Adaptive Sampling Using LS-OPT®

Anirban Basudhar[1]

[1]Livermore Software Technology Corporation, Livermore, CA

1 Introduction

LS-OPT is a design optimization and probabilistic analysis package with an interface to LS-DYNA® that provides a flexible framework to solve several types of design problems. In order to solve the problem, it runs simulations at multiple samples that are selected all at once (single iteration) or iteratively [1]. The iterative approach has two main advantages:

- it does not require prior knowledge about the sufficient number of samples and instead provides a convergence history
- it can use updated information from the previous runs to select the samples smartly and thus typically reduces the number of required simulations

While LS-OPT supports both direct and metamodel-based solution methods, the discussion in this study relates to the sampling for metamodel-based [2][3] and classifier-based methods [4][5]. The paper will present an overview of the existing sampling strategies as well as some new adaptive sampling tools that are being developed for release in LS-OPT 6.1.

Three metamodel-based optimization strategies have been available in LS-OPT for quite some time – single iteration, sequential and sequential with domain reduction [1][6]. Among these the last one falls in the category of adaptive sampling methods. It sequentially adapts the variable bounds for the sampling region based on the previous predicted optimum and thus samples in the vicinity of the optimum. Efficient Global Optimization (EGO) [7], based on the kriging metamodel [8], has been added as a fourth optimization strategy in LS-OPT 6.0. It selects one point per iteration based on the objective function prediction and the prediction variance, while the remaining samples are space-filling.

The adaptive nature of the above strategies stems from the consideration of the previous optimum and sampling density information. However, they do not give any particular emphasis to the role of design constraints in determining the sample selection, except in the vicinity of the optimum. LS-OPT does support the definition of sampling constraints that are explicit functions of the design variables. These constraints are used to restrict the sampling domain to the regions perceived as important. However, very often the constraints are not available as analytical functions and are instead obtained from the simulations. Additionally, in the present LS-OPT implementation the sampling constraints are automatically added as design constraints as well. In other words, the use of sampling constraints is limited to sampling within the explicitly defined feasible regions. This may sometimes be useful when the exact constraints are known, but may not be the best sampling strategy when the goal is something other than simply finding feasible samples or when the constraints are not available as explicit functions. In the general case the constraint boundary needs to be estimated. Also, strategies other than selecting samples in the feasible space, e.g. sampling in the vicinity of the constraint boundary, are needed [9][10][11].

To enable adaptivity of the design of experiments (DOE), the sampling constraint capability of LS-OPT has been enhanced with version 6.1 release in mind. The methodology is based on explicit design space decomposition (EDSD) [11][12] using support vector machine (SVM) classifiers [13], which have been implemented in LS-OPT 6.0. As part of the recent development for version 6.1, they can now be assigned as sampling constraints that are not necessarily design constraints. One can choose to select the feasible region or in the vicinity of the SVM constraint boundary. Sampling near the boundary is especially important in the context of reliability assessment. In the previous LS-OPT versions the sampling for reliability assessment was limited to a single iteration. This restriction has also been removed in this work with the ability to sample in the vicinity of the approximated failure boundary iteratively. These techniques are expected to reduce the number of simulations required to arrive at the solution with the same level of accuracy.

The paper is organized as follows. Section 2 lists the currently available sampling techniques in LS-OPT. This is followed by a brief introduction to classification-based design and SVM classifiers in Section 3. The new tools and methods for classfication-based adaptive sampling using LS-OPT are presented in Section 4. Examples of adaptive sampling using EDSD are demonstrated in Section 5. The tools are still in their preliminary stages of development, so the paper ends with a discussion on the future scope for development in Section 6 that includes ease of use as well as additional methods.

2 Existing Iterative Sampling Strategies and Tools in LS-OPT

2.1 Sequential Sampling

In this approach, a specified number of samples are added for each iteration. The first point at each iteration is obtained by solving the design optimization problem while the rest are typically space-filling. Thus, other than the first sample, the rest can be anywhere in the space without consideration of the objective function or the constraints. Thus, the sampling doesn't really adapt itself as more simulation results are obtained.

2.2 Sequential Sampling with Domain Reduction

The strategy is more adaptive in nature as the design problem is considered while selecting all the samples. The first sample is selected in the same manner as in the sequential approach, but in addition the other samples are also selected within a subregion in the vicinity of the previous optimum. The size of this box-shaped subregion typically decreases as the optimum converges.

2.3 Efficient Global Optimization

This is also a sequential approach in which the design problem is considered for selecting the first sample only. The only difference is that instead of minimizing the objective function, an expected improvement function (EIF) is maximized to select the first sample. The EIF considers both the mean prediction of the objective function as well as its variance. This method is based on the Kriging metamodel, which provides a measure of the prediction variance. Although the LS-OPT 6.0 implementation of EGO selects all points expet the first using a space-filling method, there is scope for improving this by considering the EIF for these samples as well.

3 Classification-based design Using LS-OPT

Sampling constraints based on a classification-based approach are being developed for version 6.1. Before presenting those tools, the basic idea of classifiers implemented in LS-OPT 6.0 is introduced in this section.

3.1 Basic Classification-based design Methodology

In both design optimization and reliability assessment one of the main tasks is the demarcation between acceptable (feasible/safe) and unacceptable (infeasible/failed) designs. In optimization, the optimum design is located in the feasible space. Similarly, in reliability assessment, the failed samples contribute to the failure probabiliy. If the boundary separating acceptable and unacceptable regions of the design space is available analytically in terms of the design variables, reliability assessment and optimization become relatively straightforward. However, in general such a boundary is not available. Instead, only the responses corresponding to specific designs are available. In metamodel-based approaches the response values at these specific points in the design space are used to construct analytical response approximations to predict the responses at any general design. These approximations are then used to demarcate the acceptable and unacceptable design space based on threshold values. However, a different approach is used in classification based design. Classification methods only require pass/fail information at a few specified samples that are used for training. This information is readily available using simulations at these samples even if the responses are binary or discontunuous. The decision boundary is constructed as the classifier that optimally separates the acceptable and unacceptable training samples. The difference between metamodel-based and classification-based methods to determine acceptability of any general design alternative is shown in Fig 1. The classification-based method takes a decision directly based on the position of the new sample in the design space whereas in the metamodel-based method, the decision is taken based on the corresponding predicted response value and threshold. As the decision-making using a trained classifier is straightforward and cheap, the decision boundary can be used as an optimization constraint or for reliability assessment. Additionally a classifier can also be used to define the sampling domain, which is the main topic of this paper. Once the classifier seperating the feasible and infeasible samples is constructed, it can be used for adaptive sampling in varios ways, e.g. sampling in the feasible region, sampling in the vicinity of the boundary (more in Section 4) etc.

