Multidisciplinary Design Optimization of Body Exterior Structures

Michel H.J.W. Paas(✉) and Hessel C. van Dijk

Ford Werke GmbH, Cologne, Germany
mpaas@ford.com, hvandijk@ford.com

Abstract. Multidisciplinary Design Optimization (MDO) uses optimization methods to solve design problems by incorporating all relevant disciplines simultaneously. In vehicle engineering the objective typically is to optimize weight while maintaining performance involving various load cases from multiple CAE attributes. The MDO process can be divided into three successive steps: (1) data generation entailing creation, submission and post-processing of various DoE's, (2) meta-modeling and optimization, and (3) validation of optimization proposals. Recent efforts have been targeted at improving the efficiency of the "meta-modeling and optimization" phase in terms of throughput time and quality of solutions by the development of:

- Automated preparation and submission of CAE models to the HPC cluster.
- Automated meta-modeling producing reproducible, high-quality meta-models based on Gaussian Processes with Automatic Relevance Determination at reduced HPC work load.
- An enhanced NSGA-II algorithm for constrained multi-objective optimization.

These developments resulted in 20% reduction in throughput times in conjunction with further weight saving potential. The viability of these improvements is illustrated by findings in recent optimization projects.

Keywords: Multidisciplinary design optimization · Vehicle structures

1 Introduction

Multidisciplinary Design Optimization (MDO) uses optimization methods to solve design problems by incorporating all relevant disciplines simultaneously. In vehicle engineering the objective typically is to optimize the weight of the body exterior while maintaining attribute performance involving various load cases from crash safety, NVH, durability & strength, forming and vehicle dynamics. The MDO process, which has been embedded in Ford's Global Product Development System, can be divided into three phases (Fig. 1): (1) data generation entailing creation, submission and post-processing of various DoE's, (2) meta-modeling and optimization, and (3) validation of optimization proposals. The software supporting this process has been developed in-house in Matlab. It is noted that in the data generation phase commercially available

A. Schumacher et al. (eds.), *Advances in Structural and Multidisciplinary Optimization*,
https://doi.org/10.1007/978-3-319-67988-4_2

software (ModeFrontier) is deployed as well. As closed loop optimization of large finite element models is impracticable, the MDO process is relying on optimization of meta-models, which act as real-time substitutes of numerically intensive finite element models. In addition, they provide detailed system knowledge and can be used for scenario analysis and design sensitivity studies. In what follows emphasis will be on efficiency improvements in the "meta-modeling and optimization phase" of the MDO process.

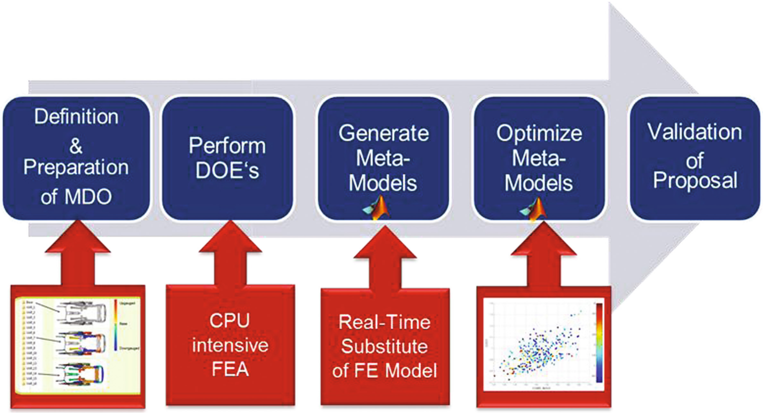

Fig. 1. MDO process

2 Gaussian Processes

Meta-model selection requires establishing a balanced trade-off between under-fitting and over-fitting the data obtained from the DOE. Over-fitting occurs when the meta-model is trying to fit the noise in the training data, which has a detrimental effect on the performance of the model on new data. Under-fitting refers to a meta-model that is too simple to model the training data and to generalize to new data [1]. The data obtained from DoE's are available as design vectors $\underline{x}i$ (i = 1, 2, …, N) with associated outputs $\underline{y}$. Then, a general model for observation y_i corrupted by independent and identically distributed Gaussian noise ε can be written as:

$$y_i = f(\underline{x}_i) + \varepsilon_i \quad \text{with} \quad \varepsilon_i = N\left(0, \sigma_n^2\right) \tag{1}$$

