 10). This is probably due to the movement of the variables in order to fulfill the mass constraints. This leads to a degenerated sample distribution with an accumulation of samples at the lower and upper bound of the mass constraints (Figure 2).

Figure 10 shows the results of the *Monte Carlo* and the *Latin Hypercube* simulations of Stage I and the result of an optimum run found by LS-OPT using the response surface methodology in Stage II. A significant additional improvement of the response E_M by the deterministic approach is observed.

5 Conclusions

The present paper shows the proceeding of a two-stage optimization, where in the first stage a stochastic approach is considered and in the second stage, starting from the best result of the first stage, a deterministic optimization using the Response Surface Method in LS-OPT is performed.

The variation of the design variables (sheet thicknesses) is restricted by a mass constraint. This is done in order to achieve only samples whose mass lies within a desired range. This ensures that needless simulations are avoided.

On the basis of a set of *Latin Hypercube* simulations, a variable screening using the ANOVA (ANalysis Of VAriance) feature in LS-OPT is performed. With this tool, those variables detected as insignificant are removed for the second stage of the optimization process.

The starting variables for the deterministic optimization in the second stage are taken out of the best run of the *Monte Carlo* simulations. A significant additional improvement of the Energy/Mass ratio by the deterministic approach is observed. In total, an improvement of 32.78 % from the initial configuration (1.205) to the optimum result of LS-OPT (1.60) is achieved (Figure 10).

The combination of stochastic and deterministic optimization has shown to be a reasonable method for obtaining sensible input values for a deterministic optimization. Furthermore, it can be expected that in many cases, the convergence of the deterministic algorithm to local extremal values with respect to the objective may be avoided.

References

- [1] F. Günther and T. Rzesnitzek. Crashworthiness optimization of van components with ls-opt. In *Proceedings of 19th CAD-FEM Users' Meeting*, 2001.
- [2] R. Myers and M. D.C. *Response Surface Methodology*. Wiley New York, 1995.
- [3] H. Rinne. *Statistische Analyse multivariater Daten: Einführung*. R. Oldenbourg Verlag München Wien, 2000.
- [4] N. Stander and K. Craig. *LS-OPT User's Manual: Design Optimization Software for the Engineering Analyst*. Livermore Software Technology Corporation, 2002.
- [5] L. Wall, T. Christiansen, and J. Orwant. *Programming Perl*. O'Reilly & Associates, 3rd edition, 2000.

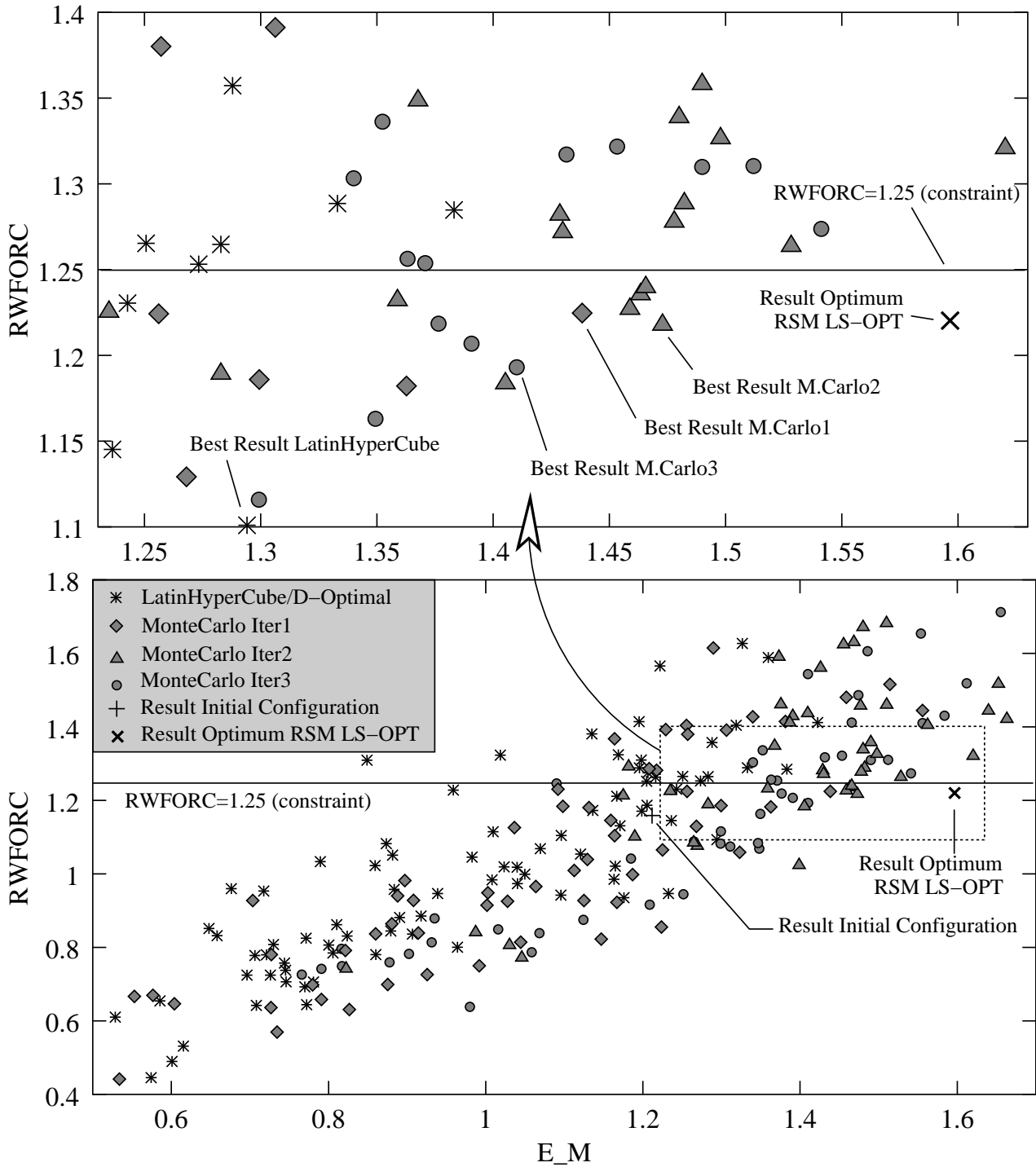


Figure 10: Response E_M vs. the corresponding rigid wall force RWFORC for *Latin Hypercube* and *Monte Carlo* Samples and for the final optimum found by the Response Surface Method (RSM) in Stage II