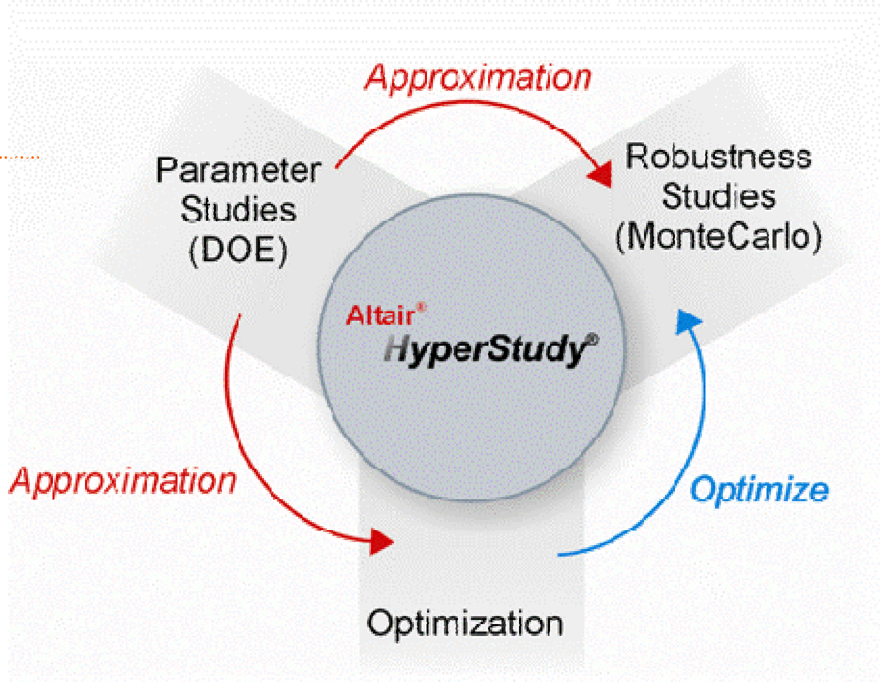
HyperStudy®



Altair Engineering

September 4th, 2009

Agenda

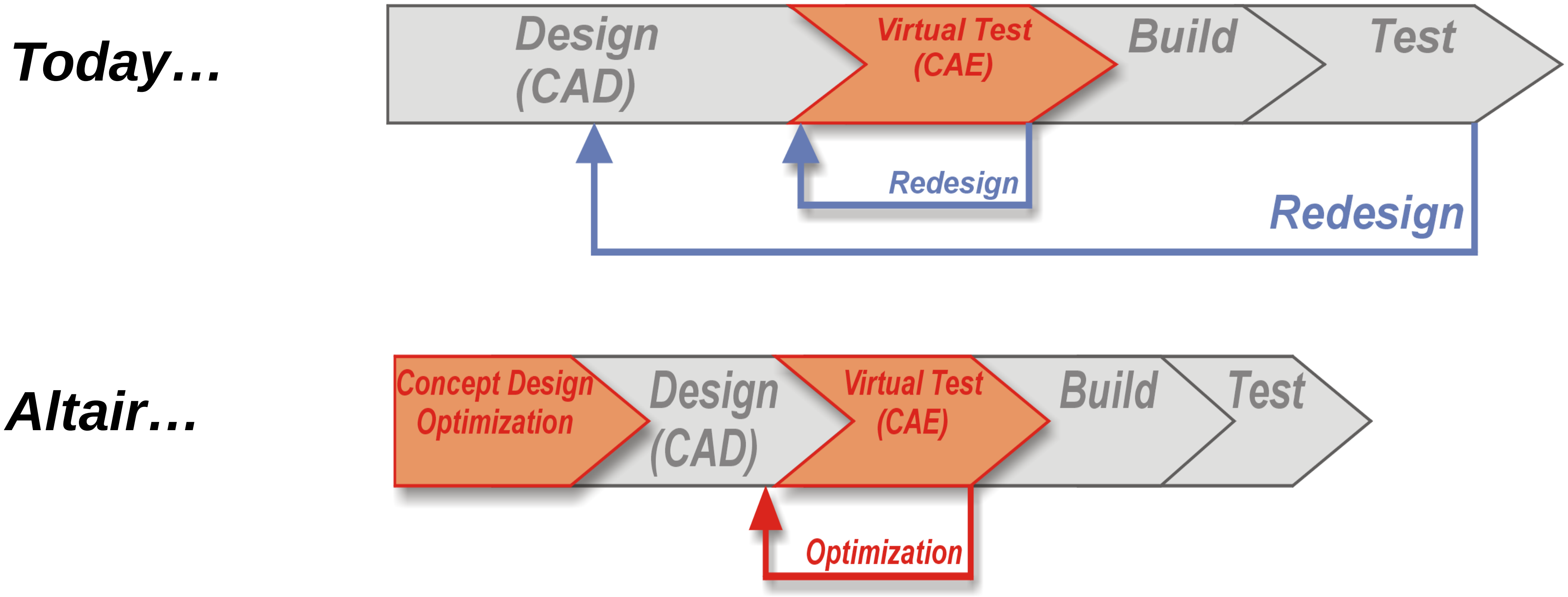
Morning Session (8:30 AM - 12:30 PM)

1. *Introduction to HyperStudy*
2. *Exercises 1.1 and 1.2*
3. *BREAK (10 Minutes)*
4. *Design of Experiments (DOE)*
5. *Exercises 2.1, 2.2 and 2.3*
6. *Approximations*
7. *Exercises 3.1 and 3.2*

Afternoon Session (1:30 - 4:30 PM)

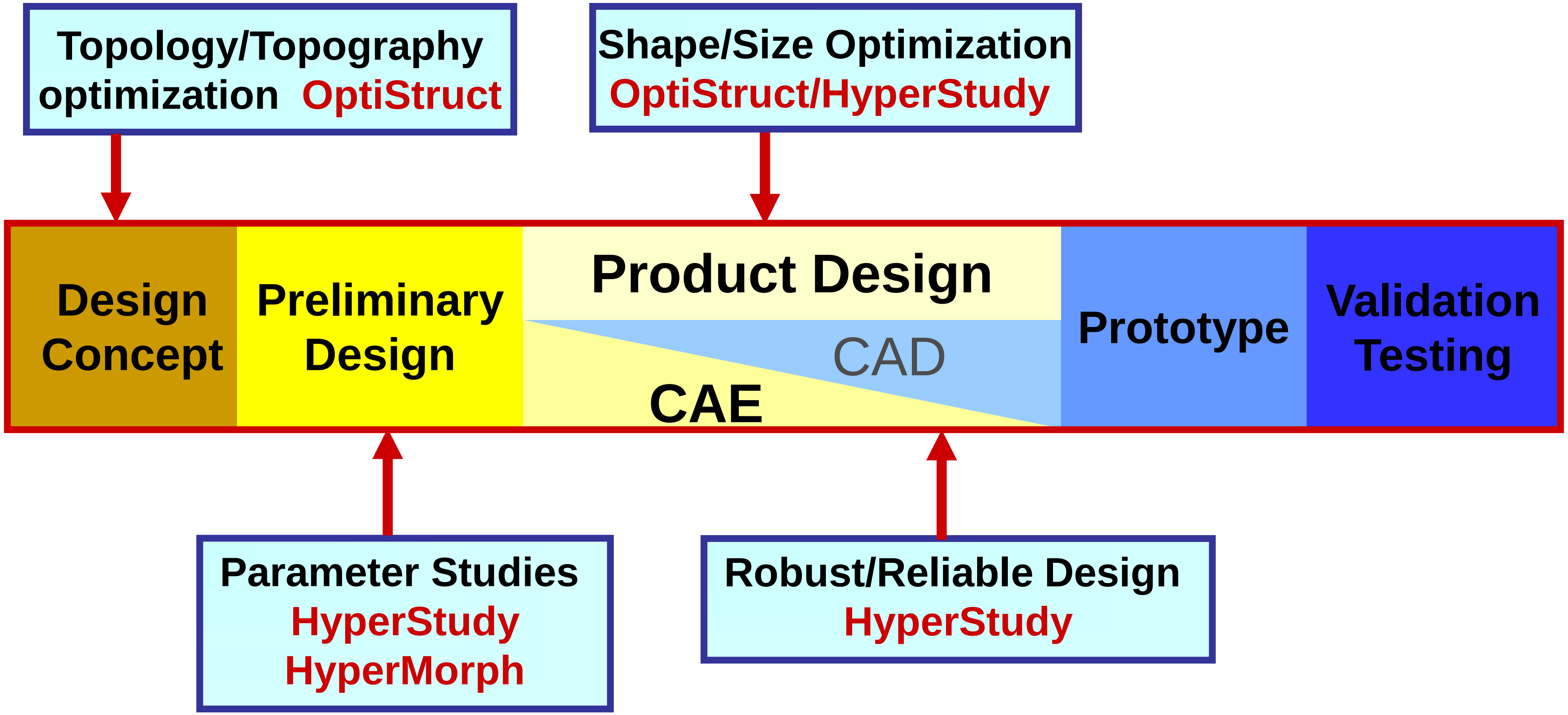
8. *Optimization*
9. *Exercises 4.1, 4.2 and (one of these 4.3, 4.4 or 4.5)*
10. *BREAK (10 Minutes)*
11. *Stochastic*
12. *Exercises 5.1 and 5.2*

Altair's Vision



Optimization is Driver in CAE Driven Design Process

Optimization with Altair HyperWorks



Chapter 1: Introduction to HyperStudy

Altair HyperStudy

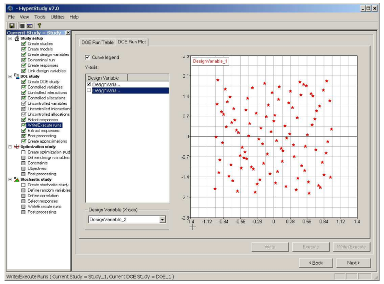
- is a software to perform Design of Experiments (DOE), approximations, optimization, and stochastic studies.
- is applicable to study the different aspects of a design under various conditions, including non-linear behaviors.
- can be applied in the multi-disciplinary optimization of a design combining different analysis types.
- is integrated with HyperWorks through direct links to the models in HyperMesh (.hm), HyperForm (.hf), and MotionView (.mdl).
- models can be easily parameterized. Aside from the typical definition of solver input data as design variables, the shape of a finite element model can also be parameterized with ease through HyperMorph.

Study Definition

	DOE	Approximation	Optimization	Stochastic
Parameters Screening	👍			
System Performance Study	👍			
Response Surface Evaluation		👍		
Optimum Design			👍	
Variation Analysis				👍
Robust Design	👍	👍	👍	👍
Reliability Design	👍	👍	👍	👍

DOE Studies

Design Space Sampling → **Run DOE** → **Evaluate results**

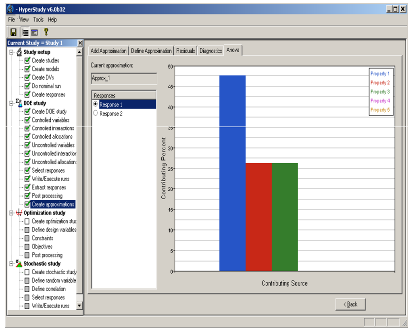


- Factorial designs
- Plackett-Burman
- Box-Behnken
- Central-Composite
- Latin HyperCube
- Hammersley
- User defined
- External Matrix

Run DOE

- Invoke solvers
- Batch submission
- Parallel job submission

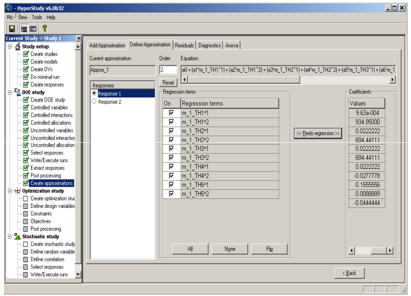
Evaluate results



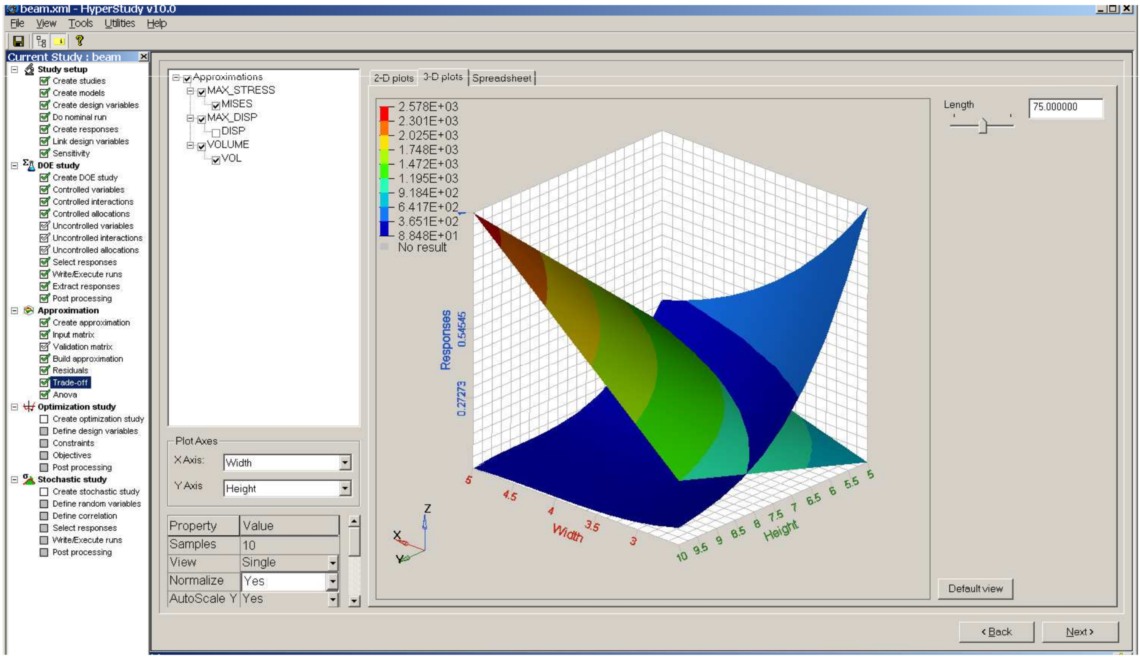
- Effects plots
- ANOVA

Approximations

Responses Approximation

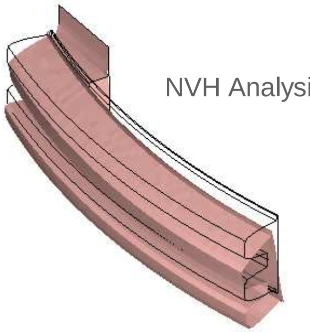


- Least square
- Moving least square
- HyperKriging

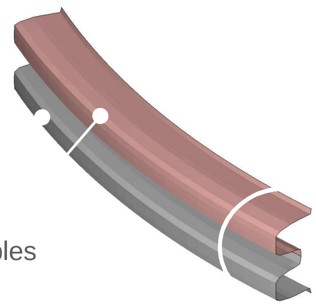


Optimization Studies

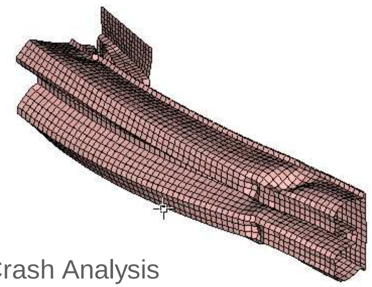
Multi-disciplinary Optimization → **Define Design Variables** → **Run Optimization** → **Evaluate results**



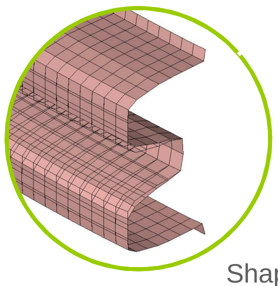
NVH Analysis



Size variables

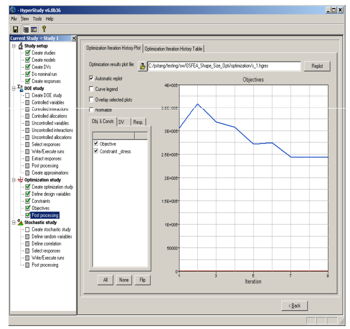


Crash Analysis



Shape variables

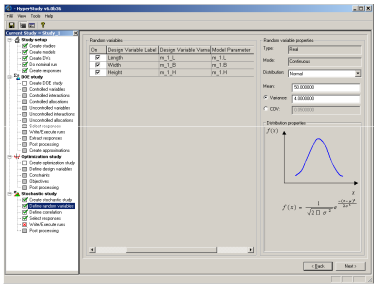
- Invoke solver directly
- From approximation
- Genetic Algorithm
- Sequential Quadratic Programming
- Adaptive Response Surface
- Method of Feasible Directions
- External optimizers



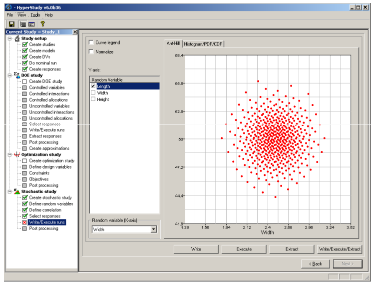
- Objective function
- Design constraints
- Design variables
- Responses

Stochastic Studies

Define input distributions → **Select sampling method** → **Run stochastics** → **Evaluate results**

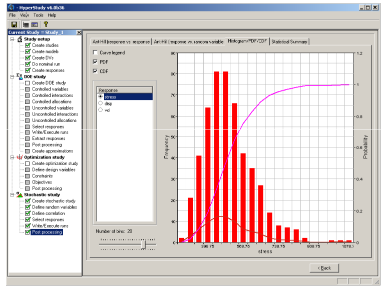


- Normal
- Uniform
- Triangular
- Exponential
- Weibull



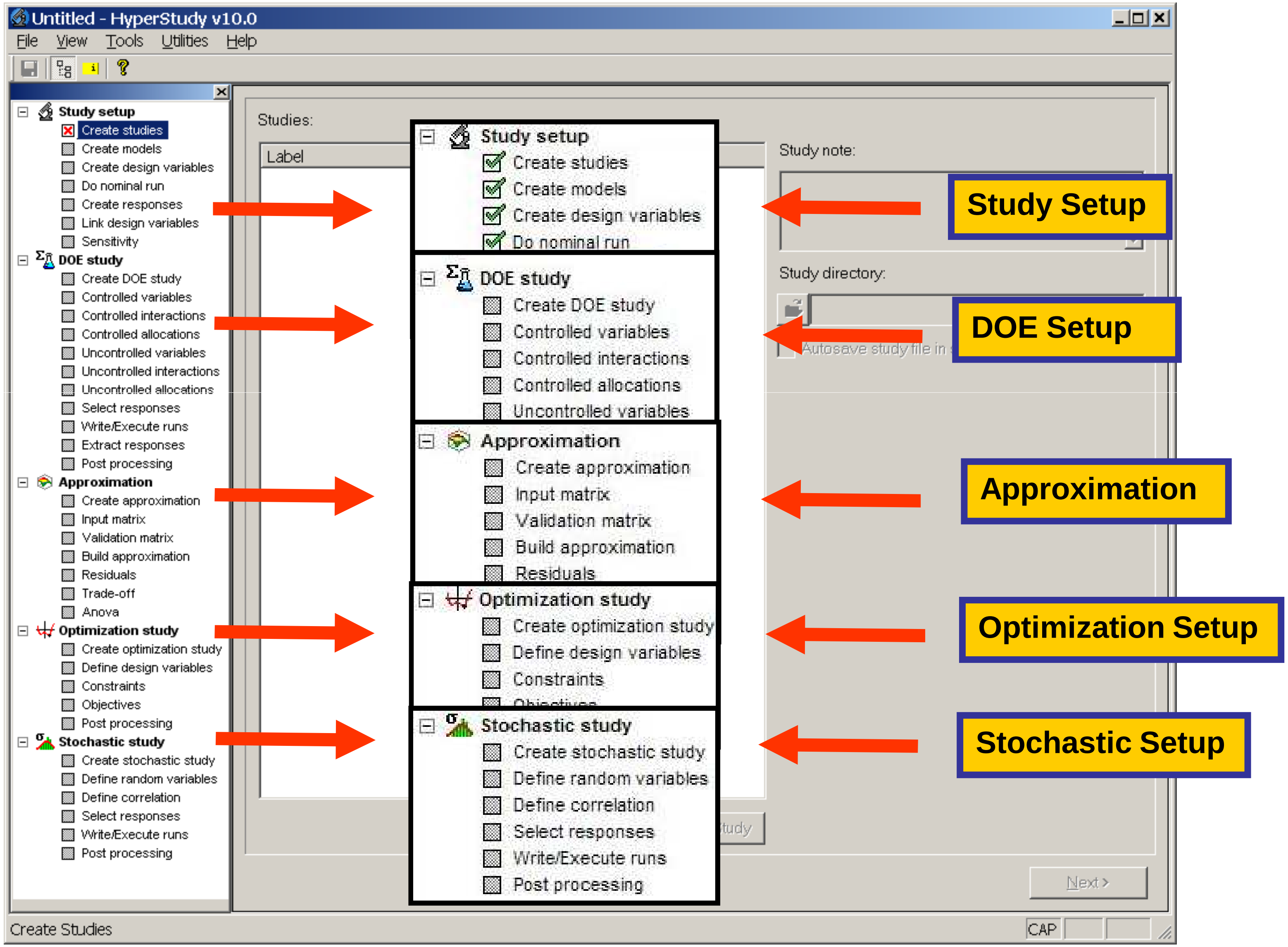
- Simple Random
- Latin HyperCube
- Hammersley

- Call solver directly
- Using response surface

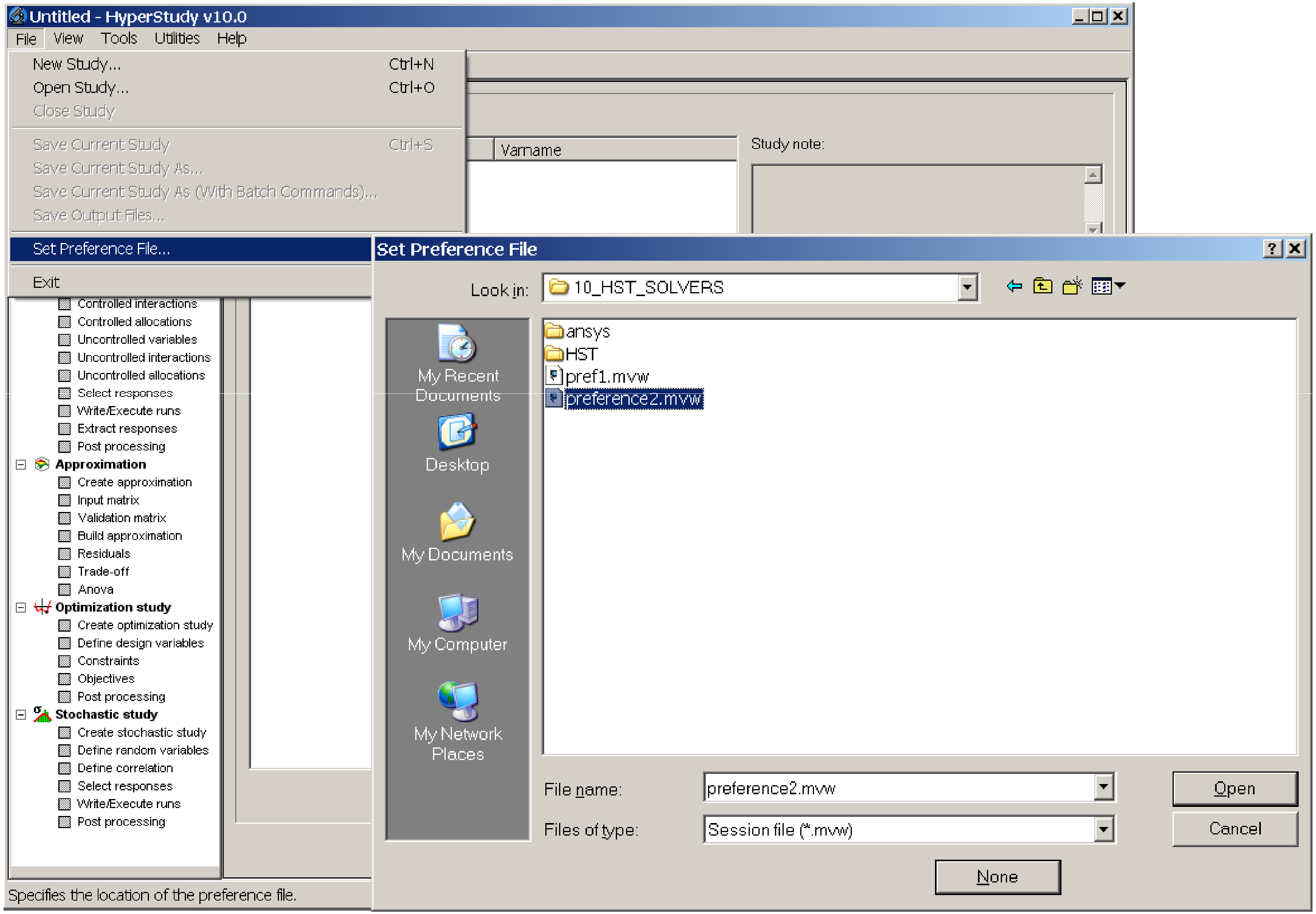


- Probability Distribution Function (PDF)
- Cumulative Distribution Function (CDF)
- Histogram Distribution
- Ant-Hill plots
- Statistical moments