Gaussian Processes (GP) can be adopted to characterize the latent response f. A Gaussian Process is a stochastic process, such that any finite sub-collection of random variables has a multivariate Gaussian distribution [2, 3]. GP regression uses the similarity between points to predict the value for an unseen point from training data. A GP model is fully defined by a mean function $m(\underline{x})$ and covariance function $K(\underline{x}, \underline{x}')$:

$$\begin{bmatrix} f(\underline{x}_1) \\ \vdots \\ f(\underline{x}_n) \end{bmatrix} \sim N\left(\begin{bmatrix} m(\underline{x}_1) \\ \vdots \\ m(\underline{x}_1) \end{bmatrix}, \begin{bmatrix} K(\underline{x}_1, \underline{x}_1) & \cdots & K(\underline{x}_1, \underline{x}_1) \\ \vdots & \ddots & \vdots \\ K(\underline{x}_n, \underline{x}_1) & \cdots & K(\underline{x}_1, \underline{x}_1) \end{bmatrix} \right) \quad \text{or} \quad f \sim GP(m(.), K(.,.)) \tag{2}$$

Without loss of generality, the mean function is usually defined to be zero. Choosing among alternative covariance functions is a way of reflecting prior knowledge about the physical process under investigation. For new predictions based on given training data it is assumed that the prediction process is jointly Gaussian distributed with the training points. Then, using the rules for conditioning Gaussians the conditional distribution of new observations $\hat{\underline{y}}$ for known training data $\underline{y}, \underline{x}$, and new $\hat{\underline{x}}$ can be expressed as:

$$\hat{\underline{y}} \Big| \underline{y}, \underline{x}, \hat{\underline{x}} \sim N\left(\hat{\mu}, \hat{\Sigma}\right) \tag{3}$$

where:

$$\hat{\mu} = K(\hat{\underline{x}}, \underline{x})(K(\underline{x}, \underline{x}) + \sigma^2 I)^{-1} \underline{y} \tag{4}$$

$$\hat{\Sigma} = K(\hat{\underline{x}}, \underline{x}) + \sigma^2 I - K(\underline{x}, \underline{x})(K(\underline{x}, \underline{x}) + \sigma^2 I)^{-1} K(\underline{x}, \hat{\underline{x}}) \tag{5}$$

Model parameter settings θ are obtained by minimizing the negative log marginal likelihood (probability of the data given the model):

$$-\log p(y|\theta) = \frac{1}{2} \log \det K(\theta) + \frac{1}{2} y^T K^{-1}(\theta) y + \frac{N}{2} \log(2\pi) \tag{6}$$

Adopting Gaussian Processes for meta-modeling and optimization has a number of distinct advantages:

- The predictive uncertainty stemming from both the intrinsic noise σ_n and the errors in the parameter estimation procedure is quantified.
- Arbitrarily complex relationships can be fitted.
- Immune to over-fitting the model to the training data.
- Prior knowledge of the data can be incorporated via covariance functions.

It is noted that the training effort needed is $O(N^3)$. Hence, for large data sets so called sparse GP approximation methods or alternative modeling techniques, such as neural networks, should be preferred.

3 Efficiency Improvement Actions

Large-scale MDO projects are extremely numerically intensive. Typically over 50 DoE's are needed to cover all performance requirements. For each DoE several hundreds of CAE models with run times of up to 25 h are generated to adequately scan the design space. In this context compression of turnaround times is of paramount importance. Recently, significant improvements in MDO efficiency could be accomplished based on:

- Automated preparation and submission of CAE models to the HPC cluster.
- Automated Meta-Modeling for fast, reproducible, and high quality meta-models.
- Faster multi-objective optimization based on an enhanced NSGA-II algorithm.

These efficiency improvements resulted in an overall lead time compression by 20% in conjunction with an increased weight savings potential.

3.1 Automated Meta-modeling

Meta-modeling is an interactive, time consuming and tedious process. Modular software for fully automated meta-modeling has been developed in Matlab. The software can be pre-configured through an interactive GUI, and progress can be monitored through a web-browser. During meta-modeling, sub-processes are triggered automatically to accomplish the best possible predictive quality. Reproducible high-quality meta-models in conjunction with a dramatic reduction in throughput time are accomplished by: data preprocessing using clustering and multivariate outlier detection, GP Regression with Automatic Relevance Determination, and cross-validation.

