An Overview of LS-OPT® Functionality

with a Preview of Version 4.2

Nielen Stander*

David Björkevik**, Christoffer Belestam**

Katharina Witowski[†]

Trent Eggleston[‡]

*Livermore Software Technology Corporation, Livermore, California, USA

**Engineering Research AB, Linköping, Sweden

[†]Dynamore AG, Stuttgart, Germany

[‡]Self

Summary:

This paper reviews the features of LS-OPT® in the areas of design optimization, multi-disciplinary optimization, reliability-based optimization and multi-objective optimization. Relatively new features for solver job distribution are also discussed. A preview is given of the next release, Version 4.2.

Keywords:

Optimization, Design Optimization, Reliability, Robust Design, Multi-Objective Optimization, MDO, RBDO, MOO, job distribution, computer cluster, experimental design, surrogate model, metamodel

1   Introduction

LS-OPT® [1] is a simulation-based optimization program with a dedicated interface to LS-DYNA® [2]. Other interfaces are available such as a frequency extraction and mode tracking interface that works with MSC NASTRAN. LS-OPT specializes in the following optimization areas:

1. Multi-Criteria Design Optimization. This array of features, which forms the general framework for optimization, allows the handling of multiple design objectives and constraints to produce a Pareto Optimal design set.
2. Reliability-based Design Optimization. This set of features allows the incorporation of uncertainty into the design. A user can for instance specify a probability of failure as part of the constraint specification. The standard deviation of an output response can also be specified and used as any one of the design objectives in order to ensure a robust response.
3. Outlier Analysis allows the display of sources of variation of the design response as a fringe plot super-imposed on the finite element mesh.
4. Parameter Identification. Special features are provided for calibrating systems or materials using test results.

The principle methodology used in LS-OPT is Response Surface Methodology, originally based on polynomial approximations but now also available with other types of surrogate functions, namely Feedforward Neural Networks, Radial Basis Function Networks and Kriging. Radial Basis Function Networks have been selected as the default option since they are highly accurate predictors based on cross-validation. Various types of point selection schemes are available for design sampling and have been matched to the respective surrogate model types.

Optimization can be done either directly or using surrogate models. Reliability-based design optimization requires surrogate models in order to approximate the probability of failure and/or robustness.

Because of a differentiation in needs and resources amongst LS-OPT users, a small number of strategies have been devised to simplify the optimization setup. The following section discusses these strategies as well as the Multi-objective Optimization feature.

2   LS-OPT Methodology

2.1   Surrogate-based Design Optimization

Function evaluation for applications typically served by LS-DYNA can be extremely computer-intensive. These typically involve finite element models featuring element counts in the range of 4-6 million. Hence direct methods in which 5000-10000 simulations may be required for convergence are typically not suitable, even using present day computer clusters perhaps featuring 2000+ processors. The purpose of surrogate-based optimization is to reduce the number of simulations by approximating the design. The creation of a surrogate model also enables features such as Reliability-based Optimization.

2.1.1 Strategies

Optimization strategies have been implemented to address a diversity of design needs, accuracy requirements and availability of computing resources. These are schematically displayed in Fig. 1.

1. *Single Stage:* This is typically used when the user has a fixed computational budget. A single iteration is run and the user only needs to specify how many runs can be afforded. The optimization finishes once all the runs have been concluded and the surrogate models have been built based on the results. The user can of course continue the optimization by adding solver runs should the need arise for further refinement of the surrogate model. RBF networks have been selected as the default surrogate.
2. *Sequential:* This iterative method is typically specified if the user targets a specific accuracy of the surrogate model. Multiple iterations are run and in each iteration a set of design points are added using an Adaptive Space Filling scheme to locate points in sparse areas. In Version 4.2, the PRESS cross-validation error can be specified with a threshold value so that iterations will continue until this value is reached. This method adds points globally so that the final surrogate can serve as

a global design model. This method is particularly suitable for multi-objective optimization. RBF networks have been selected as the default surrogate model.