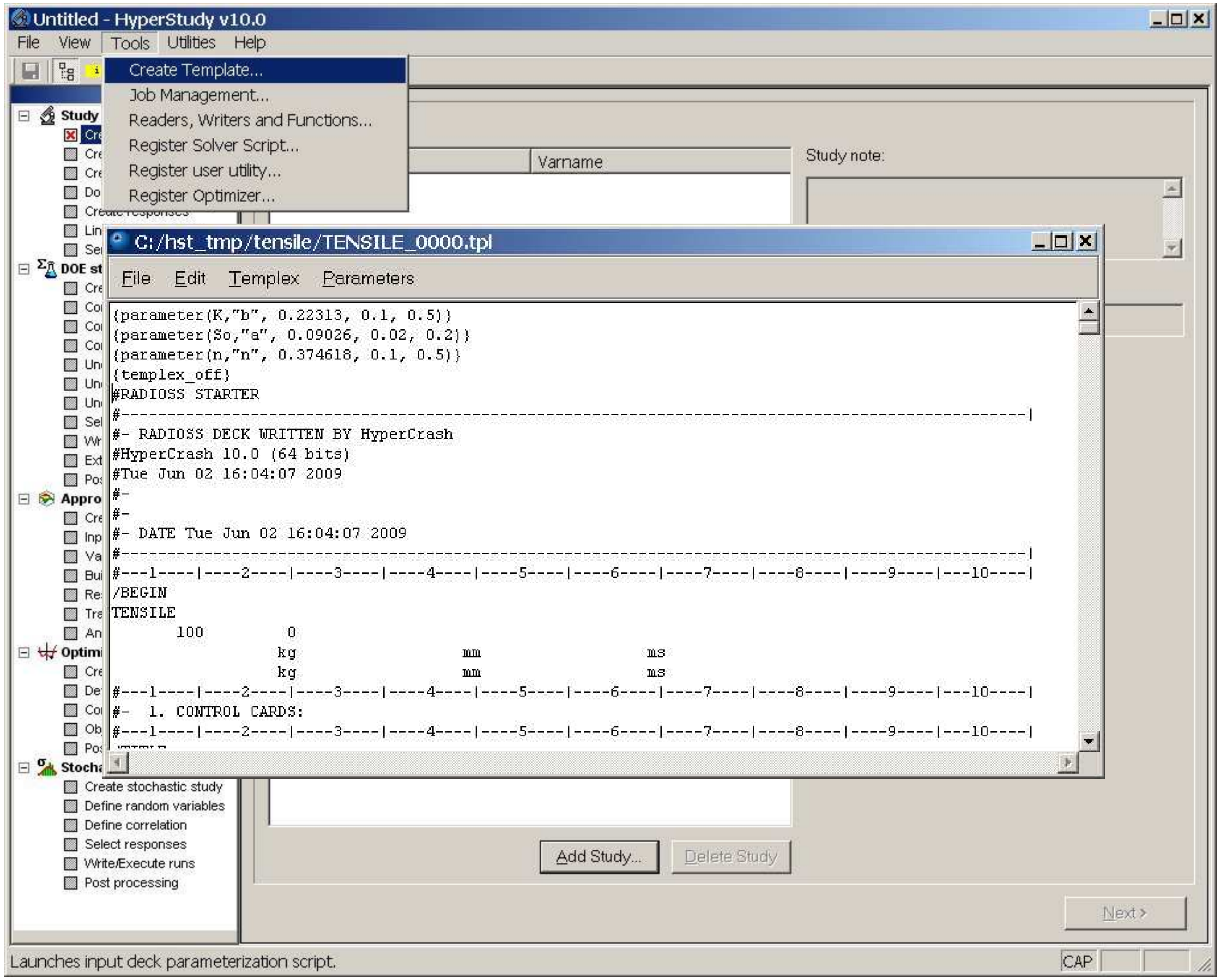
HyperStudy Process Flow



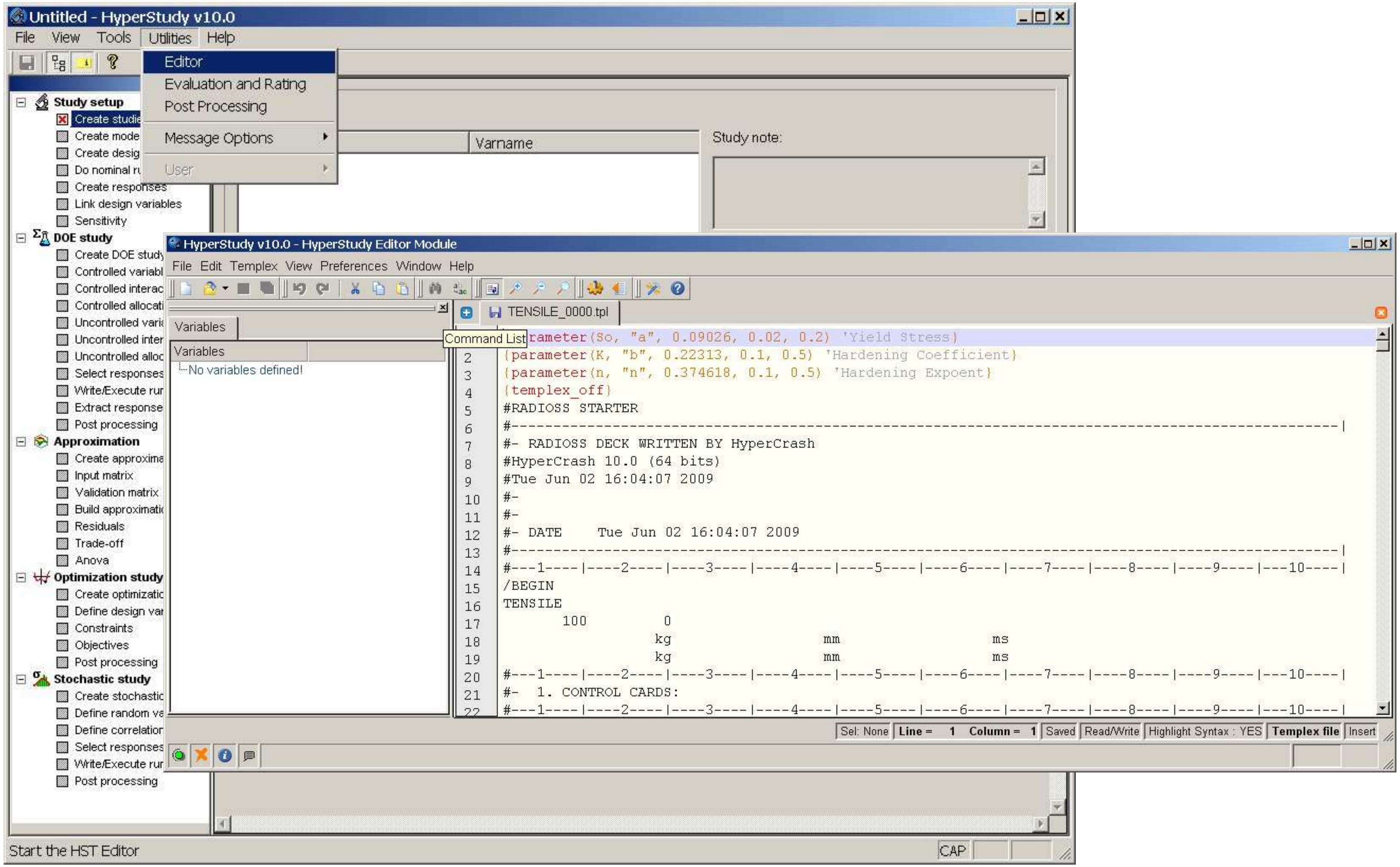
Preference file



Old Editor

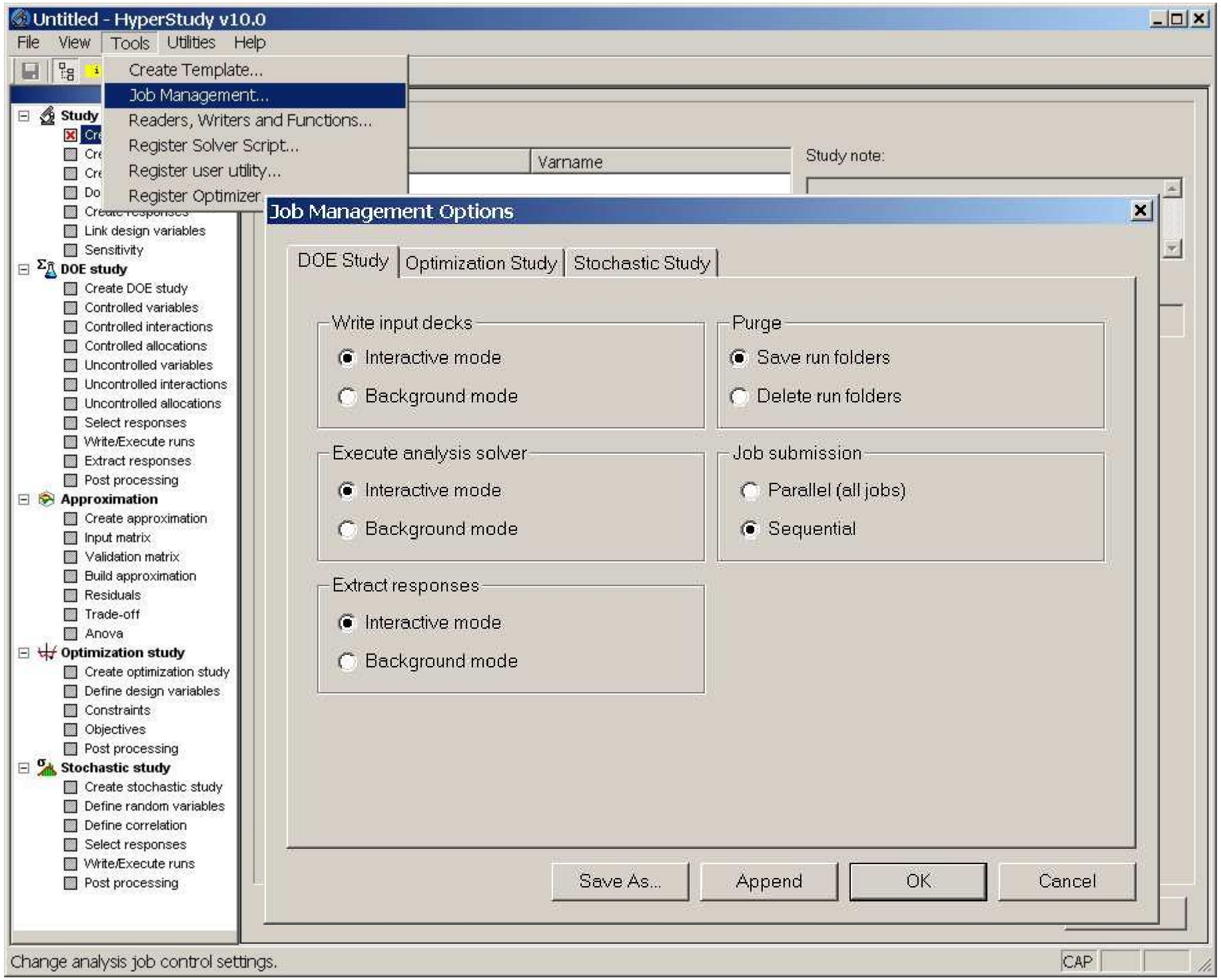

Untitled - HyperStudy v10.0
File View Tools Utilities Help
Create Template...
Job Management...
Readers, Writers and Functions...
Register Solver Script...
Register user utility...
Register Optimizer...
Varname
Study note:
C:/hst_tmp/tensile/TENSILE_0000.tpl
File Edit Templex Parameters
{parameter(K,"b", 0.22313, 0.1, 0.5)}
{parameter(So,"a", 0.09026, 0.02, 0.2)}
{parameter(n,"n", 0.374618, 0.1, 0.5)}
{templex_off}
#RADIOSS STARTER
#- RADIOSS DECK WRITTEN BY HyperCrash
#HyperCrash 10.0 (64 bits)
#Tue Jun 02 16:04:07 2009
#- DATE Tue Jun 02 16:04:07 2009
/BEGIN
TENSILE
100 0
kg mm ms
kg mm ms
#- 1. CONTROL CARDS:
Create stochastic study
Define random variables
Define correlation
Select responses
Write/Execute runs
Post processing
Add Study...
Delete Study
Next >
Launches input deck parameterization script.
CAP

New Editor

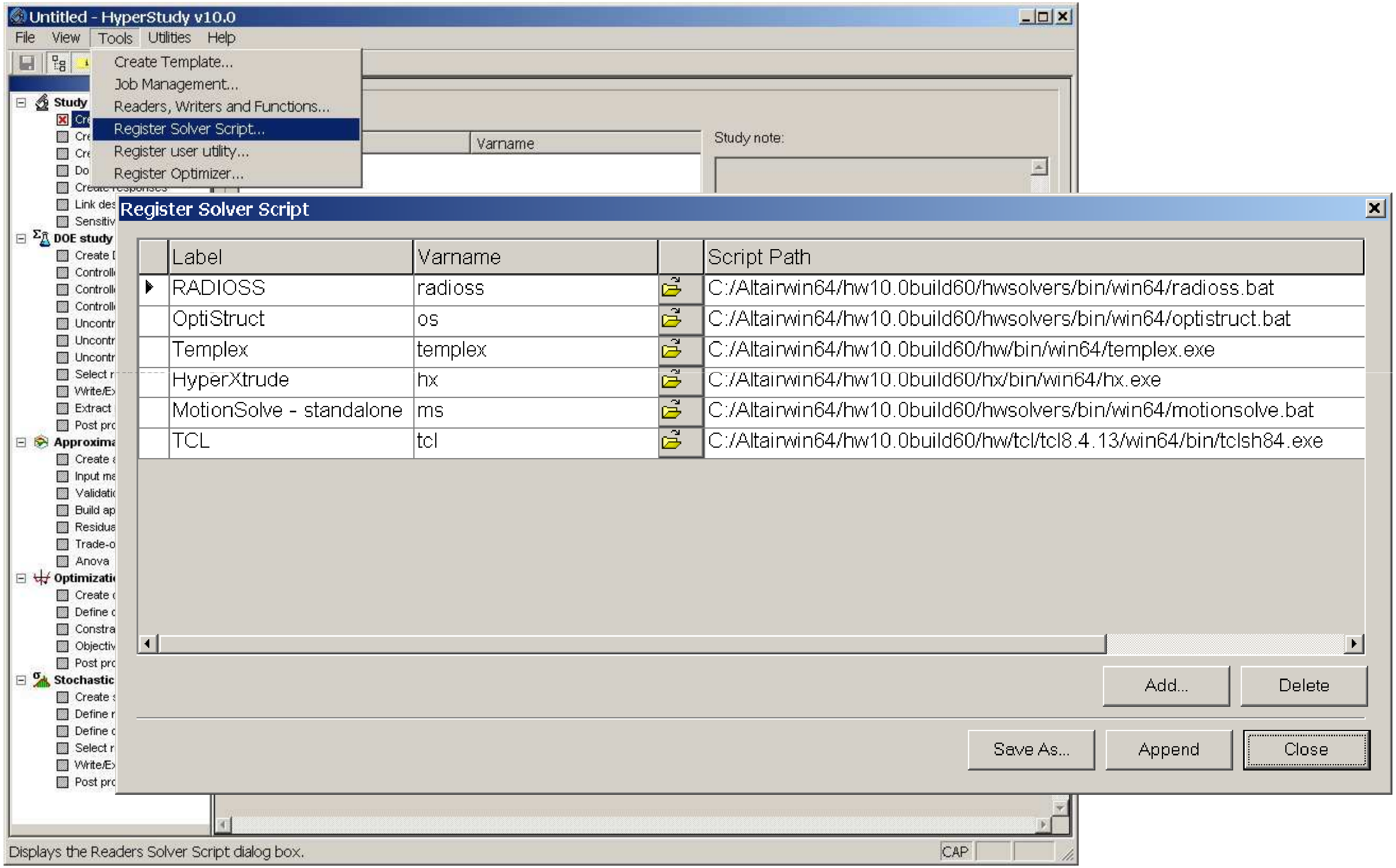


Untitled - HyperStudy v10.0
File View Tools Utilities Help
Editor
Evaluation and Rating
Post Processing
Message Options
User
Study setup
Varname
Study note:
DOE study
Approximation
Optimization study
Stochastic study
HyperStudy v10.0 - HyperStudy Editor Module
File Edit Templex View Preferences Window Help
TENSILE_0000.tpl
Variables
No variables defined!
Command List
{parameter(K, "b", 0.22313, 0.1, 0.5) 'Hardening Coefficient}
{parameter(n, "n", 0.374618, 0.1, 0.5) 'Hardening Expoent}
{templex_off}
#RADIOSS STARTER
#- RADIOSS DECK WRITTEN BY HyperCrash
#HyperCrash 10.0 (64 bits)
#Tue Jun 02 16:04:07 2009
/BEGIN
TENSILE
#- 1. CONTROL CARDS:
Start the HST Editor

Job Management

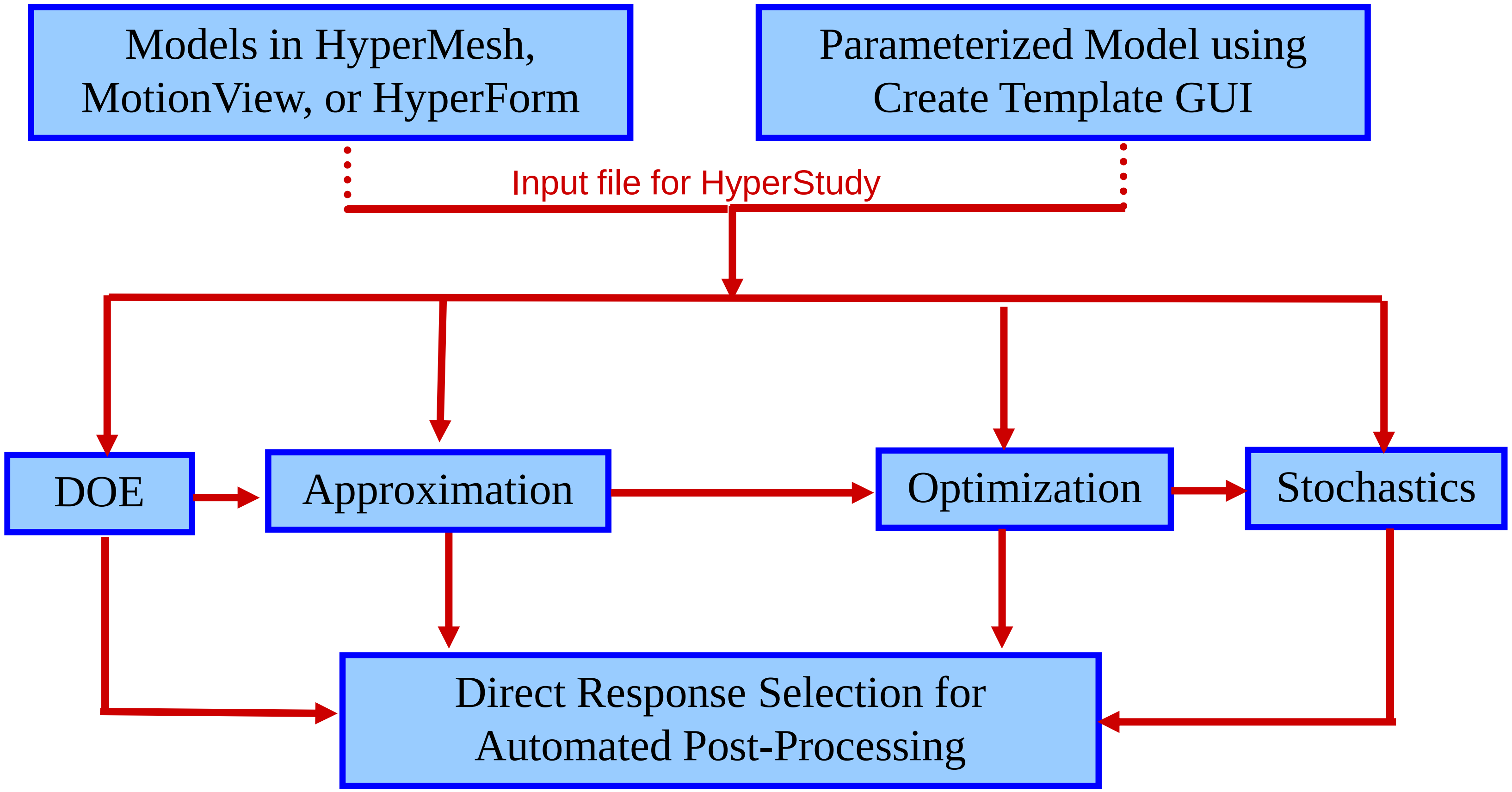


Solver Script



Parametrization of a FEA model

The process of selecting design variables / factors / random variables for Optimization / DOE / Stochastic studies can be summarized as:



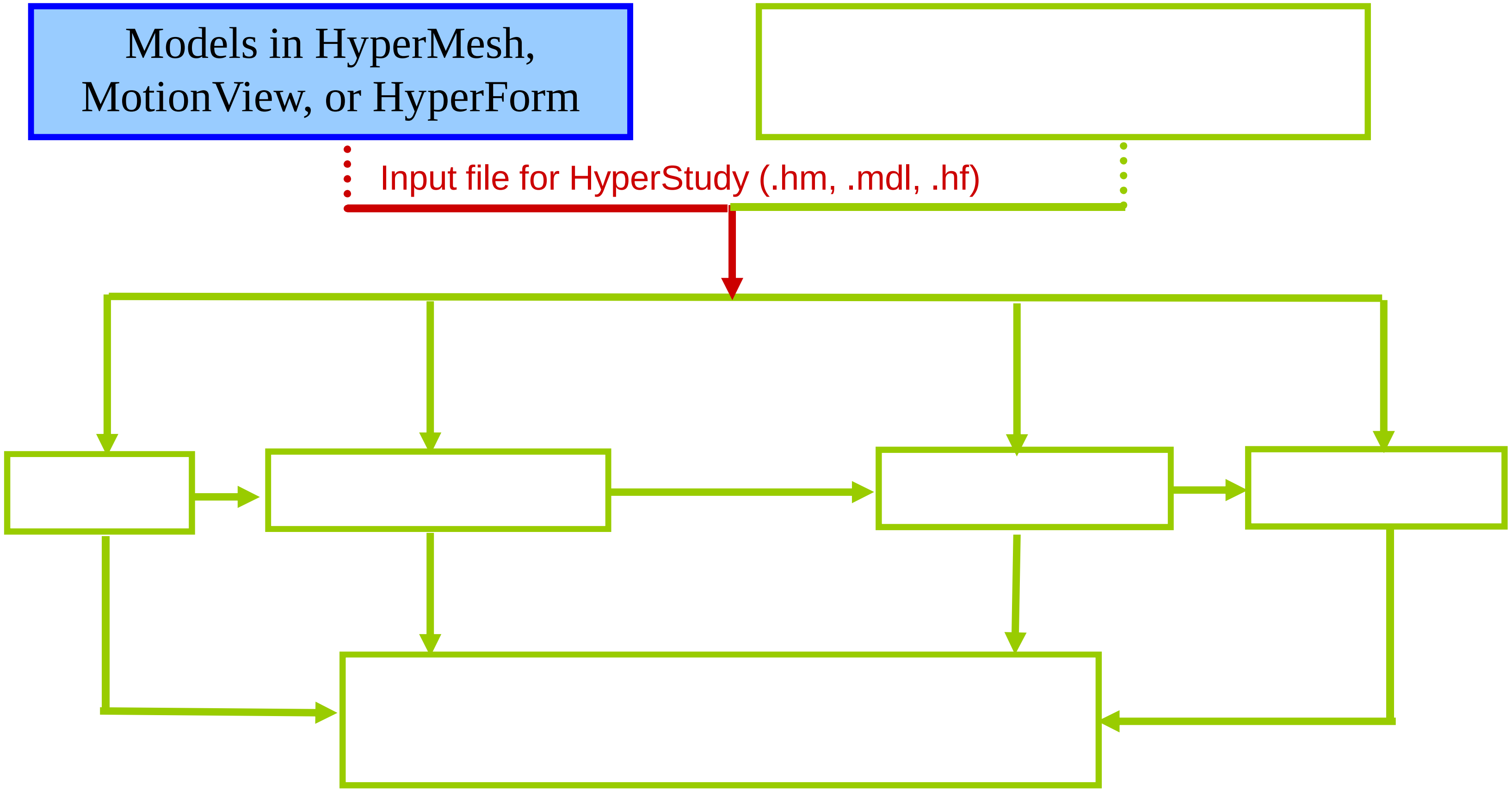
Parametrization

Direct linking to Hypermesh-MotionView-HyperForm provides HyperStudy direct access to simulation models and to the features such as thickness, concentrated masses, shape changes which are used as the design variables in DOE, optimization or stochastic studies .

When HyperStudy is launched in a standalone mode, the design variables need to be identified using a common method as HyperStudy interacts with solvers of different physics.

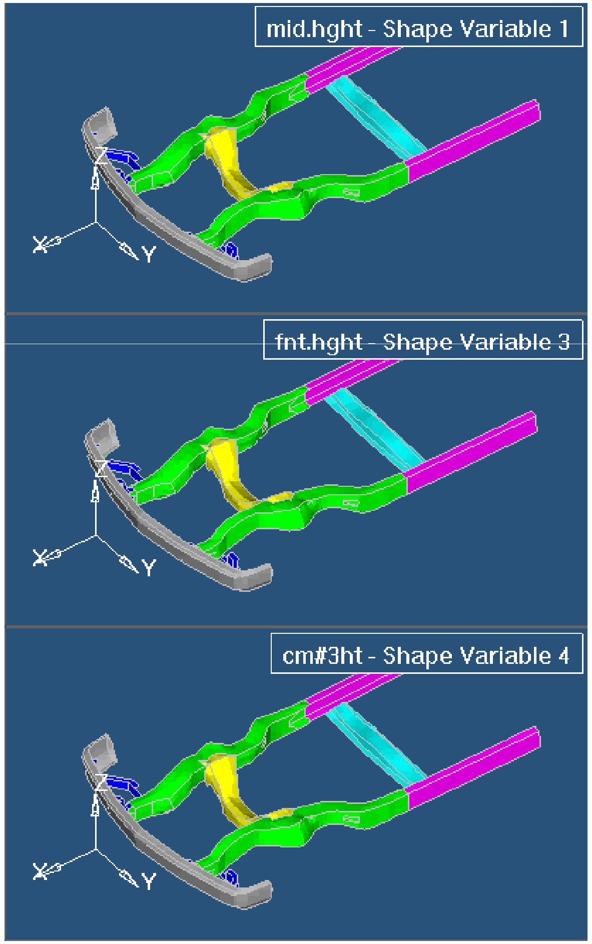
HyperStudy uses TEMPLEX to parameterize the solver input deck. The solver input deck is taken and "parameter" statements are added, which in turn is linked to the property such as thickness, concentrated masses. This file with "parameter" statements is called as "parameterized input deck". This deck forms the input file for HyperStudy.