Data preprocessing using clustering and outlier detection

Data preprocessing involves removal of failed runs and redundant RV's. First hierarchical clustering is applied to the data to identify and remove highly covariant responses from the data set. Hierarchical clustering is a sequential clustering algorithm [1]. An agglomerative approach is adopted starting with data points as individual clusters. Then, in subsequent steps the closest pair of clusters based on an average distance matrix using correlations for distance measures is merged, thus creating a set of nested clusters organized as a hierarchical tree. In Fig. 2 an example is displayed of a dendrogram showing clusters in CAE model responses based on a distance metric in terms of correlation coefficients. Clusters of highly correlated response variables are highlighted. In order to prevent redundancies, only one response is retained in each cluster with highly covariant elements.

Next multivariate outliers are identified and removed from the data set. Outlier detection is the identification of observations which do not fit to the remainder of the data set [4]. For this purpose it is assumed that the data are multivariate normally distributed. Multivariate outliers can't be detected by applying outlier detection rules on each variable separately, but rather they are detected by measuring the Mahalanobis distance of each data point. The Mahalanobis distance is a distance measure that de-correlates correlated variables by inverse Cholesky transformation. As classical estimators of mean and covariance are highly sensitive to outliers, robust estimation is accomplished by deployment of the Minimum Covariance Determinant estimator (MCD) [5, 6]. Outliers can be found by comparing each data point with a critical value of the χ^2 distribution, as the sum of squares of N independent normal random variables follows a χ^2 distribution with N degrees of freedom. These outliers are subsequently removed from the data set. An example of multivariate outlier detection in DoE

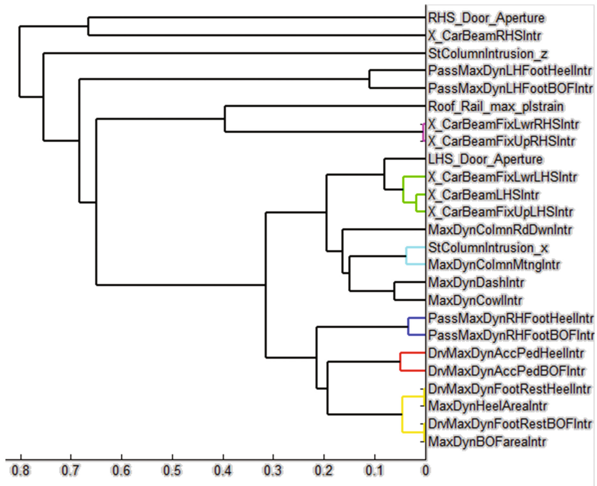

Fig. 2. Dendrogram

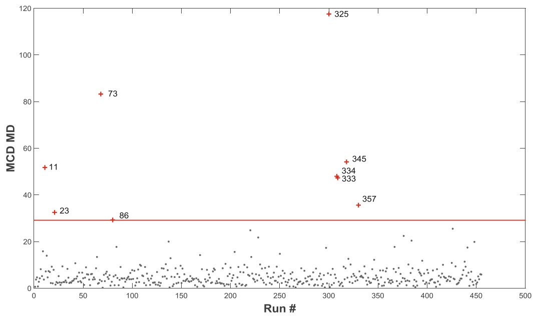

Fig. 3. Mahalanobis distances for DoE responses

responses is given in Figs. 3 and 4, with Fig. 3 showing the runs which are designated as outliers based on the MCD estimator, and Fig. 4 showing the corresponding matrix scatter plot including outliers.

In addition, the implementation of multivariate outlier detection has been extended with clustering based on Gaussian Mixtures [3], which can be deployed to prevent undesired deletion of clusters of CAE runs exhibiting multi-modal responses, e.g. local instabilities.