3. *Sequential with Domain Reduction:* This approach is intended for problems in which the user seeks to converge to a single design or design region. Therefore, a domain reduction scheme (the classical Sequential Response Surface Method (SRSM)) is used to converge to a point in the design space. There are two main variants, depending on the surrogate model selected. For linear polynomials (the default), only the design results of the current iteration are used. However, as a second option, surrogate models such as the RBF networks can also be used in which case all the design points, including points from previous iterations, are utilized. This method is also suitable for Reliability-based or Robust Design Optimization since the surrogate accuracy in the neighborhood of the optimum design, where design variation is being introduced, has been maximized through the focusing of the sampling in this particular region of interest (subregion).
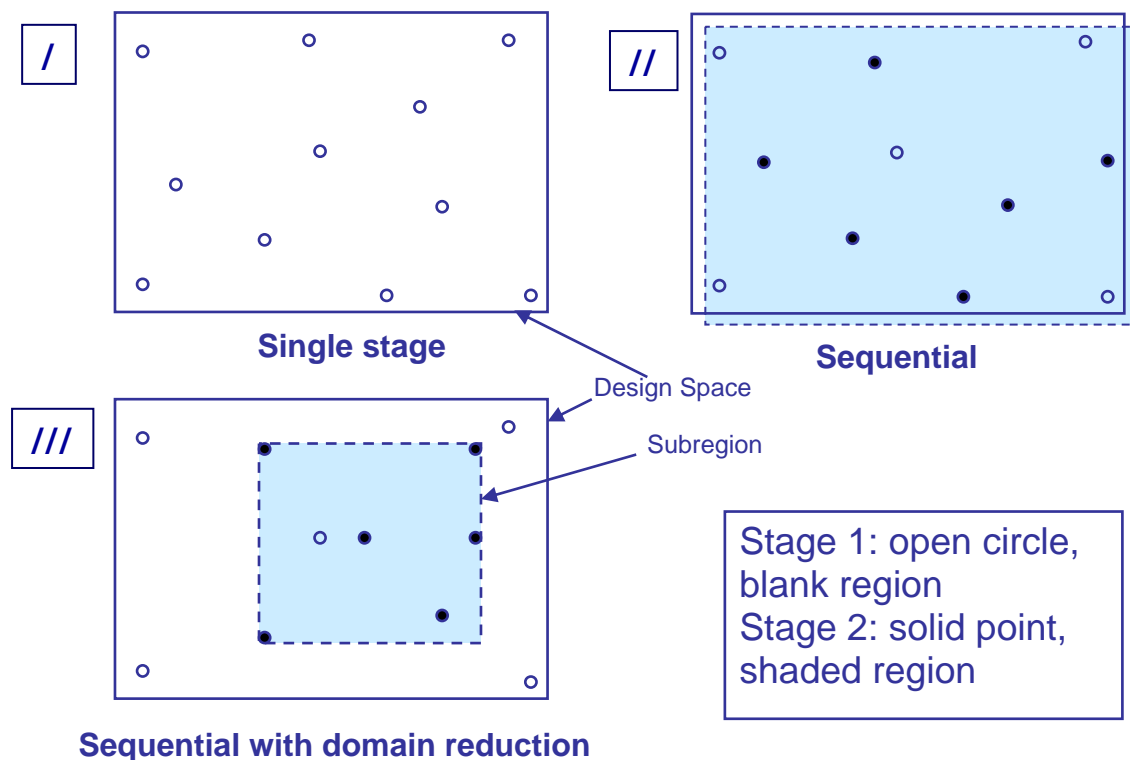

Single stage

Sequential

Sequential with domain reduction

Design Space

Subregion

Stage 1: open circle, blank region
Stage 2: solid point, shaded region

Fig. 1: Optimization strategies available in LS-OPT

2.2   Direct Optimization

Two main options are available namely the Genetic Algorithm (GA) and Particle Swarm Optimization (PSO). The latter will be available in Version 4.2.

2.3   Multi-Objective Optimization (MOO) (Direct or Surrogate-based)

2.3.1 Algorithms

Most practical engineering problems involve multiple design objectives and constraints. The optimization of systems with more than one objective function is called multi-objective optimization. These objectives are often in conflict. Contrary to the single-objective optimization problem (SOP), the multi-objective optimization problem (MOP) does not result in a single optimum solution. Instead, it results in a set of optimum solutions that represent different trade-offs among the objectives. These solutions are known as Pareto optimal solutions or constitute the Pareto optimal solution set. The function space representation of the Pareto optimal solution set is known as the Pareto optimal front (POF). LS-OPT provides a range of solvers to compute the Pareto Optimal Front. These are:

- NSGA-II (Non-dominated sorting Genetic Algorithm)
- SPEA-II (Strength Pareto Evolutionary Algorithm)
- SMPSO (Speed-Constrained Multi-objective Particle Swarm) (v4.2)