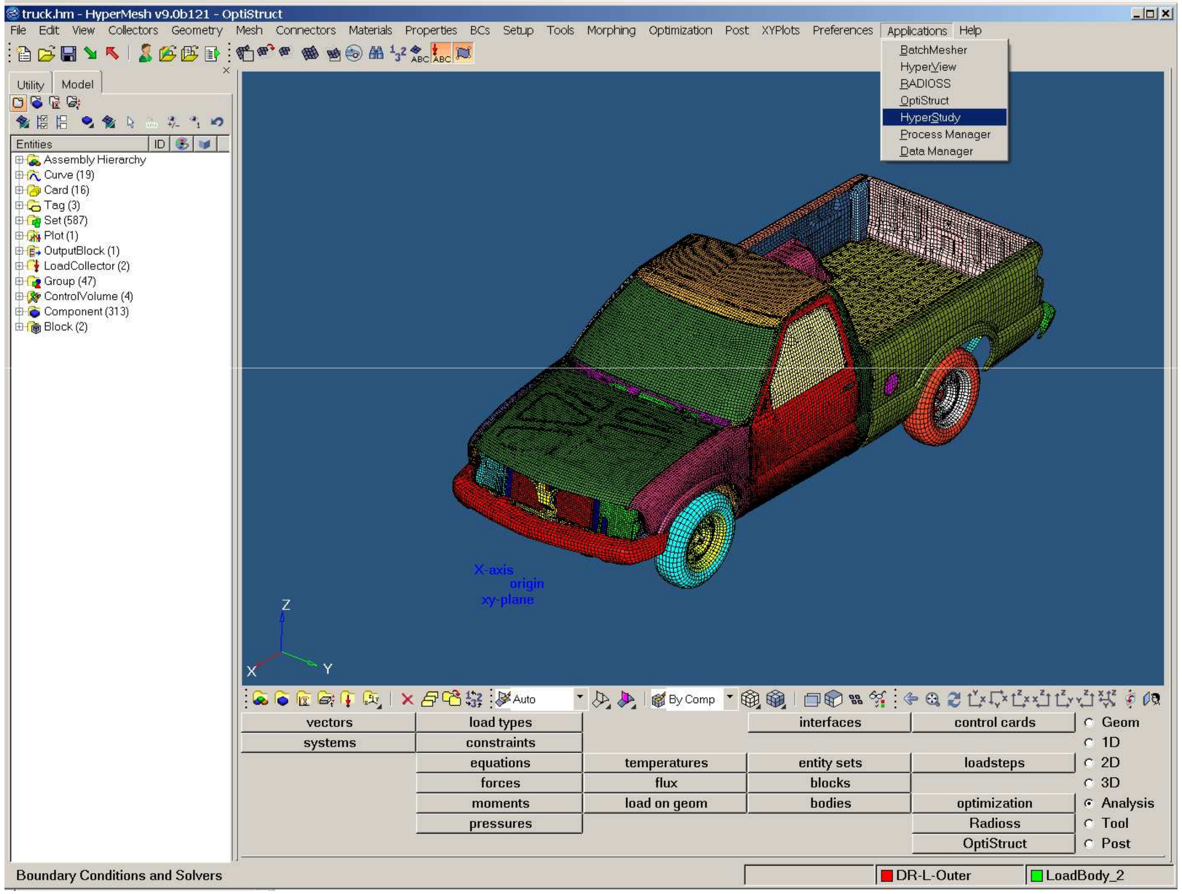
HyperStudy/HM-MV-HF link



HyperMesh-HyperStudy Link

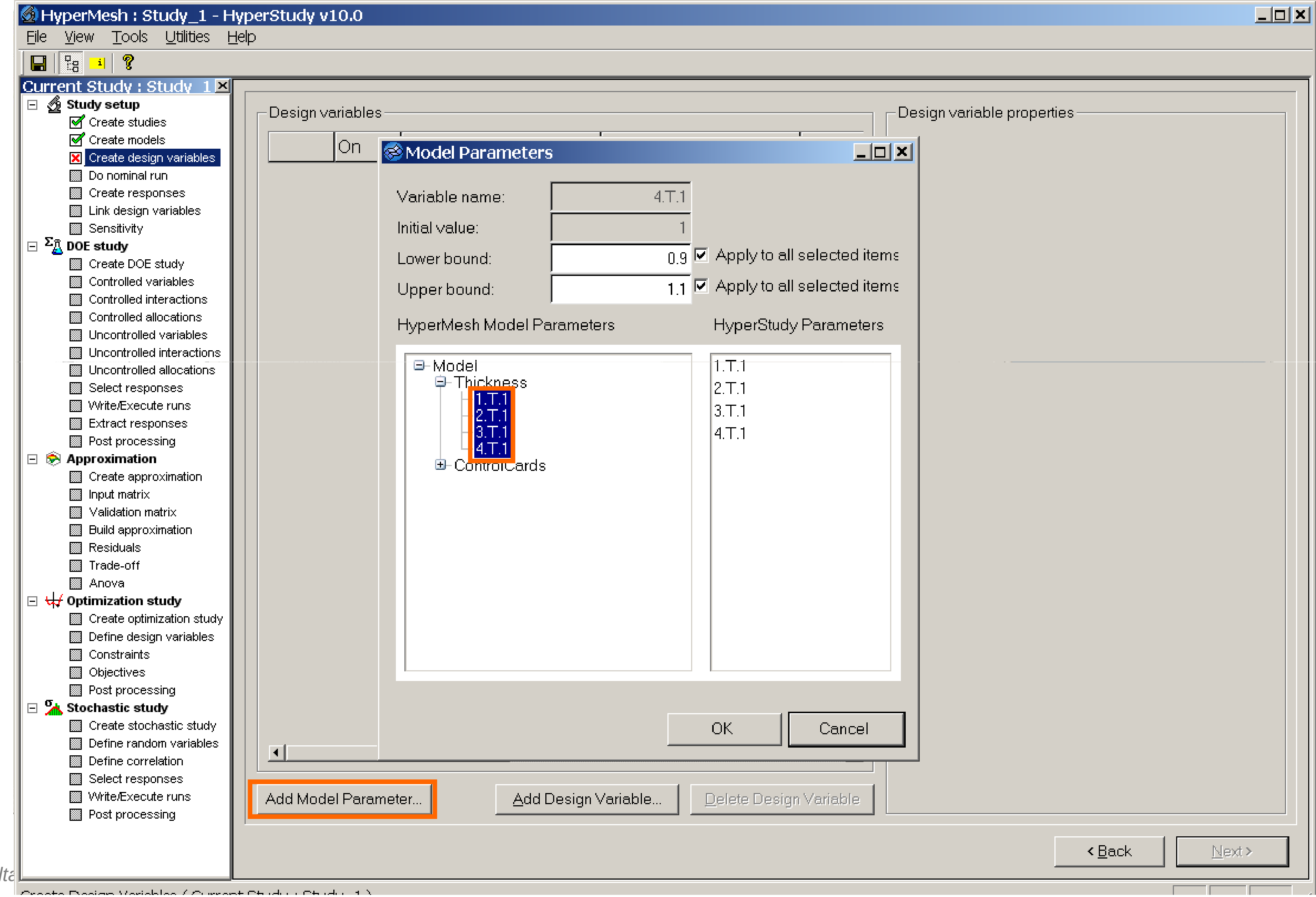
Shapes in HyperMesh





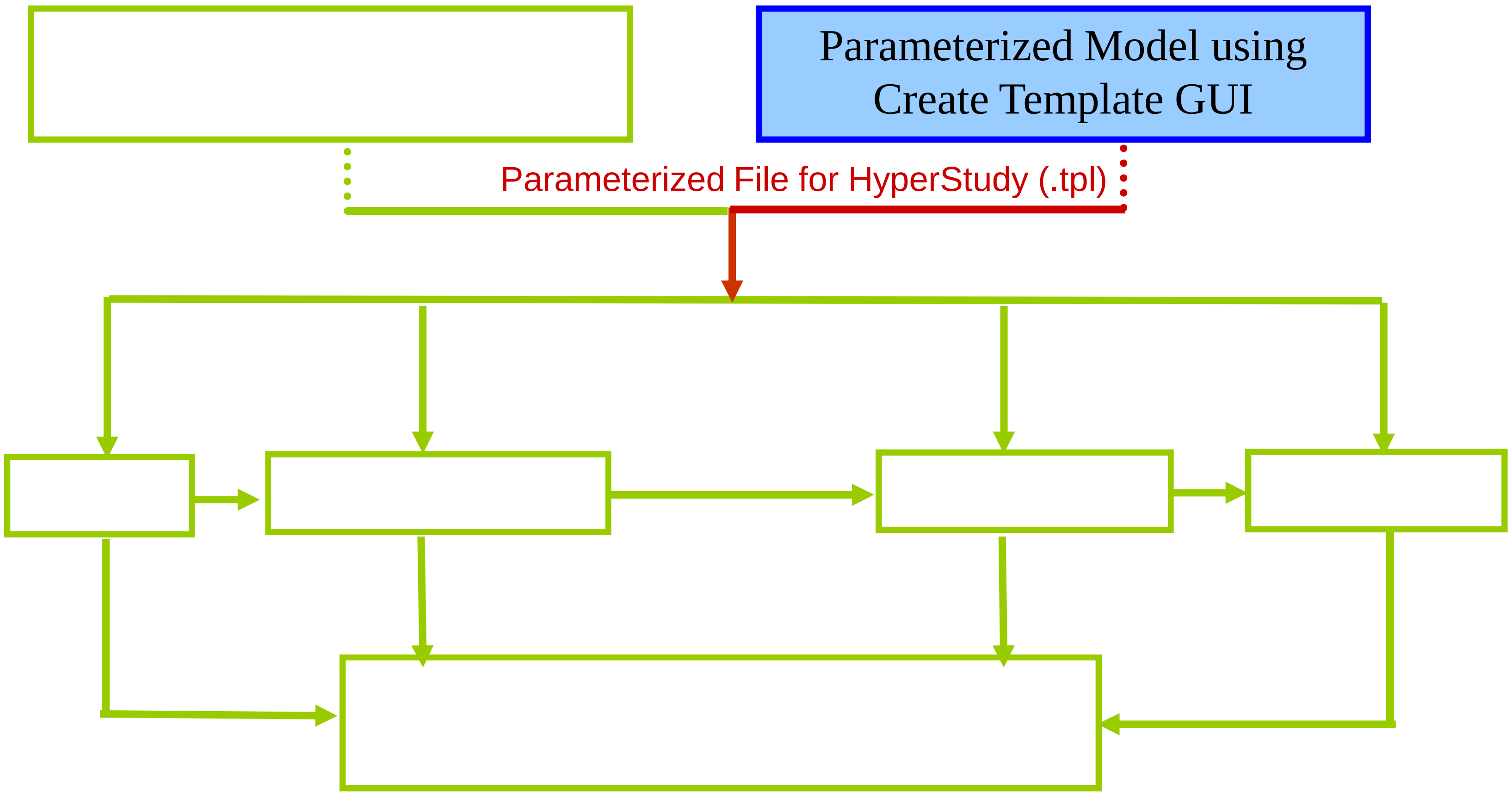
Model Parameterization

HyperMesh-HyperStudy Link



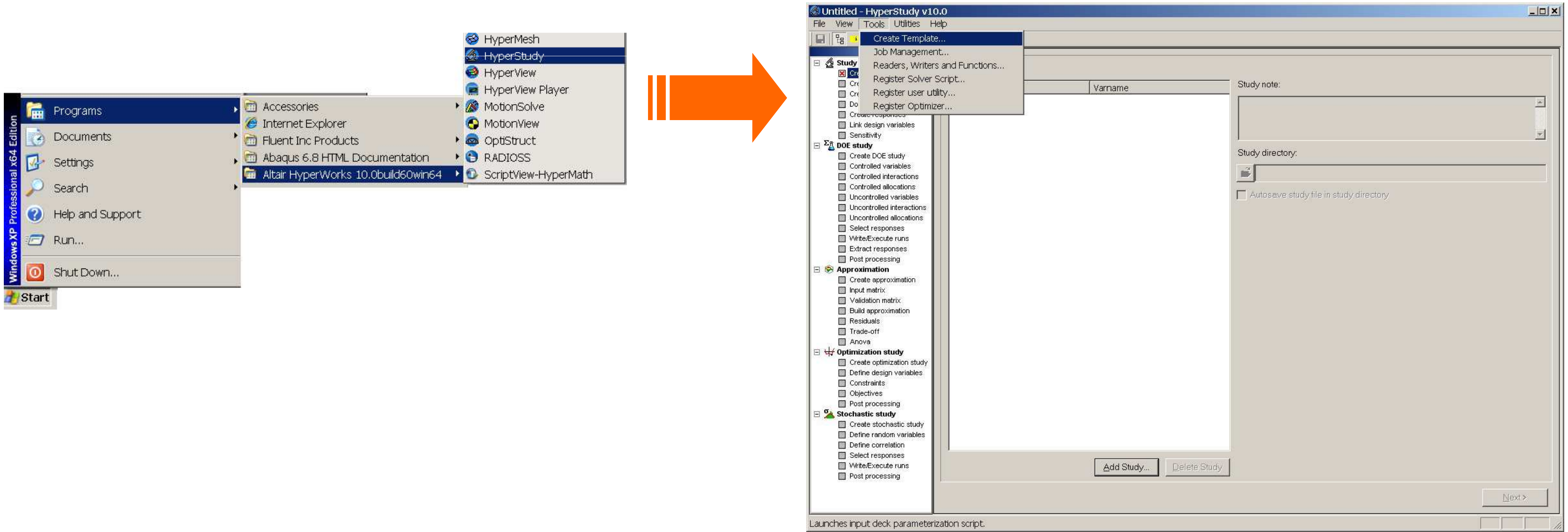
Exercise 1.1:
Create Design Variables Through HyperMesh

Parametrization Using Create Template GUI



Parametrization using Create Template GUI

A parameterized solver input deck is required when HyperStudy is launched in standalone mode.



What is Templex?

Templex is a general-purpose text and numeric processor. Templex generates output text based on guidelines defined in a template file.

The input files can be created using any text editor or word processor that can save a file in ASCII format.

Templex solver can be used to process the template file and the output can be sent to a monitor, to another file, HyperGraph or to other output streams.

Templex is built into HyperStudy as an expression builder for creating responses for Objectives and Constraints.

Templex can also be used as a standalone solver on mathematical expressions to perform Design of Experiments (DOE) / Optimization / Stochastic studies.

Important Templex Rules

- Statements, expressions, and variables must be placed between braces, **{ }**, so that ***Templex*** will evaluate them. Otherwise they are treated as plain text and not processed.
- After ***Templex*** processes a parameterized template file, the plain text is copied to the specified output stream verbatim.
- When ***Templex*** encounters an expression within braces, it evaluates the expression and sends the resulting value to the output stream. If the item between the braces is a ***Templex statement***, Templex carries out the statement instructions. If the Templex statement generates output, the resulting value is sent to the output stream too.
- Variable names can be up to **eighty characters long**. They consist only of letters, numbers and underscores; no other characters are allowed. Variable names must always **begin with a letter** and are **case sensitive**.
- In addition to scalars ***Templex*** supports **one- and two- dimensional arrays**. In ***HyperStudy*** to define the responses the results from the analysis runs are represented as **one-dimensional arrays "VECTORS"**. The index starts with 0 as in C language.
- The ***template file*** must have **at least one parameter statement** to be able to read into the HyperStudy.

Create Template GUI

The parameterized input deck is a **Templex template, *.tpl, file.**

To create the *.tpl file, the following are added to the analysis input file :

1. Templex parameter statements,
2. Field variable calls, and
3. External-file-include statements

Create Template GUI

1. Templex parameter statements are used for individual field changes and shape vector (DESVAR) nodal perturbations. All parameter statements are placed at the top of the *.tpl file.

The templex parameter function has the following format:

{parameter(varname, label, nominal, minimum, maximum)}

Example:

{parameter(TH1, ”Thickness 1”, 1.6, 0.5, 3.5)}

Create Template GUI

2. Field variable statements must be inserted into the analysis deck for each field to be optimized, such as component thickness, density, etc.

The formats for field variable statements is as follows:

{variable,%field_width.decimal_places}

'Variable' is the same as in the corresponding parameter statement.

The 'field width' value depends on the solver being used so that the proper spacing is maintained in the analysis deck. Each solver will have it's own filed width: LS-DYNA is 10, OptiStruct is 8, and HyperForm is 8. The 'decimal_places' value is a user preference.

Example

{TH1, %8.5f}

Example: LS_Dyna Input Deck

Original LS-DYNA Deck

```
$HMNAME    PROPS     1beam
        1         71.0                  20.0       0.0              0
1.6      1.6       1.6     1.6        0.0
```

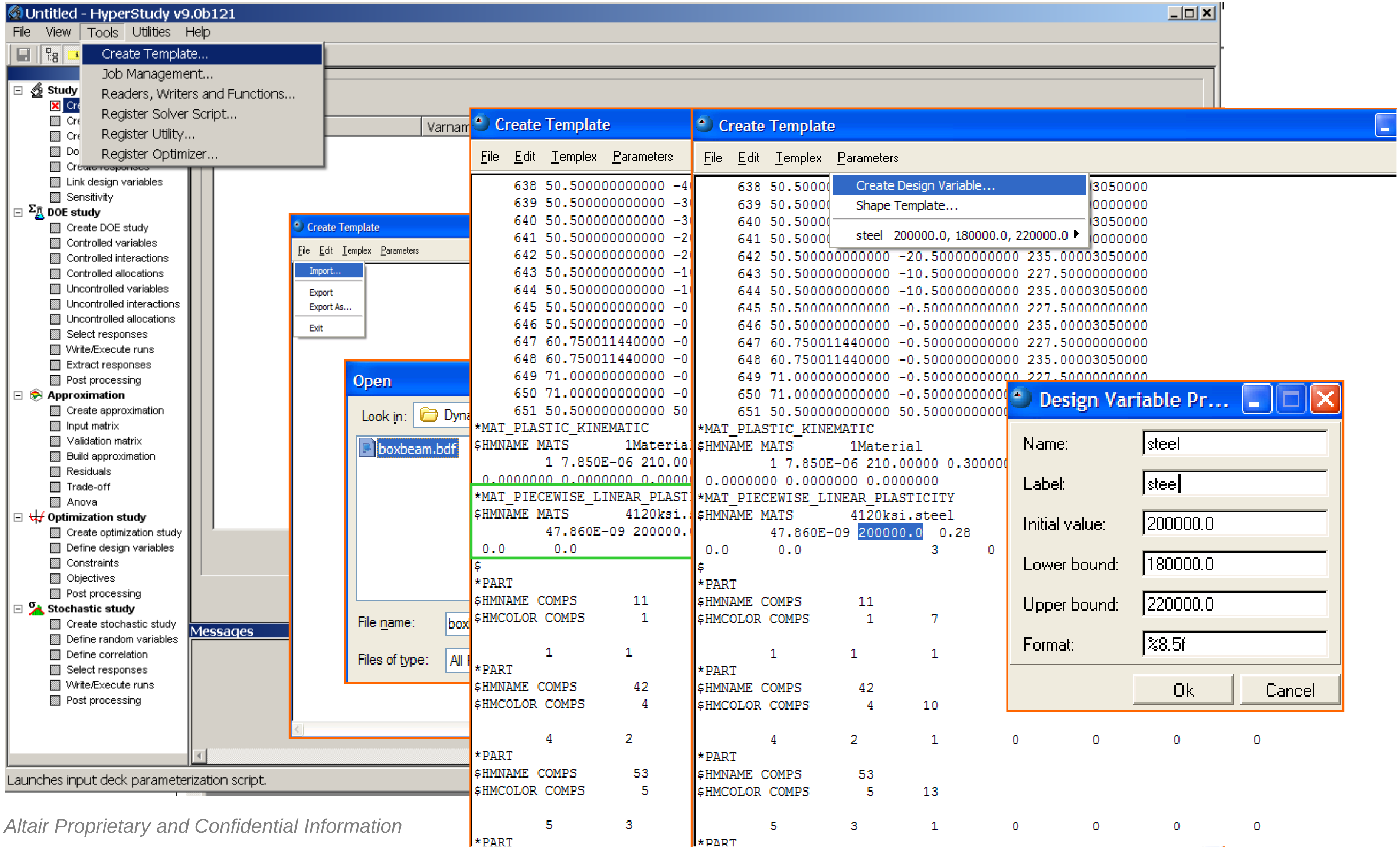
Parameterized LS-DYNA Deck (.tpl)

```
$HMNAME    PROPS     1beam
        1         71.0                  20.0       0.0              0
{TH1,%10.5f}{TH1,%10.5f}{TH1,%10.5f}{TH1,%10.5f}0.0
```

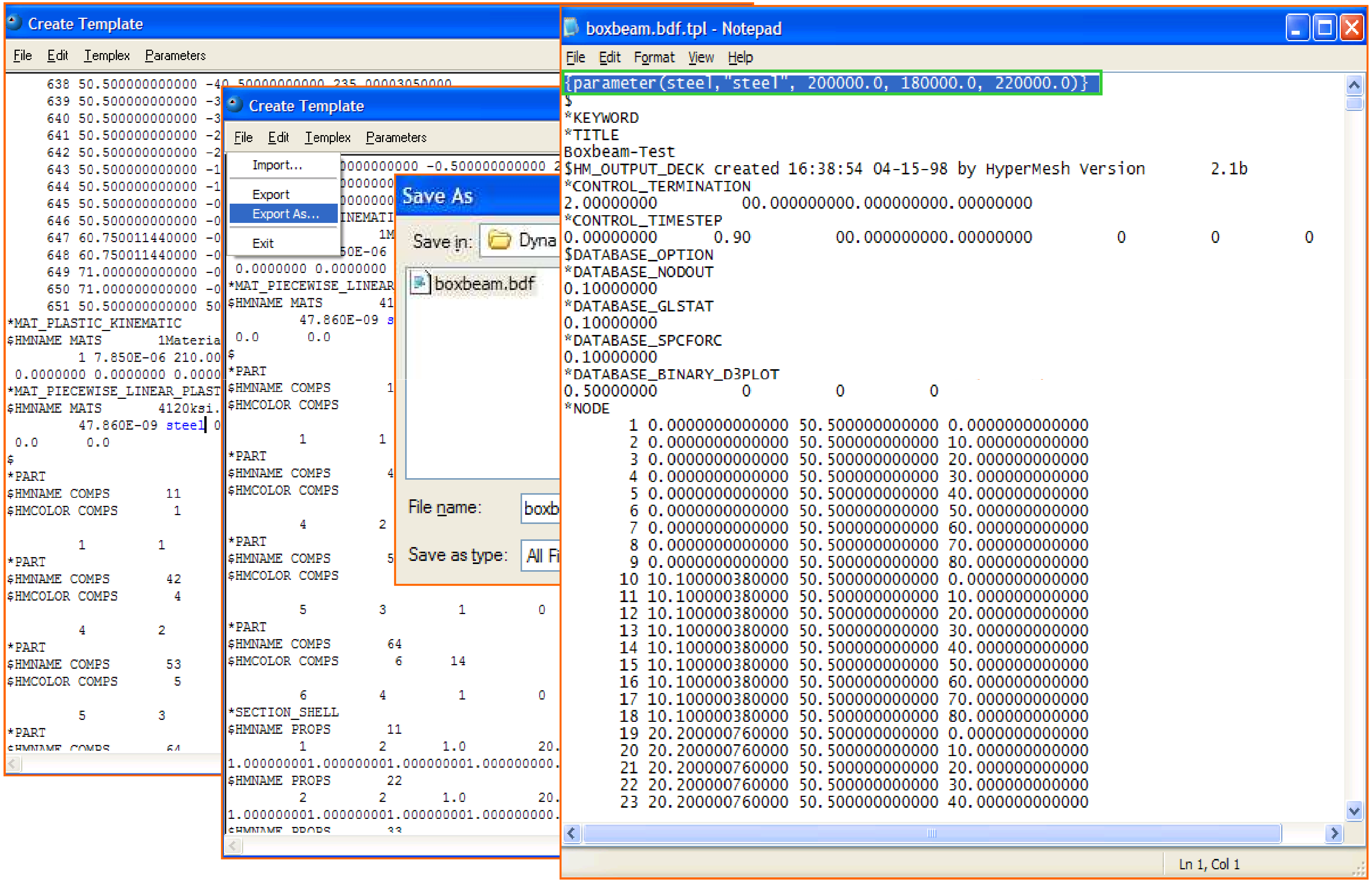
3. If the design variables in optimization are shape vectors, further editing of the deck is necessary such as external-file-include statements.

Example: Nastran Input Deck

HyperStudy offers a simple GUI to parameterize input decks.



Example: Nastran Input Deck



Exercise 1.2:
Create Design Variables using the Create Template GUI

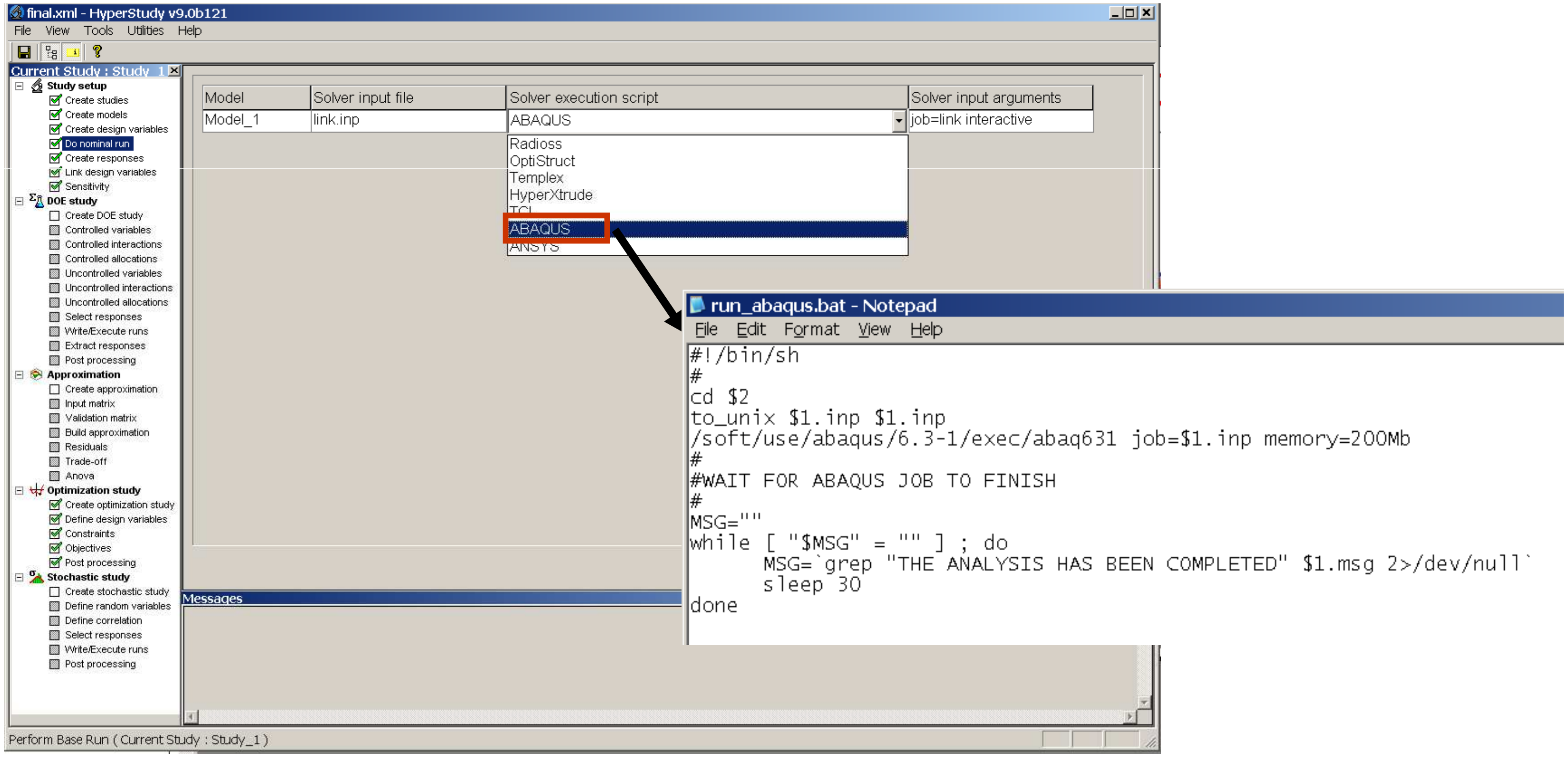
Direct Solvers & Results Access

- Use of HyperWorks result readers: no configuration needed
- Supported Solvers
  - ABAQUS
  - ADAMS
  - ANSYS
  - DADS
  - LS-DYNA
  - MADYMO
  - MARC
  - NASTRAN
  - PAM-CRASH
  - RADIOSS
  - SIMPACK
- HyperWorks Solvers
  - OptiStruct
  - MotionSolve
  - HyperForm
- General Result Formats
  - MS Excel (.csv file)
  - Any XY data
- File Parser
- Additional interfaces
  - Import Reader Language
  - External Readers Programmed in C

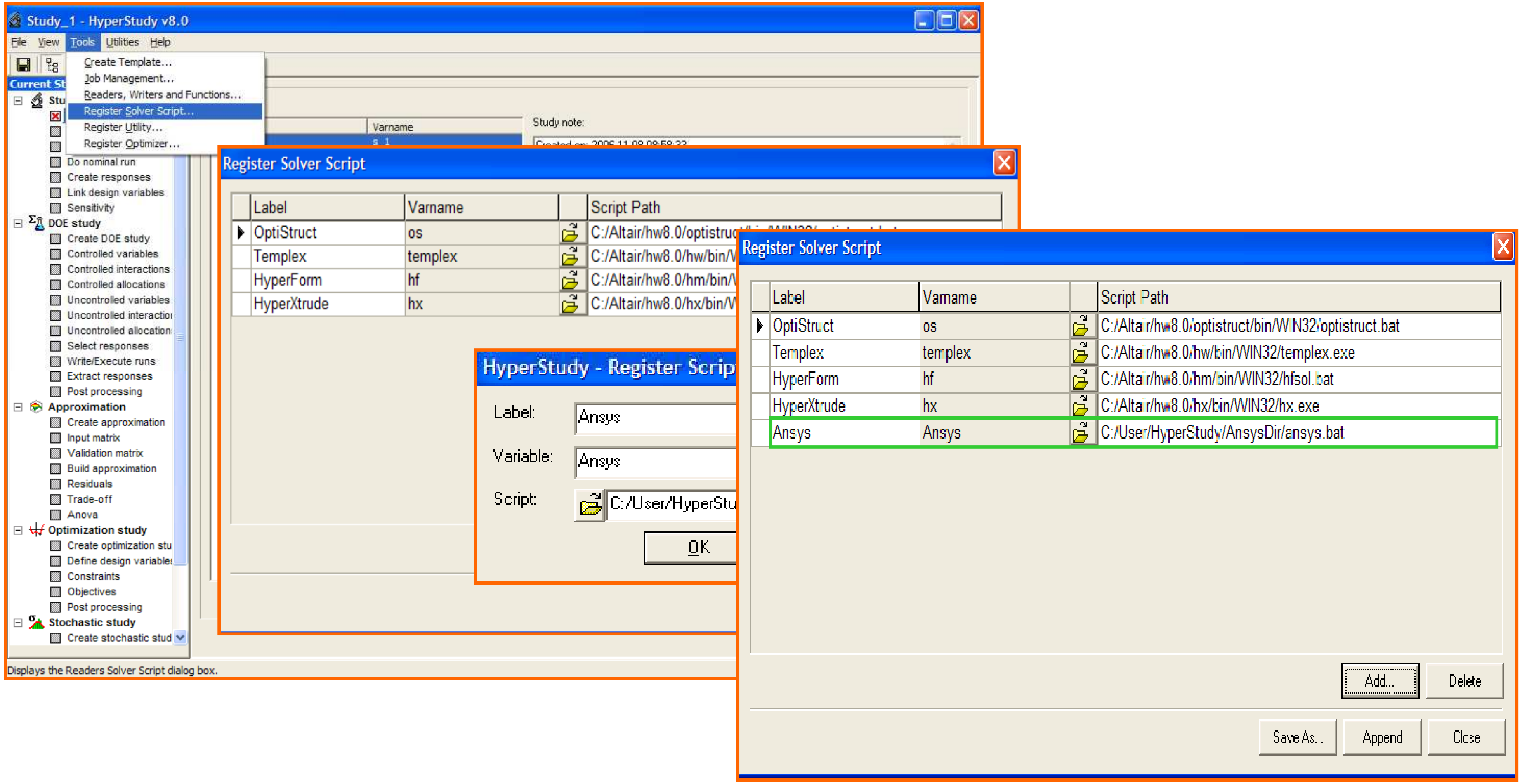
Interfacing with Solvers

By default, HyperStudy has 4 solvers: OptiStruct, Templex, HyperForm and HyperXtrude. Additional, solvers may be added through the

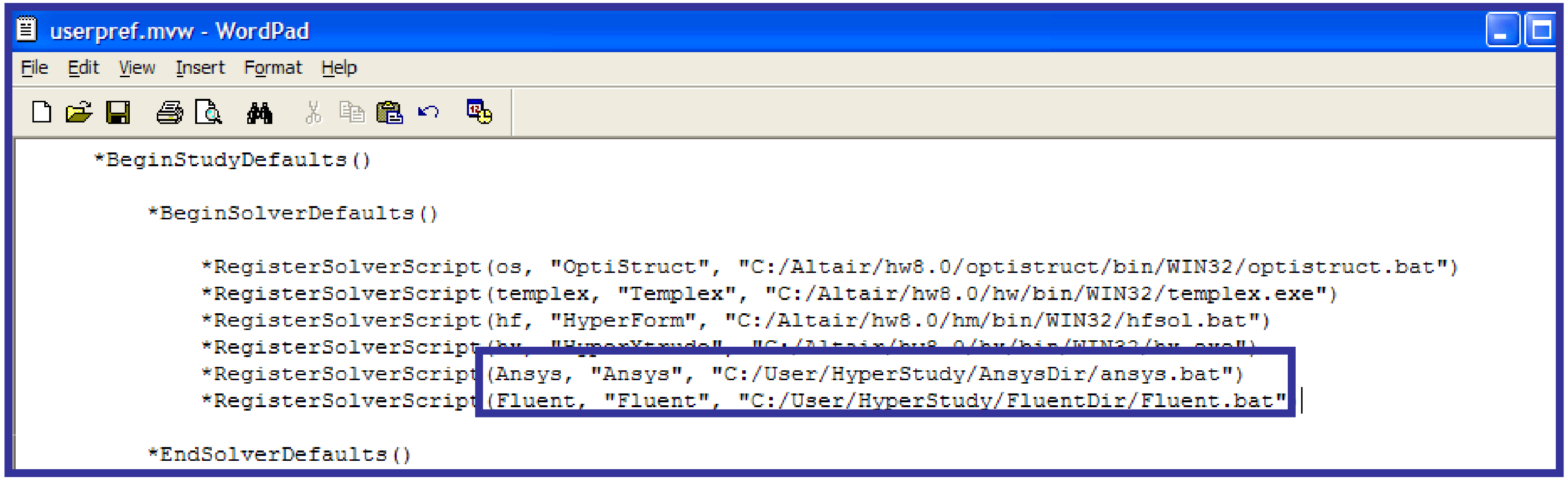
- Register Solver Script GUI or
- Preference file located in <install_dir>/hw/prefinc directory.