GP regression with Automatic Relevance Determination

In Gaussian Processes with Automatic Relevance Determination (ARD) length scale parameters are introduced for each input dimension in the covariance function. For example, the ARD version of the squared exponential covariance is given by:

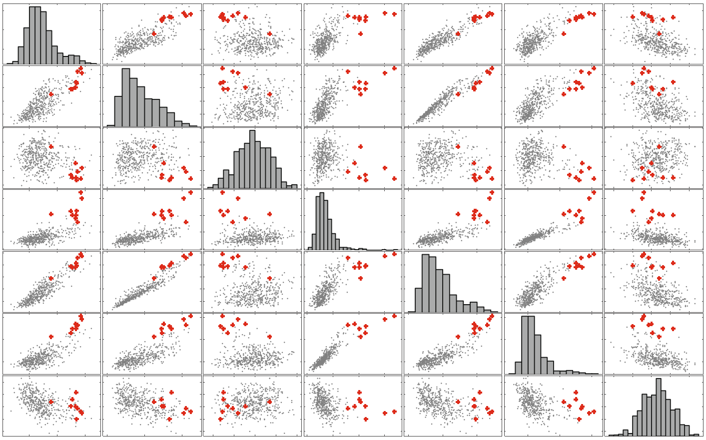

Fig. 4. Multivariate outlier detection: outliers marked as '+'

$$K(x, x') = \sigma_o^2 \exp\left(-\frac{1}{2}\sum_{d=1}^{D}\frac{\left(x_d - x'_d\right)^2}{\lambda_d^2}\right) \tag{7}$$

with λ_d the characteristic length scale measuring the distance for being correlated along x_d. If the reciprocal length scale $1/\lambda_d$ is small the covariance will become almost independent of the associated inputs x_d [2, 7]. By discarding the irrelevant inputs, the dimensionality of the data is reduced enabling smaller sized DoE's and smaller prediction errors. A practical application from NVH is displayed in Fig. 5. The reciprocal ARD length scales for multiple responses are indicative of the relevance of the input dimensions based on Eq. (7).

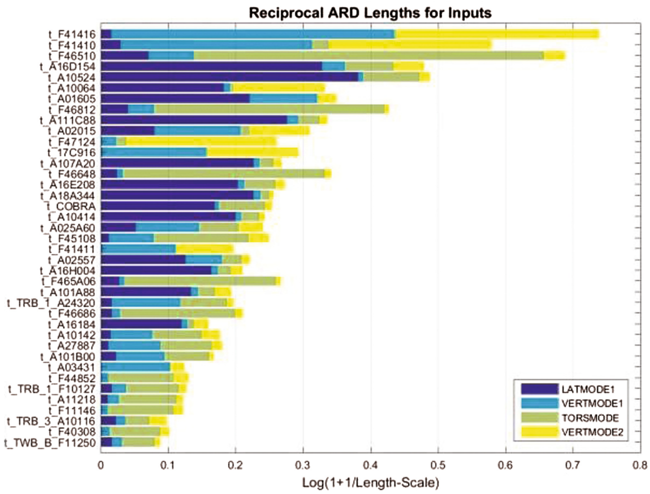

Fig. 5. Reciprocal ARD length for multiple outputs

Cross-validation

In automated meta-modeling the predictive quality is verified by utilizing cross-validation [1], where all data points are used for validation once. A major advantage of cross-validation is that no DoE runs are wasted for validation of predictive quality. However, the validation process can take a long time as multiple training passes are required depending on the number of folds (training sessions) in the data set. In general *N/K* folds on *N* – *K* points each are conducted, with *N* the number of training data, and *K* the number of validation data. The limit case is Leave-One-Out cross-validation, where $N - 1$ points are used for training and 1 point for validation.

By implementing a closed form solution for Leave-One-Out cross-validation the standard process of conducting *N* training sessions could be circumvented [3]. For Leave-One-Out cross-validation the predictive quality in terms of Coefficient of Determination R^2 can be expressed as:

$$R^2_{LOO} = 1 - \frac{1}{N}\sum_{i=1}^{N}\frac{\left[\hat{B}y\right]_i^2}{\left[\hat{B}\right]_{ii}^2}; \hat{B} = \left[K + \sigma_n^2 I\right]^{-1} \tag{8}$$

3.2 Enhanced NSGA-II for Multi-objective Optimization

In MDO multiple potentially conflicting objectives must be minimized simultaneously. The best trade-offs among the objectives must exhibit Pareto optimality. A Pareto optimal solution implies that while moving from one Pareto solution to another, any improvement in one objective requires a degradation of at least one other objective [8]. The set of all feasible non-dominated solutions is referred to as the Pareto optimal set, and the set of corresponding objective function values is referred to as the Pareto front. The final solution selected from the Pareto set is always a trade-off between critical performance parameters, e.g. vehicle weight vs intrusions. In general, viable multi-objective optimization procedures should exhibit following characteristics:

- Strong approach to the true Pareto front.
- Wide coverage of the true Pareto front.
- Pareto set with uniformly distributed solutions.