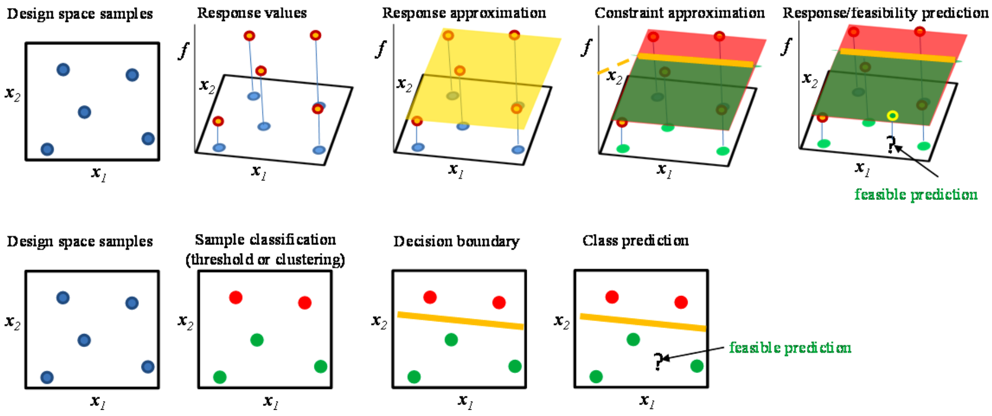

Fig.1: Summary of basic classification method (bottom) and comparison to metamodeling (top).

3.2 Support Vector Machine Classification

SVMs belong to a class of macine learning techniques 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 separaing hyperplane. The separating boundary demarkaing 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).

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

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 ($s(\mathbf{x}) = 0$).

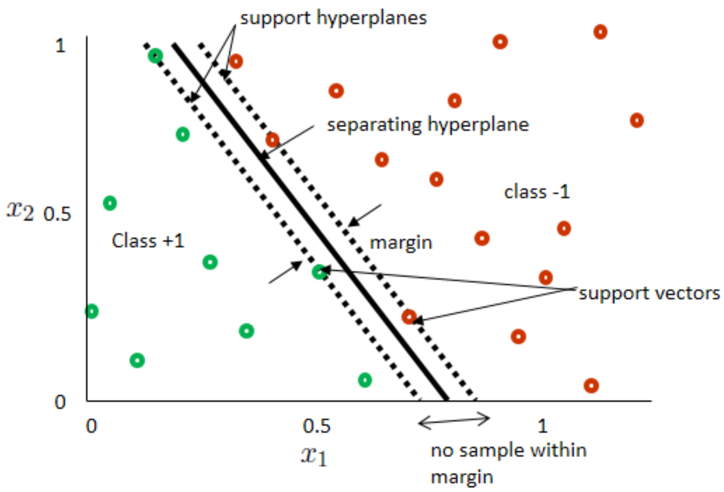

Fig.2: Linear classification using SVM.

4 Classifier-based Adaptive Sampling Using LS-OPT

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:

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

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

$$\begin{aligned} &\max_{\boldsymbol{x}} \|\boldsymbol{x} - \boldsymbol{x}_{nearest}\| \\ s.t. \quad &\left| b + \sum_{i=1}^{N} \alpha_i y_i K(\boldsymbol{x}, \boldsymbol{x}_i) \right| \leq \varepsilon \end{aligned} \tag{3}$$

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 constraint, with some tolerance, can be defined as:

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

Additionally, the fraction of samples to be selected within the classifier constrained regions can also be specified. The remaining samples are selected to fill the entire space without the EDSD sampling constraints.

5 Examples

Two examples are presented in this section – one for optimization and the other for failure probability calculation using Monte Carlo analysis. The EDSD-based adaptive strategy is compared to the sequential space-filling strategy for these examples.

5.1 Design Optimization with Mode Tracking (Discontinuous Frequency Constraint)

This example consists of an LS-DYNA simplified car model (Fig 3) for which the mass is minimized while constraining the first torsional mode to be greater than a certain limit. The thickness values of the bumper (tbumper) and the rail (trailb) are the optimization variables. Thus the optimization problem is:

$$\begin{aligned} &\min_{x_1,\ x_2} \quad Mass \\ s.t. \quad &\nu_{torsion} > 2.2 \end{aligned} \tag{5}$$

where $\nu_{torsion}$ is the first torsional mode frequency, x_1 and x_2 are the thicknesses of bumper and rail, respectively. The first torsional mode for the baseline design was identified manually before automatically detecting the closest mode shape for the other samples, based on the modal assurance criterion (MAC). As the torsional mode is tracked, the mode number can switch from one design to another. This is also typically accompanied by a sudden jump in the frequency, leading to a response discontinuity (Fig 3).

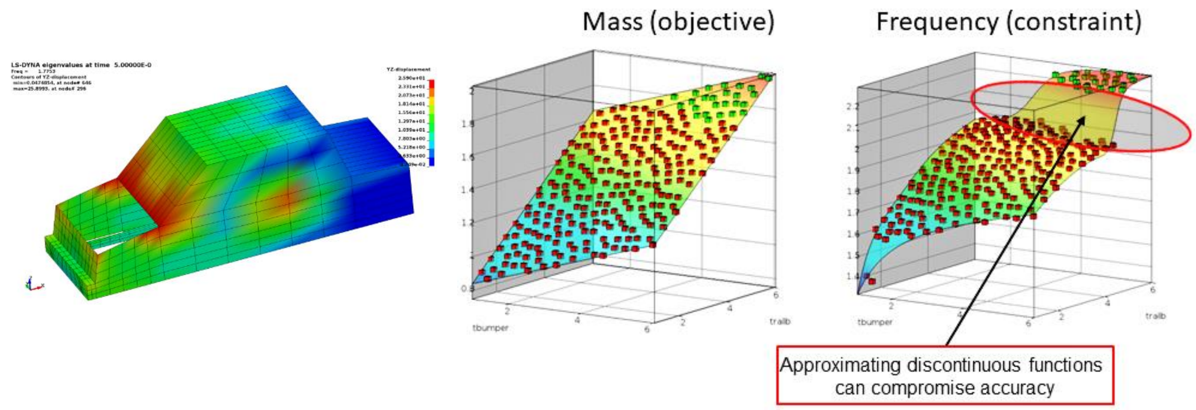

Fig.3: Simplified car LS-DYNA model for modal analysis (left). Mass and discontinuous frequency response (right).