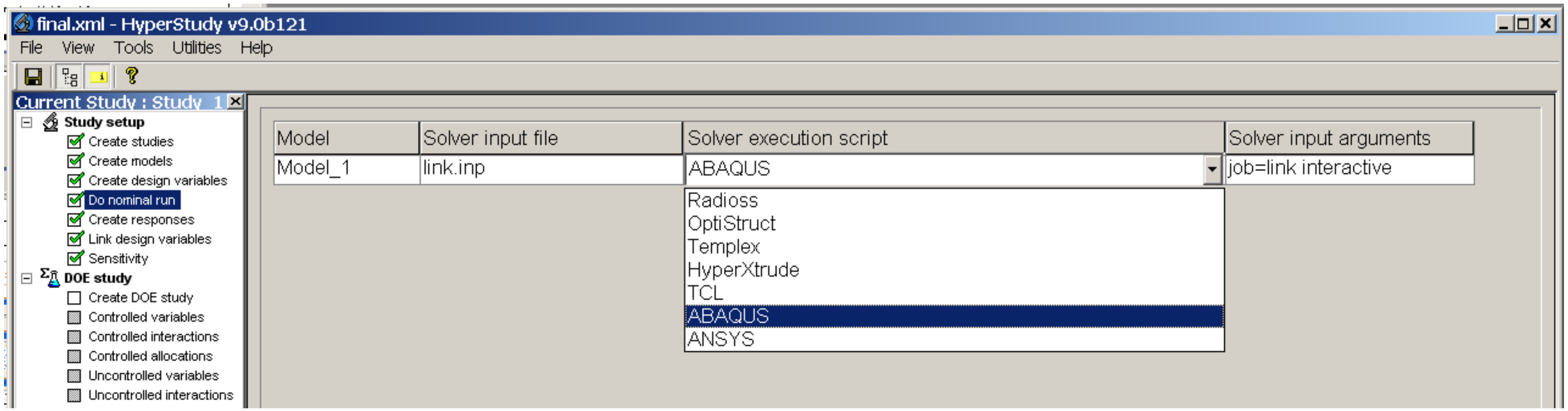


Interfacing with Solvers through the GUI

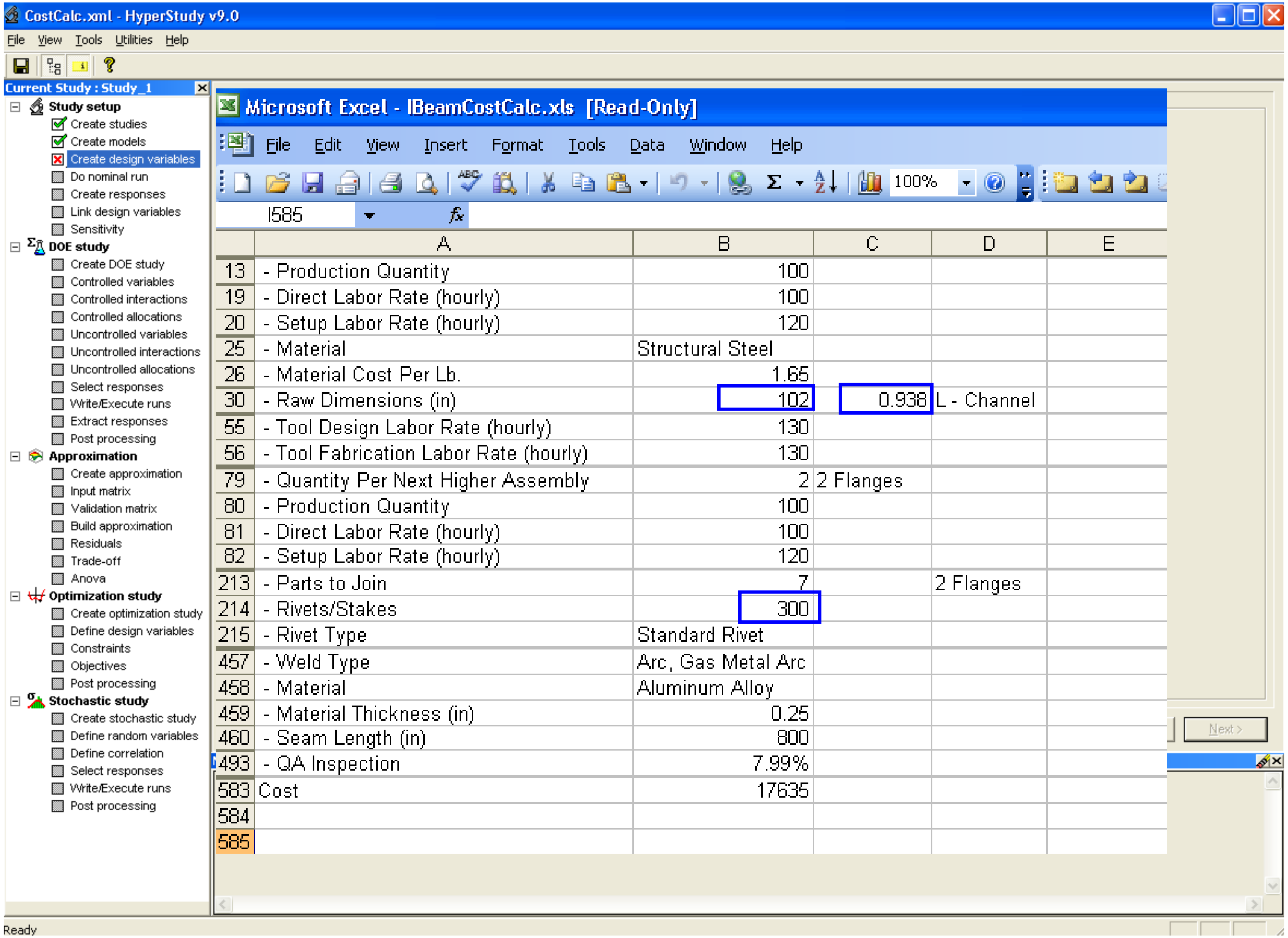


Interfacing with Solvers through the preference file





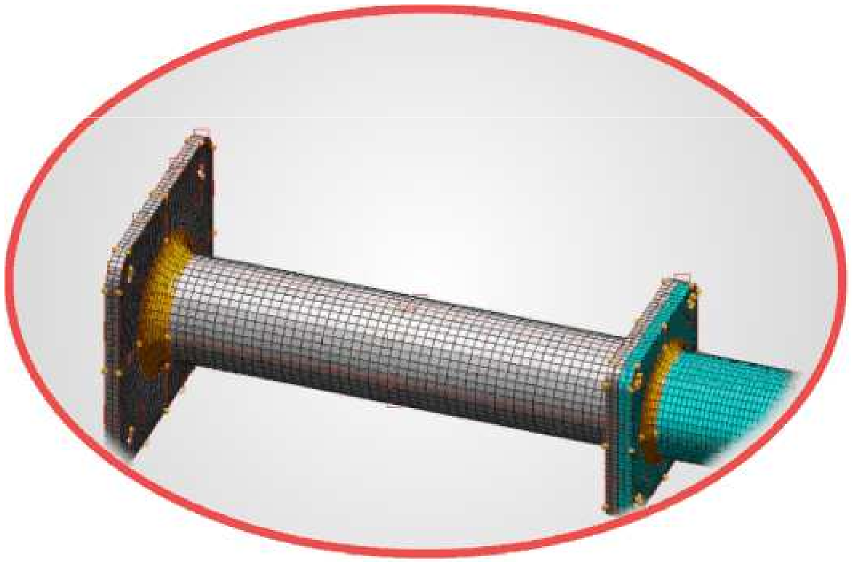
Integration with Excel

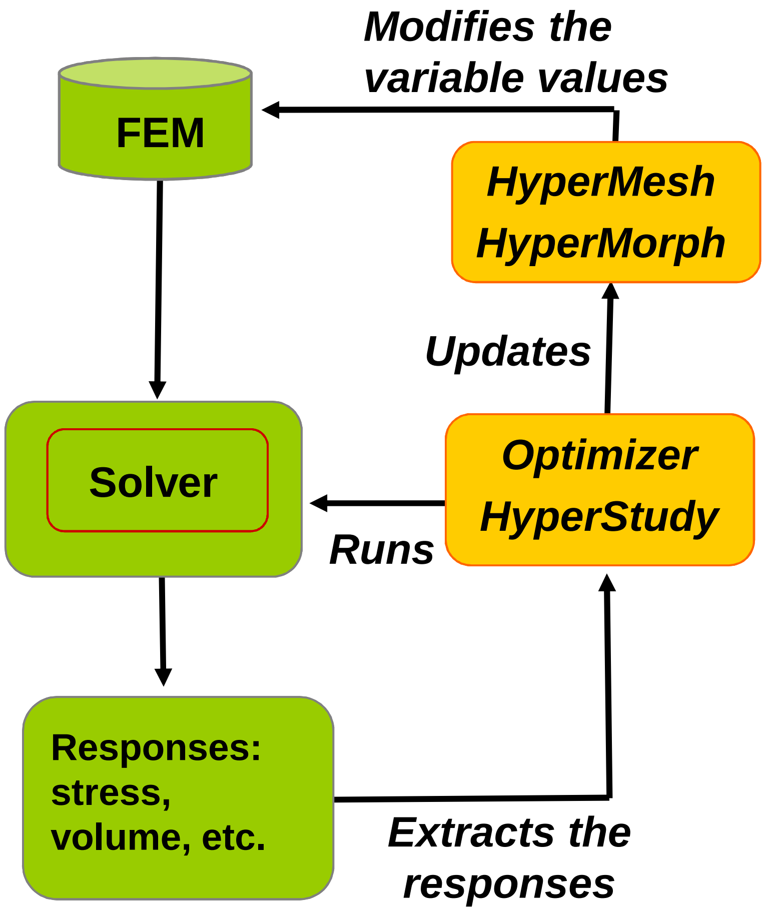


	A	B	C	D	E
13	- Production Quantity	100			
19	- Direct Labor Rate (hourly)	100			
20	- Setup Labor Rate (hourly)	120			
25	- Material	Structural Steel			
26	- Material Cost Per Lb.	1.65			
30	- Raw Dimensions (in)	102	0.938	L - Channel	
55	- Tool Design Labor Rate (hourly)	130			
56	- Tool Fabrication Labor Rate (hourly)	130			
79	- Quantity Per Next Higher Assembly	2	2 Flanges		
80	- Production Quantity	100			
81	- Direct Labor Rate (hourly)	100			
82	- Setup Labor Rate (hourly)	120			
213	- Parts to Join	7		2 Flanges	
214	- Rivets/Stakes	300			
215	- Rivet Type	Standard Rivet			
457	- Weld Type	Arc, Gas Metal Arc			
458	- Material	Aluminum Alloy			
459	- Material Thickness (in)	0.25			
460	- Seam Length (in)	800			
493	- QA Inspection	7.99%			
583	Cost	17635			
584					
585					

HyperStudy/HyperMorph Coupling

- Morphing: automatic mesh parameterization
- Morphing shapes used as design variables in HyperStudy and OptiStruct .
- Mesh based, no CAD data needed





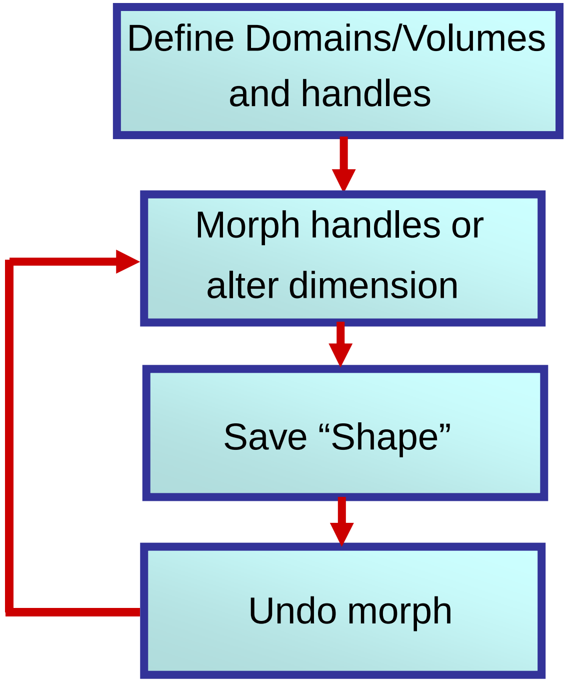
Shape Variable Generation using HyperMorph

Shape design variables necessary for DOE/Optimization/Stochastic studies can be easily handled using HyperMorph module within HyperMesh.

The finite element model is morphed to a given shape and then saved as “shape”. Multiple “shapes” can be saved in HyperMorph. These saved “shapes” are then used by HyperStudy as design variables.

Domains

global domains	1D domains
local domains	2D domains
global and local	3D domains
edge domains	auto functions
general domains	



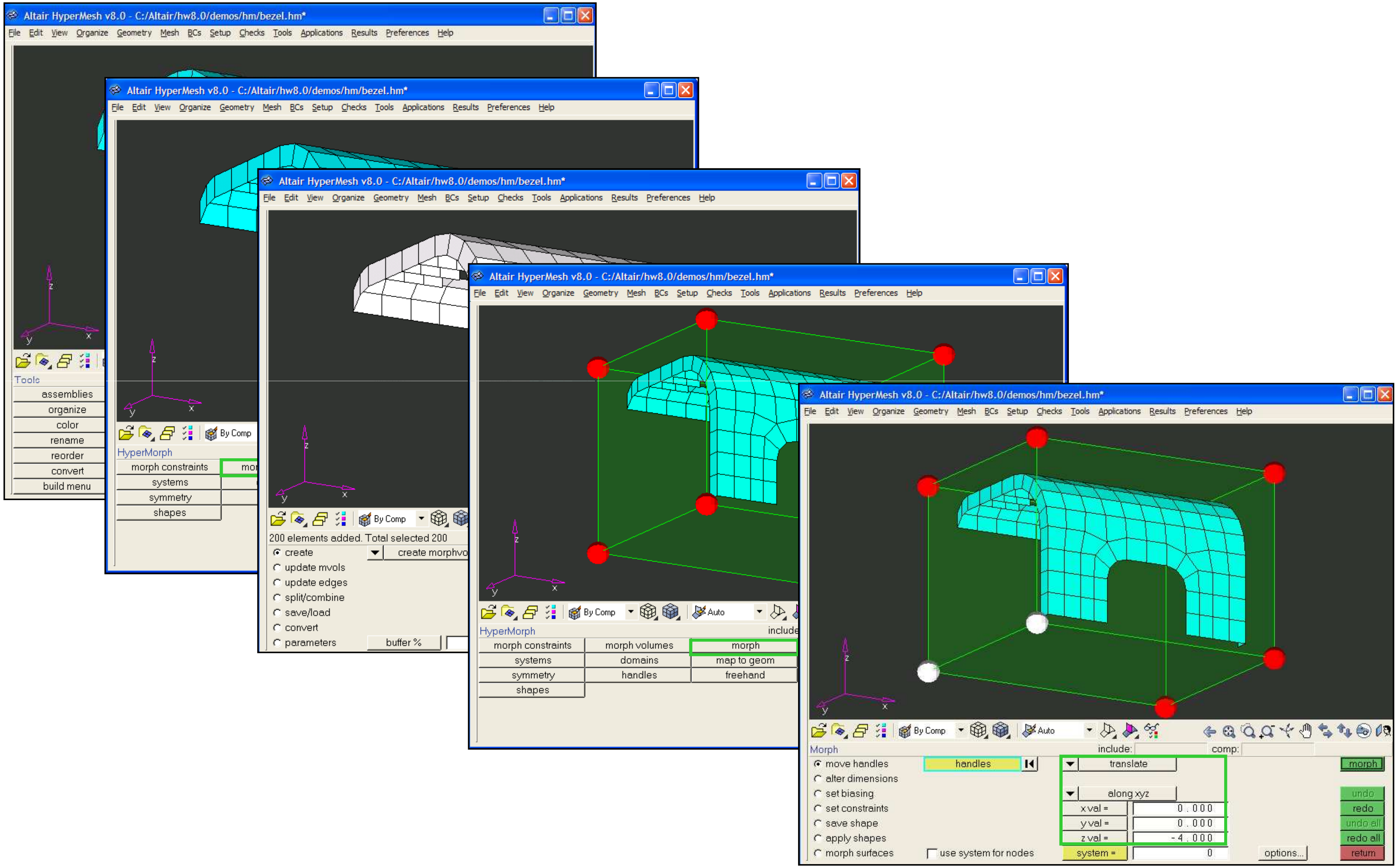
Morph Volumes

Morph to Geometry

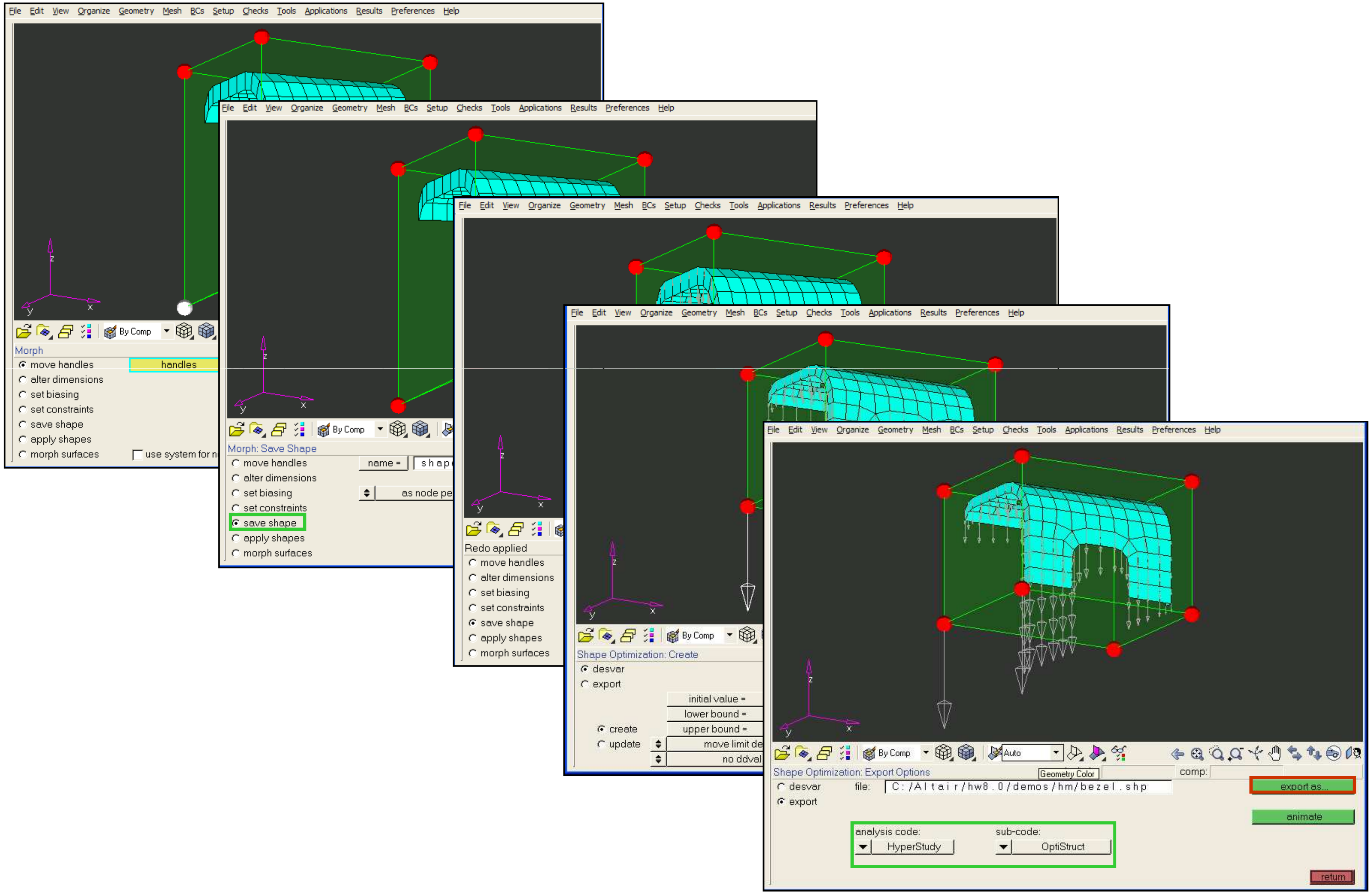
Handles

- Generated automatically
- Can be added by hand
- Global
- Local
- Biasing allows for C1 continuity

Shape Variable Generation using HyperMorph



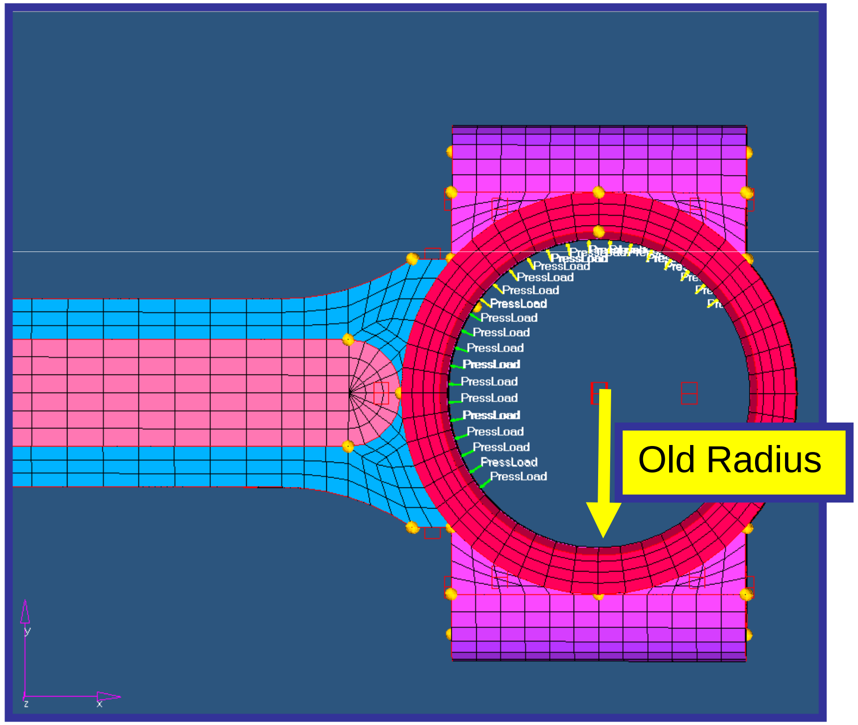
Shape Variable Generation using HyperMorph

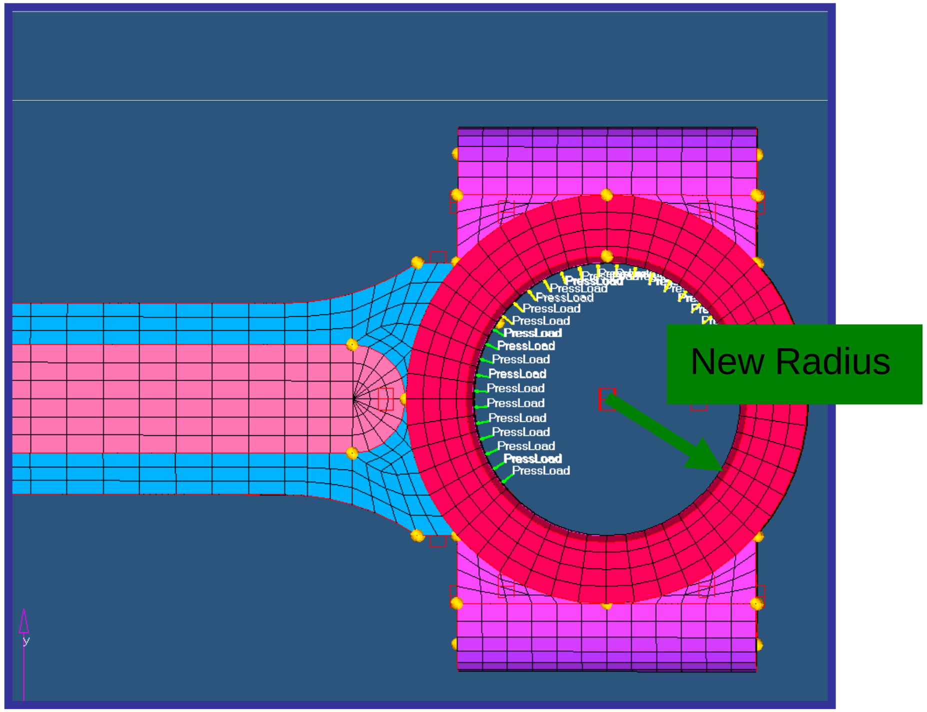


Example

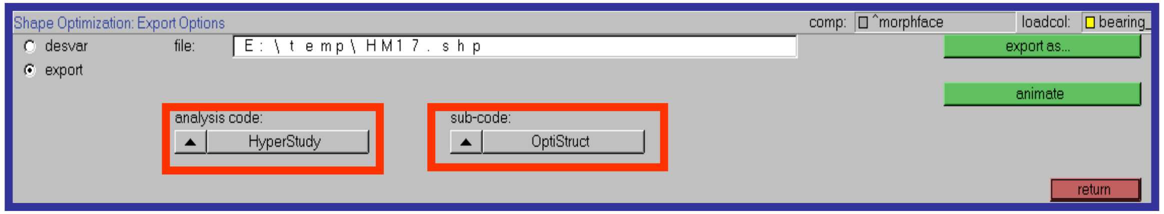
Design Variable:

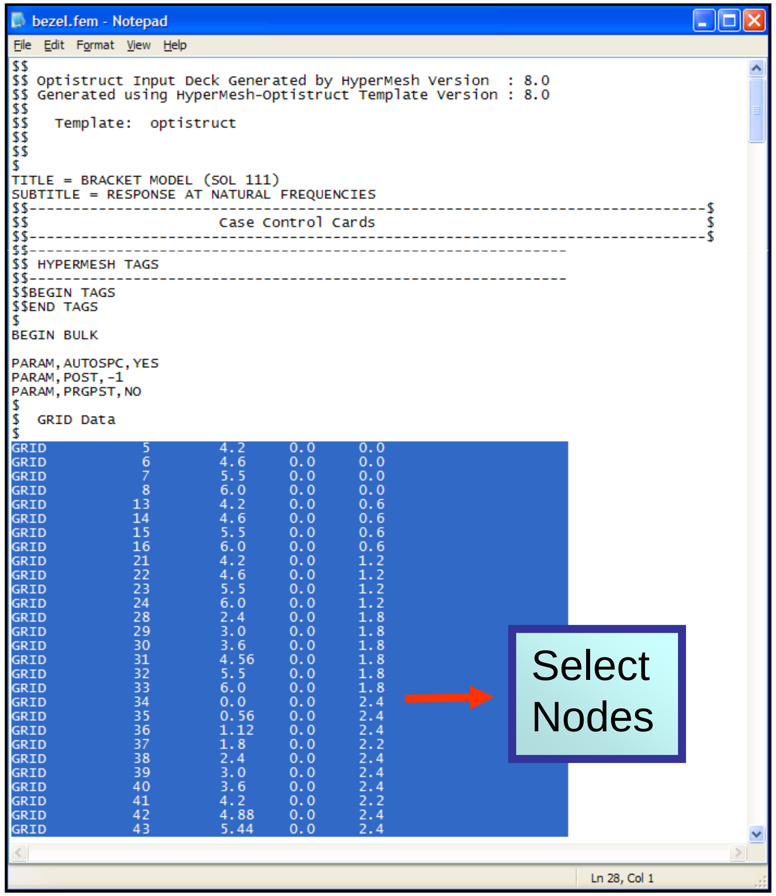
Boss Radius

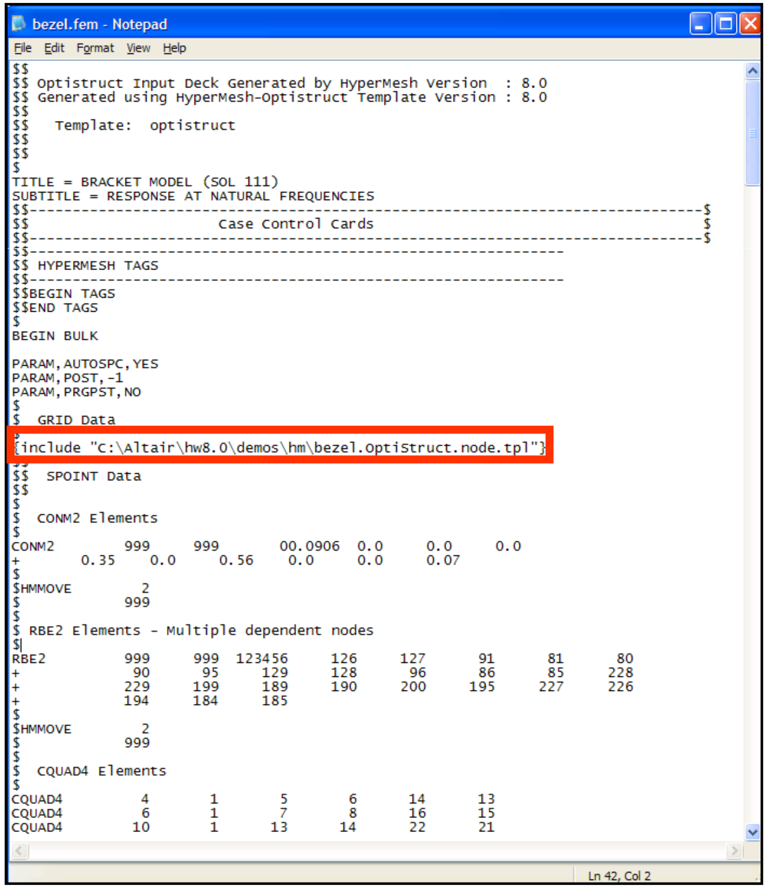




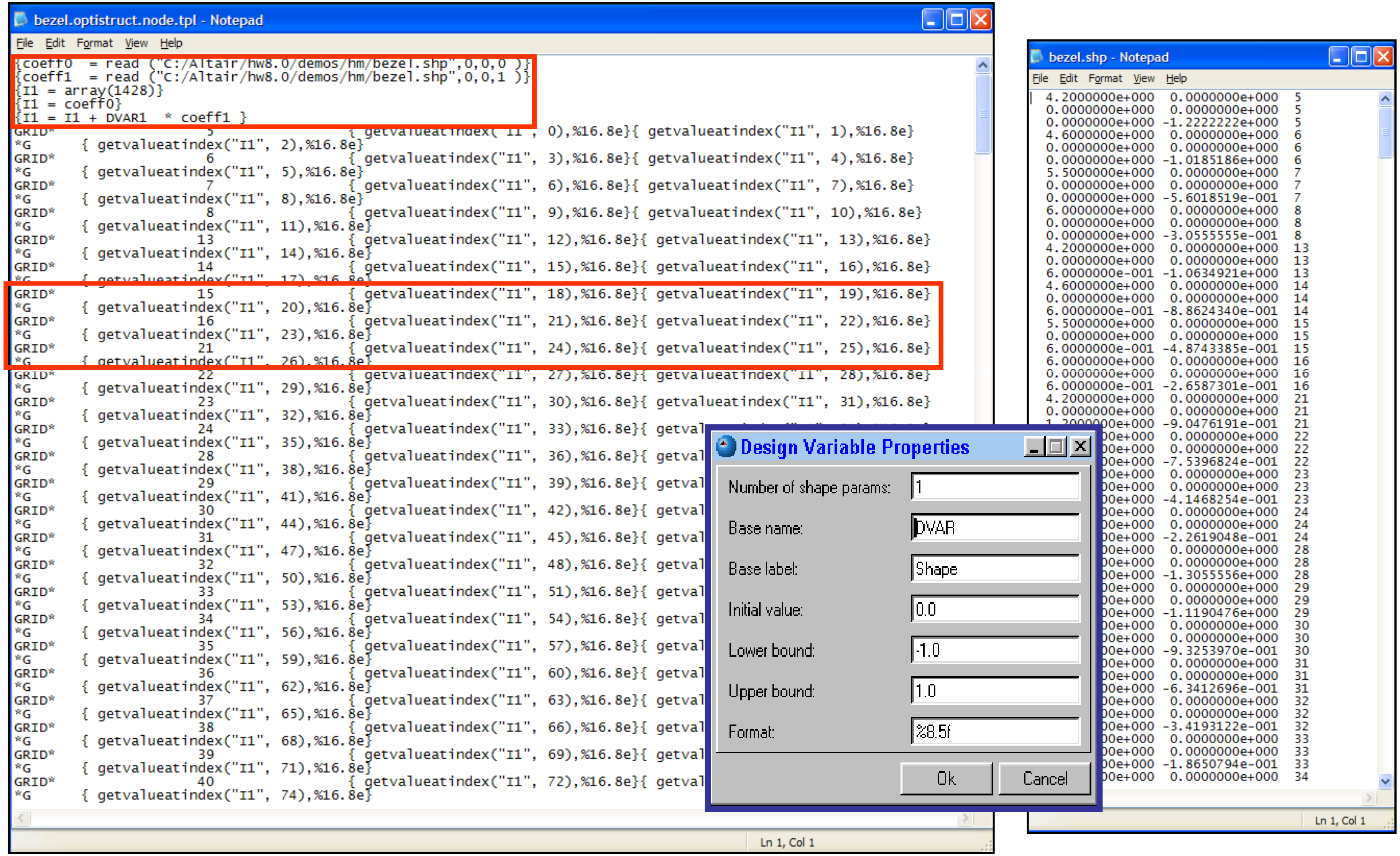
Exporting Shape Variables





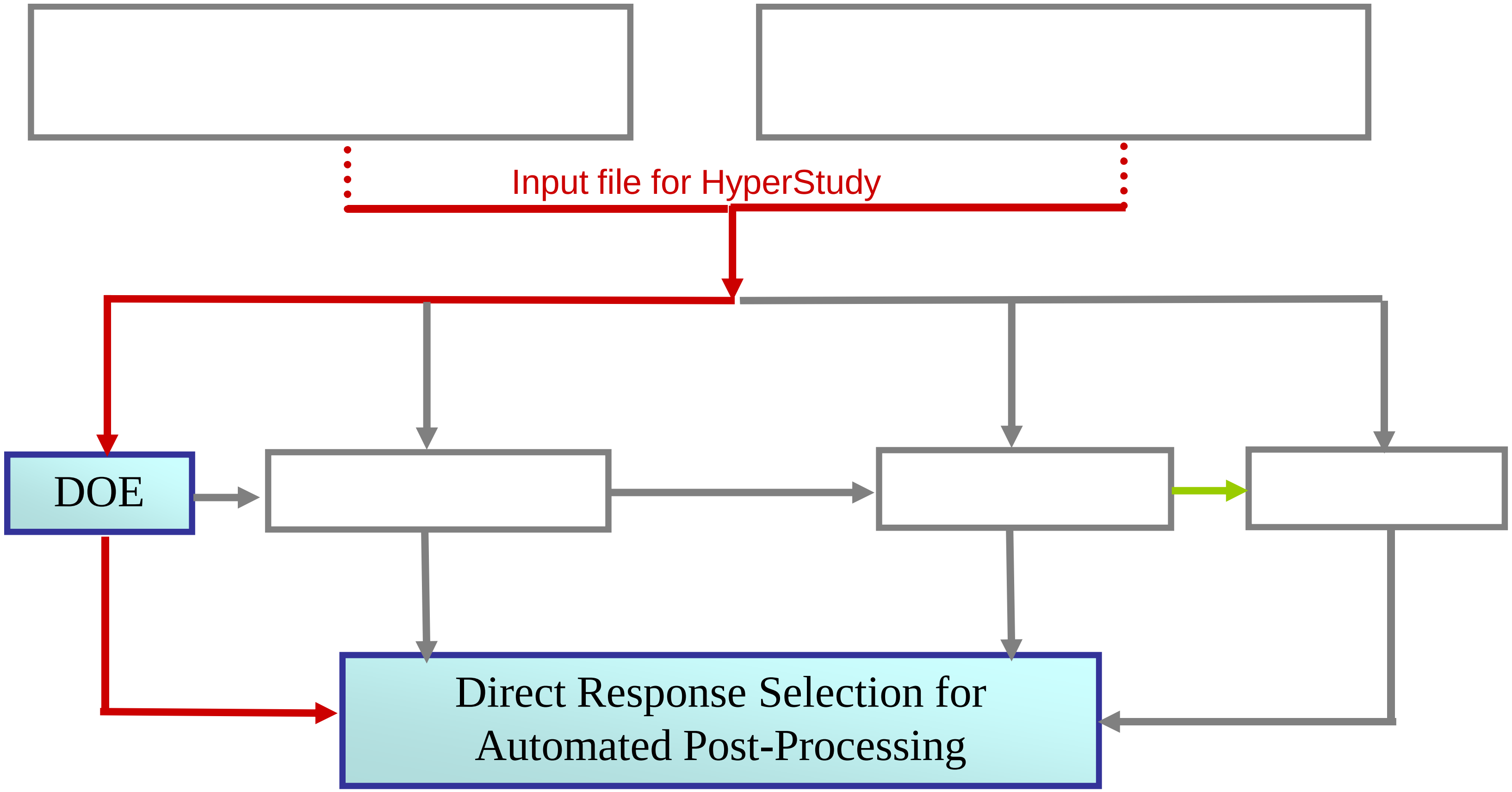


Exporting Shape Variables



Chapter 2: Design of Experiments (DOE)

Design of Experiments (DOE)



Design of Experiments (DOE)

What is DOE?

Design of Experiments (DOE) can be defined as a series of tests in which purposeful changes are made to the input variables of a process or system so that the reasons for changes in the output responses can be identified and observed.

Objectives of DOE Study

- To determine which factors are most influential on the responses.
- To determine where to set the influential controlled input variables so that:
  - The response is close to the desired nominal value.
  - Variability in output response is small.
  - The effects of the uncontrolled variables are minimized.
- To construct an approximate model that can be used as a surrogate model for the actual computationally intensive solver.

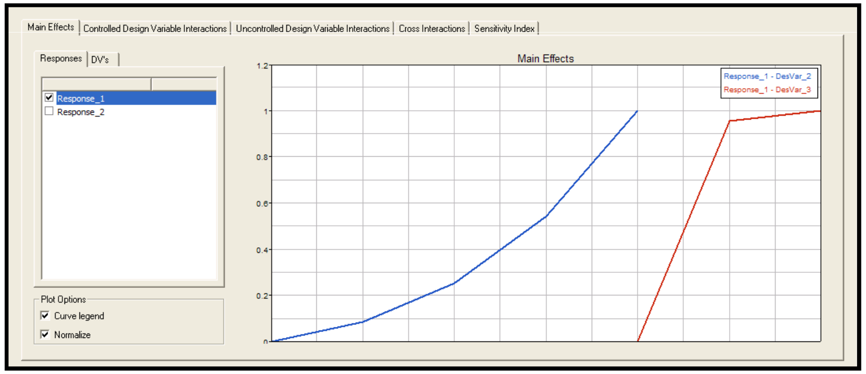
DOE Definitions

- A ***Factor*** is a input parameter (or design variable) of the system. Factors can be controlled or uncontrolled
- A ***Level*** is a discrete (or continuous) value of the factor.
- A factor can be either ***Discrete*** i.e., slow (-) or fast (+) (Ex. a variety of seed, type of paint etc.) or ***Continuous*** (Ex. temperature, a sinusoidal input for frequency response analysis etc.)
- ***Controlled factors*** are design variables that can be realistically controlled in the production (real world) environment. Examples include gauge thickness of sheet steel, shape of a support bracket, and mold temperature.
- ***Uncontrolled factors (Noise)*** are variables that cannot be realistically controlled in the production (real world) environment, but can be controlled in the lab. Examples include ambient temperature and occupant seating positioning.

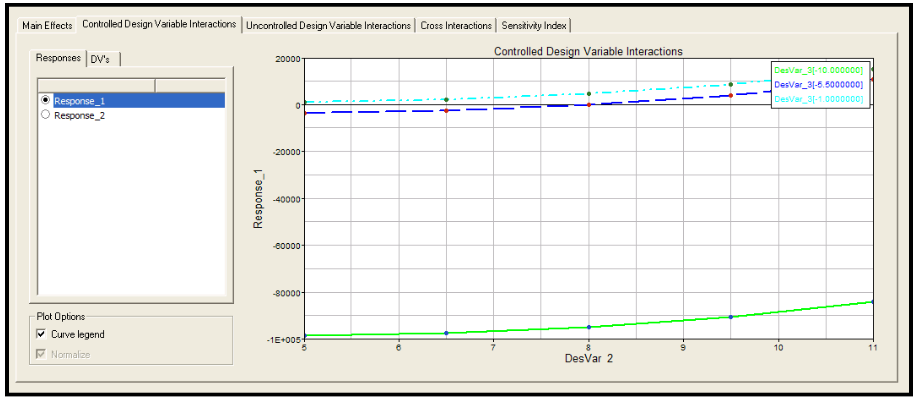
DOE Definitions

- ***Main Effects*** are the influence of the factors on the responses
- ***Interaction Effect*** is the interdependence among the factors. Due to this interdependence a difference in the response occurs when the factors are changed simultaneously as against what would have occurred when the factors were changed individually

Main Effects



Interaction Effects



DOE Example

Key DOE concepts are illustrated by considering an airbag design. Here we want to study how the **Speed of Bag Inflation (S)**, **Size of Vent (V)** and the **Size of Bag (B)**, influence the HIC of an air bag and also find out how these factors interact with each other.

FACTOR (Design Variable)	**LABEL**	**LEVEL (-)**	**LEVEL (+)**
Speed of Bag Inflation	S	Slow	Fast
Size of Vent	V	Small	Large
Size of Bag	B	Small	Large

DOE Example

Initially, consider the simplified problem where the vent size (V) is constant and consequently the are only two factors, S and B. For this, a simple full factorial DOE matrix is given as:

ROW	S	B	S X B
1	-	-	+
2	+	-	-
3	-	+	-
4	+	+	+
Divisor	2	2	2

The following DOE study will result in a polynomial expression that relates HIC to the two factors: Speed of the air bag and Size of the Bag.

Example: $HIC = a0 + a1 * S + a2 * B + a3 * S * B$

DOE Study Types - Applications

- **DOE Types Available in HyperStudy**
  - Full Factorial
  - Fractional Factorial
  - Central Composite
  - BoxBehnken
  - Plackett Burman
  - Latin Hypercube
  - Hammersley
  - Run Matrix
  - User Defined
- DOE for Screening
- DOE for Factorial Studies
- DOE for Response Surface (RSM) evaluation

DOE for Screening

Objective

- A simple DOE study (ex. two level design with no interactions) will provide a global understanding of the complete system i.e give the magnitude and direction of effects
- This initial screening exercise will allow parameters which do not influence the system to be discarded thus reduce the number of factors and runs.
- Lower precision

Types

- Fractional Factorial
- Plackett-Burman
- D-Optimal

DOE for Factorial Study

Objective

- Fewer factors
- Main effects and some factor interaction effects
- Linear model

Types

- Full Factorial
- Fractional factorial
- D-Optimal

DOE for Response Surface

Objective

- Fewer factors
- Model of relationships
- Accurate prediction
- Optimization

Types

- Box-Behnken
- Central composites
- D-Optimal

DOE Process

1. Assign factors (dv’s) from .tpl, .hm,.mdl, .hf file.
2. Perform nominal run to create responses for DOE study.
3. Select the **DOE type** for controlled and/or uncontrolled factors.
4. Divide the factors into controlled and uncontrolled if needed.
5. Export the solver input files for the specified runs as required by the DOE matrix.
6. Solve the above exported files.
7. Extract the responses for the above solved files.
8. Study the main effects, interaction effects, sensitivity index.

Exercise 2.1: DOE Study of a Rail Joint using OptiStruct

Exercise 2.2: DOE Study of a Cantilever Beam using Templex

Design Variables

Width: $20 < b < 40$

Height: $30 < h < 90$

Lenght: $50 < L < 100$

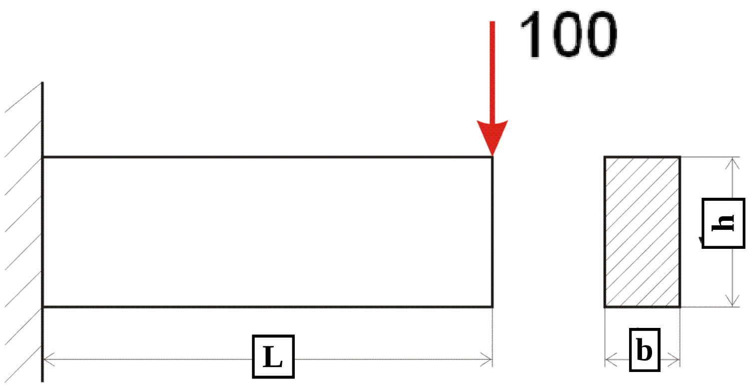
Design Space: *All beam elements*

Responses:

$$\sigma_{max}(L,b,h) = \frac{Mc}{I}$$

$$U_{max}(L,b,h) = \frac{PL^3}{3EI}$$

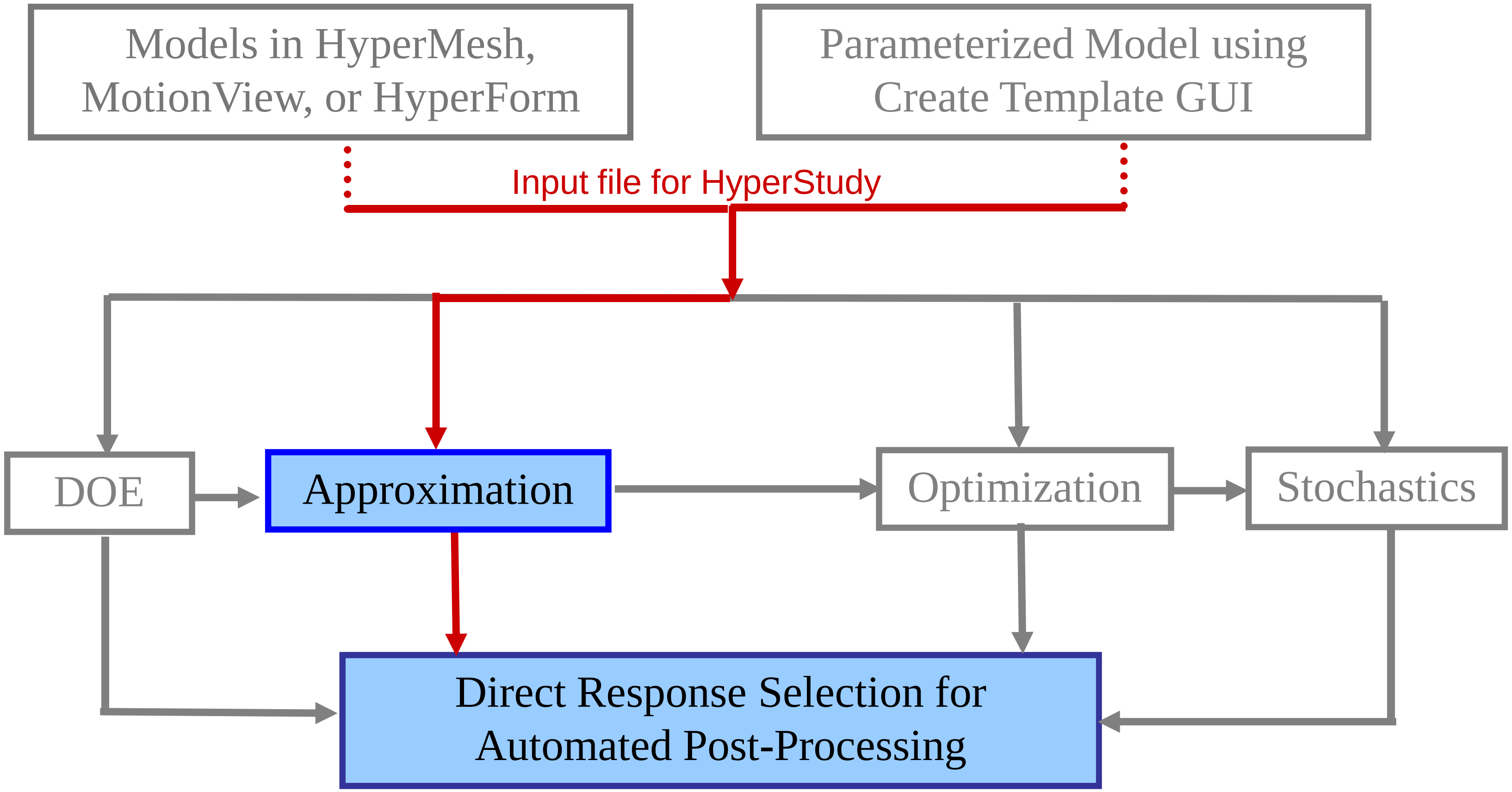
$$Vol(L,b,h) = Lbh$$



Exercise 2.3: DOE with MotionSolve Data Mining

Chapter 3: Approximations

Approximation



Approximation

- Approximations are surrogate models that represent the actual responses.

- **Why the need for approximations?**

- Some simulations are computationally expensive which makes it impractical to rely on them exclusively for design studies. Use of approximations in such cases lead to substantial savings of computational resources.
- Optimization can fall into local minimum or maximum when the responses are nonlinear. Using approximate responses, the user can avoid this issue.

- **The challenge:**

- When using approximations, the issue of a trade off between accuracy and efficiency is ever present.
- The question is how approximate the representation of the design space can be while remaining accurate enough.

Approximations Definitions

- ***Regression*** is the polynomial expression that relates the response of interest to the factors that were varied. It is only as good as the levels used when performing the study. For example, a two-level parameter only has a linear relationship in the regression. Higher order polynomials can be introduced by using more levels. Note that using more levels results in more runs.

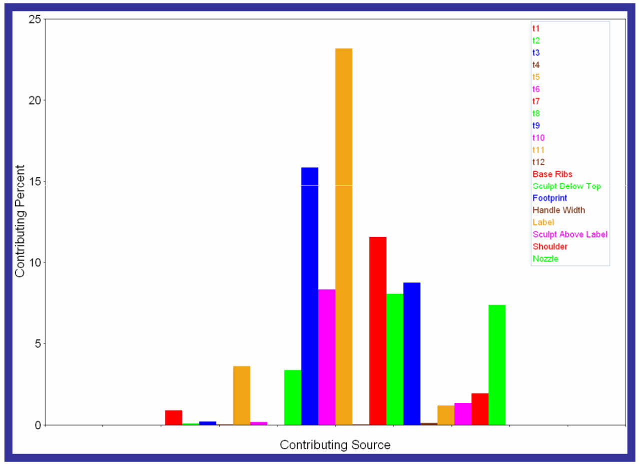
- ***Linear Regression model***
  - F(X) = a0 + a1X1 + a2X2 + (error)

- ***Interaction Regression Model***
  - F(X) = a0 + a1X1 + a2X2 + a3X1X2 + (error)

- ***Quadratic Regression Model (2nd order)***
  - F(X) = a0 + a1X1 + a2X2 + a3X1X2 + a4X12 + a5X22 + (error)

Approximations Definitions

Anova (Analysis of Variance) is a representation of the contributing percent of each design variable for the selected response.



Approximation Types

- Least Squares Regression
- Moving Least Squares Method
- HyperKriging

Approximation Types

Least Squares Regression (LSR)

LSR attempts to minimize the sum of the squares of the differences (*residuals*) between responses generated by the approximation and the corresponding simulation results.

HyperStudy allows the creation of least squares regressions for any polynomial order.

Moving Least Squares Method (MLSM)

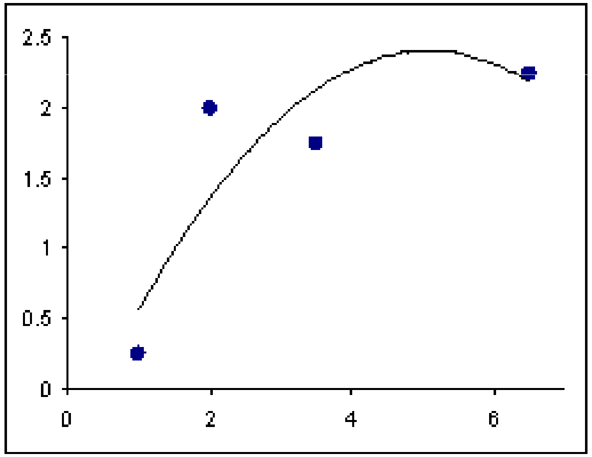
MLSM is a generalization of a conventional weighted least squares model building. The main difference is that the weights, associated with the individual DOE sampling points, do not remain constant but are functions of the normalized distance from a DOE sampling point to a point x where the approximation model is evaluated.

HyperStudy provides the choice of first, second, and third order functions for the Moving Least Square fit.

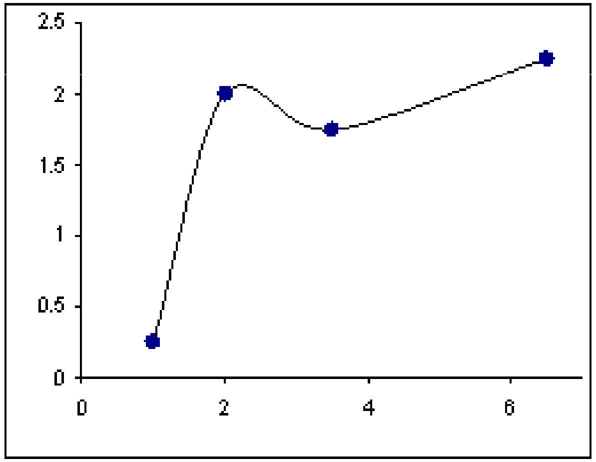
Approximation Types

HyperKriging

An approximation function, built using Kriging, has the property that it agrees with the original response function at the DOE points, i.e. this approach produces an interpolation model. This makes the technique suitable for modeling highly nonlinear response data that does not contain numerical noise. HyperKriging is computationally more expensive that the LSR and MLSM method.