2.3.2 Convergence

A common problem with multi-objective optimization is the detection of convergence of the algorithm. In contrast to single objective optimization problems which have a single solution, convergence of MOO problems cannot be judged by monitoring a single design or design response. Instead, convergence to a complete and accurate POF is much more complex as both optimality (the closeness of each point in the non-dominated set to the POF) and diversity (the completeness or resolution of the non-dominated set) play an important role. This means that convergence to the POF has to be judged using more than one metric. Several convergence metrics are available in Version 4.2, the most important of which are [3][4]:

1. *Dominated hypervolume [3]*. This approach computes the hypervolume bounded by the POF on the one side and the Nadir point on the other. The Nadir is defined as the vector of maximal objectives which is updated from iteration to iteration.
2. *Standard deviation of the crowding distance [3]*. The crowding distance is defined as half the perimeter of the largest hypercube around a point that does not encompass any other solution.
3. *Spread of the POF [3]*. The spread of the front is calculated as the diagonal of the largest hypercube in the function space that encompasses all points. A large spread is desired to find diverse trade-off solutions.
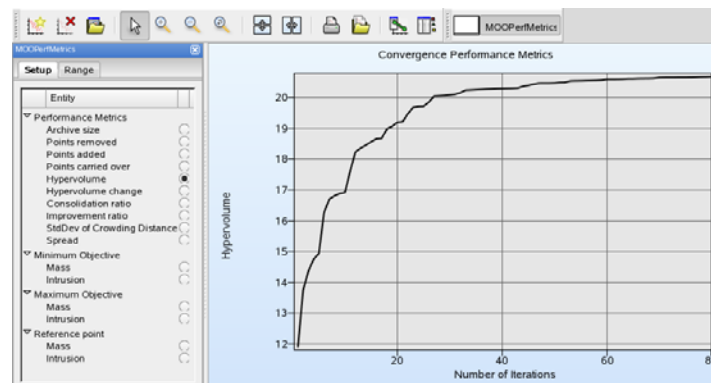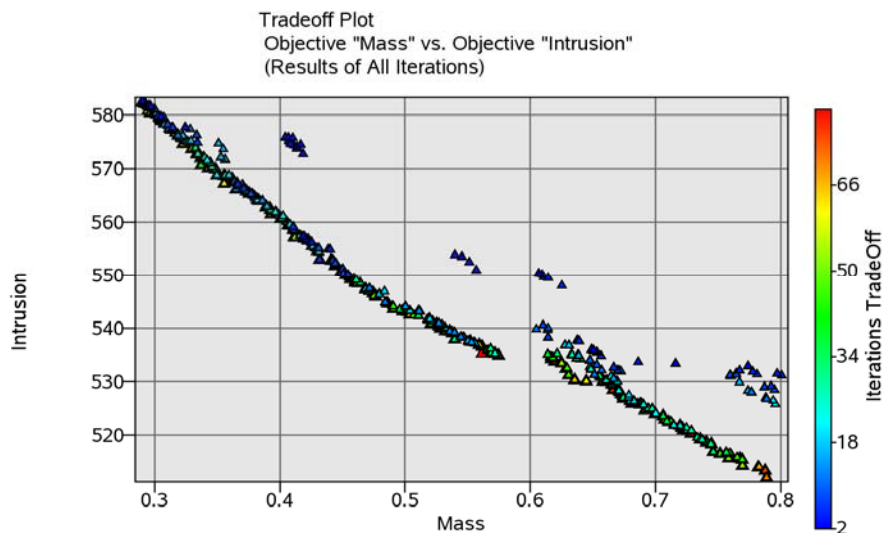

Fig.2: History of the Dominated Hypervolume convergence metric of an Intrusion vs. Mass non-dominated point set. The upper diagram shows the evolution of the non-dominated set as a function of generation.

Other histories such as archive size, points dominated, points survived, consolidation ratio, and improvement ratio [4] can also be plotted in Version 4.2. As an example, the dominated hypervolume convergence is displayed in Fig. 2 above.

2.3.3 Integrated Pareto display

Because of the difficulty of viewing the Pareto front for more than 2 or 3 objectives, LS-OPT provides an integrated display using four display types. These are the scatter plot (useful up to 3 dimensions), parallel coordinate plot, Hyper-radial visualization and Self-Organizing Maps. The various plots are connected so that when highlighting a particular design point or group in one display, the same set will be highlighted across the other plot types. An example is shown in Fig. 3 below.

