Sequential Optimization & Probabilistic Analysis Using Adaptively Refined Constraints in LS-OPT®

Anirban Basudhar
Livermore Software Technology LLC, Ansys Group, USA
Katharina Witowski
Dynamore GmbH, Germany
Imtiaz Gandikota
Livermore Software Technology LLC, Ansys Group, USA

Abstract

This paper presents some of the sequential optimization and probabilistic analysis methods in LS-OPT with particular emphasis on the use of classifiers for accuracy and efficiency improvement. Classifiers were first introduced in LS-OPT 6.0 for the handling of constraints. This paper provides a review of the basic classification-based constraint handling method and its applications and advantages for specific types of problems. Additionally, the application of classifiers is extended to adaptive sampling using EDSD (explicit design space decomposition) sampling constraints in LS-OPT 6.1. The different adaptive sampling options and approaches are presented through the examples. Another aspect of this paper is the extension of the probabilistic analysis method in LS-OPT from single iteration to sequential. The sequential analysis can be performed with or without EDSD sampling constraints, but sampling constraints, if used, are can guide the samples adaptively to important regions. Although the EDSD sampling constraints are defined using support vector machine (SVM) classifiers, the adaptive samples are useful in enhancing the constraint boundary accuracy even if it is defined using metamodels.

Overview of LS-OPT Optimization and Probabilistic Analysis Methods

The optimization and probabilistic methodologies or tasks in LS-OPT [1] are broadly divided into direct and metamodel-based methods. Direct methods are in general robust and can solve all types of complex problems if sufficient simulations are performed. Unfortunately, the associated computational cost can often be prohibitive. Metamodels, on the contrary, attempt to build computationally inexpensive surrogates for simulation models using only a few samples [2]. The focus of this paper is on such surrogate-based methods. However, the fidelity of the inexpensive surrogates is determined by the number and quality of the samples, as well as by the complexity of the problem at hand. Therefore, special sampling strategies are sometimes needed, some of which have been part of LS-OPT for quite some time [1]. There is, however, scope for improving these strategies as this is still an evolving research area. This section provides an overview of the pre-existing metamodel-based methods in LS-OPT, before diving into some of the potential enhancements in the following sections.

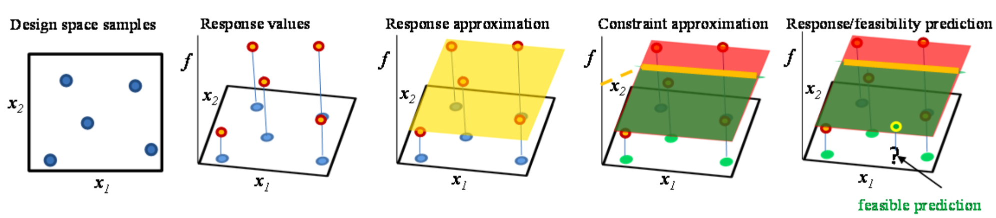


Fig.1: Response and feasibility prediction using a metamodel. The predicted response can be used as an objective function or as a constraint for optimization or probabilistic analysis.

Metamodel-based Optimization

There are four metamodel-based optimization strategies in LS-OPT – single iteration, sequential, sequential with domain reduction and efficient global optimization [1]. The single iteration method requires a prior total number of samples to be evaluated. These response values at these samples is fitted to a metamodel. Once the metamodel is trained, a core optimizer is used to solve the approximated optimization problem. The sequential approach adds samples in batches over several iterations; the single iteration optimization is repeated to gradually improve the approximation accuracy and converge to a solution. Both these approaches are rather naïve in their sampling strategy; there is no consideration of the nature of objective and constraint functions while selecting the samples for evaluation. The domain reduction method [3] uses a more intelligent and informed strategy which adaptively adjusts the region of interest based on the current approximations. The EGO method [4] is an adaptive approach when applied as a serial sampling method but is relatively naïve in multiple parallel sampling based on its current LS-OPT implementation. Planned improvements to the parallelization strategy should however mitigate this limitation.

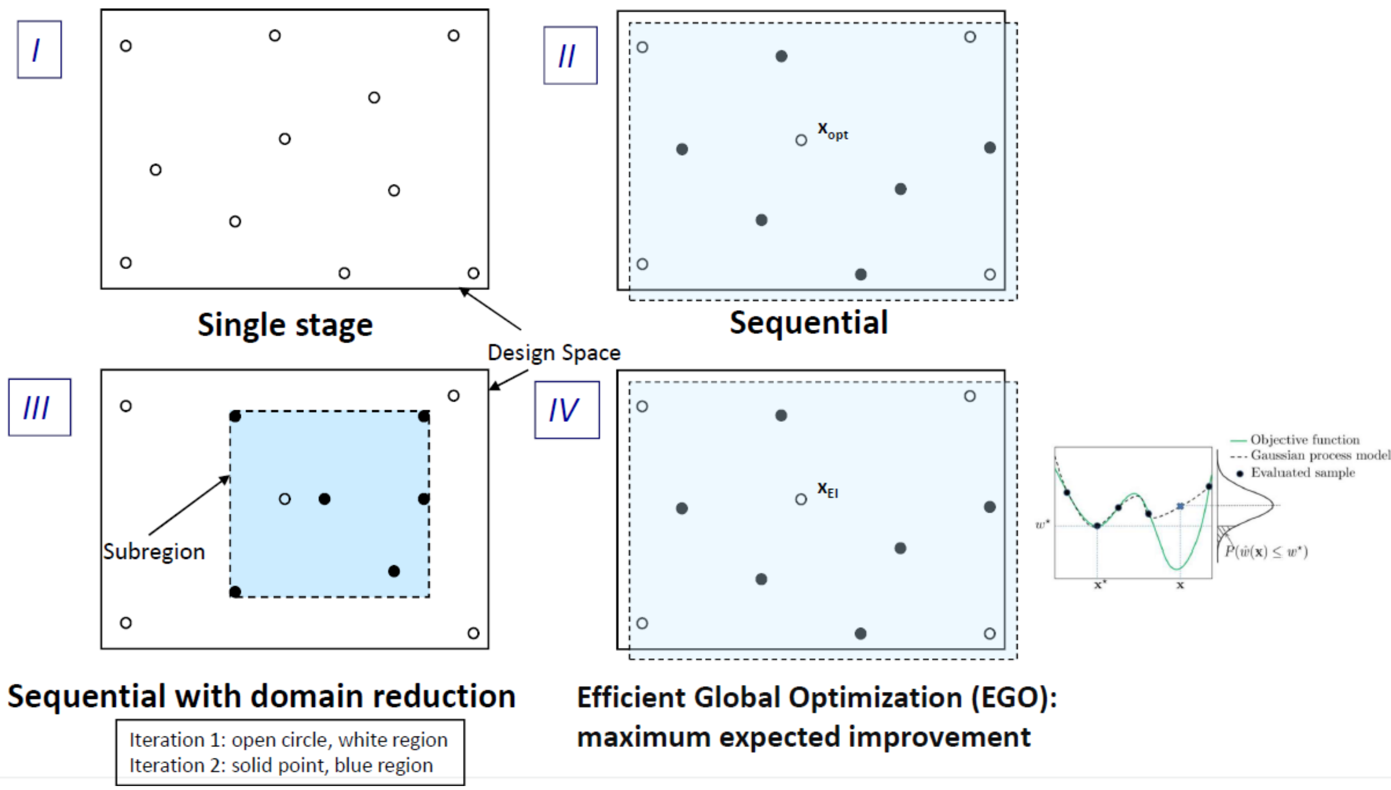


Fig.2: Metamodel-based optimization strategies in LS-OPT.