Least squares quadratic regression



Kriging model

Approximations Process

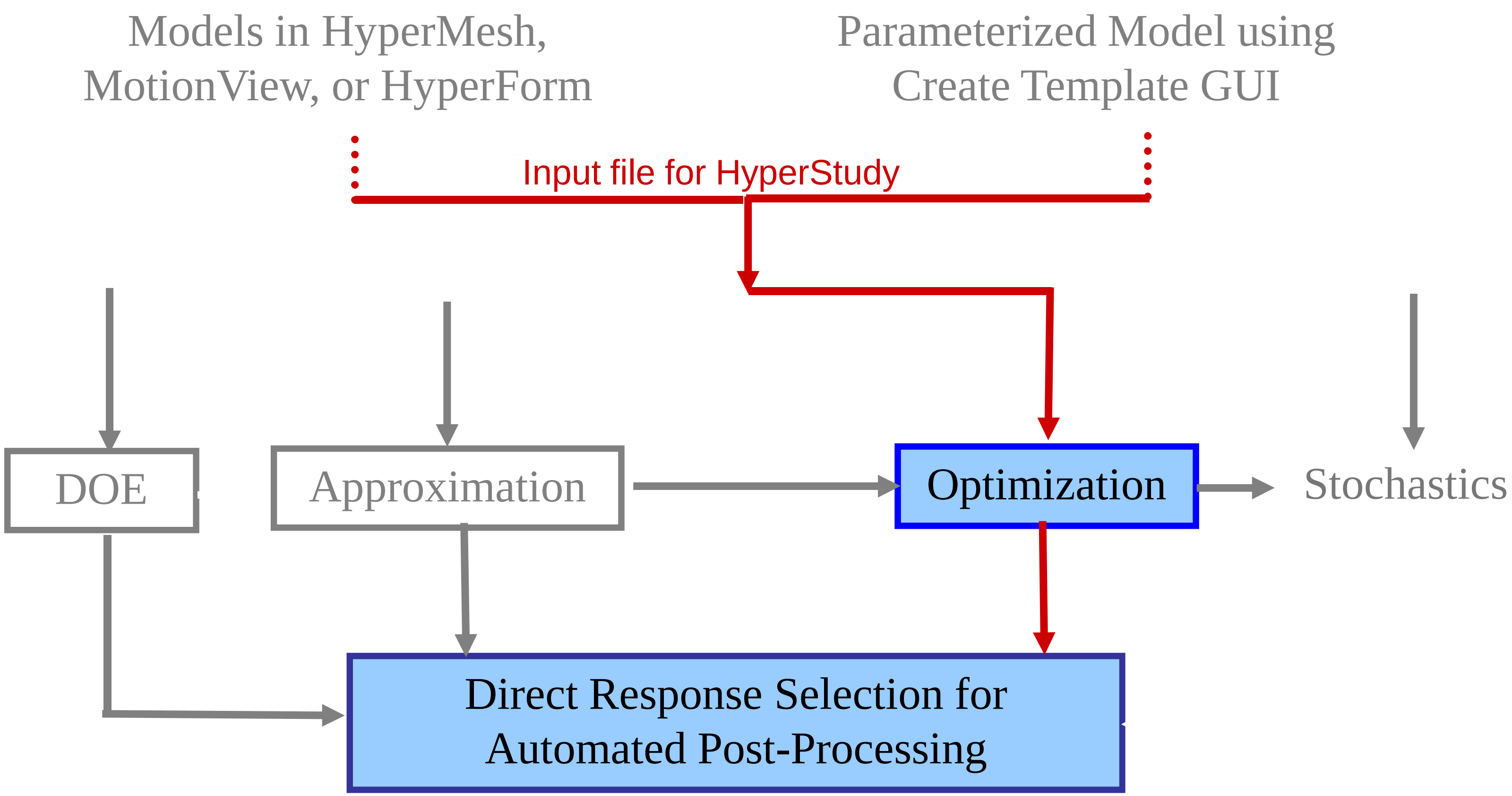
1. Select approximation types for responses
2. Import the run matrix
3. Build approximation models for each response
4. Check the residuals
5. Do trade-off studies and view ANOVA results

Exercise 3.1: Approximation Study of a Rail Joint using OptiStruct

Exercise 3.2: Approximation Study of a Cantilever Beam using Templex

Chapter 4: Optimization

Optimization



Optimization Problem Formulation

Objective: $\min f(\mathbf{x})$

min cost ($)

Constraints: $g(\mathbf{x}) \leq 0.0$

$\sigma < \sigma_{allowable}$

Design Space: $\text{lower } x_i \leq x_i \leq \text{upper } x_i$

2.5 mm < thickness < 5.0 mm

number of bolts $\in$ (20, 22, 24, 26, 28, 30)

Optimization Definitions

Design Variables: System parameters that can be changed to improve the system performance.

beam dimensions, material properties, diameter, number of bolts

Objective Function: System responses that are required to be minimized (maximized). These responses are functions of the design variables.

mass, stress, displacement, frequency, pressure drop

Constraint Functions: System requirements that need to be satisfied for the design to be acceptable. These functions are also functions of the design variables.

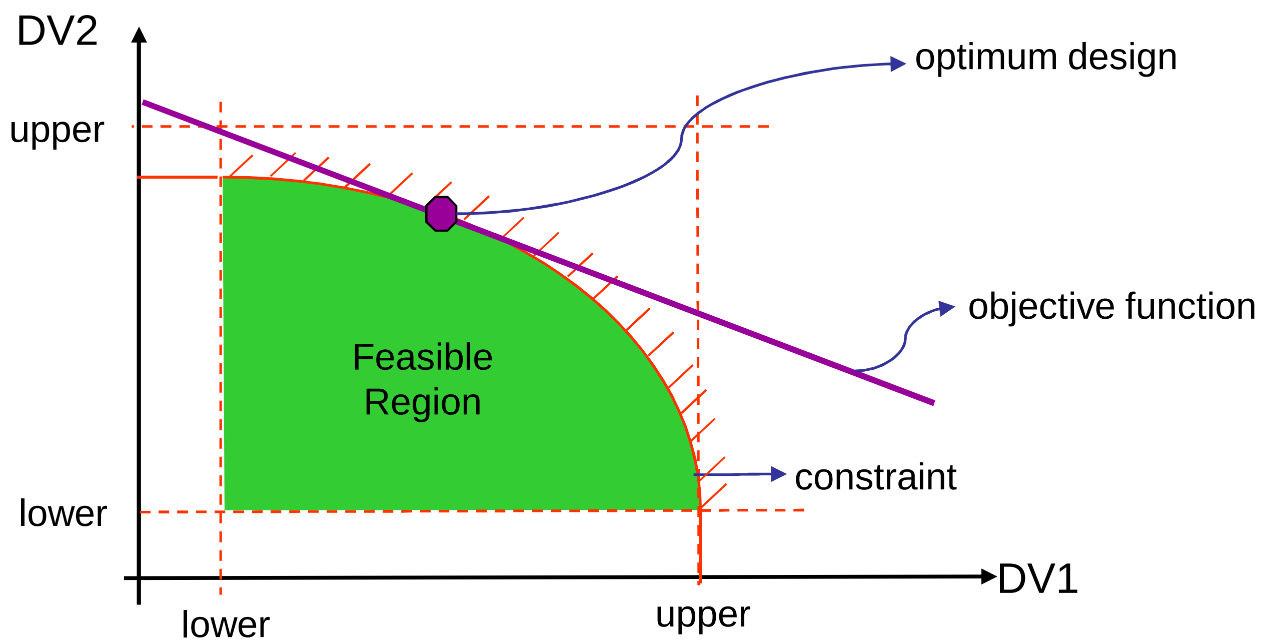
displacement, frequency, pressure drop, cost

Optimization Definitions

Feasible Design: Design that satisfies all the constraints.

Infeasible Design: Design that violates one or more constraints.

Optimum Design: Set of design variables along with the minimized (maximized) objective function that satisfy all the constraints.



Optimization Methods

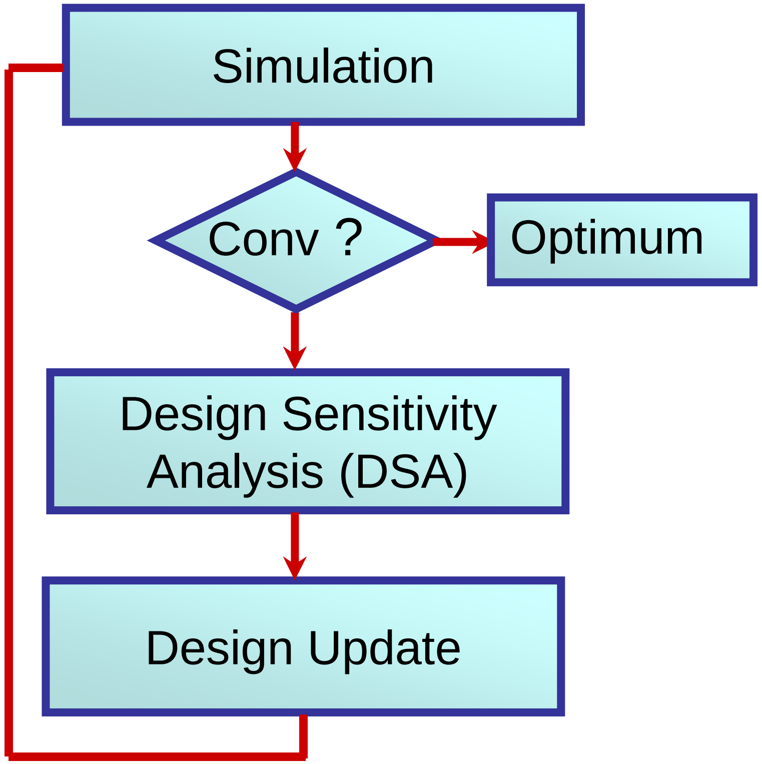
Optimization methods can be classified in three categories:

1. Gradient Based: are effective when the sensitivities (derivatives) of the system responses w.r.t design variables can be computed easily and inexpensively.

2. Response Surface Based: are very general in that they can be used with any analysis code including non-linear analysis codes. Global optimization methods use higher order polynomials to approximate the original structural optimization problem over a wide range of design variables.

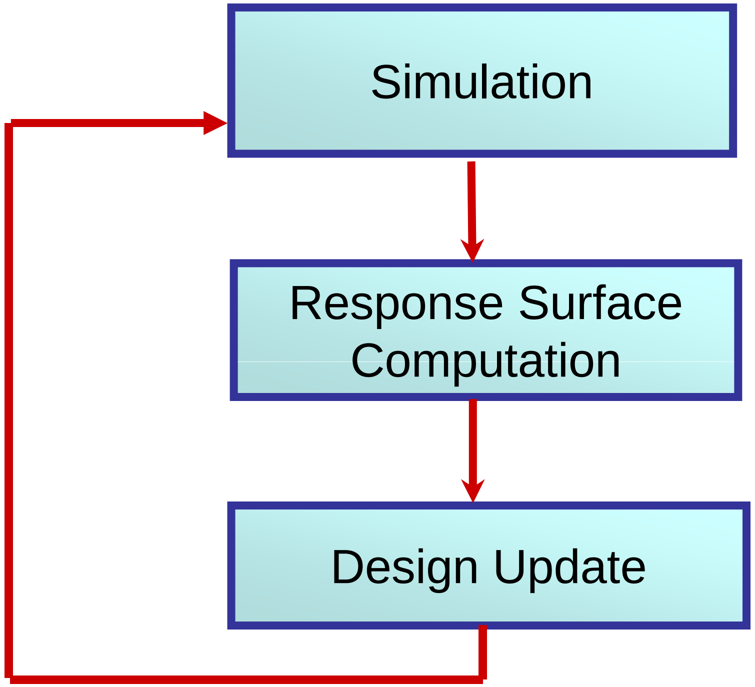
3. Exploratory Methods: are suitable for discrete problems such as finding the optimum number of cars to manufacture. These methods do not show the typical convergence of other optimization algorithms. Users typically select a maximum number of simulations to be evaluated. These algorithms are good in search on nonlinear domains however they are computationally expensive.

Gradient Based Methods



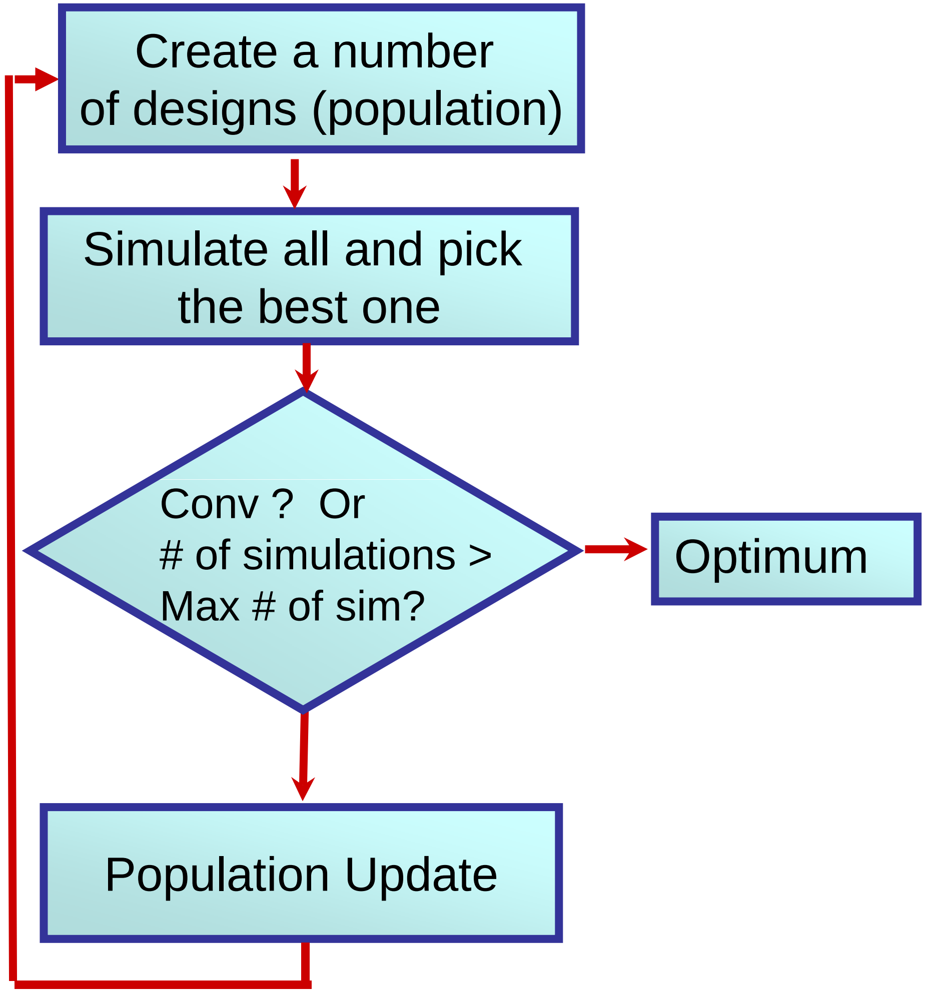
- Design sensitivity analysis (DSA) must be available
- Applied for linear static and dynamic problems
- Mostly integrated with FEA Solvers
- Not feasible for Non-Linear solvers

Surface Approximation Methods



- Sequential Response Surface update
  - Linear step
  - Quadratic response surface
- Non-linear physics
- Experimental Analysis
- Wrap-Around Software
- HyperOpt is solver neutral

Exploratory Methods



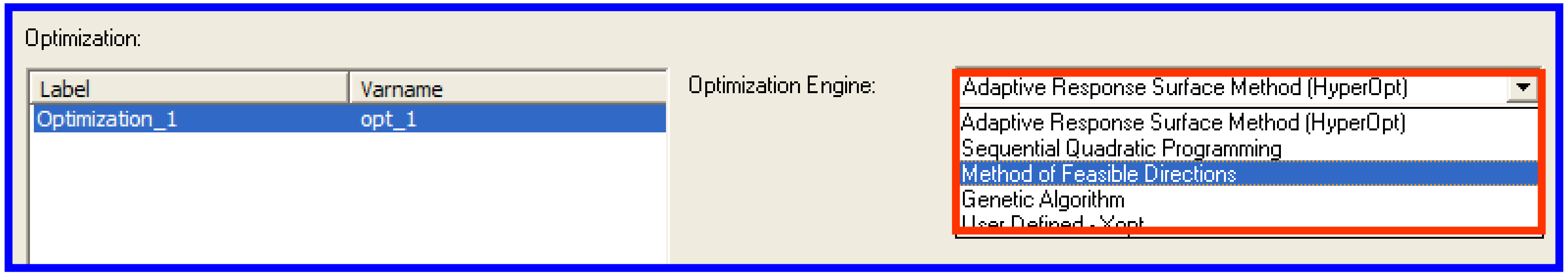
- Gradients are not needed
- Suitable to nonlinear simulations
- Computationally expensive as it may require large number of analysis
- Convergence not guaranteed

Optimization Methods in HyperStudy

HyperStudy offers four optimization engines: Adaptive Response Surface, Method of Feasible Directions, Sequential Quadratic Programming, Genetic Algorithm. It also allows for User-defined Method.

With user-defined method, user's may interface their optimization code with HyperStudy to perform optimization.

Also, the approximations can be used as surrogate model to perform optimization.

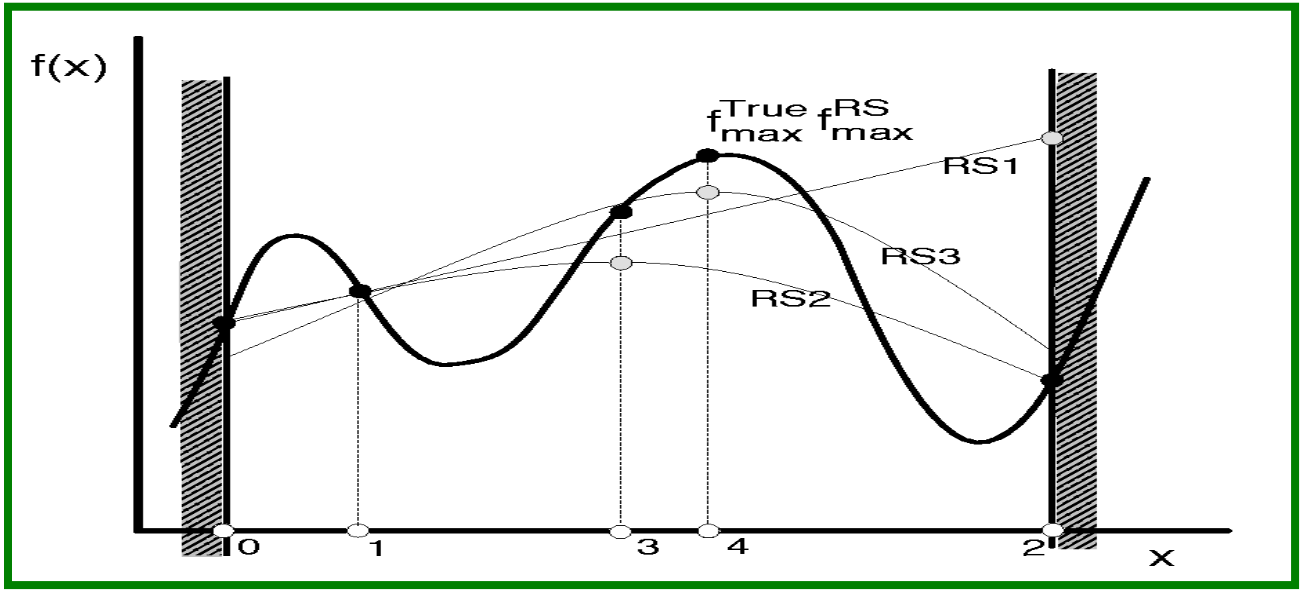


Adaptive Response Surface Method (HyperOpt)

In this approach, the objective and constraint functions are approximated in terms of design variables x using a second order polynomial.

The polynomial coefficients are determined using a least squares fit of the functions on to the previous design points (actual nonlinear analysis results).

In general, more designs are available than are required for an exact least squares fit, making the system over-determined. HyperOpt uses a very efficient algorithm to estimate a response surface to be closer to certain designs of interest. HyperOpt also uses move limits to make the optimization algorithm robust.



Method of Feasible Directions

The method of feasible directions is one of the earliest methods for solving constrained optimization problems.

The fundamental principle behind this method is to move from one feasible design to an improved feasible design. Hence, the objective function must be reduced and the constraints at the new design point should not be violated.

Sequential Quadratic Programming

Sequential quadratic programming (SQP) is a method for solving constrained optimization problems.

The fundamental principle behind this method is to create a quadratic approximation of the Lagrangian and to solve that quadratic problem to define the search direction s.

The constraints are linearized during the search.

This quadratic problem can be solved by a variety of methods. The solution of the problem yields the search direction along which the next design that improves the objective function and does not violate the constraint can be found.

Genetic Algorithm

A genetic algorithm is a machine learning technique modeled after the evolutionary process theory.

Genetic algorithms differ from conventional optimization techniques in that they work on a population of designs. These designs are then evaluated for their fitness, which is a measure of how good a particular design is. Following Darwin's principle of survival of the fittest, designs with higher fitness values have a higher probability of being selected for mating purposes to produce the next generation of candidate solutions.

In addition to reproduction, selected individual designs go through crossover and mutation. The designs that result from this process (the children) become members of the next generation of candidate solutions.

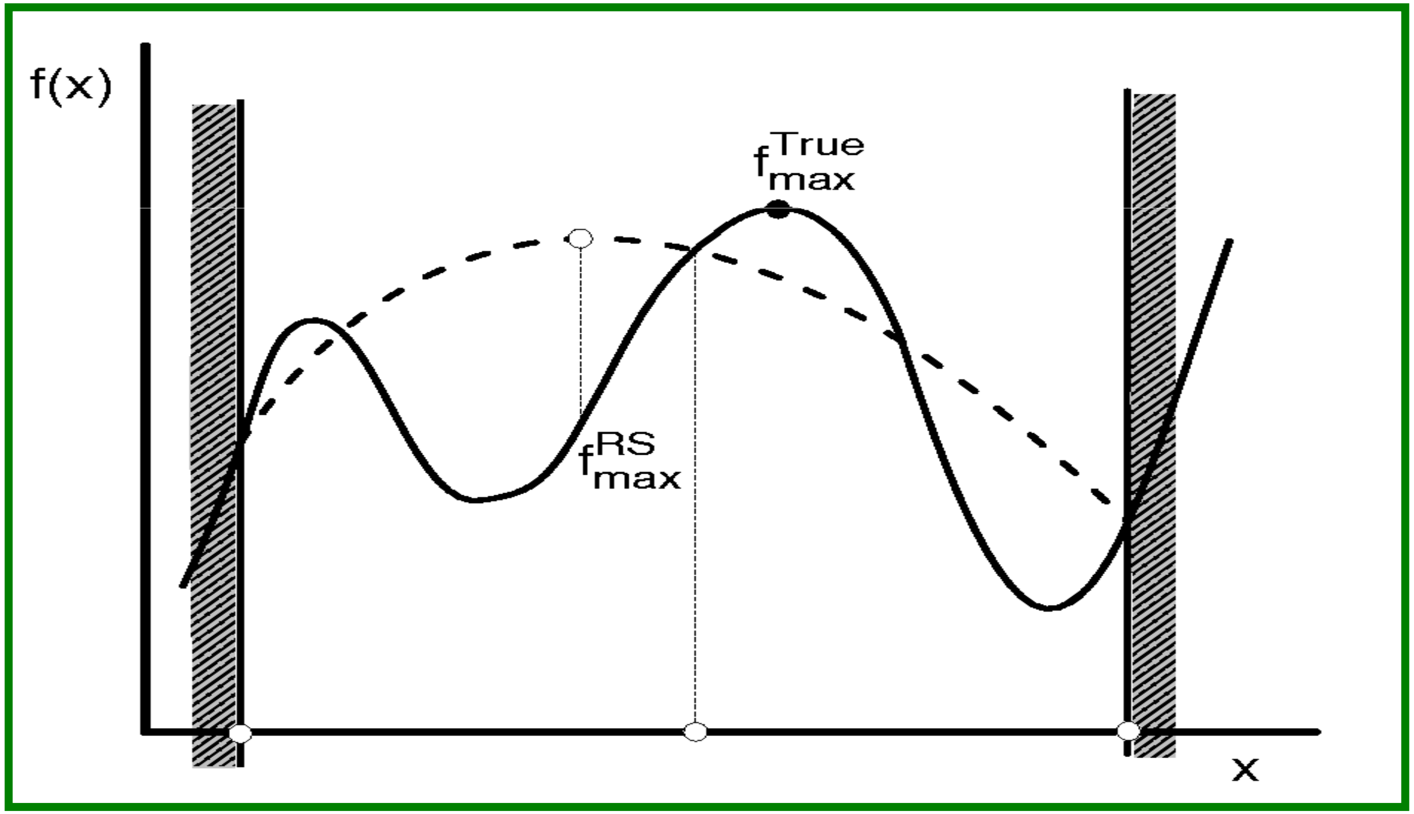
This process is repeated for many generations in order to artificially force the evolution of a population of designs that yield a solution to a given problem.

Genetic algorithms do not show the typical convergence of other optimization algorithms. Users typically select a maximum number of iterations (generations) to be evaluated. A number of solver runs is executed in each generation, with each run representing a member of the population.

Using Approximations as a Surrogate Model
Ordinary Response Surface Method

Ordinary response surface methods are based on DOE studies

Response surface is fitted (quadratic polynomial) with a fixed number of designs analyzed, consequently it is not sufficient to approximate and optimize highly non-linear functions

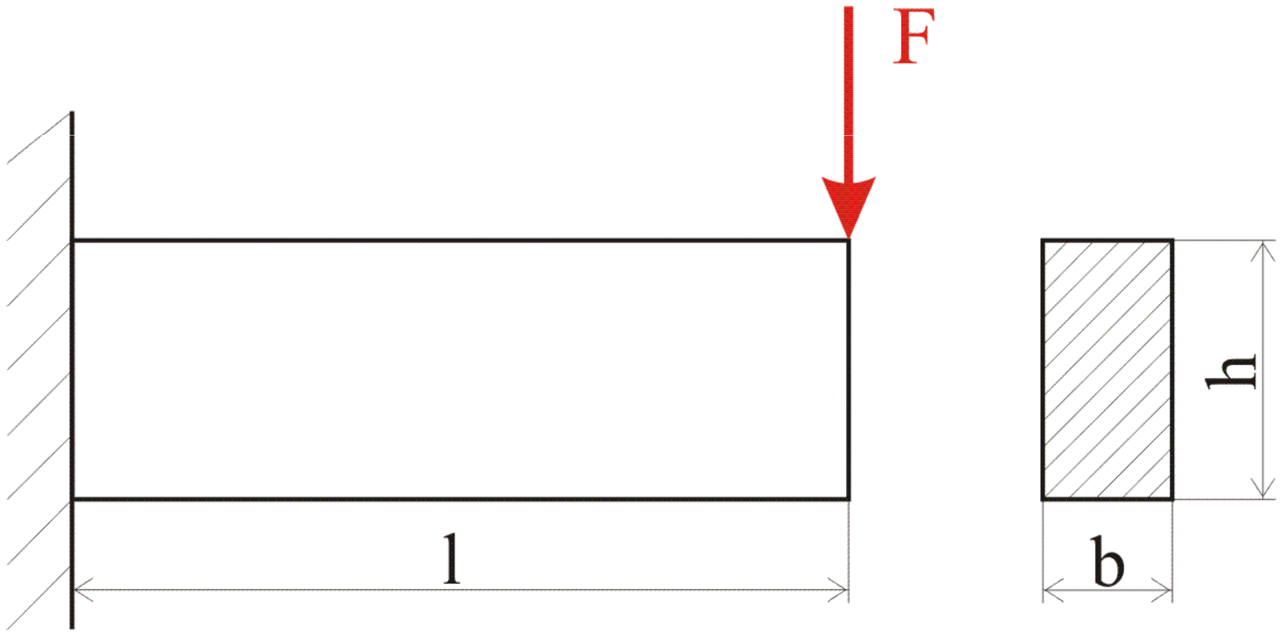


Optimization Process

1. Assign factors (dv’s) from .tpl, .hm,.mdl, .hf file
2. Perform nominal run to create responses.
3. Select the Optimization Algorithm
4. Select the Design Variables for optimization study
5. Define Objective function and optimization constraints
6. Launch Optimization
7. Postprocess the optimization results

Optimization Problem Example

- A cantilever beam is modeled with 1D beam elements and loaded with force F=2400 N. Width and height of cross-section are optimized to minimize weight such that stresses do not exceed yield. Further the height h should not be larger than twice the width b.