The most popular heuristic optimization procedures to attain these characteristics deploy population based evolutionary algorithms (EA), which are capable of coping with non-convex, discontinuous, and multi-modal solutions spaces. An enhanced version of NSGA-II, a computationally fast and Pareto dominance based multi objective EA [9], has been embedded in Ford's MDO process. Enhancements include vectorization and constraint handling. NSGA-II has been benchmarked against several alternative heuristic procedures for various cross attribute weight optimization problems. In general, it was observed that NSGA-II exhibited a wider coverage area in the objective function space and converged closer to the true Pareto front. In Fig. 6 NSGA-II has been compared to the aggregation based MOEA/D [10]. The Pareto front shows normalized mass vs. total constraint violation as specified by Eqs. (9) and (10). It is noted that a comparable Pareto front is rendered 8 times faster for NSGA-II, which can be attributed to vectorization of the NSGA-II algorithm.

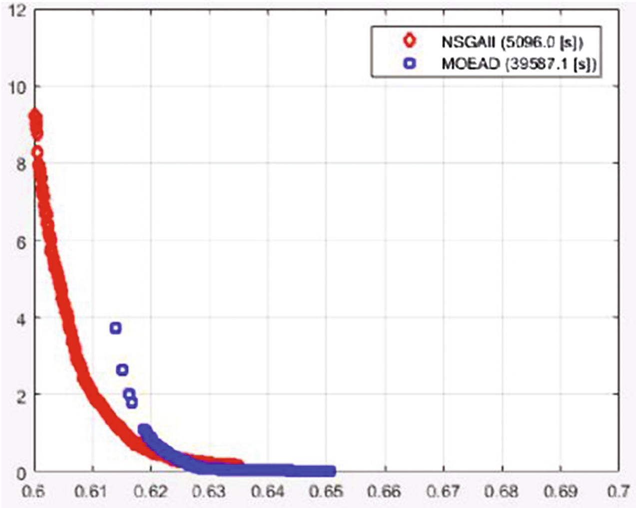

Fig. 6. Pareto fronts established by NSGA-II and MOEA/D

Enhanced NSGA-II is capable of properly handling constraints in the design variables. In addition, inconsistent settings for constraints and ranges in the design variables are revealed.

Handling of compliance with multiple performance targets can be achieved by imposing a penalty on normalized response values violating lower and upper constraint boundaries X_L and X_U

$$C_i(x) = \max(\mathrm{sign}(\mathrm{X}_L - x), 0) * \frac{x - X_L}{|X_L| + \gamma|X_U|} + \max(\mathrm{sign}(x - \mathrm{X}_U), 0) * \frac{x - X_U}{|X_U| + \gamma|X_L|} \tag{9}$$

with scaling factor $\gamma << 1$. The total violation of performance targets is captured by one penalty function via a multiplicative criterion, which reaches zero if all constraints are met:

$$C_T = \prod_{i=1}^{Nconstr} (C_i + 1) - 1 \tag{10}$$

4 Optimization Under Uncertainty

Compliance with performance targets must be validated for the final solution(s) selected from the Pareto front based on CAE models for all relevant load cases. Due to predictive uncertainty in meta-modeling, validation may result in violation of certain performance targets. Predictive uncertainty can be attributed to selection of inappropriate regression techniques, lack of data from the DoE, a too large set of design variables or excessive noise in the data (lack of CAE model robustness). In order to mitigate the risk of constraint violations in confirmation runs predictive uncertainty should be accounted for in optimization. For this purpose, the NSGA-II algorithm been extended to optimization under uncertainty. By combining optimization under uncertainty with Gaussian Processes fully probabilistic predictions for each point in the design space are available, which can be used to add a tolerance in terms of predictive uncertainty $\boldsymbol{\sigma}$ to the expected value prediction. Then, the optimization problem can be formulated as [11]:

$$\min_{\boldsymbol{X}}[\boldsymbol{\mu_{01}}(\boldsymbol{X})+\boldsymbol{\alpha\sigma_{01}}(\boldsymbol{X}),\ \boldsymbol{\mu_{02}}(\boldsymbol{X})+\boldsymbol{\alpha\sigma_{02}}(\boldsymbol{X}),\ldots,\ \boldsymbol{\mu_{0\mathrm{M}}}(\boldsymbol{X})+\boldsymbol{\alpha\sigma_{0\mathrm{M}}}(\boldsymbol{X})] \tag{11}$$