One thing to note is that design problems are often constrained, and therefore accuracy of the constraint boundary is also important in addition to the objective function [5-8]. For certain types of problems, a classification-based approach can be useful for defining and refining the constraints [9-14].

Metamodel-based Probabilistic Analysis

The probabilistic analysis methods in LS-OPT include single iteration Monte Carlo analysis for failure probability calculation, multi-level framework-based reliability and tolerance analysis and DynaStats – a tool for spatial-temporal stochastic analysis for LS-DYNA® models [1]. While DynaStats is a very interesting tool for result visualization and the multi-level framework gives a great deal of flexibility for reliability calculation [15], the later also adds some complexity to the problem setup.

LS-OPT Constraint Handling and Probabilistic Analysis Enhancements

This section presents the contributions of this work, which can be grouped in two categories – allowing sequential sampling and convergence study for probabilistic analysis and developing a classifier-based constraint handling and adaptive sampling method.

Sequential Metamodel-based Probabilistic Analysis (Version 6.1)

Being limited to single iteration analysis, the single level reliability capabilities of LS-OPT have been fairly basic for a while. As part of this work, a sequential Monte Carlo strategy has been added to the next LS-OPT version 6.1 in order to alleviate this limitation. The sequential approach facilitates incremental sample addition (Fig 3) and convergence study of the failure probability (Fig 4). However, the methodology of sample addition can have a great impact on the sampling quality and the failure probability estimate. A classifier-based method has been developed in LS-OPT 6.1 to adaptively guide samples to important regions and improve the failure boundary estimate.

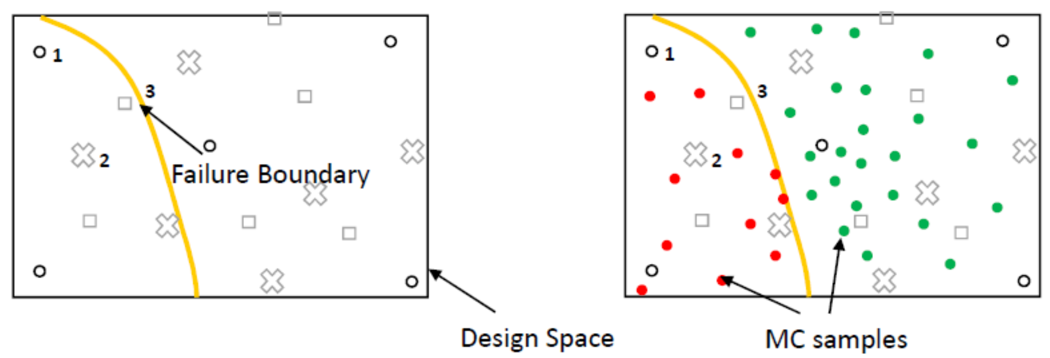


Fig.3: Sequential sampling (left) and Monte Carlo analysis using the predicted failure boundary (right). The numbers indicate the iteration.

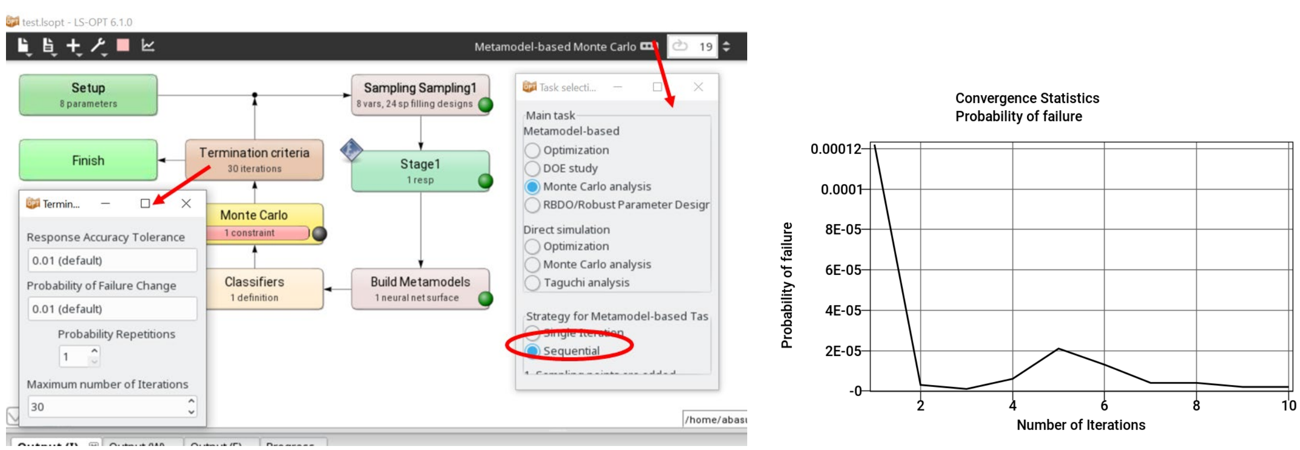


Fig.4: LS-OPT GUI for sequential probabilistic analysis (left) and an example of failure probability convergence (right).

Classifier-based Constraint Boundary (Version 6.0) and its Applications

The basic idea of classifiers implemented in LS-OPT 6.0 [1] is introduced in this section along with its applications. One of the applications is to use them for adaptive sampling [11-13], which is the focus of this paper. A separate section is dedicated later to classifier-based sampling constraints implemented in version 6.1.

In both design optimization and reliability assessment one of the main tasks is the demarcation between acceptable (feasible/safe) and unacceptable (infeasible/failed) designs. Classification methods use the pass/fail information at a few specified samples used for training an optimal boundary that separates the two categories.

The difference between metamodel-based and classification-based methods to determine acceptability of any general design alternative is shown in Fig 5. The classification-based method takes a decision directly based on the position of the new sample in the design space, whereas a metamodel takes the decision based on the corresponding predicted response value and threshold. Some of the applications of classifiers are listed below:

- Pass/fail information is readily available even for binary responses, e.g. failed simulation, lack of quantification ability etc., making classification the method of choice for such applications.
- A classifier considers the feasibility information during training itself, because of which the emphasis is on the accuracy near the decision boundary. This is particularly useful in cases where noise and response discontinuities hamper metamodel construction.
- A single classifier can be constructed for multiple failure modes related to different simulations, thereby providing an opportunity to skip many simulations or to terminate simulations early without any loss of useful data.
- A classifier can also be used to define the sampling domain, which is the main topic of this paper.