Optimization Problem Example

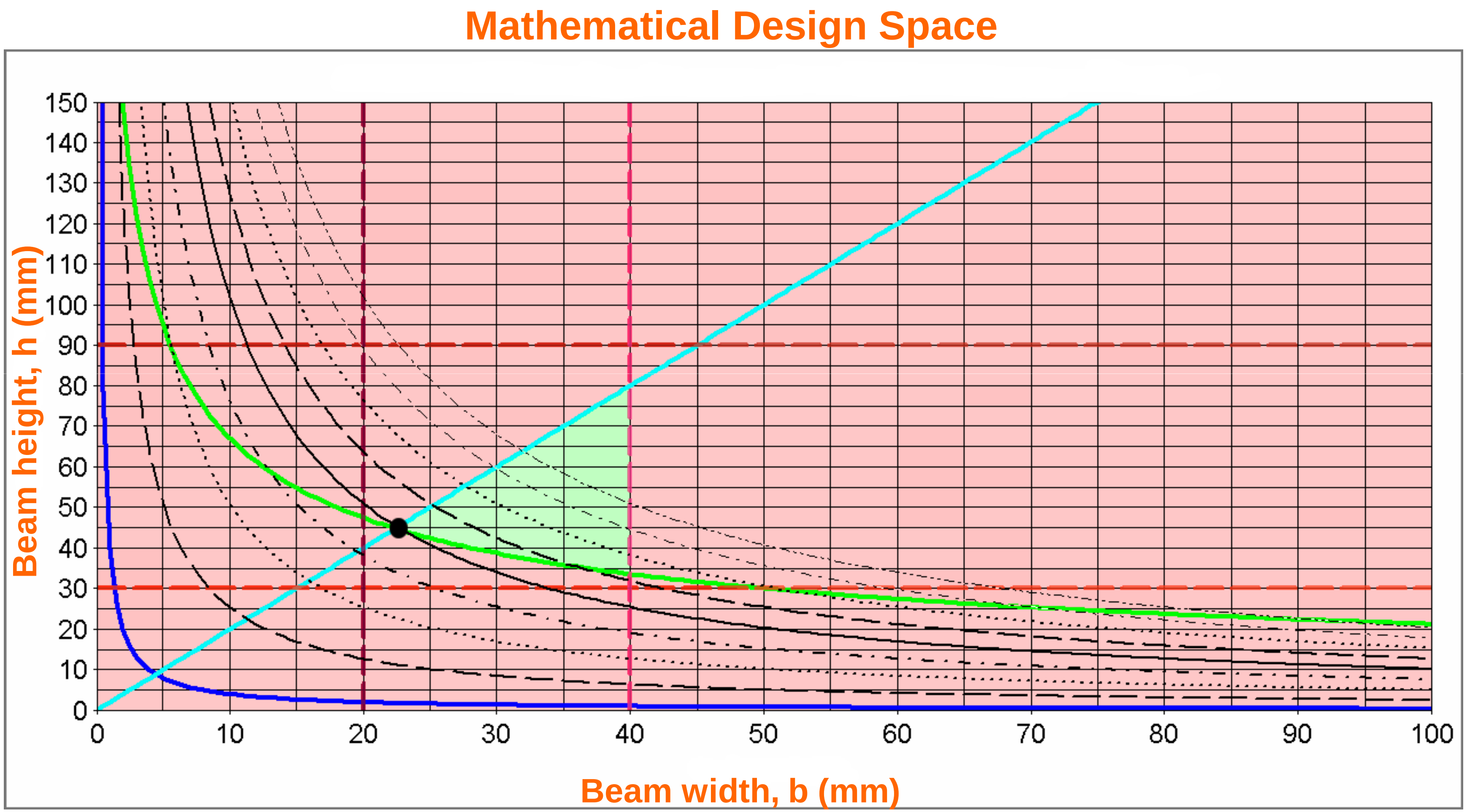
- Objective
  - Weight: *min m(b,h)*
- Design Variables
  - Width: $b^L < b < b^U$, $20 < b < 40$
  - Height: $h^L < h < h^U$, $30 < h < 90$
- Design Region: *All beam elements*
- Design Constraints:

$\sigma(b,h) \leq \sigma_{max}$, *with* $\sigma_{max} = 160$ MPa

$\tau(b,h) \leq \tau_{max}$, *with* $\tau_{max} = 60$ MPa

$h \leq 2*b$

Optimization Problem Example



HyperStudy Case Studies and Examples

Case Studies Examples

- Optimization studies are increasing in popularity. It provides added value to the baseline study, it helps the engineer in understanding the physics of the problem, it allows exploration in the design space.
- The technology has been applied to full body crash. An example consists of an objective to minimize mass using footwell intrusion displacements as constraints and several material thickness values as design variables
- The technique has been applied to headform assessments. Initially to identify the worst attitude of impact. Once this is established the rib thickness of the crashbox is determined
- Four case studies are presented. Two relating to structural crashworthiness, two relating to occupant safety .
- In addition fours examples are given. Two of these examples relate to computational fluid dynamics and two of them relate to fatigue.

Case Study 1: Rail Optimization

- A box rail of length lx=800mm is clamped at the right side and is impacted by a moving stone wall of mass 1,000kg and initial velocity 2m/s from the left side
- The rail is required to absorb the maximum internal energy within an analysis time of ts=30ms. The displacement of the stone wall (x-direction) should be less than 50mm. The problem is typical of the designs encountered in car bumpers and crash boxes
- A size optimization is performed with six design variables consisting of variable thickness patches. The initial thickness of all design variables was 2mm and could vary between 0.2mm and 3mm

Case Study 1: Rail Optimization

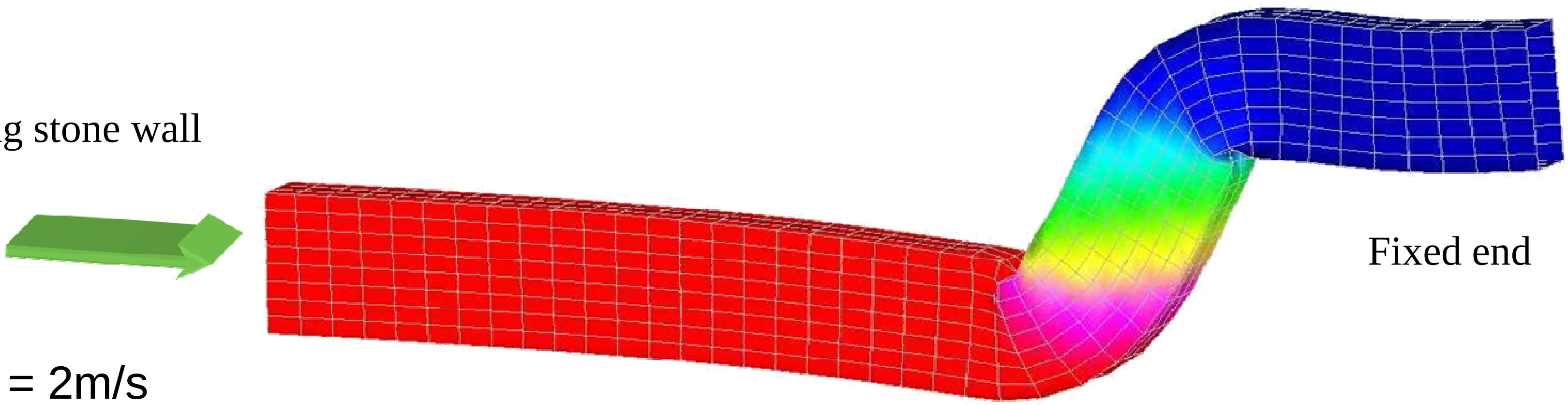
- The HyperOpt optimization (Sequential Optimization) is compared against local approximation and DOE approaches. Although structural behaviour is similar, HyperOpt maximizes the internal energy
- Design iteration history is presented showing the variation of the objective with the constraint

Case Study 1: Rail Optimization

Problem Setup

max Internal Energy

Δl_x = 50mm



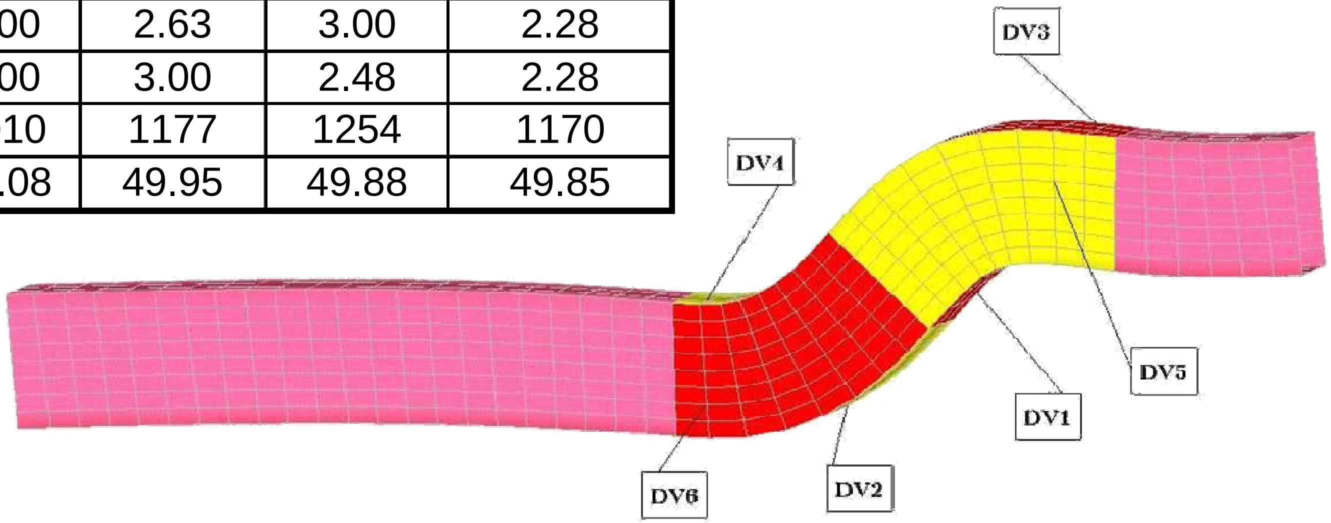
V_0 = 2m/s

m = 1000kg

Case Study 1: Rail Optimization

Size Optimization

	Initial	Ord. RS	Seq. RS	Local app.
DV1 [mm]	2.00	1.70	2.02	2.25
DV2 [mm]	2.00	1.92	1.57	2.95
DV3 [mm]	2.00	0.20	1.56	2.11
DV4 [mm]	2.00	1.95	1.65	1.94
DV5 [mm]	2.00	2.63	3.00	2.28
DV6 [mm]	2.00	3.00	2.48	2.28
E_{max} [Nm]	1010	1177	1254	1170
Δl_x [mm]	52.08	49.95	49.88	49.85



Case Study 2: Roof Crush

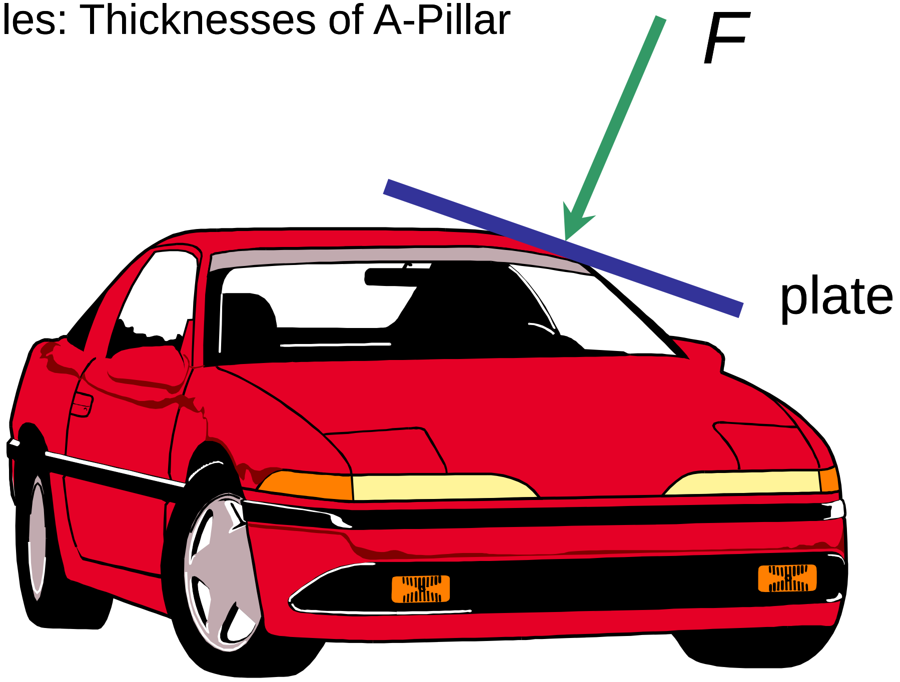
- The US standard FMVSS216 requires a certification which simulates the roll over of a vehicle
- The certification test consists of moving an undeformable plate into an A-pillar at a constant velocity and the load / displacement response monitored
- The objective is to minimize the mass of the design and still ensure that the applied load meets the certification requirement. The design variables consist of five material thicknesses in the A-pillar and the roof frame

Case Study 2: Roof Crush

Objective : minimize mass

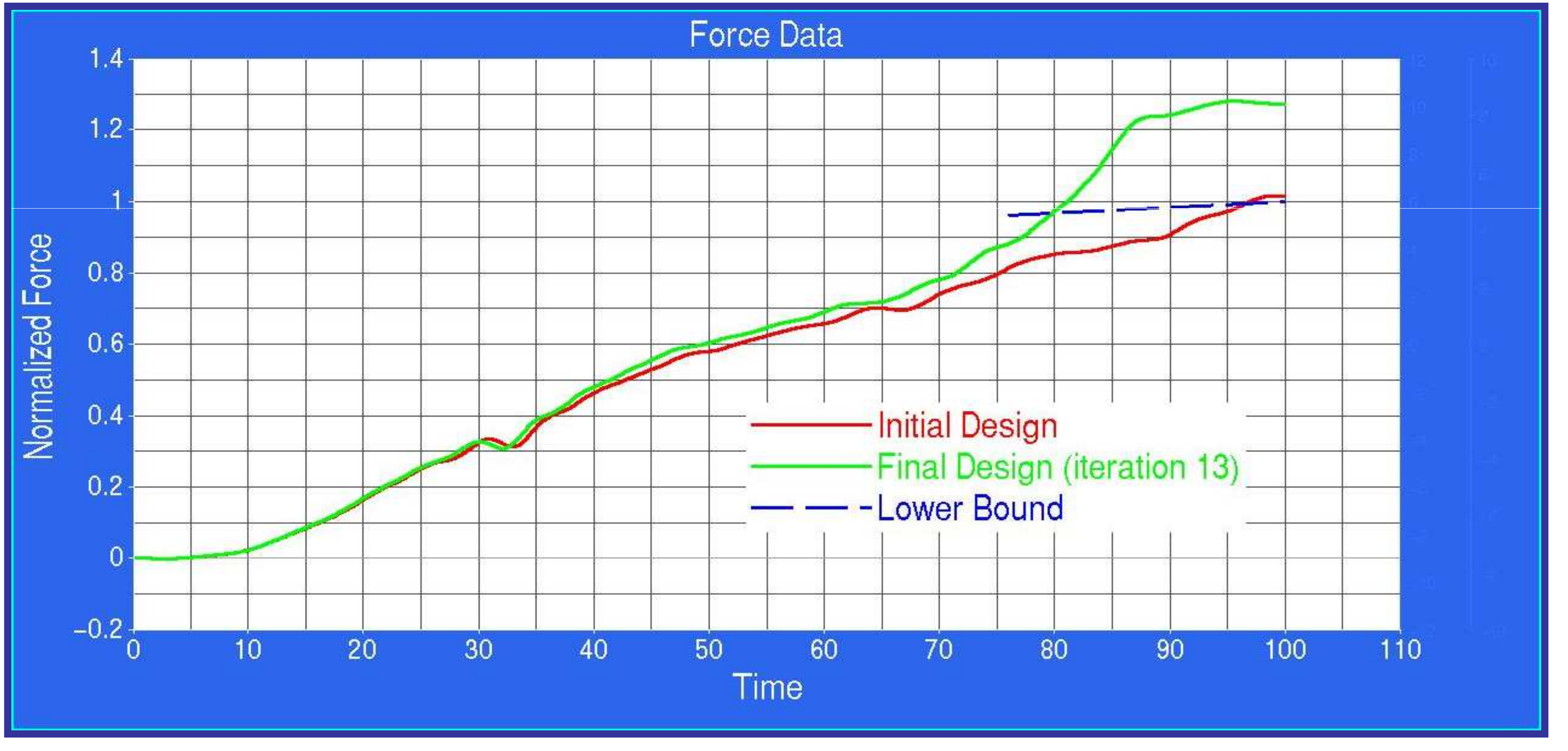
Constraint : max $F(d) > F_{limit}$

Design variables: Thicknesses of A-Pillar



Case Study 2: Roof Crush

Variation of the Normalized Constraint with Time

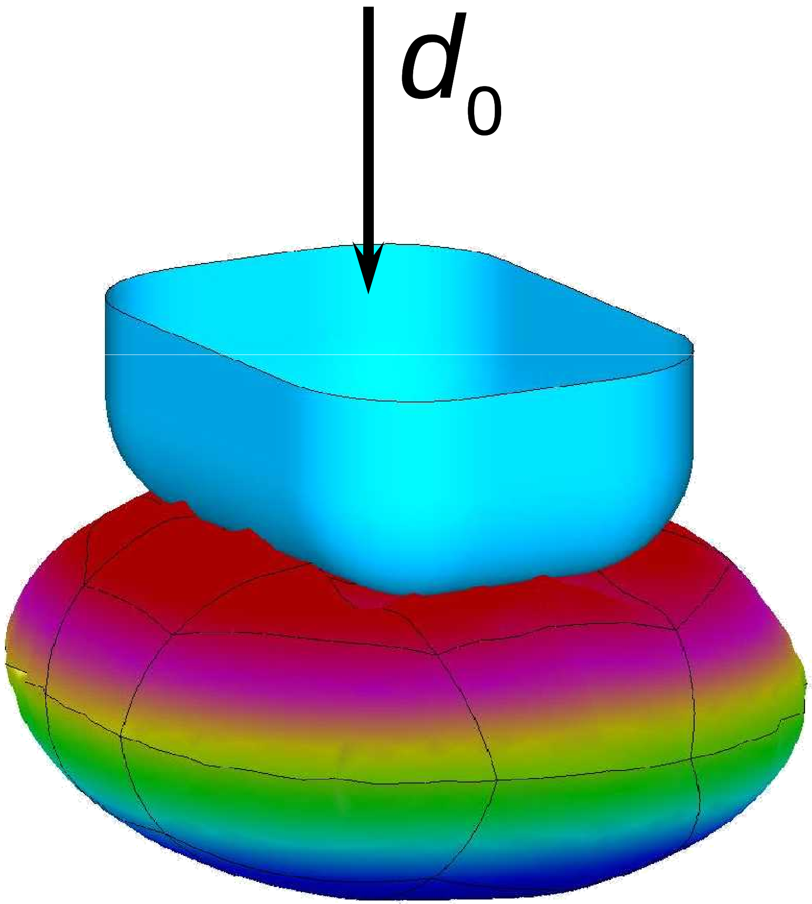


Altair Proprietary and Confidential Information

Case Study 3: Airbag

- Airbag impactor test requires the HIC not be exceed 1000. The impactor should not displace more than 260mm. The design variables are vent orifice area, fabric permeability, ignition time and mass flow rate of the gas
- The test set-up consists of allocating an initial velocity to the impactor
- After twenty-two iterations the HIC has been considerably reduced

Case Study 3: Airbag



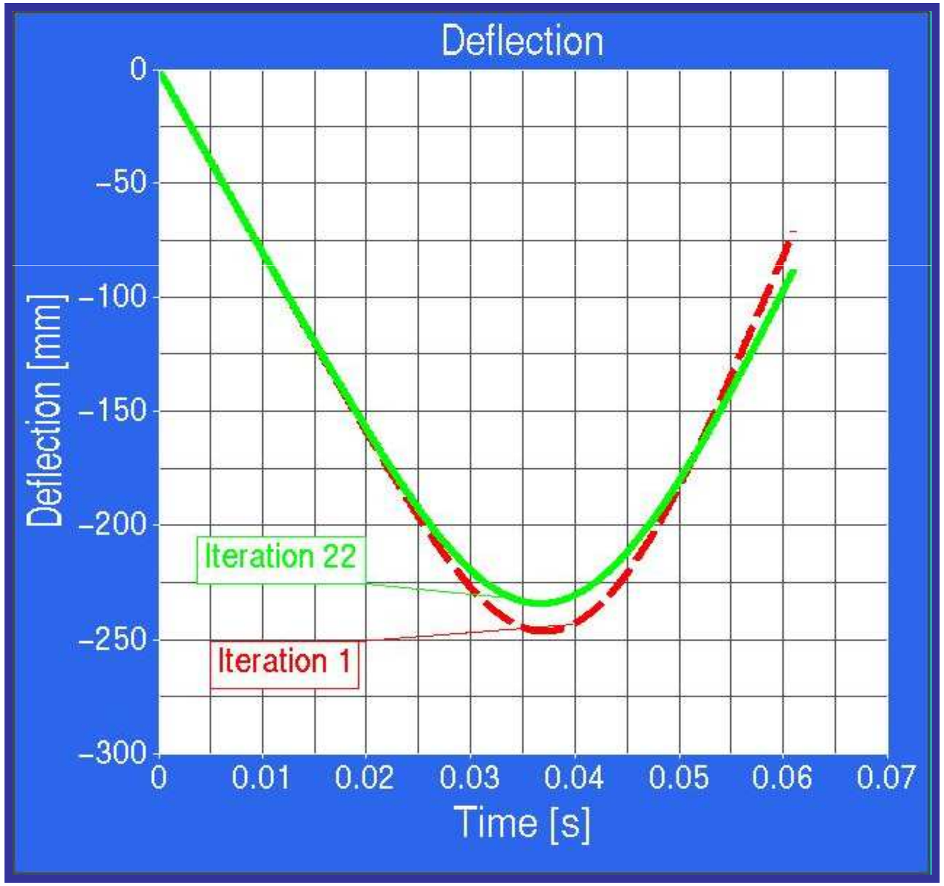
Objective : minimize head injury criterion (HIC)

$$HIC = \max_{t_2 - t_1 \le 36ms} \left(\frac{1}{t_2 - t_1} \int_{t_1}^{t_2} \ddot{d}_{impactor}\, dt \right)^{2.5} (t_2 - t_1)$$

Constraint : Deflection d_{max} < 260mm

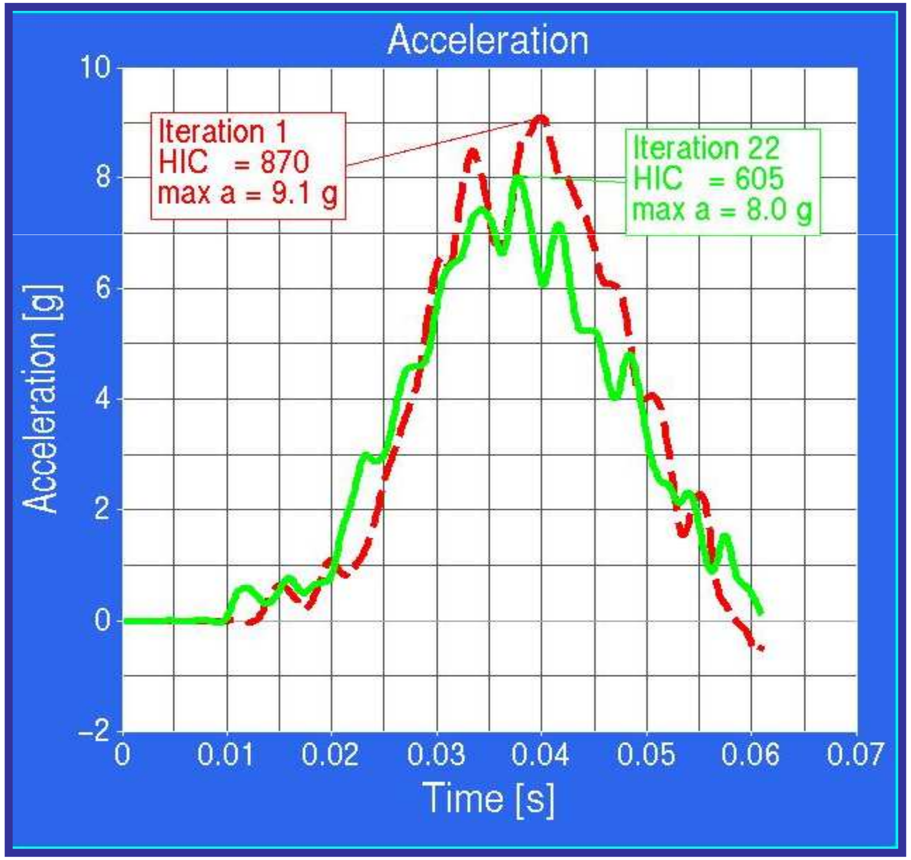
Design Variables:
- Ignition time
- Fabric permeability
- Mass flow rate
- Vent orifice area

Case Study 3: Airbag



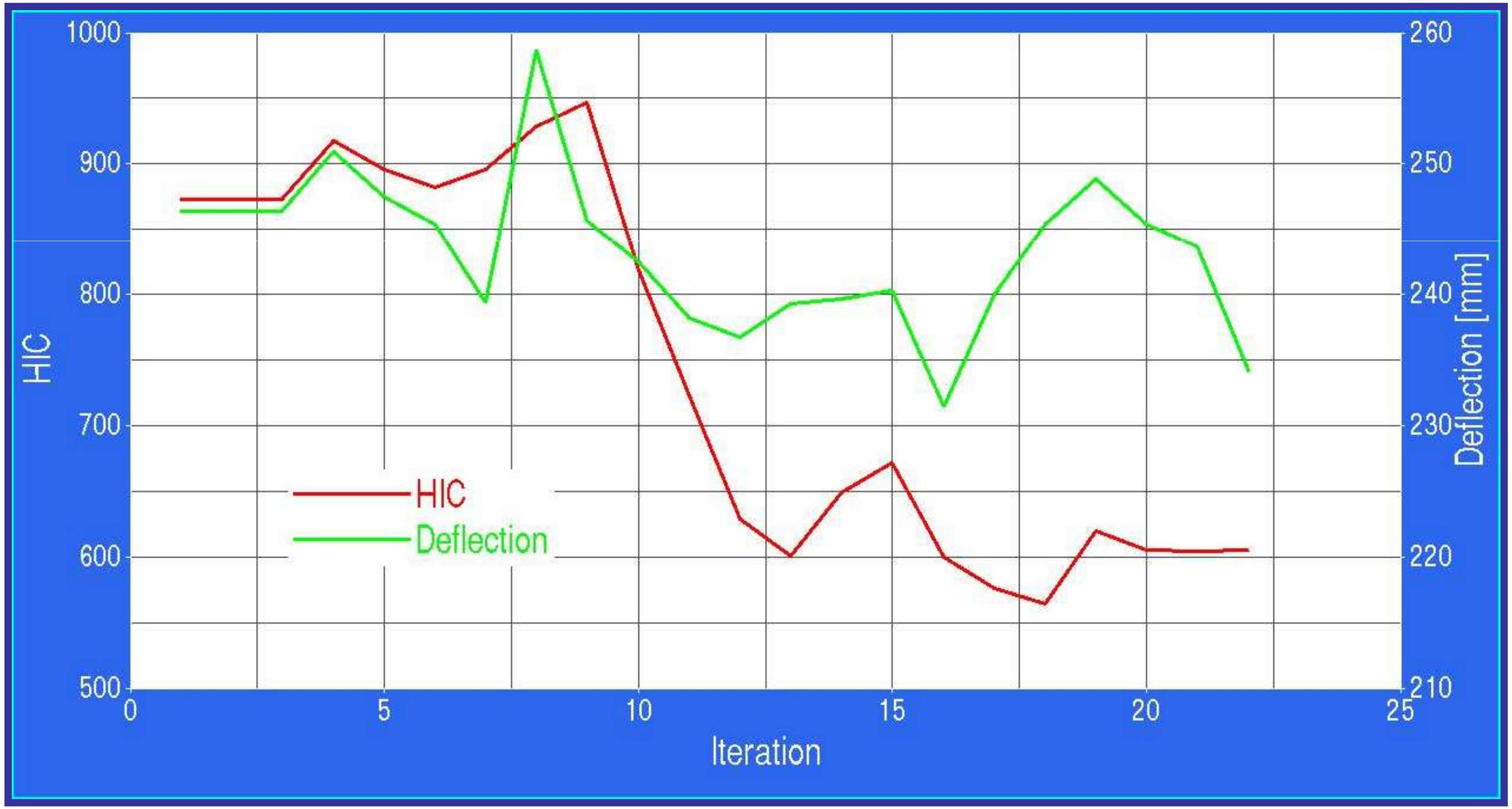
Case Study 3: Airbag

Objective with Time



Case Study 3: Airbag

Variation of the Objective / Constraint with Design Iteration History

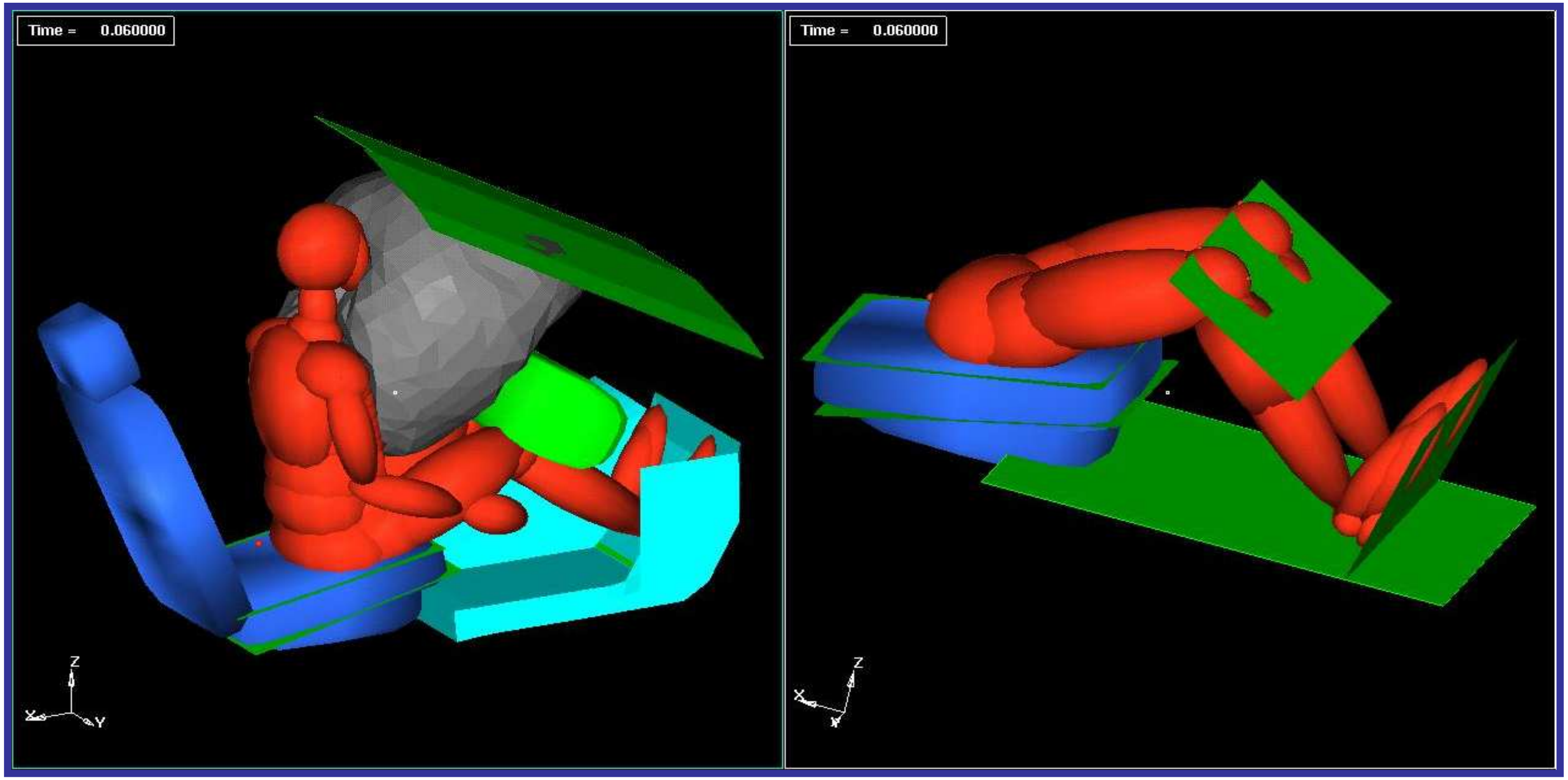


Case Study 4: Knee Bolster Design

- This study considers the occupant behavior to determine the initial package space for a knee bolster. The model consists of the lower extremities of the dummy.
- The dummy is loaded by an acceleration pulse $a_x(t)$ resulting from the crash test in accordance with FMVSS208
- The objective is to minimize the femur load with a limit on the knee penetration into the knee bolster. The design variables consist of geometric and stiffness parameters