Due to the discontinuities in the frequency response, this example is suited for classifier-based constraint handling [14]. Therefore, the optimization is solved using an SVM decision boundary. First a space-filling sequential approach is used to update the SVM boundary and the optimum iteratively. This is compared to an adaptive sampling strategy with samples selected within the SVM margin to get higher accuracy near the boundary. The fraction of samples within the margin is varied (0.5, 0.75, 1); rest of the samples are space-filling. All the cases are run for 25 iterations with 10 samples per iteration. The best computed point history is given in Table 1. The initial constraint boundary is shown in Fig 4. The optimization histories are also shown in Fig 5. Additionally the updated constraint boundaries and the samples are shown in Fig 6 – Fig 9. It can easily be seen that most of the samples are close to the constraint boundary, resulting in high local accuracy, when the samples are added in the margin.

Strategy \ Samples	Initial	50	100	150	200	250
Sequential Space-filling	1.1493	1.76644	1.76644	1.75527	1.75424	1.75354
50% SVM Margin	1.1493	1.77290	1.75446	1.75336	1.75314	1.75314
75% SVM Margin	1.1493	1.75626	1.75333	1.75311	1.75311	1.75311
100% SVM Margin	1.1493	1.75447	1.75347	1.75324	1.75314	1.75314

Table 1: Comparison of the evolution of the optimal mass value using different sampling strategies. The red text indicates an infeasible staring design.

It is interesting to note in Fig 5 that the sequential space-filling strategy cannot find a feasible solution for the initial few iterations. It is likely to be due to the local inaccuracy close to the boundary. Fig 4 shows the lack of samples in the vicinity of the initial boundary, giving it considerable flexibility to translate without misclassifying the existing samples. The same is true over the initial few iterations. The sudden drops in the mass during some intermediate iterations can be attributed to the SVM trying to locate new potentially feasible regions that have not been sampled previously.

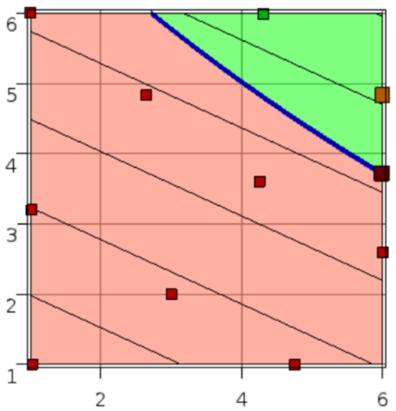

Fig.4: Modal Analysis Example. Initial Constraint boundary approximation and sampling.

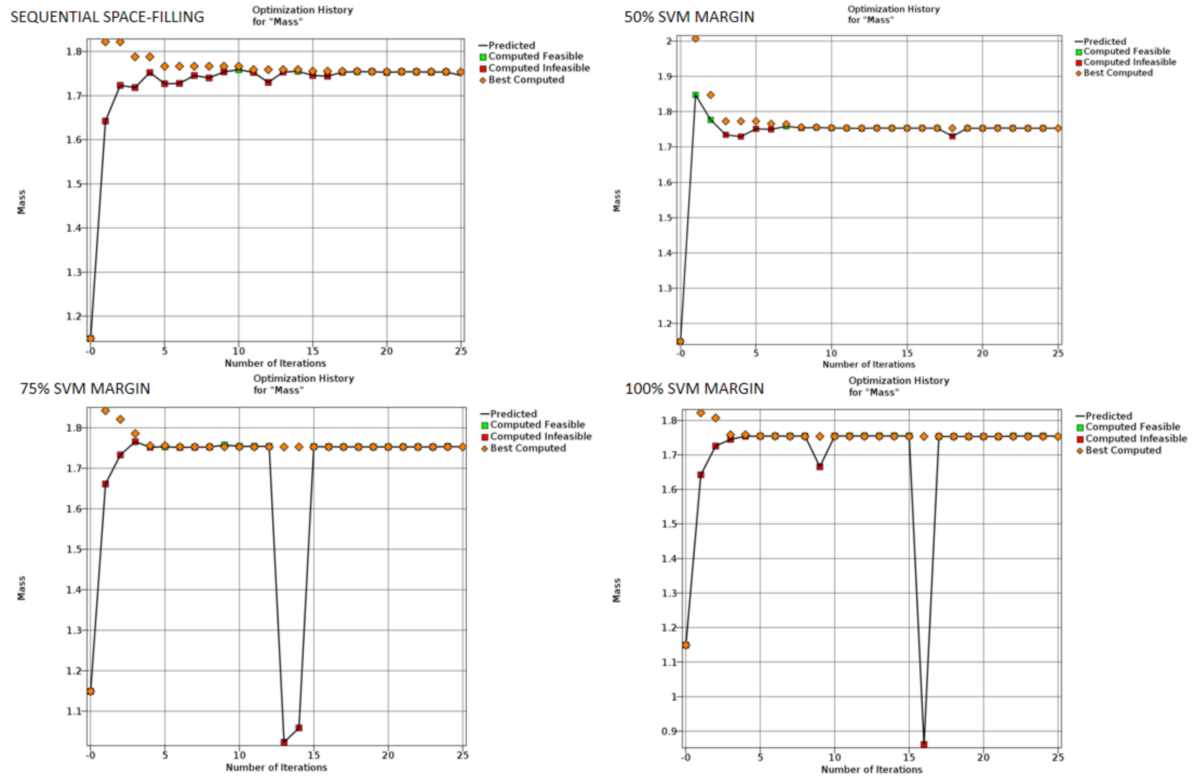

Fig.5: Optimization history of modal analysis example.