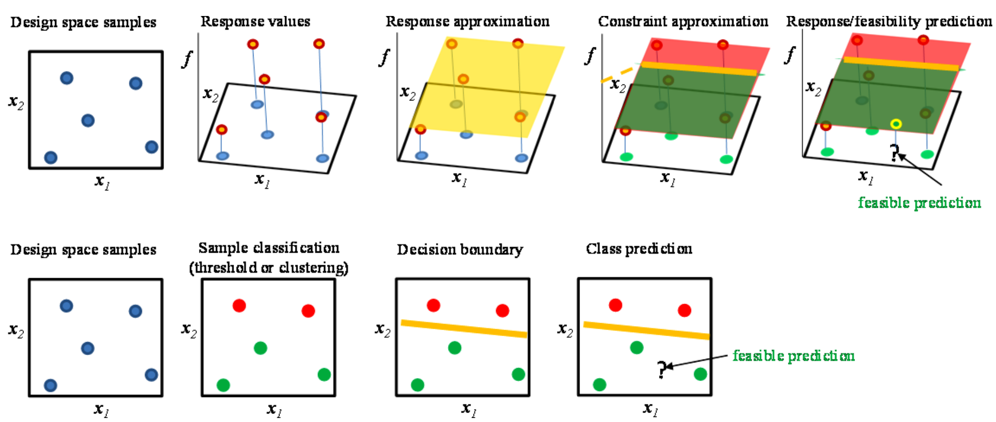


Fig.5: Summary of basic classification method (bottom) and comparison to metamodeling (top).

Support Vector Machine Classification

Support Vector Machine (SVM) [16] is a type of macine learning technique that can be used for both classification and regression. The basic idea of SVM classification in the context of linear binary separators is to maximize the margin between two hyperplanes (lines in a two-dimensional space) that are parallel and equidistant on either side from the separating hyperplane. The separating boundary demarcating the samples belonging to two classes, typically labelled as +1 and -1, is referred to as the SVM decision boundary and the two parallel hyperplanes are known as the support hyperplanes. The SVM decision boundary is constructed such that there is no sample belonging to either class in the margin between the support hyperplanes. The SVM value is equal to zero at the decision boundary and +1 and -1 at the two support hyperplanes. The same idea is extended to nonlinear decision boundaries using a kernel function. In such cases the decision boundary and the supporting boundaries are linear in a higher dimensional feature space, but they are nonlinear in the original variable space or input space. The SVM values at the decision boundary and the two support boundaries are still 0, +1 and -1. The general SVM boundary for the nonlinear case is obtained as s(x)=0, where s(x) is given in Eq. (1).

$$s(\mathbf{x}) = b + \sum_{i=1}^{N} \alpha_i y_i K(\mathbf{x}, \mathbf{x}_i) \qquad \textbf{(1)}$$

Here, $y_i = \pm 1$ (e.g. red vs green) is the class label, α_i is the Lagrange multiplier for i^{th} sample and b is the bias. The kernel K maps the design space and the feature space (the high-dimensional space consisting of basis functions as the dimensions, where the classifier is linear). In this work, a Gaussian kernel is used to construct SVM boundaries ($s(\mathbf{x}) = 0$).

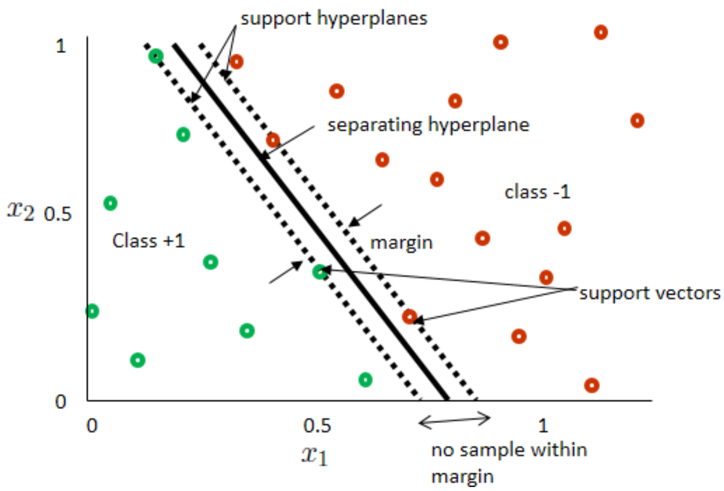


Fig.6: Linear classification using SVM.

Classifier-based Sampling Constraints for Adaptive Decision Boundary Refinement (v 6.1)

As mentioned above, the margin between the supporting boundaries does not have any samples. Additionally, the samples nearest to the decision boundary are the ones that influence it the most. Thus, it would be reasonable to say that sampling within the margin, which is in the vicinity of the decision boundary and lacks existing samples, could provide useful information to update the decision boundary. This is especially useful in the context of reliability assessment where an accurate approximation of the boundary is needed. In general, the samples can be constrained to lie in the vicinity of the decision boundary as:

$$|s(x)| = \left| b + \sum_{i=1}^{N} \alpha_i y_i K(\boldsymbol{x}, \boldsymbol{x}_i) \right| \leq \varepsilon \quad \textbf{(2)}$$

LS-OPT 6.1 will allow such constraints to be defined using EDSD sampling constraints. It is also be desirable to select a sample in sparsely populated regions. A new sample can then be obtained using constrained space-filling sampling as [12]:

$$\max_{\boldsymbol{x}} \|\boldsymbol{x} - \boldsymbol{x}_{nearest}\|$$

$$s.t. \quad \left| b + \sum_{i=1}^{N} \alpha_i y_i K(\boldsymbol{x}, \boldsymbol{x}_i) \right| \leq \varepsilon \quad \textbf{(3)}$$

Apart from sampling in the vicinity of the boundary, it may sometimes be useful to sample the region belonging to one of the classes, e.g. the feasible region or the non-dominated region in the context of multi-objective optimization. Such a [14], with some tolerance, can be defined as:

$$s(x) = b + \sum_{i=1}^{N} \alpha_i y_i K(\boldsymbol{x}, \boldsymbol{x}_i) \leq \varepsilon$$

$$or$$

$$s(x) = b + \sum_{i=1}^{N} \alpha_i y_i K(\boldsymbol{x}, \boldsymbol{x}_i) \geq \varepsilon \tag{4}$$

Equations 2-4 adaptively select samples by exploiting the previous classifier boundary estimate. An example of EDSD constraint definition in LS-OPT is shown in Fig 7.