With $\boldsymbol{\mu}$ the expected value and $\boldsymbol{\alpha\sigma}$ the associated standard deviation scaled by $\boldsymbol{\alpha}$; $\alpha = 0$ implies standard constraint handling, whereas $\alpha > 0$ implies more stringent constraint handling. The resulting non-dominated solutions are referred to as the mean value penalty Pareto front. In Fig. 7a it is shown that risk of constraint violation in performance confirmation is mitigated by the mean value penalty Pareto front

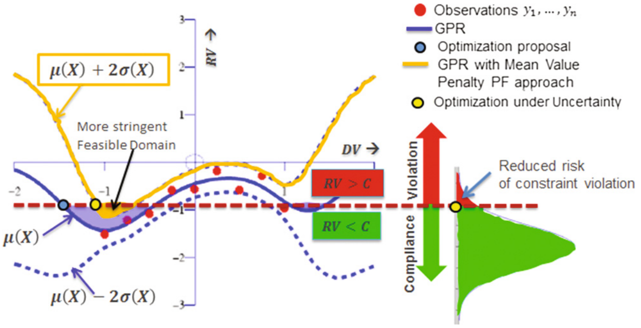

Fig. 7a. Risk of constraint violation for mean value penalty PF optimization

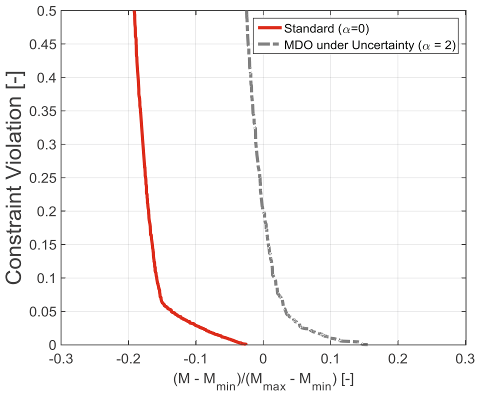

Fig. 7b. Mean value penalty PF optimization for $\boldsymbol{\alpha}$ = 0 and $\boldsymbol{\alpha}$ = 2.

optimization. In Fig. 7b it is shown that the mean value penalty Pareto front as established by optimization under uncertainty based on Eq. (11) with $\alpha = 2$ yields more conservative solutions than the Pareto front determined based on standard optimization ($\alpha = 0$).

5 MDO Applications

5.1 Cross-Attribute Weight Optimization of Body-In-White Structures

In order to support efficient Body-In-White (BIW) development of passenger cars, multi-disciplinary design optimization is conducted aiming at weight reduction whilst maintaining relevant attribute performance levels. A large-scale MDO typically involves in excess of 50 load cases, 150 design variables and 1000 responses. In Table 1 an example of attribute load cases considered in BIW weight optimization is shown. Potential design variables are gauges, material grades, and shape parameters. In addition, tailor rolled blanks, adhesives and spot weld groups can be included as design parameters. Tailor rolled blanks comprise sections of non-constant thickness and

Table 1. Example of load cases considered in BIW weight optimization

Crash safety	NVH	Durability & strength	Closures CAE	Closures CAE	Chassis CAE	Forming
FOF	Modes	SB28 2nd row	FSD Pole Intrusion	RSD Pole Intrusion	Body Y-Low	Part mass
FON	Torsional Stiffness	SB28 2nd Row Ch. Seat	FSD Open Slam	RSD Open Slam	FRT SBFR Modes	Part Feasibility
FP2L	Dynamic Stiffness	SB28 1st Row	FSD Close Slam Sheet	RSD Sag Ajar Pos.	GSS	
FPL	Equivalent Stiffness	SB28 1st Row HA	FSD Close Slam SW	Sag Open Pos.	GEDL	
RD6	Seat Att. Stiffness	Dash Cowl Fatigue	FSD Sag Drop-Off Ajar	Fr Rigidity Def. B-Pill	FRT SBFR Stiffness	
RD8	IP Att. Stiffness	Full Body Fatigue	FSD Sag Drop-Off Open	Fr Rigidity Def. C-Pill		
RD8A		PTL Brake	FSD Fr Rigidity Mid	Fr Rigidity PS B-Pill		
SI9		PTL Clutch	FSD Fr Rig. Rr	Fr Rig.PS C-Pill		
SP6		VJS Rear Loc1	FSD Fr Rig Perm Set Mid			
SP7		VJS Rear Loc2	FSD Fr Rig Perm Set Rr			
Roof Crush		VJS Rear Loc3	High Speed Def.			