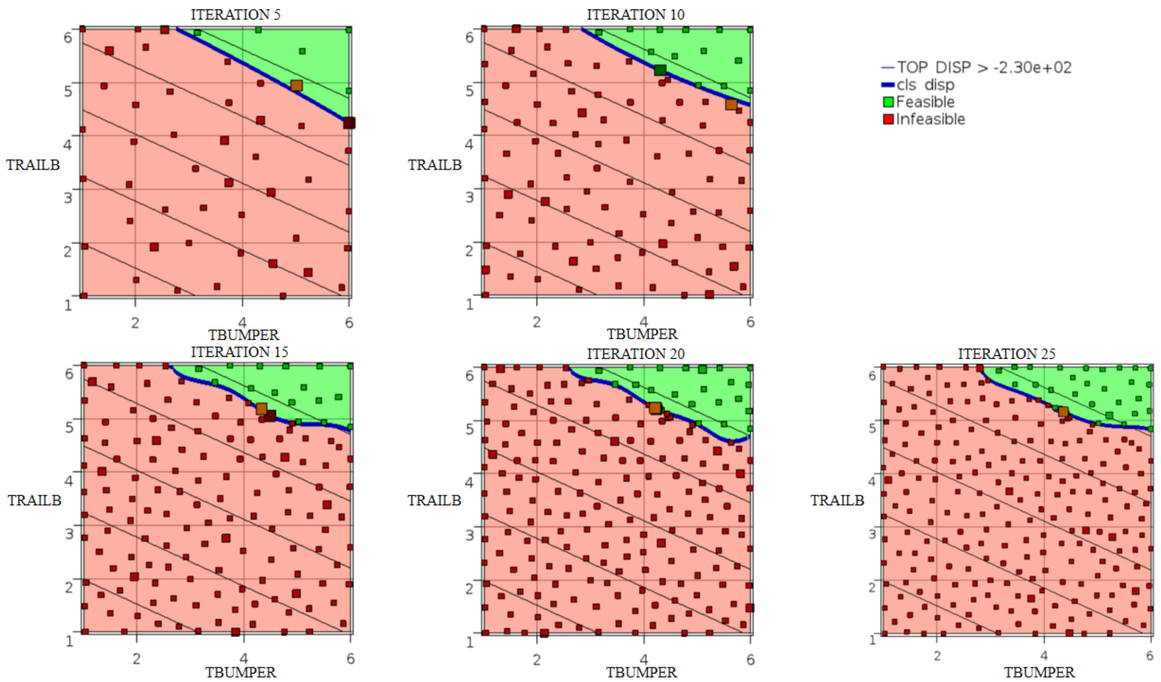

Fig.6: Modal Analysis Example. Constraint boundary evolution with sequential space-filling sampling

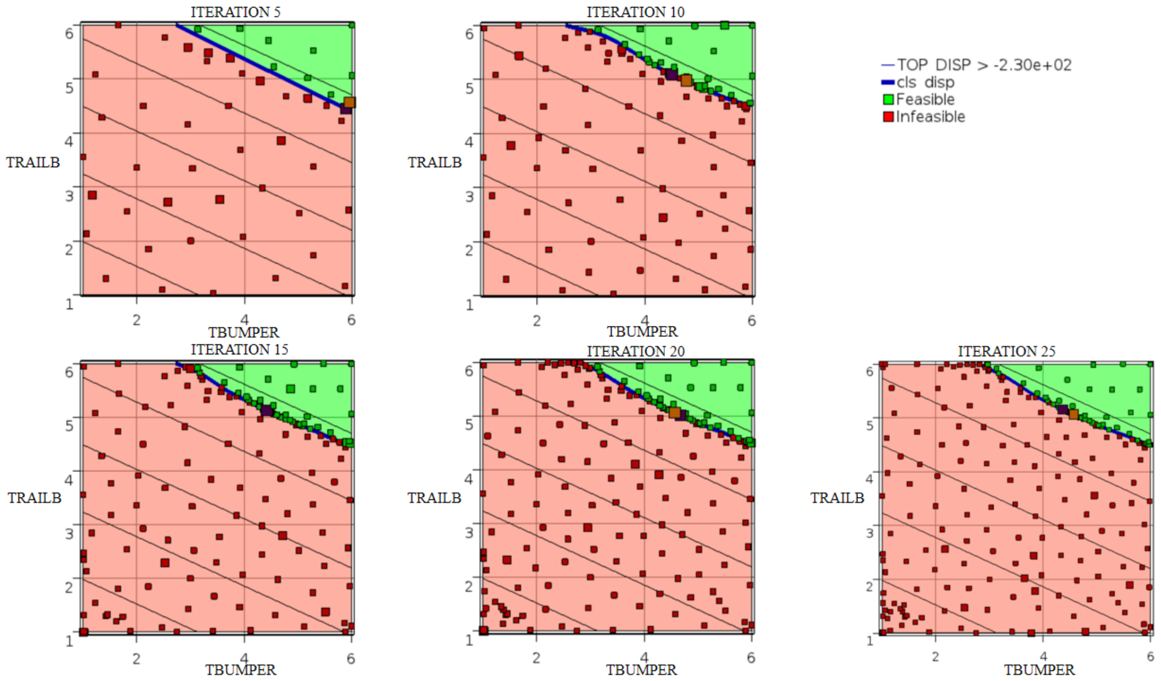

Fig.7: Modal Analysis Example. Constraint boundary evolution with 50% SVM margin sampling