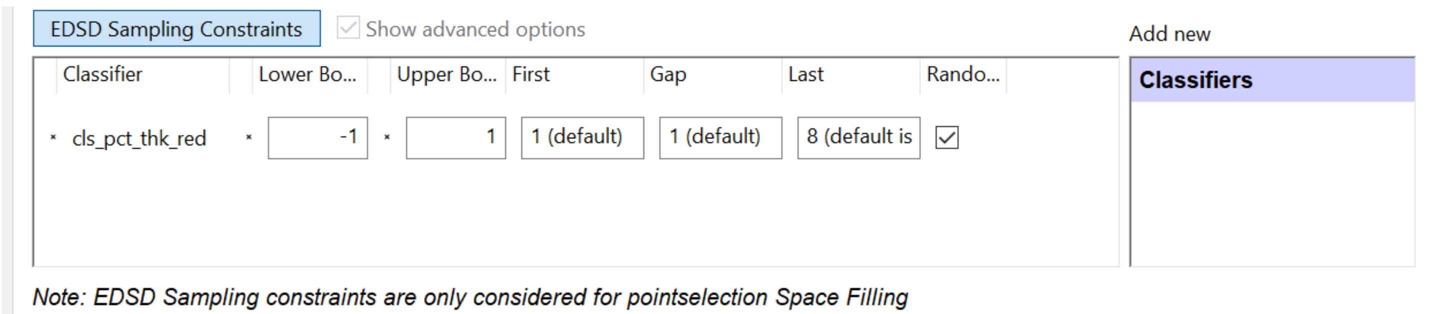


Fig.7: EDSD Classifier Sampling Constraint Panel.

As can be seen in Fig 7, an EDSD sampling constraint definition has several associated options. Foremost, at least one of the bounds must be specified. In the context of reliability assessment, it is common to specify both the bounds so that the samples are selected in the vicinity of the decision boundary on either side (Fig 8 left). In some applications a user may want to select new samples in the predicted feasible region (Fig 8 right), in which case only one of the bounds is needed. Both the cases are shown in Fig 8 - sampling near the boundary using Equation 3 and sampling the desired feasible or Pareto optimal region (green) using constraints in Equation 4.

In addition to the bounds, a user may specify the first and last sample of an iteration for which the sampling constraint would be in effect; rest of the samples are space-filling outside of the ε band. This should allow the exploration of disjoint regions in the design space and prevent excessive margin thinning and SVM locking [12].

Additionally, it is possible to randomize the bounds; in that case, the actual bounds for a particular sample selection are obtained using a uniform distribution between the specified maximum bounds. This allows some of the samples to be almost on the boundary while the rest are in its vicinity within the specified bounds.

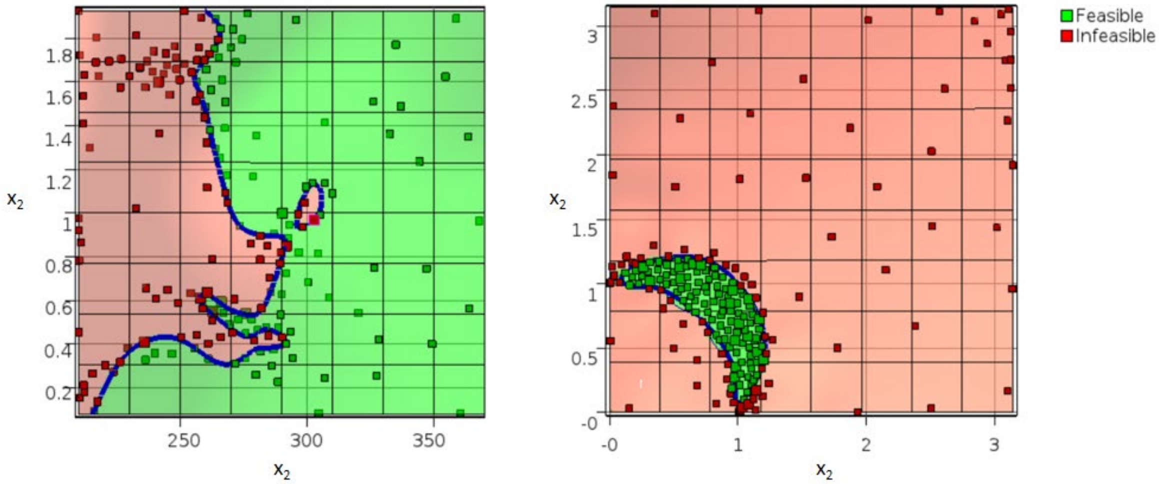


Fig.8: Examples of adaptive sampling using SVM-based EDSD: sampling near the boundary (left) and sampling in the feasible space (right).

Examples

Example 1: Simulation Filtering for Multi-disciplinary Analysis Constraint Boundary Definition

This example consists of two disciplines - vehicle side impact and modal analysis. The objective of this example is to demonstrate the ability to sequentially filter multi-disciplinary samples, allowing huge computational savings using the classifier-based constraint handling approach.

The LS-DYNA finite element models [32] of the vehicle used in the study (Fig 9) consist of approximately 30,000 elements and therefore, are generalized models intended only for demonstrating classifier concepts discussed in this study. Most of the elements in the model were defined using shell section with ELFORM 2 formulation of LS-DYNA.

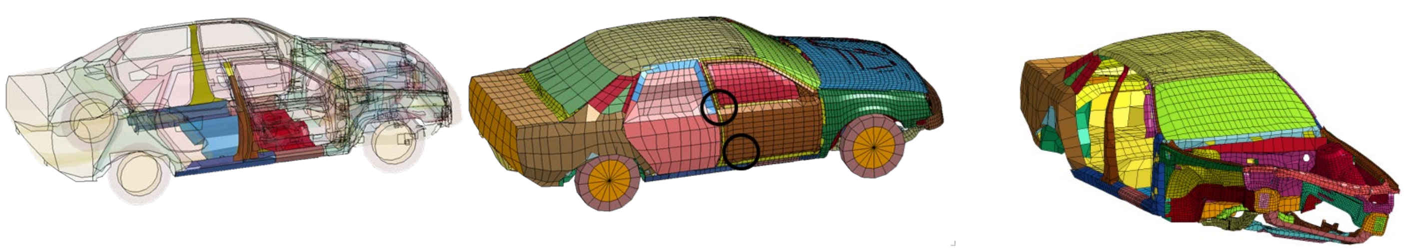


Fig.9: Ex 1. Design parts (darker shade, left), b-pillar and door intrusion location (center), torsional mode (right).

The vehicle model was subjected to a side impact against the rigid cylindrical pole (radius 150 mm) at a velocity of 29 mph. The modal analysis was used to obtain the first torsional mode frequency. The shell thicknesses of a few design parts, selected based on their contribution to overall internal energy absorption, were selected as the design variables for the Monte Carlo analysis. To simplify the setup into a two variable example to enable visualization, the thickness of the b-pillar and the lower beams was defined as “tbeam” and the thickness of the two floor parts was defined as “tfloor”. The intrusion distance of b-pillar, lower beam, and the door were defined as the design constraints for the Monte Carlo analysis, in addition to a frequency constraint. The intrusion distance was calculated as the difference of displacements of the nodes in the intrusion location and a reference node. The design parts and the location of intrusions are shown in Fig 9. In the Monte

Carlo analysis, the two noise variables, tbeam and tfloor, are normally distributed with a mean and standard deviation of 4, 0.4 mm and 2.5, 0.25 mm, respectively. The constraints in the problem are as follows.

$$(g_1 \leq 585mm),(g_2 \leq 710mm),(g_3 \leq 638.23mm),(g_4 \geq 41.6\,Hz) \tag{5}$$

where g_1, g_2, g_3 are the intrusion distances of the b-pillar, lower beam, and the door and g_4 is the first torsional frequency.