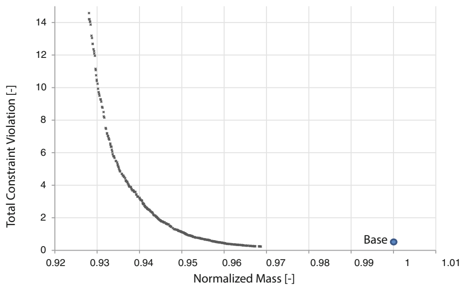

Fig. 8. Pareto front for load cases acc. to Table 1.

provide challenging constraints in terms of allowable thickness increments, thickness gradients, and nesting symmetry.

In real world applications of vehicle structure optimization, various scenarios are usually investigated, e.g. by imposing commonality and manufacturing constraints for relevant parts, or by relaxing certain performance targets. Pareto optimal solutions can be obtained for each scenario using weight and total constraint violation as given by Eq. (10). For example, in Fig. 8 the Pareto front for normalized mass vs total constraint violation associated with the load cases in Table 1 is displayed. It can be observed that a 4% weight reduction is feasible at equivalent attribute performance levels. Using meta-model based optimization, fast turn-around times of the scenarios considered can be warranted. Hence, MDO is a viable approach for supporting the decision making process.

5.2 Optimization of Weight and Adhesive Application in Body-in-White Structures

Adhesives are applied to increase the bonding stiffness between vehicle parts. Smart application of adhesives allows for gauge reductions without compromising stiffness targets. As adhesive application incurs additional cost a trade-off is required between the incremental length of adhesive application and BIW weight. In Fig. 9a, adhesive groups are displayed which can be activated via a binary switch. Total adhesive length and weight of affected parts were minimized while maintaining cross attribute performance involving crash safety, durability and NVH load cases. In Fig. 9b it is illustrated that an incremental weight reduction can be achieved without performance deterioration by adding adhesives lines.

5.3 Optimization of Margin and Flushness

The hood of a vehicle should align with the fender plane. To meet this requirement, the hood inner panel must be tuned by increasing local stiffness such that the requirements for maximum displacements of the hood are met with minimal weight. The optimization problem has been addressed by dividing the hood inner panel into a grid of squares with tunable sheet metal thicknesses (Fig. 10a). The maximum displacement of the hood inner panel could be accurately described as evidenced by the LOO R-squared of 0.95 acc. to Eq. (8). Subsequent optimization reveals the trade-off between weight

Fig. 9a. Selection of adhesive groups

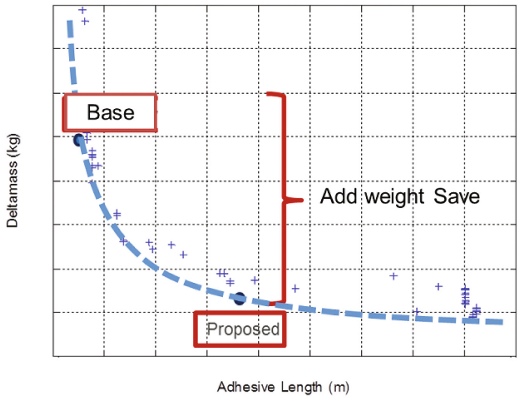

Fig. 9b. Trade-off: weight vs. additional adhesive length at equivalent performance