Scatter plot Parallel Coordinate
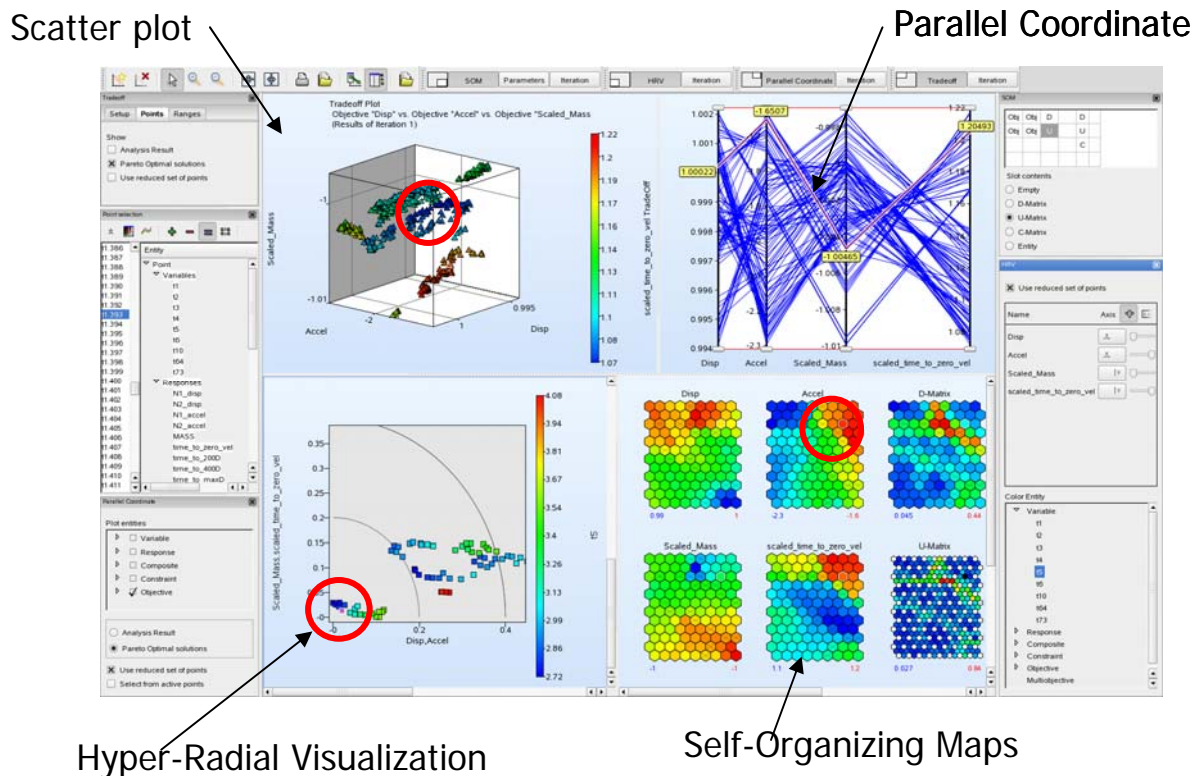
Hyper-Radial Visualization Self-Organizing Maps

Fig. 3: Integrated Pareto front representing 4 objective functions displayed in LS-OPT. A selected design is circled in red and highlighted in the Parallel Coordinate plot.

3   Reliability-based Optimization

LS-OPT includes the full complement of features for Reliability and Robustness analysis. This includes the definition of noise variables using a wide variety of statistical distributions (Normal, Uniform, Lognormal, Weibull, Beta, Binomial, Truncated normal and user options for both the probability density function and cumulative distribution function). These distributions can also be assigned to design variables for optimization. Integration of these features into the surrogate-based optimization allows for reliability-based optimization in which the probability of failure can be incorporated into design constraints. The standard deviation of the response can be used to maximize robustness using the so called Robust Design Optimization.

4   LS-DYNA integration

LS-OPT is strongly integrated with LS-DYNA. Some of the specific interfaces are the following:

- Checking of LS-DYNA keyword files (*DATABASE_) to ensure that the desired responses are available in the database.
- Automatic importation of design parameters from LS-DYNA keyword files (*PARAMETER_).

