optiSLang as algorithmic toolbox for RDO

In the field of product engineering, the introduction of technical innovations as well as the requirements for improvement of product performance are accelerating and increasing. Thus, product life cycles and development times are constantly getting shorter. Therefore, there is a high demand for virtual product development (Computer Aided Engineering, CAE) and CAE-based optimization tools. At the same time, quality requirements of reliability and robustness are increasing. Robust and optimized product designs are needed. Considering this, CAE-based Robust Design Optimization (RDO) becomes more and more important. To address these challenges, optiSLang offers you an algorithmic toolbox to meet all requirements.

With optiSLang, you are able to:
- couple any (CAE) solver software to your workflow and use it for automation of your process chain
- manage an unlimited number of parameters due to automatic input reduction
- handle design failure up to 50%, non-linear CAE-problems and noisy responses
- standardize your process for quality assurance

Sensitivity analysis, optimization and robustness evaluation with a minimum amount of user input and solver runs for your effective virtual product development.

The main benefits of optiSLang’s sensitivity analysis are:
- Automatic identification of the MOP with maximal progression quality by determination of best approximation model (e.g. polynomial, Moving Least Squares) for each output in the optimal subspace of important input parameters
- Automatic reduction of inputs of those being important to describe the output variation
- Assessment of the prediction quality of the MOP by optiSLang’s Coefficient of Prognosis (CoP) including quantification of input importance
- Quantification of numerical noise of the outputs
- Estimation of the optimization potential: are the chosen input parameters in the right range to achieve the objective? Are objectives conflicting? Can the MOP be used to approximate the solver output for further optimization?

Optimal design selection
Within the optimization framework, several algorithms for single and multi-objective optimization are available:
- Gradient-based methods
- Adaptive Response Surface Method
- Nature-inspired algorithms such as evolutionary algorithms or particle swarm optimization

The choice of the most suitable and efficient optimization method is automatically provided by optiSLang workflow wizards. This choice is based on the results of the sensitivity analysis and the optional user-defined input, like the number of objectives, parameters and failed designs as well as the level of solver noise.

There are two main optimization strategies:
1. The search of the best design is performed on the MOP as a very efficient pre-search without any additional solver calls. Then, the best design of the MOP can be used as the final design after validation with a single solver call or as a start design for a subsequent optimization
2. Standard optimization with direct solver calls using the reduced input parameter set of the sensitivity analysis

Evaluate tolerances
To match quality requirements of designs exposed to given material and environmental variations or geometry tolerances, their scatter of all important responses needs to stay within acceptable ranges. To prove that, a variance based robustness evaluation can be performed.

In this context, the first step is the assignment of scattering input parameters by definition of their scatter range, distribution and correlation among each other. Then, the robustness space is scanned by advanced sampling methods and the resulting response scatter is quantified. By using the MOP ap-
Several strategies for robustness evaluation are possible to be realized in optiSLang:

1. Robustness evaluation after determination of the best design
2. Sequential optimization and robustness evaluation
3. Fully coupled robust design optimization

Example: Tuning Fork in ANSYS classic

Introduction

With the simple example of a tuning fork, the whole workflow will be presented comprising integration, sensitivity analysis, optimization and robustness evaluation in optiSLang. A modal analysis with a fixed support of the tuning fork and an undamped oscillation is performed within the parametric modeling environment ANSYS classic.

Integration

The integration is created via batch-script that executes ANSYS classic in batch mode and ASCII-input and -output files. This integration is fully supported in optiSLang and therefore very simple to generate.

Definition of input and output parameters

For the sensitivity analysis and optimization, six input parameters were considered. Namely, these are the geometry parameters rod length, width, grip length, width, radius and depth (Fig. 1). At the same time, four output parameters are evaluated: the frequencies obtained by the modal analysis (frequency 1, 2 and 3) and the mass.

Optimization task

The aim of the optimization was to equalize the main frequency 1 to 440 Hz and the higher frequencies to their duplicates and triplicates respectively. Thus, a single objective target optimization was performed by taking the quadratic deviation to the target value and by weighing the importance of the targets. The objective function which should be minimized is \( f(x) = (\text{frequency}_1-440)^2 + ((\text{frequency}_2-880)/2)^2 + ((\text{frequency}_3-1320)/3)^2 \). As a constraint, the mass should be lower than 12 g.

Sensitivity analysis

As a pre-optimization step and for the identification of the most important inputs, a sensitivity analysis was performed. By obtaining an MOP for each output, an input parameter reduction is possible. For instance, 98 % of frequency 1 can be explained by the predefined settings of the optimizer. 200 design points were evaluated until the algorithm was converging. In a second step, the best design of this MOP pre-search was used as a start design for an optimization with Adaptive Response Surface Method (ARSM) with direct solver calls. The search window of the start range for the ARSM was reduced to 10 % to generate a local search. The post-processing window of these results is illustrated in Fig. 3.

The benefits of using optiSLang for this optimization are:

- The optimization was performed much faster: without the MOP, 200 solver calls would have been necessary for the first optimization step. This results in 33 minutes of time saved with calculation durations of 10 seconds per design.
- The properties of the best design were considerably improved. By comparing with the original design, the mass was reduced from 29.2 g to 9.9 g and the frequencies matched the target values (Fig. 3). The starting values were 414, 417 and 959 Hz.

Robustness evaluation

For the robustness evaluation of the best design, besides geometry parameters, material parameter variations (here: E-module) were also considered in order to recognize critical inputs and to reduce uncertainties. All geometry parameters were varied with 0.5 % scatter relative to the mean and 1 % for the E-module with normal distribution respectively.

For a robust design, ± 7 Hz variation of the first frequency were defined to be acceptable. As a result of the robustness evaluation (Fig. 4 (A)), 13 % of the designs are not in this range indicating that the tuning fork design is not robust. To reduce the output variation, either the mean value of the output or the scatter of the most important inputs needs to be decreased. As the first option is meaningless for the frequency of a tuning fork, the second strategy was implemented here. The analysis of the importance of the inputs by their sensitivity measures (Fig. 4 (B)) indicates that the scatter of the four most important geometry parameters (grip width, rod width, grip length, rod length) induce ~70 % of the total output variation whereas the E-module induces 26 % of the latter. Consequently, the scatter of these four geometrical inputs was reduced to 0.2 % and the E-module to 0.8 % relative variation of their mean in a second robustness analysis. Finally, only 1 % of the designs were not in the acceptable variation range of ± 7 Hz. Thus, by actively influencing only the most important inputs, a robust design could be obtained.

Author // Stephanie Kunath (Dynardo GmbH)

This tuning fork example can be provided to you as a tutorial with example files.