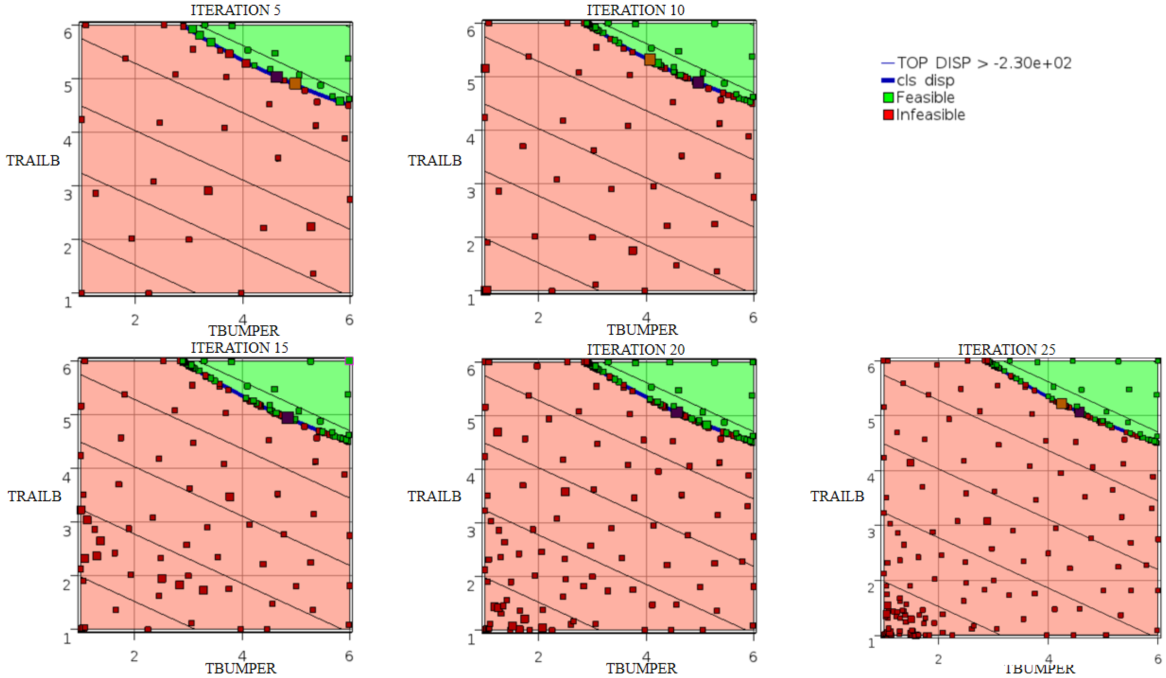

Fig.8: Modal Analysis Example. Constraint boundary evolution with 75% SVM margin sampling

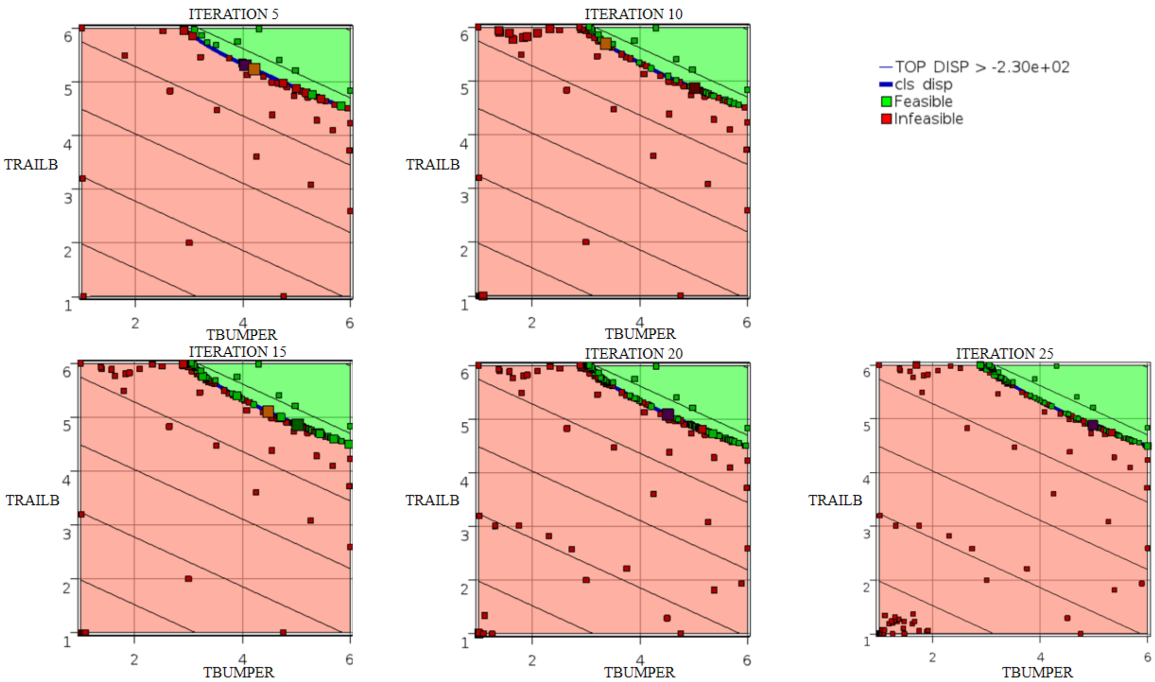

Fig.9: Modal Analysis Example. Constraint boundary evolution with 100% SVM margin sampling

5.2 Reliability Assessment for Tube Impact Example

The previously available reliability assessment methods in LS-OPT were restricted to single iteration direct and metamodel-based Monte Carlo analysis. This restriction will be removed in LS-OPT 6.1. This example presents a sequential reliability analysis of a tube impact problem. The z-dispacement at the top end of the tube is constrained to be less than 230 mm. The tube thickness THK and the yield strength scale factor SIGY are Normally distributed random vriables N(1,0.05). The thickness is in mm and SIGY is dimensionless.

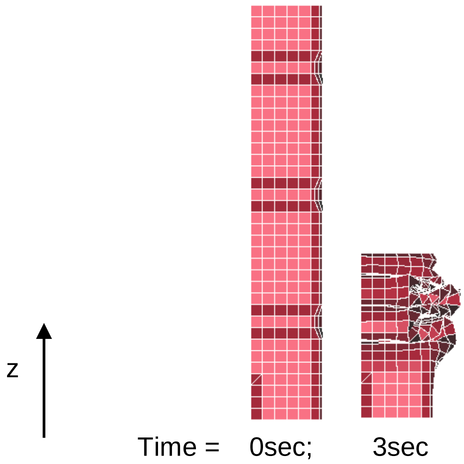

Fig.10: Tube Impact Example. The displacement at the top end is constrained

The failure probability is obtained using sequential space-filling sampling as well as using EDSD-based adaptive sampling within the SVM margin. The sampling space is a box around the mean design with bounds of $\pm 4\sigma, \sigma$ being the standard deviation. The EDSD-based adaptive sampling can also be used to complement a metamodel-based approach, which is suited for this example due to the smooth response. To demonstrate this, the sampling is guided by the SVM classifier, but the failure probabilities are calculated using feedforward neural network approximations constructed using those samples. An initial DOE of 10 spacefilling samples is used, and 10 more per iteration are added sequentially with a limit of 20 iterations. The failure probabilities are listed in Table 2. These metamodel-based probabilities are calculated with 10^6 samples, so the confidence intervals are quite small.

Strategy \ No. of samples	10	50	100	150	200
Sequential Space-filling	0.386	0.385	0.380	0.377	0.375
50% SVM Margin	0.386	0.377	0.378	0.378	0.378
75% SVM Margin	0.386	0.379	0.379	0.380	0.380
100% SVM Margin	0.386	0.379	0.381	0.379	0.380

Table 2: Tube Impact failure probability evolution using different update strategies.