- Monitoring of solver progress, e.g. from a Windows client machine for LS-DYNA running on a cluster.
- Result extraction of most LS-DYNA response types
- Displaying LS-DYNA history plots in Viewer
- D3plot database compression (node and part selection)
- Displaying outlier information on FE mesh as a fringe plot (LS-PrePost display)
- LS-DYNA *CASE is supported. Responses can be tied to a particular LS-DYNA Case.
- *INCLUDE and *INCLUDE_PATH files are automatically parsed, copied and/or transmitted

5   System identification

A problem in system identification is the calibration of a system or material with respect to hysteretic curves. Version 4.2 includes a curve matching feature which allows matching of hysteretic curves. An example showing the convergence history of the residual as well as a display of the output curve vs. the test curve is shown in Fig. 4.
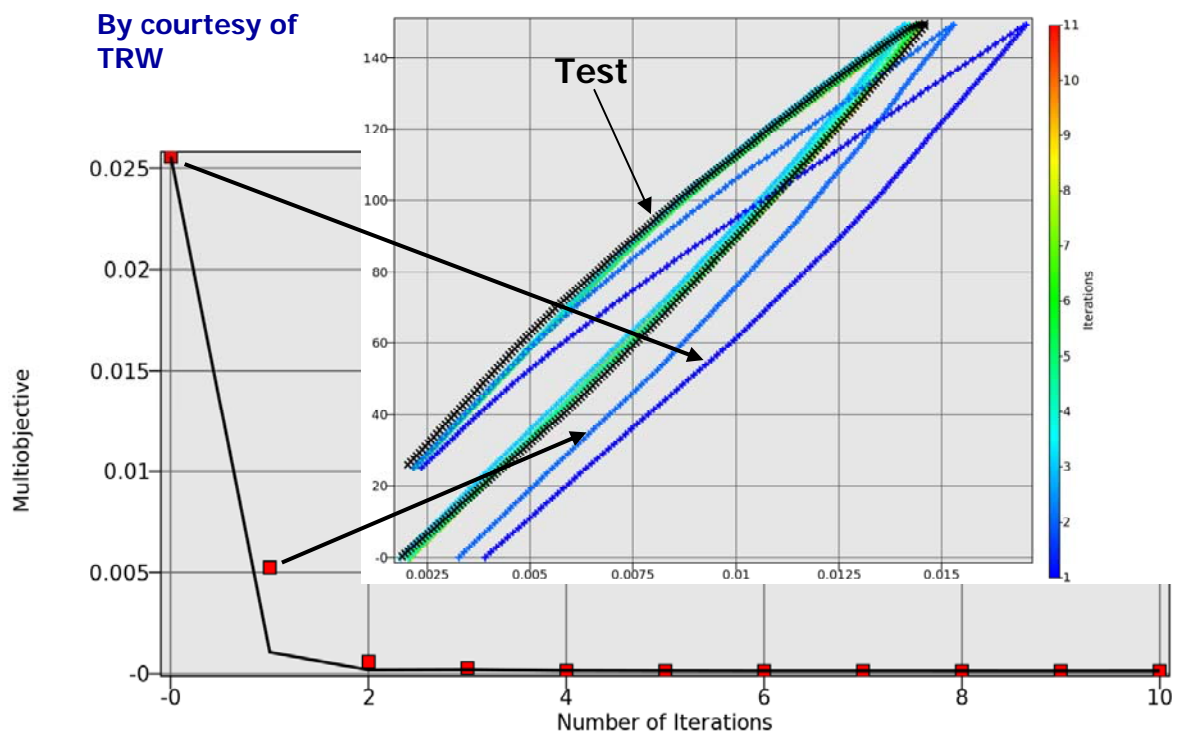

Fig.4: System identification problem: Optimization history of the residual as well as a display of the curve matching with respect to a test curve.

6   Solver job distribution

A typical work environment is a Windows or Linux client machine running LS-OPT with solver jobs running on a Linux cluster. Typical requirements are

1. The transparency of the job progress as well as error termination handling.
2. The ability to view the solver job logs.
3. The ability to transmit environment variables to the point where the solver is running. This allows system administrators to develop a standard variable-based script to execute different solvers, activate the licensing environment, etc. These environment variables can be defined in LS-OPT for the various solver cases and transmitted to the solver job execution.

LS-OPT allows several modes of job execution based on the following variables:

1. The availability or absence of a queuing system.
2. The desirability to allow remote solver jobs to initiate TCP/IP connections back to the client machine where LS-OPT is running. These connections can be used for job monitoring, input file transfer to the job location and result database file transfer from the job location to the client LS-OPT) machine.
3. The desirability of sharing a file system between the remote solver jobs and LS-OPT.

A relatively new product, *LSTCVM,* for job proxy support has been made available to address the above requirements. It consists of a separately available executable that is typically installed by a systems administrator on a computer cluster. Configuring the system on the client side is simple and can be done by any user desiring a connection to the cluster through the LSTCVM server. The LSTCVM server is compatible with late revisions of Version 4.1 (Rev. 62075). Using this server, the user can now conveniently interface a laptop running LS-OPT on Windows or Linux with a Linux-based cluster.