A common sampling set of 400 designs is used for both the crash analysis and the modal analysis. An SVM classifier is used to approximate the decision boundary. The classification-based approach allows filtering of samples going from one analysis to the next. The steps are as follows (Fig 10 – Fig 13).

1. As the modal analysis is significantly less time consuming, run all 400 points for this discipline and identify the samples that violate the frequency constraint (red).

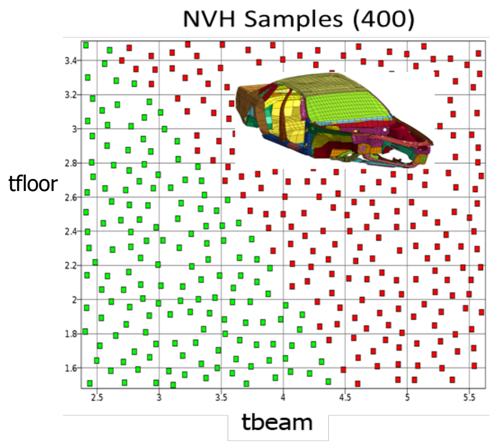


Fig.10: Ex 1. Modal analysis at all the samples to identify feasible and infeasible ones for that discipline.

2. Do not consider the 246 infeasible modal analysis samples for crash simulation

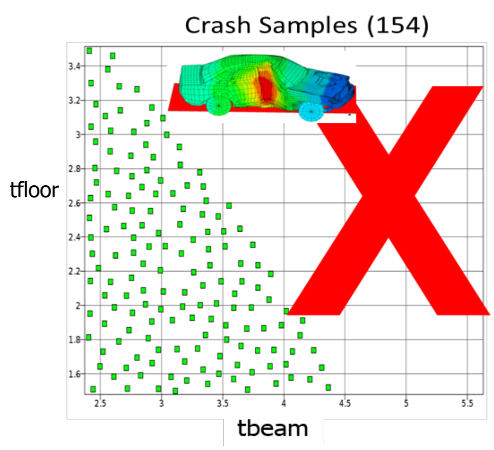


Fig.11: Ex 1. Samples for crash simulation (only the ones that are feasible based on modal analysis).

3. Run the crash analysis at 154 previously feasible samples and identify the crash constraint violations.

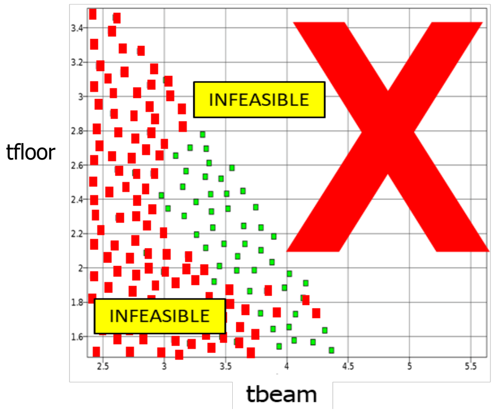


Fig.12: Ex 1. Feasible(green) and infeasible (red) crash samples.

4. Include the infeasible samples from the previous discipline and construct the system classifier.

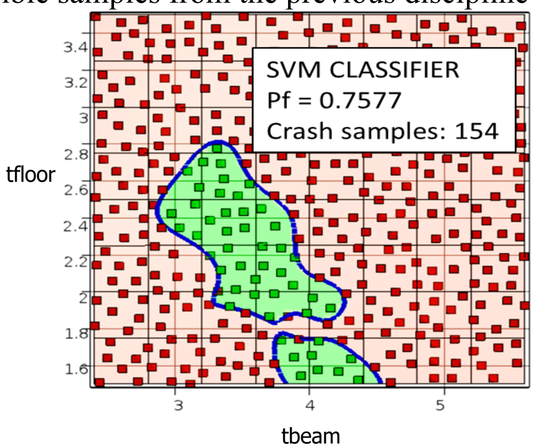


Fig.13: Ex 1. Combining all the infeasible samples (crash + modal analysis) to obtain the complete data set for system feasibility based on all disciplines. SVM classifier built using 400 samples although only 154 crash simulationes are performed.

Clearly the classifier-based approach allowed filtering out of 246 crash simulations out of 400, resulting in significant computational cost savings for this example. The results of a direct Monte Carlo Analysis (Importance Sampling based using 20000 samples) are shown for reference (Fig 14). The error in failure probability using the classifier constraint is only 0.28%.

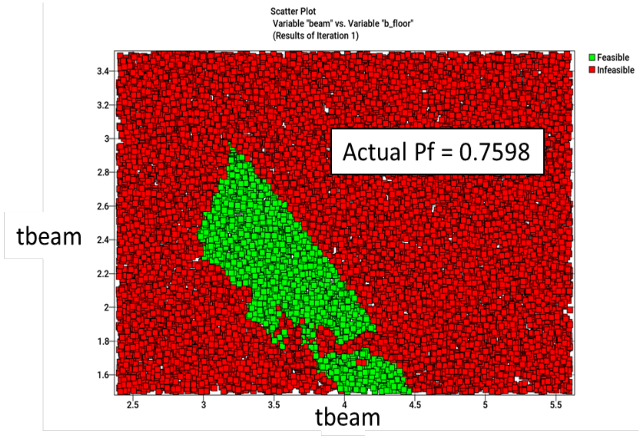


Fig.14: Ex 1. Result of direct Monte Carlo analysis showing the design space feasibility and the actual failure probability.

The result is also compared to a feed forward neural network metamodel using all 400 samples for both the disciplines (Fig 15). The error in this case is 2.01%. It is important to note that 246 additional crash simulations were needed to arrive at the metamodel-based solution.

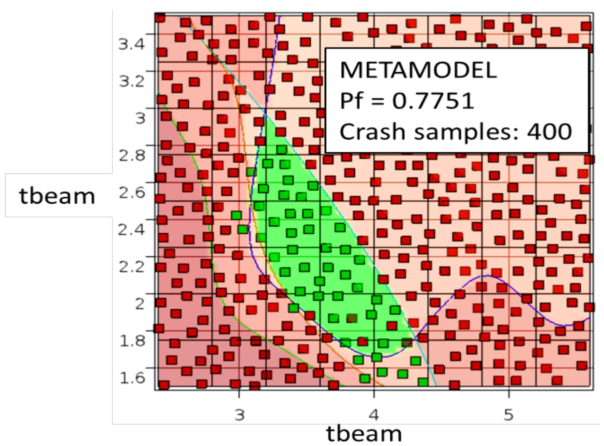
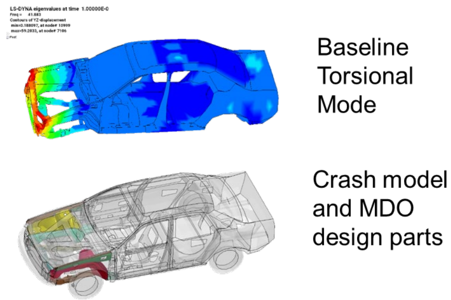


Fig.15: Ex 1. FFNN based failure boundary using 400 NVH and crash simulations. The misclassification of some of the training samples near the boundary is very apparent.