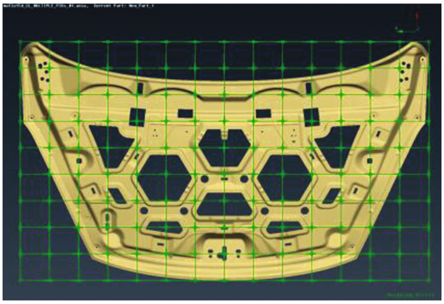

Fig. 10a. Hood inner panel divided in grid of thickness zones

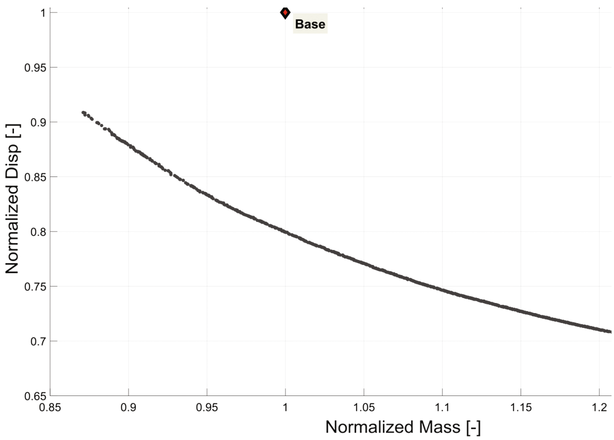

Fig. 10b. Pareto front of mass vs. maximum displacement of hood inner panel

and margin flushness as characterized by the maximum displacement of the hood inner panel (Fig. 10b).

6 Discussion

Multidisciplinary design optimization (MDO) is deployed in all major vehicle programs at Ford to reduce weight and cost while meeting performance targets. Recently, significant improvements in efficiency of the MDO process have been accomplished by the development of: automated preparation and submission of CAE models to the HPC cluster, automated meta-modeling, and multi-objective optimization based on a fast NSGA-II algorithm. These improvements resulted in 20% reduction in throughput times in conjunction with better solutions, i.e. increased weight savings in cross-attribute structural optimization. Future developments will focus on continuous improvements of the MDO process. Potential topics include the application of adaptive sampling techniques in conjunction with Gaussian Processes to further reduce the computational effort for creating meta-models, and mixtures of Gaussian processes for reliably predicting multi-modal responses. Furthermore, the deployment of the in-house developed software tools will be extended to a wide range of problems involving data analysis.

References

1. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning, 2nd edn. Springer, Heidelberg (2008)
2. Rasmussen, C.E., Williams, C.K.I.: Gaussian Processes for Machine Learning. MIT Press, Cambridge (2006)

3. Barber, D.: Bayesian Reasoning and Machine Learning. Cambridge University Press, Cambridge (2012)
4. Ben-Gal, I., Maimon, O., Rockach, L. (eds.): Outlier Detection. In: Data Mining and Knowledge Discovery Handbook: A Complete Guide for Practitioners and Researchers. Kluwer Academic Publishers, Boston (2005)
5. Rousseeuw, P.J., Van Driessen, K.: A fast algorithm for the minimum covariance determinant estimator. Technometrics **41**, 212–223 (1999)
6. Riani, M., Atkinson, A.C., Cerioli, A.: Finding an unknown number of multivariate outliers. J. Roy. Stat. Soc. Ser. B **71**, 447–466 (2009)
7. Neal, R.M.: Bayesian Learning for Neural Networks. Lecture Notes in Statistics, vol. 118. Springer, New York (1996)
8. Marler, R.T., Arora, J.S.: Survey of multi-objective optimization methods for engineering. Struct. Multi. Optim. **26**, 369–395 (2004)
9. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multi-objective genetic algorithm: NSGA-II. Trans. Evol. Comput. **6**(2), 182–197 (2002)
10. Zhang, Q.: MOEA/D: a multi-objective evolutionary algorithm based on decomposition. IEEE Trans. Evol. Comput. **11**(6), 712–731 (2007)
11. Petrone, G.: Optimization under uncertainty. Ph.D. thesis, University of Naples, Italy (2012)

Springer

http://www.springer.com/978-3-319-67987-7

Advances in Structural and Multidisciplinary Optimization
Proceedings of the 12th World Congress of Structural and Multidisciplinary Optimization (WCSMO12)
Schumacher, A.; Vietor, Th.; Fiebig, S.; Bletzinger, K.-U.; Maute, K. (Eds.)
2018, XXII, 2115 p. 1225 illus. In 2 volumes, not available separately., Hardcover
ISBN: 978-3-319-67987-7