The actual failure probability is calculated using 20000 direct Monte Carlo samples. The 95% confidence interval of the failure probability is $[0.375, 0.388]$ with a mean estimate of 0.381. It is interesting to note that the mean failure probability calculated with space-filling sampling sampling is equal to the lower bound of this interval, whereas the adaptively updated values are closer to the mean. Fig 11 shows the initial boundary. The updated boundaries and samplings for the four cases are show in Fig 12 – Fig 15.

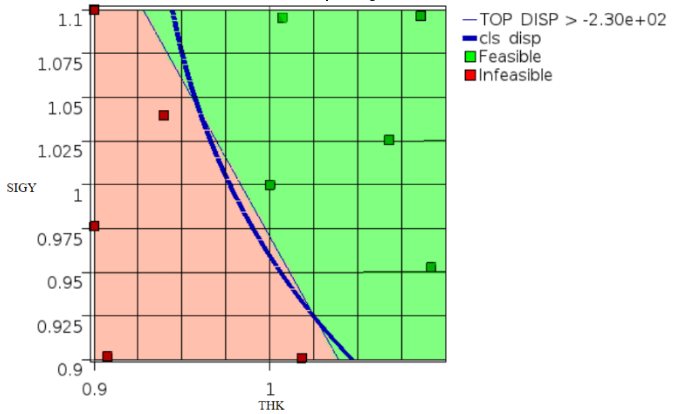

Fig.11: Tube Impact Example. Initial failure boundary using 10 space-filling samples.

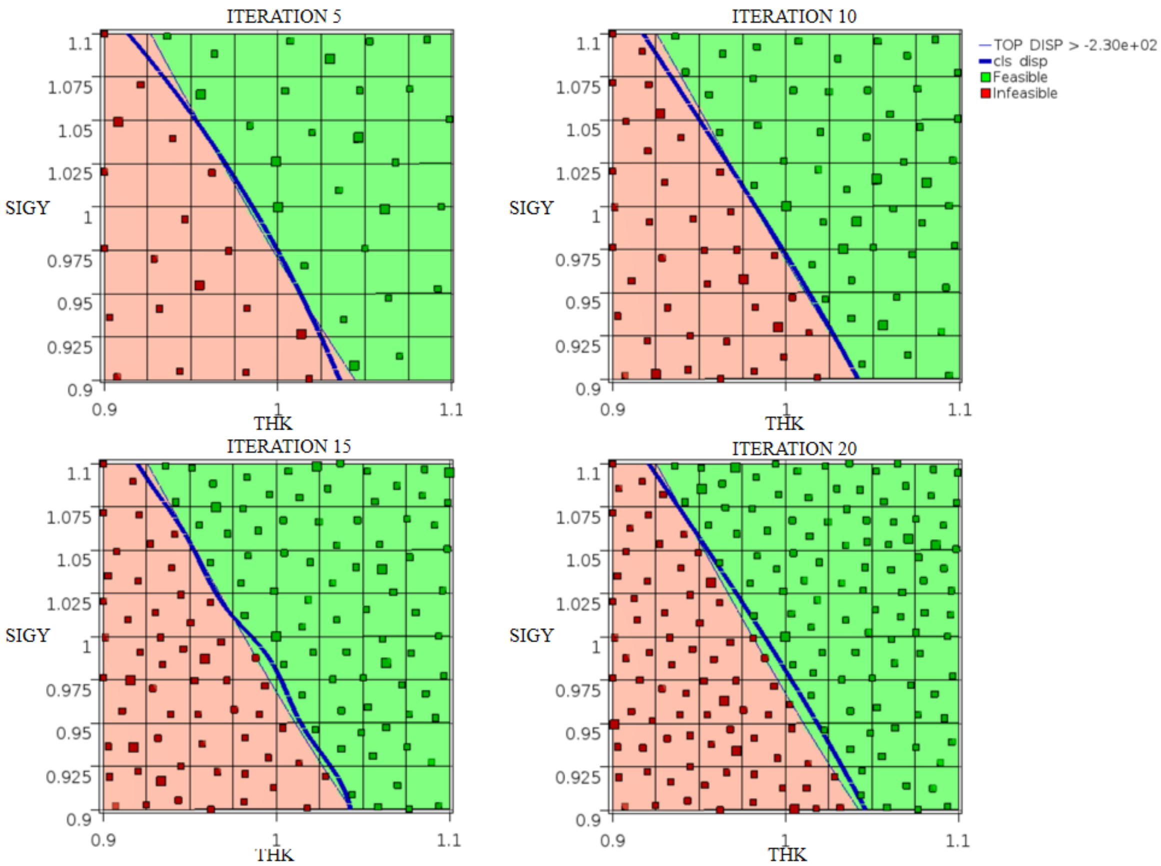

Fig.12: Tube Impact Example. Failure boundary evolution with sequential space-filling sampling.