Example 2: Iterative 7-dimensional Multi-disciplinary Optimization with Filtered Simulations

This example uses the same finite element models as in example 1 and the objective again is to demonstrate the adaptive filtering of samples from one discipline to another. However, the problem dimensionality is scaled up to seven. Additionally, the application in this example is mass optimization subject to multi-disciplinary constraints. The optimization method is sequential with domain reduction, which also allows us to study the sample filtering as a function of iterations. The two disciplines in this example are frontal crash and modal analysis (Fig 16).



$$
\begin{aligned}
&\min && Mass\,(x_{1,2..7}) \\
&s.t. && 41.38Hz < b_{freq} < 42.38Hz \\
& && Stage\ 1\ pulse > 13.94g \\
& && Stage\ 2\ pulse > 19.17g \\
& && Stage\ 3\ pulse > 21.3g
\end{aligned}
$$

where

- $x_{1,2..7}$ are the design part thicknesses
- b_{freq} is the first torsional frequency

Fig.16: Ex 2. BIW and crash models (left). Optimization problem formulation (right).

Fig 17 shows the LS-OPT setup to solve this problem. The optimization is run with 40 samples per iteration and is terminated after 20 iterations with an overall crash simulation saving of 297 (37%). Fig 18 shows a projection of the scatter plot with modal frequency feasibility information and the simulation savings at each iteration.

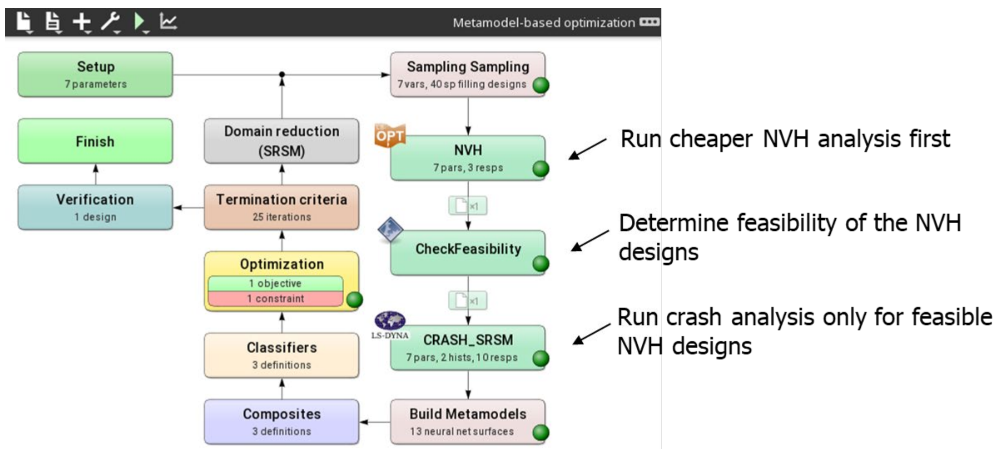


Fig.17: Ex 2. FFNN based failure boundary using 400 NVH and crash simulations. The misclassification of some of the training samples near the boundary is very apparent.

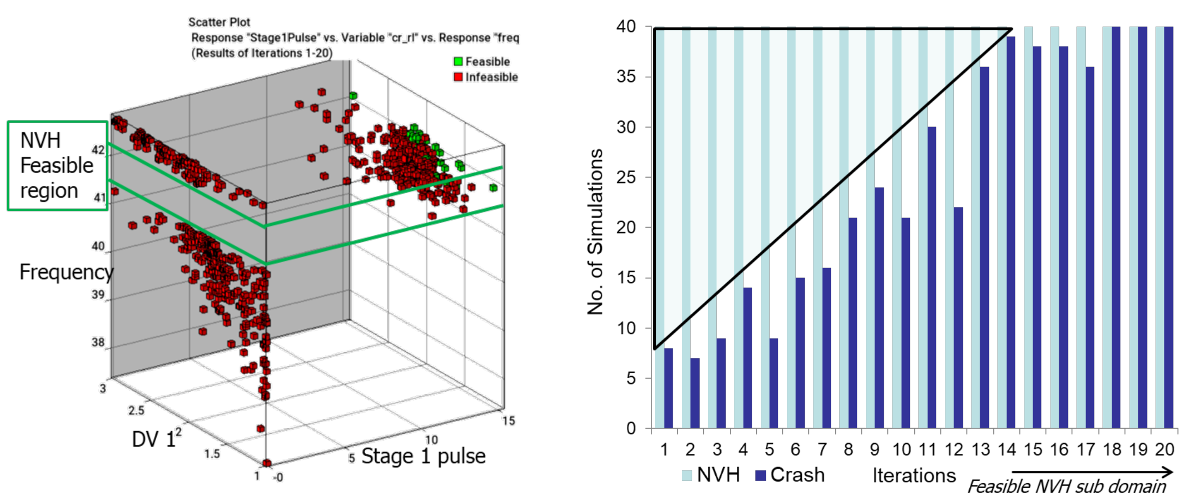


Fig.18: Ex 2. Scatter plot showing a projection of the feasible and infeasible space based on NVH (left). Crash simulation savings (region above the bars) vs iteration plot (right).

Example 3: Adaptive Sampling for Reliability Assessment of Simple Car Modal Analysis

In this example a modal analysis is performed on an LS-DYNA simplified car model (Fig 19). Two thicknesses tbumper and trailb are considered as random variables. The variable tbumper has a Truncated Normal distribution with bounds [1,6] mm, mean = 3.5 mm, standard deviation = 0.5 mm and the variable trailb has a Truncated Normal distribution with bounds [1,6] mm, mean = 3.5 mm, standard deviation = 0.6 mm. First, the torsional mode is identified for a baseline design tbumper = 3.5 mm, trailb = 2 mm. Two constraints are then defined based on the first torsional mode frequency of the perturbed Monte Carlo designs and the modal assurance criterion (MAC) value with respect to the baseline. The constraints are as follows.

$$Frequency \geq 1.6, MAC \geq 0.7 \quad \textbf{(6)}$$

As the torsional mode is tracked, the mode number can switch from one design to another. This is typically accompanied by a sudden jump in the frequency, leading to a discontinuity (Fig 19). The problem is solved using a FFNN approximation of the frequency as well as with SVM classification and the results are compared (Fig 20 and Fig 21) with and without EDSD sampling constraint based adaptive sampling. 20 samples are added per iteration. The EDSD constraints are defined such that the first 16 samples are within +/-2 SVM margin and the rest are space-filling.