7   New features in Version 4.2

Apart from the several Version 4.2 features mentioned in the preceding paragraphs, e.g. convergence criteria for MOO, accuracy-based stopping criteria, new optimization algorithms (PSO), etc. some other rather important features are also part of the current development.

7.1   Constrained Space Filling

Although this feature has existed in past versions as the "Move" feature in the constraints panel, optimization has now been introduced to drive this feature with the purpose of improving robustness and accuracy. A two-dimensional example is shown in Fig. 5. for both continuous and discrete-continuous examples. This feature is particularly useful in cases where design variables are inter-constrained, such as in shape optimization problems.
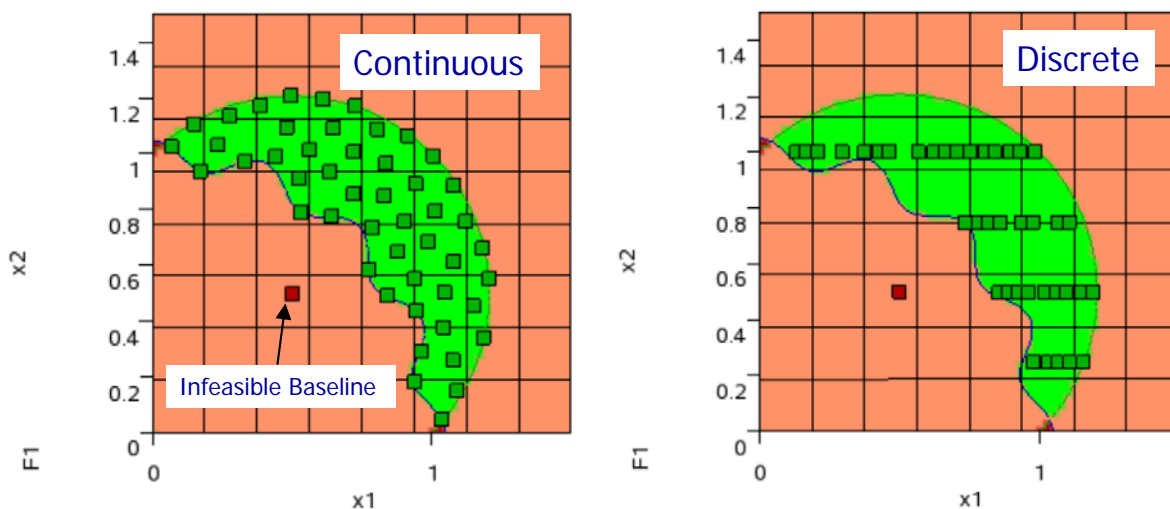

Fig. 5: A constrained space filling point selection demonstrating continuous and discrete-continuous sets in two dimensions.

7.2   Process modeling

A major new feature, process modeling, will be introduced over the next two versions 4.2 and 5.0. This feature will provide the ability to run an entire process such as encountered when combining manufacturing with design in the optimization process. While Version 4.2 will introduce most of the functionality within the current GUI framework, Version 5.0 will feature a completely new Graphical User Interface to fully exploit the process modeling features.

8   Summary

LS-OPT has become a mature product suitable for industrial application in design and parameter identification. A comprehensive set of optimization features allows most types of optimization including

multiple objectives and the inclusion of uncertainty in the design. A comprehensive post-processor is available to display and export pictorial results. A strong focus in LS-OPT development has been the facilitation of solver job distribution. Hence, recently added features have contributed to a greatly simplified facility for distributing and monitoring jobs in just about any type of configuration possible. Yet, several major features, including a new GUI, are under way for future versions to be released over the next year or two.

9 Literature

[1]   Stander *et al*. LS-OPT Users Manual, Version 4.1, Livermore Software Technology Corporation, August, 2010

[2]   Hallquist, J. LS-DYNA Users Manual, Version 971, Livermore Software Technology Corporation, May 2007.

[3]   Stander, N., Goel, T. An assessment of geometry-based convergence metrics for multi-objective evolutionary algorithms, *Proceedings of the 13th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, Paper AIAA-2010-9232,* 13-15 September, 2010, Fort Worth, Texas, USA.

[4]   Goel, T. Stander, N. A study of convergence characteristics of multi-objective evolutionary algorithms, *Proceedings of the 13th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, Paper AIAA-2010-9233,* 13-15 September, 2010, Fort Worth, Texas, USA.