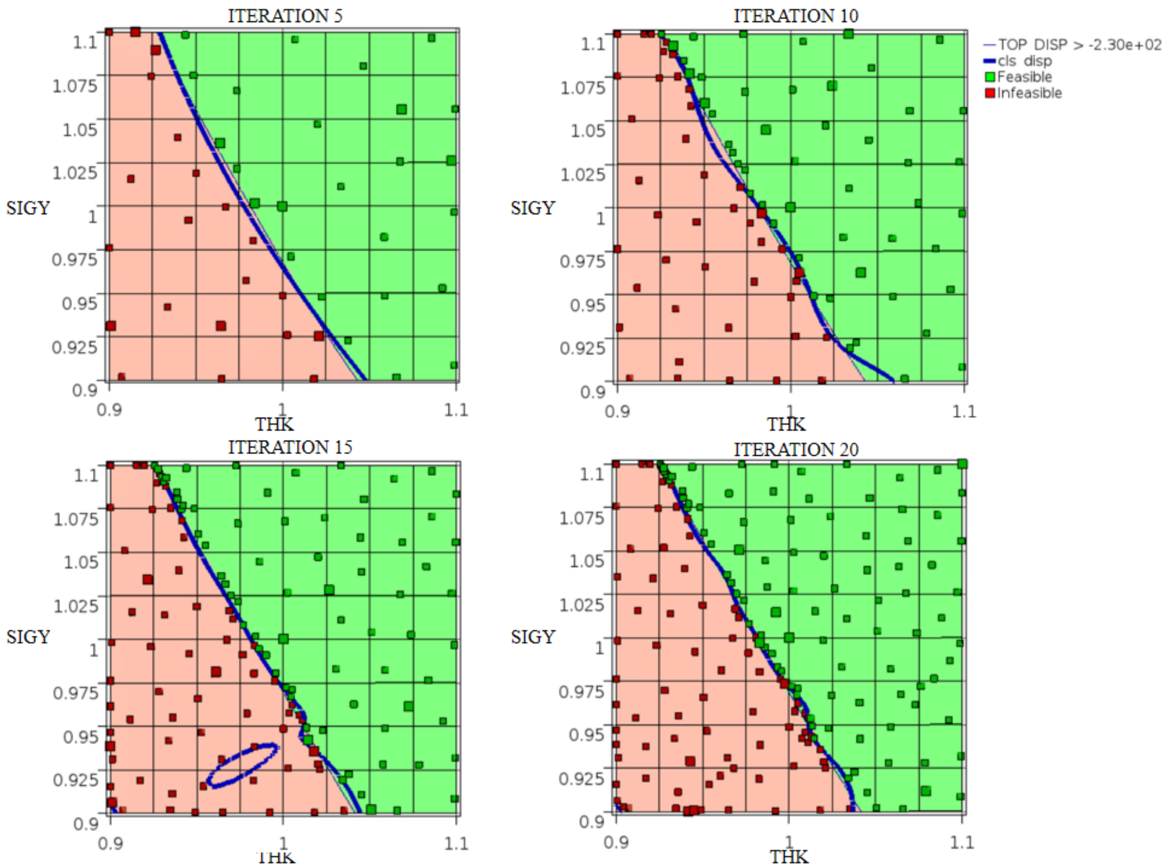

Fig.13: Tube Impact Example. Failure boundary evolution with 50% SVM margin sampling

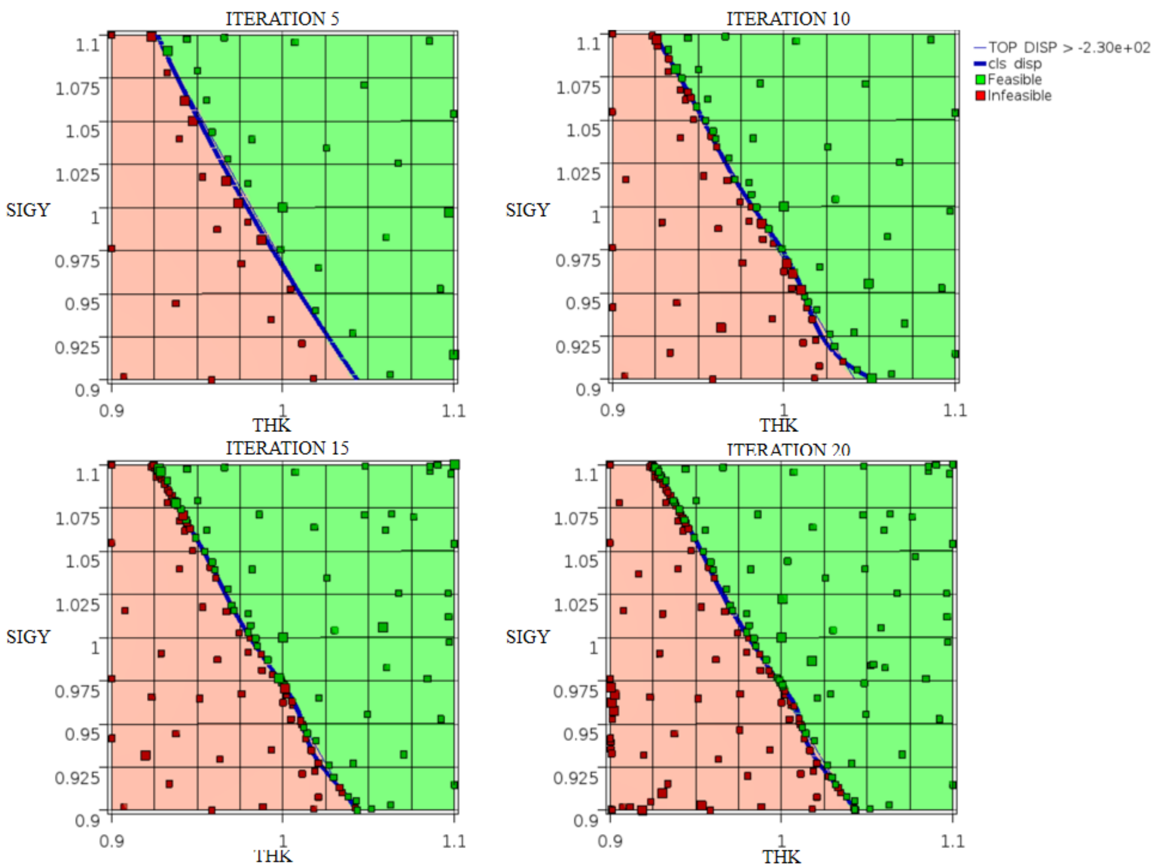

Fig.14: Tube Impact Example. Failure boundary evolution with 75% SVM margin sampling