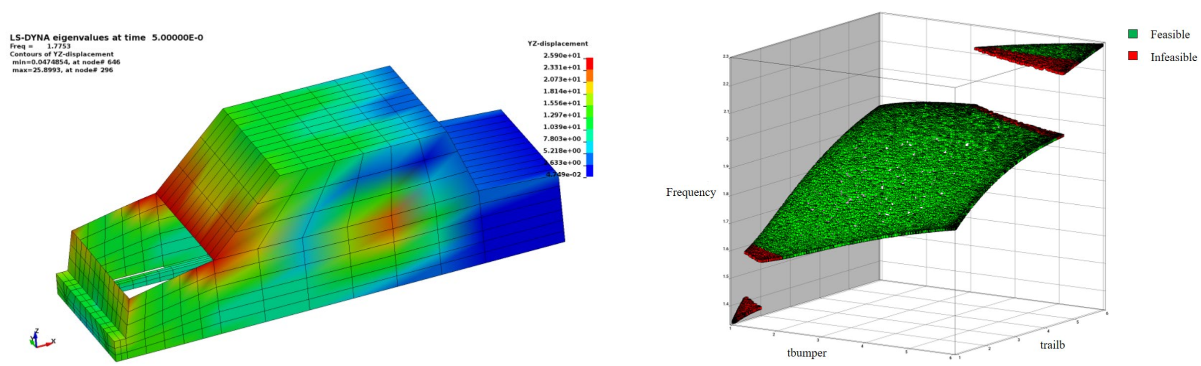


Fig.19: Simplified car LS-DYNA model for modal analysis

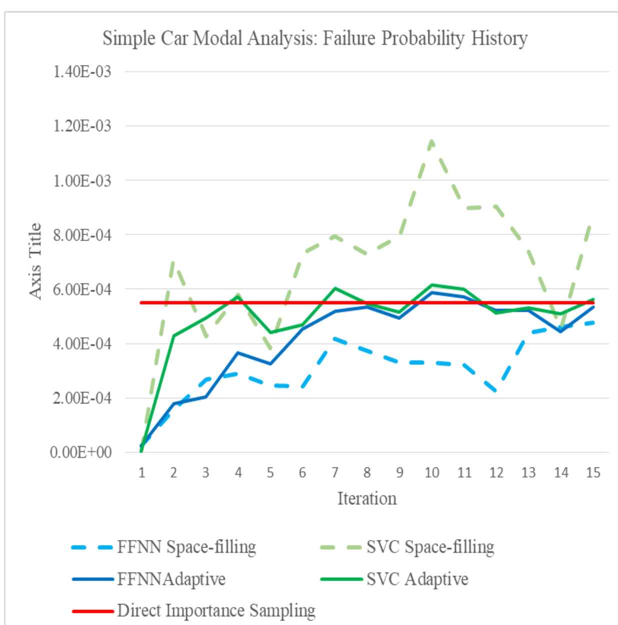


Fig.20: Ex 3. Comparison of failure probabilities. Both the SVC and FFNN based methods with adaptive sampling have lower errors and variance compared to their space-filling counterparts.

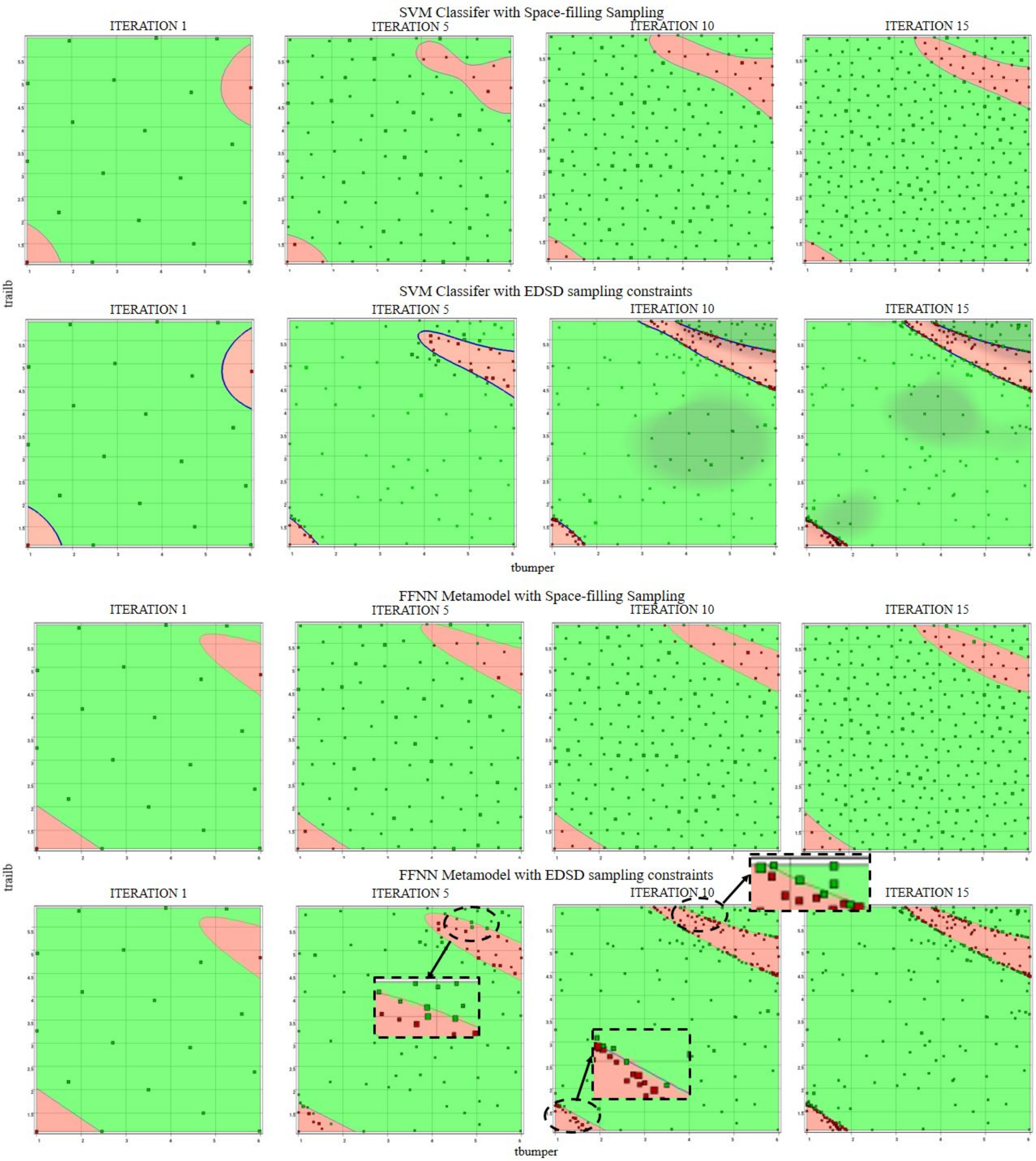


Fig.21: Failure boundary prediction using SVC and FFNN with and without EDSD constraints. The FFNN metamodel-based method has some misclassification because of the discontinuous nature of the response.