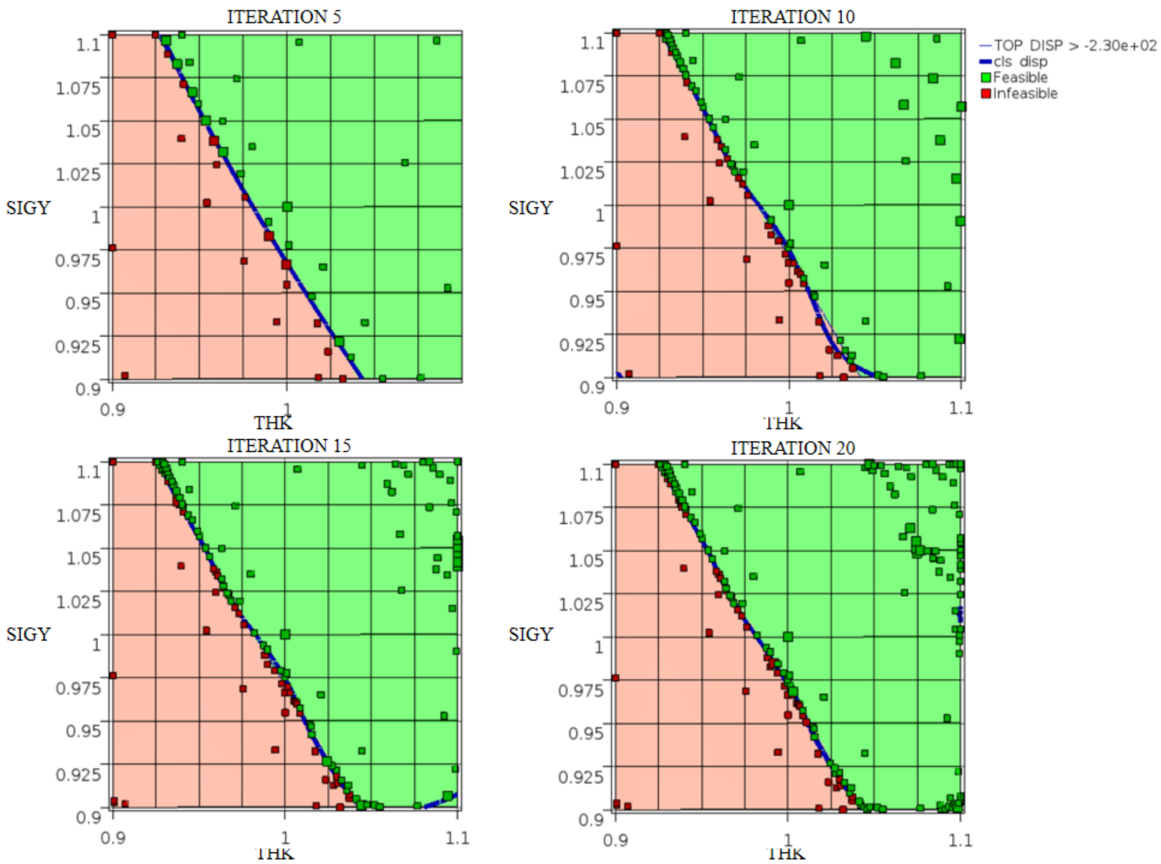

Fig.15: Tube Impact Example. Failure boundary evolution with 100% SVM margin sampling

6 Conclusions

LS-OPT provides a general purpose optimization and probabilistic analysis framework that is quite flexible in terms of the applications. It attempts to solve the problems in a robust and efficient manner to provide the best solution as well as to save computation time for the solver simulations. Adaptive sampling is an essential part of this endeavour to attain higher efficiency. Previously, adaptive sampling in LS-OPT was mainly limited to a single strategy for optimization, while there was no such strategy for failure probability calculation.

To overcome the above limitation, the sampling constraints facility in LS-OPT has been enhanced. While it was limited to explicitly defined constraints earlier, and thus inapplicable in most problems, the new tool allows sample selection based on classifier-based constraint boundary approximations. Thus it allows the samples to be adaptively selected based on different criteria even when the constraints are not available explicitly, which is often the case (i.e. constraint responses obtained from simulation results or experiments). The new approach also gives higher flexibility in terms of the sampling criteria.

Additionally, the sequential sampling approach (adaptive or space-filling) has been expanded to Monte Carlo simulation-based probability of failure calculation, instead of being limited to optimization. This allows one to study the convergence of the failure probability. The adaptive approach also enhances the accuracy and the efficiency of the reliability analysis.

This work is still in progress and more flexibility as well as automation of the adaptive strategy will be provided as it develops further. More examples will also be tested with higher level of difficulty and higher dimensionality. Some post-processing tools also need to be developed further for iterative probabilistic analysis. Additionally, the EGO strategy will also be enhanced in the future to enable more efficient parallel sampling. A non-domination classification criterion will also be introduced to facilitate adaptive sampling for multi-objective optimization [15].

7 References

[1] Stander, N., Roux, W.J., Basudhar, A., Eggleston, T, Craig, K.-J. LS-OPT Users Manual, Version 6.0. April 2019.

[2] Myers, R.H., Montgomery, D.C. *Response Surface Methodology. Process and Product Optimization using Designed Experiments*. Wiley, 1995.

[3] Simpson, T. W., Toropov, V. V., Balabanov, V. O., and Viana, F. A. C. Design and Analysis of Computer Experiments in Multidisciplinary Design Optimization: A Review of How Far We Have Come - or Not. In 12th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference. Reston, VA, 2008.

[4] Basudhar, A., Missoum, S. and Sanchez, A.H. Limit state function identification using support vector machines for discontinuous responses and disjoint failure domains. *Probabilistic Engineering Mechanics*, *23*(1), pp.1-11, 2008.

[5] Hurtado, J.E. Filtered importance sampling with support vector margin: a powerful method for structural reliability analysis. *Structural Safety*, *29*(1), pp.2-15, 2007.

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

[8] Jones, D.R., Schonlau, M. and Welch, W.J. Efficient global optimization of expensive black-box functions. *Journal of Global optimization*, *13*(4), pp.455-492, 1998.

[8] Cressie, N. The origins of kriging. *Mathematical geology*, *22*(3), pp.239-252.

[9] Basudhar, A. and Missoum, S., 2008. Adaptive explicit decision functions for probabilistic design and optimization using support vector machines. *Computers & Structures*, *86*(19-20), pp.1904-1917, 1990.

[10] Bichon, B.J., Eldred, M.S., Swiler, L.P., Mahadevan, S. and McFarland, J.M. Efficient global reliability analysis for nonlinear implicit performance functions. *AIAA journal*, *46*(10), pp.2459-2468, 2008.

[11] Basudhar, A. and Missoum, S. An improved adaptive sampling scheme for the construction of explicit boundaries. *Structural and Multidisciplinary Optimization*, *42*(4), pp.517-529, 2010.

[12] Basudhar, A. Computational optimal design and uncertainty quantification of complex systems using explicit decision boundaries, PhD dissertation, The University of Arizona, 2011.

[13] Vapnik, V. and Vapnik, V. Statistical learning theory Wiley. *New York*, pp.156-160, 1998.

[14] Basudhar, A., Witowski, K. and Gandikota, I. Classification-based Optimization and Probabilistic Analysis Using LS-OPT. LS-DYNA User's conference, Dearborn, MI, 2018.

[15] Basudhar A. Multi-objective Optimization Using Adaptive Explicit Non-Dominated Region Sampling. In11th World Congress on Structural and Multidisciplinary Optimization 2015.