Example 4: Adaptive Sampling for 5-dimensional Tube Crush Reliability Assessment

A reliability analysis of a steel tube being crushed is performed in this example. The effect of both a variation in material thickness and a variation in the material parameters is investigated. The random variables are thickness T1 N(1,0.05), yield stress scale factor SIGY N(1.1,0.05), density DENSITY N(7.83e-6,1e-6), Young's modulus E N(200,10), Poisson's ratio PR N(0.3,0.005). The geometry is shown in the figure in its original and partially deformed state. The z-displacement of the upper tube boundary is constrained to be greater than -230 mm.

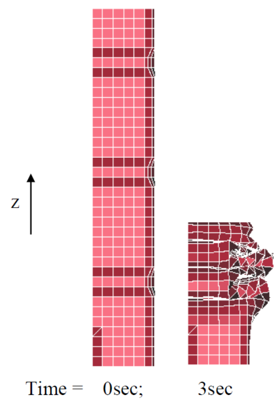


Fig.22: Ex 4. Tube crush. The displacement is in the negative z direction.

The problem is solved using an SVM classifier as well as an FFNN metamodel. 10 samples are added in each iteration with the first 8 samples within the empty SVM margin +/-1 and the last two points within SVM value +/-2. The reliability analysis is continued for 30 iterations. Fig 23 shows that the SVM classifier is outperformed by the FFNN metamodel for this example; this can be attributed to the very simple (almost linear, mildly quadratic) nature of the response function that can be easily approximated accurately using a metamodel. Ways to improve the efficiency of classification methods for such smooth problems need to be investigated; possible remedies include hybrid (classifier + metamodel) decision boundary estimators and probabilistic classifiers. A more elaborate study with respect to EDSD parameters also needs to be carried out. The failure probability calculated using direct importance sampling with 10000 samples is considered to be the actual solution (0.109)

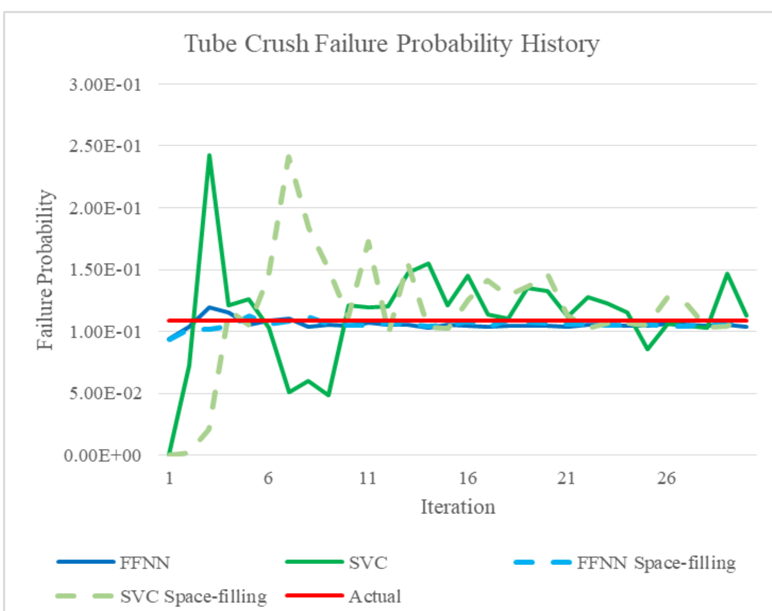


Fig.23: Ex 3. Comparison of failure probabilities. Both the SVC and FFNN based methods with adaptive sampling have lower errors and variance compared to their space-filling counterparts.

Conclusion

This paper presents two new features in LS-OPT – sequential reliability analysis and classifier-based EDSD sampling constraints. The former is an obvious improvement compared to the prior single iteration analysis, as it allows one to study the convergence of the failure probability instead of relying on an arbitrary number of samples. The EDSD sampling constraints facilitate adaptive sampling to obtain an accurate decision boundary, whether classifier-based or metamodel-based, more efficiently. In addition, advantages of a classifier-based boundary are also presented in the case of multi-disciplinary analysis (sample filtering to reduce cost) and discontinuous responses (more accurate boundary). For smooth responses metamodel-based constraint approximation can be more efficient, and therefore, a hybrid method will be investigated in the future to incorporate the good qualities of both approaches simultaneously.

References

[1] Stander, N., Roux, W.J., Basudhar, A., Eggleston, T., Craig, K.J. *LS-OPT User's Manual 6.0*, 2018.

[2] Wang, G. Gary, and S. Shan. "Review of Metamodeling Techniques for Product Design with Computation-intensive Processes." *Proceedings of the Canadian Engineering Education Association* (2011).

[3] Stander, N., Craig, K.J. On the robustness of a simple domain reduction scheme for simulation-based optimization, *Engineering Computations*, 19(4), pp. 431-450, 2002.

[4] Jones, Donald R., Matthias Schonlau, and William J. Welch. "Efficient global optimization of expensive black-box functions." *Journal of Global optimization*13.4 (1998): 455-492.

[5] Bichon, Barron J., et al. "Efficient global reliability analysis for nonlinear implicit performance functions." *AIAA journal* 46.10 (2008): 2459-2468.

[6] Hurtado, Jorge E. "Filtered importance sampling with support vector margin: a powerful method for structural reliability analysis." *Structural Safety* 29.1 (2007): 2-15.

[7] Basudhar, Anirban, et al. "Constrained efficient global optimization with support vector machines." *Structural and Multidisciplinary Optimization* 46.2 (2012): 201-221.

[8] Lee, Herbert KH, et al. "Optimization subject to hidden constraints via statistical emulation." *Pacific Journal of Optimization* 7.3 (2011): 467-478.

[9] Basudhar, Anirban, Samy Missoum, and Antonio Harrison Sanchez. "Limit state function identification using support vector machines for discontinuous responses and disjoint failure domains." *Probabilistic Engineering Mechanics*23.1 (2008): 1-11.

[10] Missoum, Samy, Palaniappan Ramu, and Raphael T. Haftka. "A convex hull approach for the reliability-based design optimization of nonlinear transient dynamic problems." *Computer methods in applied mechanics and engineering*196.29 (2007): 2895-2906.

[11] Basudhar, Anirban, and Samy Missoum. "Adaptive explicit decision functions for probabilistic design and optimization using support vector machines."*Computers & Structures* 86.19 (2008): 1904-1917.

[12] Basudhar, Anirban, and Samy Missoum. "An improved adaptive sampling scheme for the construction of explicit boundaries." *Structural and Multidisciplinary Optimization* 42.4 (2010): 517-529.

[13] Bourinet, J-M., F. Deheeger, and M. Lemaire. "Assessing small failure probabilities by combined subset simulation and support vector machines."*Structural Safety* 33.6 (2011): 343-353.

[14] Basudhar, A. Multi-objective Optimization Using Adaptive Explicit Non-Dominated Region Sampling. *11th World Congress on Structural and Multidisciplinary Optimization*, Sydney, Australia, 2015.

[15] Basudhar, Anirban, Nielen Stander, Imtiaz Gandikota, Åke Svedin, and Katharina Witowski. Design Tolerance Optimization using LS-OPT. German LS-DYNA Forum, 2014.

[16] Vapnik, V.N. *Statistical learning theory*. Vol. 1. New York: Wiley, 1998.