Case Study 4: Knee Bolster Design

Substitute Model

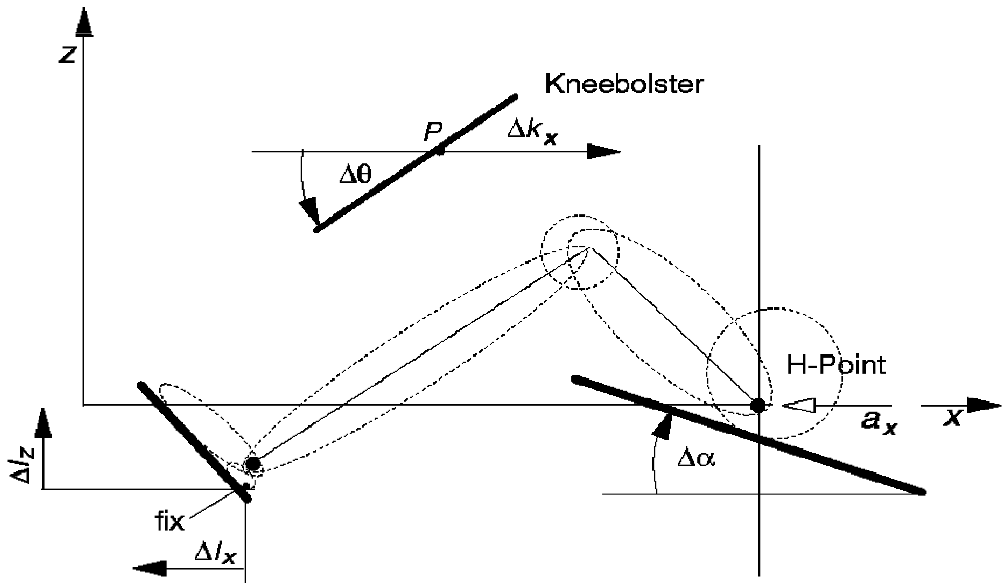


Optimization Problem

Objective : Minimize Femur Load: $F_{femur}(\Delta\theta, \Delta k_x, \Delta l_x, \Delta l_z, \Delta\alpha) \Rightarrow \min$

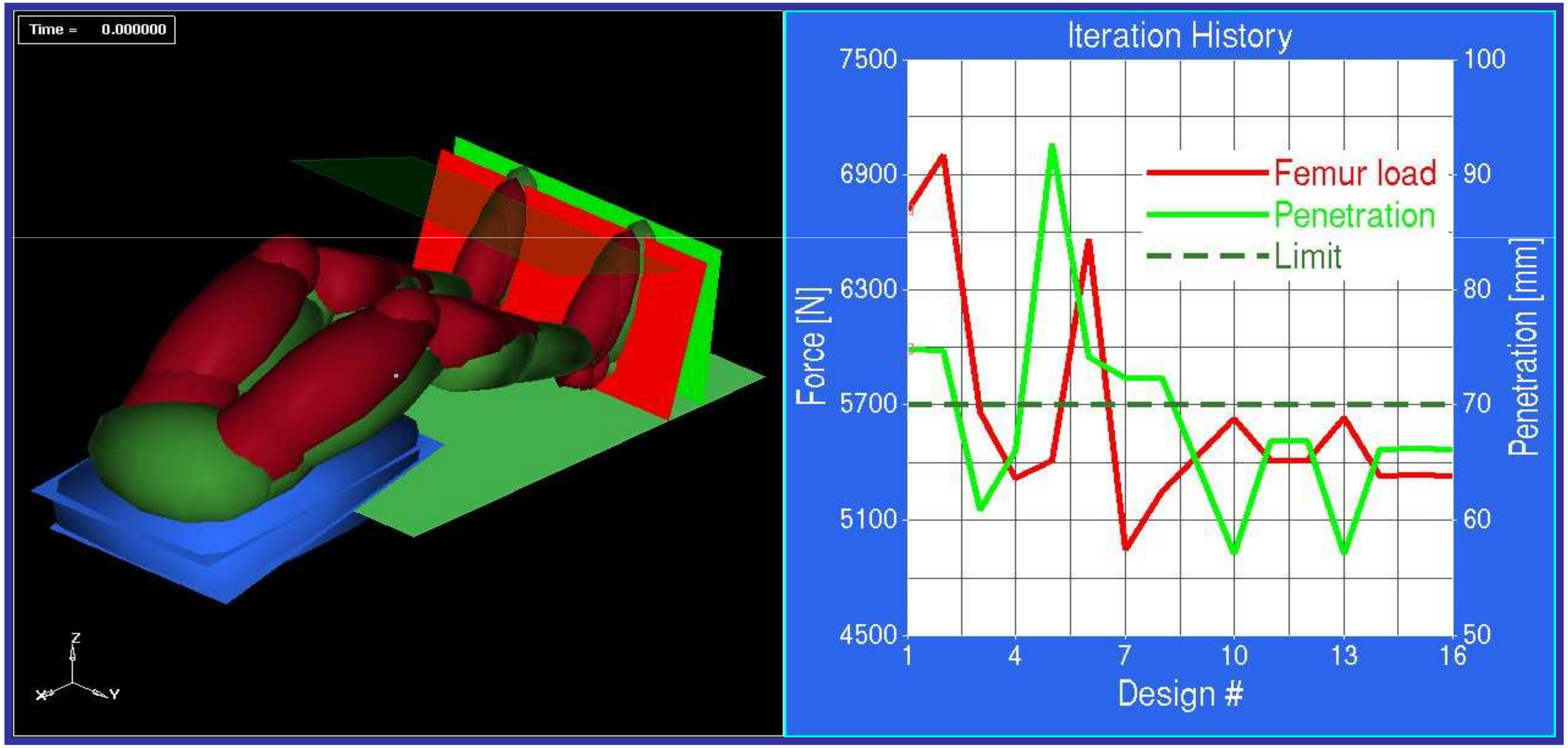
Constraint : Knee Penetration: $\delta(\Delta\theta, \Delta k_x, \Delta l_x, \Delta l_z, \Delta\alpha) \leq 0.07\,m$

Design Variables : Geometric and Stiffness Parameters $(\Delta\theta, \Delta k_x, \Delta l_x, \Delta l_z, \Delta\alpha)$



Case Study 4: Knee Bolster Design

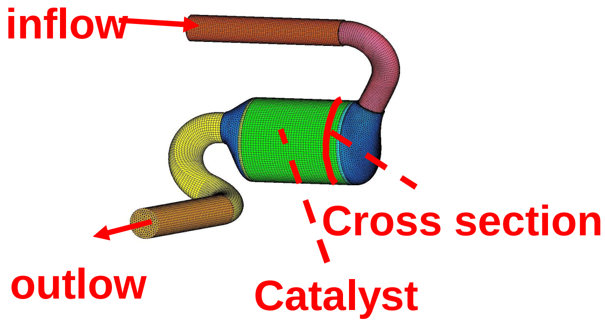
Results



Design Variable Results

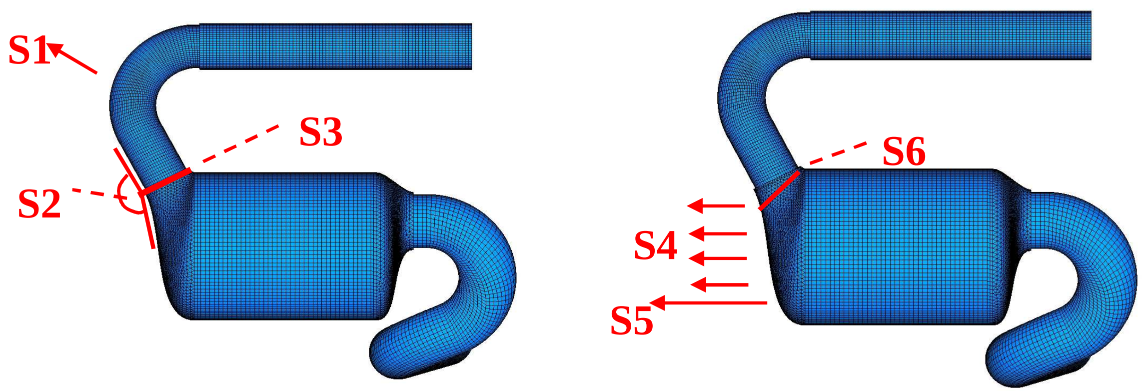
DESIGN VARIABLE	LOWER BOUND	UPPER BOUND	OPTIMAL VALUE
$\Delta\vartheta^\circ$	-10.00	10.00	-10.00
Δk_x	-0.10	0.07	-0.10
Δl_x	-0.05	0.03	0.03
Δl_z	-0.05	0.05	0.05
$\Delta\alpha^\circ$	-5.00	5.00	-4.65

Case Study 5: Exhaust System Design



- **Objective:** Uniform flow velocity in section S
  - Average velocity
  - Local deviation:
  - Uniformity-Index:
- **Constraint:** Min. pressure drop between inflow and outflow

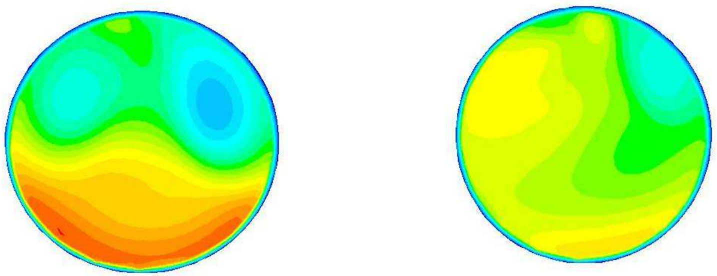
6 Shape Variables



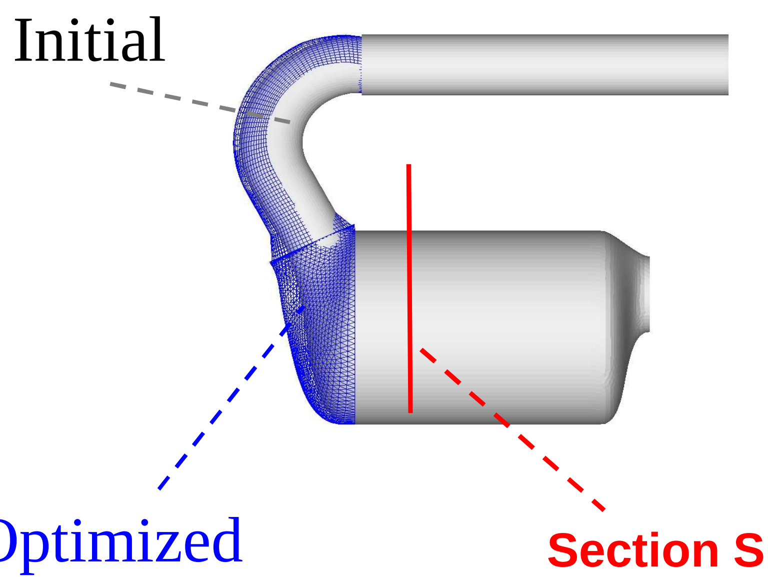
Results

γ +12.0%

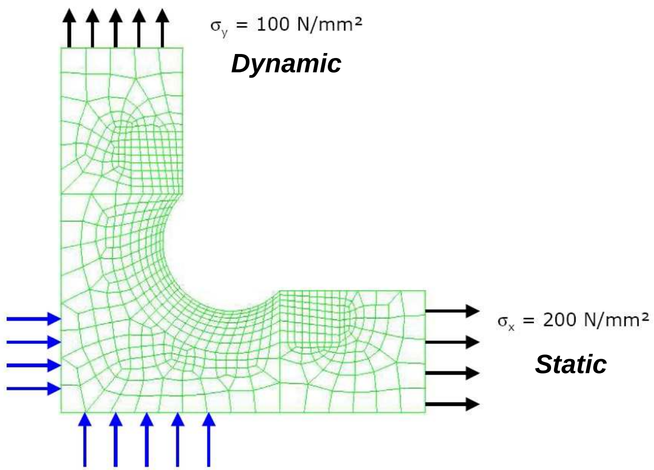
Δp -16.0%



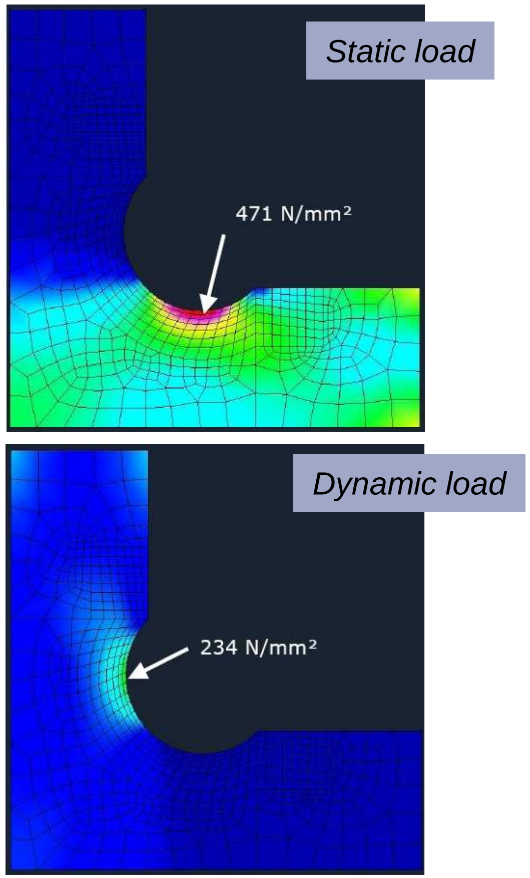
Flow Velocity in Section S



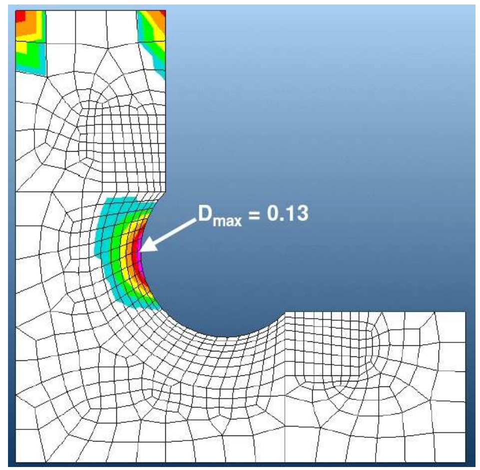
Case Study 6: Notch Design



Loads & Boundary Conditions



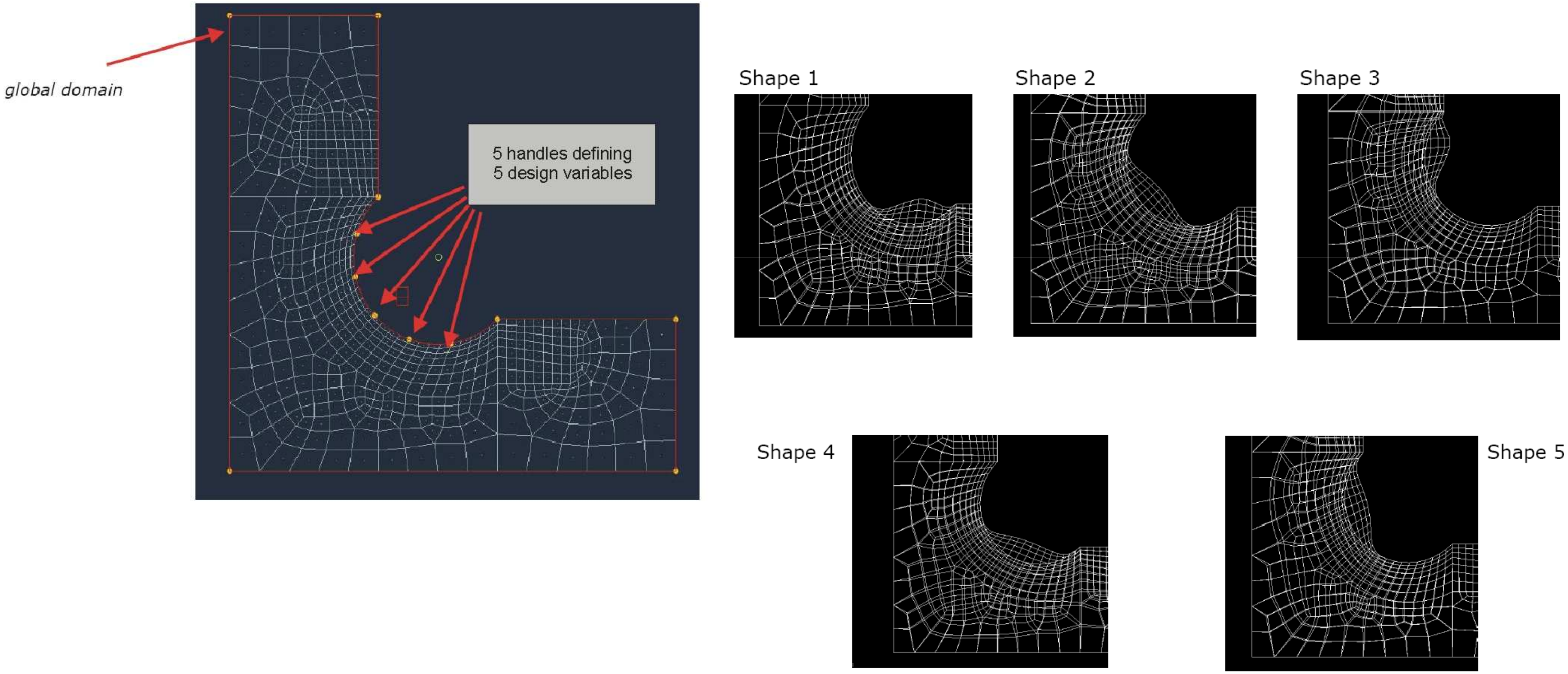
Stresses



Damage

Courtesy of MAGNA STEYR

Design Variables for Shape Optimization

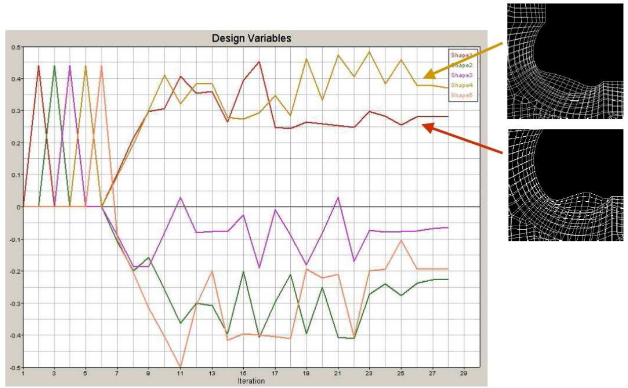


Courtesy of MAGNA STEYR

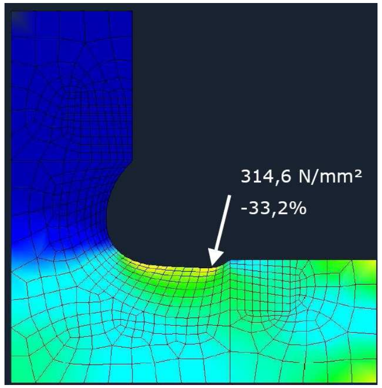
Case Study 6 : Notch Design

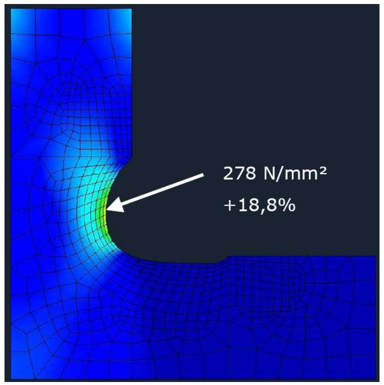
- Optimization Problem
  - Minimize maximum stress at notch edge
  - Both load cases considered
  - Shapes scaled +/- 1
- Damage deteriorated!

Design Variable History



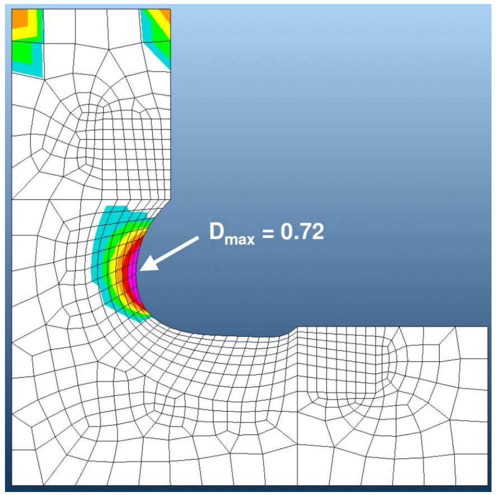
Stress Based Optimization





Stresses

Damage

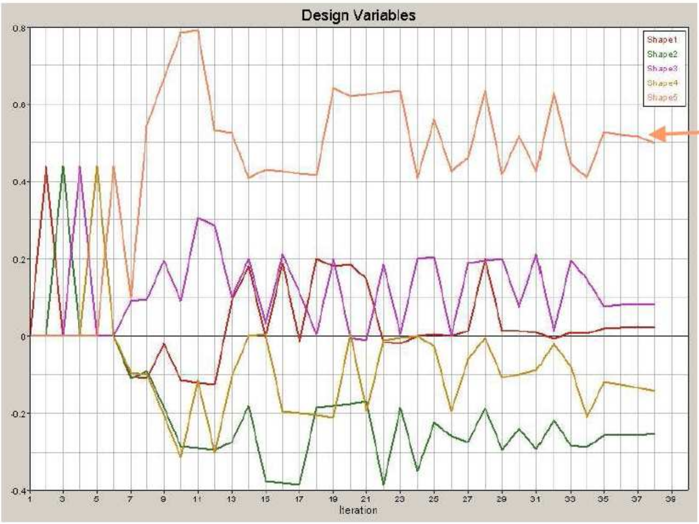


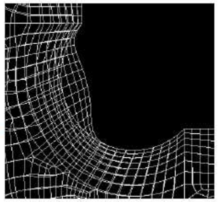
Courtesy of MAGNA STEYR

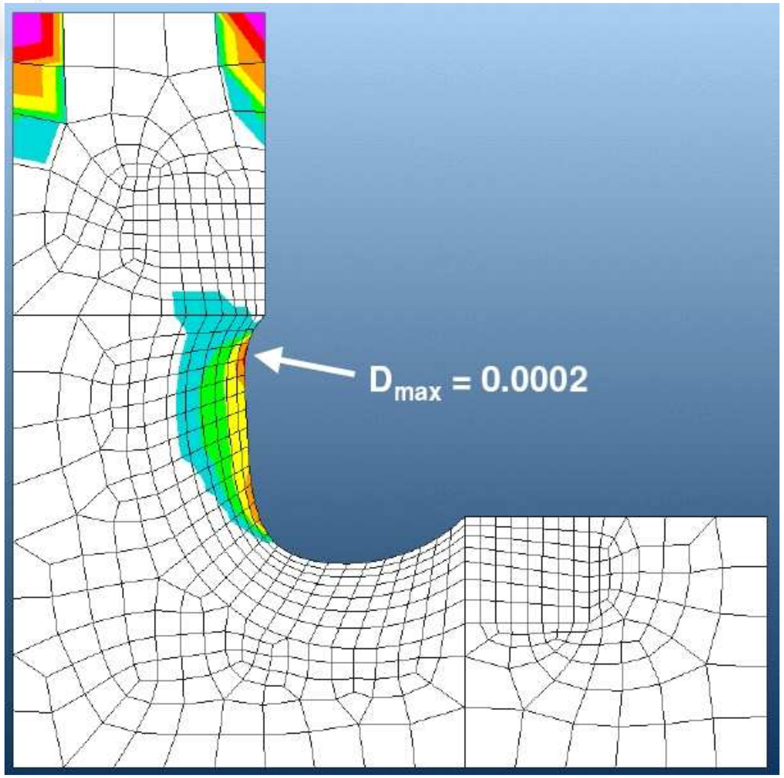
Case Study 6 : Notch Design

- HyperStudy identified dynamic load case to be sensitive to damage
- New contour different to stress optimized contour
- Damage significantly reduced

Damage Based Optimization







Courtesy of MAGNA STEYR

GENERAL DISCUSSION

- Optimization commences once a clean baseline analysis has been performed (eg. negligible hourglassing and sliding energies)
- Care with shape variables: large perturbations can result in a reduced time step size
- Mass scaling used with shape optimization should be used with caution. Particularly if the objective is to minimize mass
- For size optimization, numerical contact can be effected by changing the thickness. Workaround by keeping the contact thickness constant
- Technology can be simultaneously applied to multiple loadcases (eg. central / oblique impact attitudes)

GENERAL DISCUSSION

- Changes in design variables can result in a completely different system response (eg. components not in contact may now contact)
- Original analysis duration may extend to capture the systems response optimization
- Computational time can be a barrier with each iteration requiring a complete explicit analysis
- Convergence is quickly achieved
- Optimized design can be conveniently restarted with new design variables or further optimization starting at a new position in the response surface

Exercise 4.1: Shape and Size Optimization of a Rail Joint

Exercise 4.2: Size Optimization Study using RADIOSS

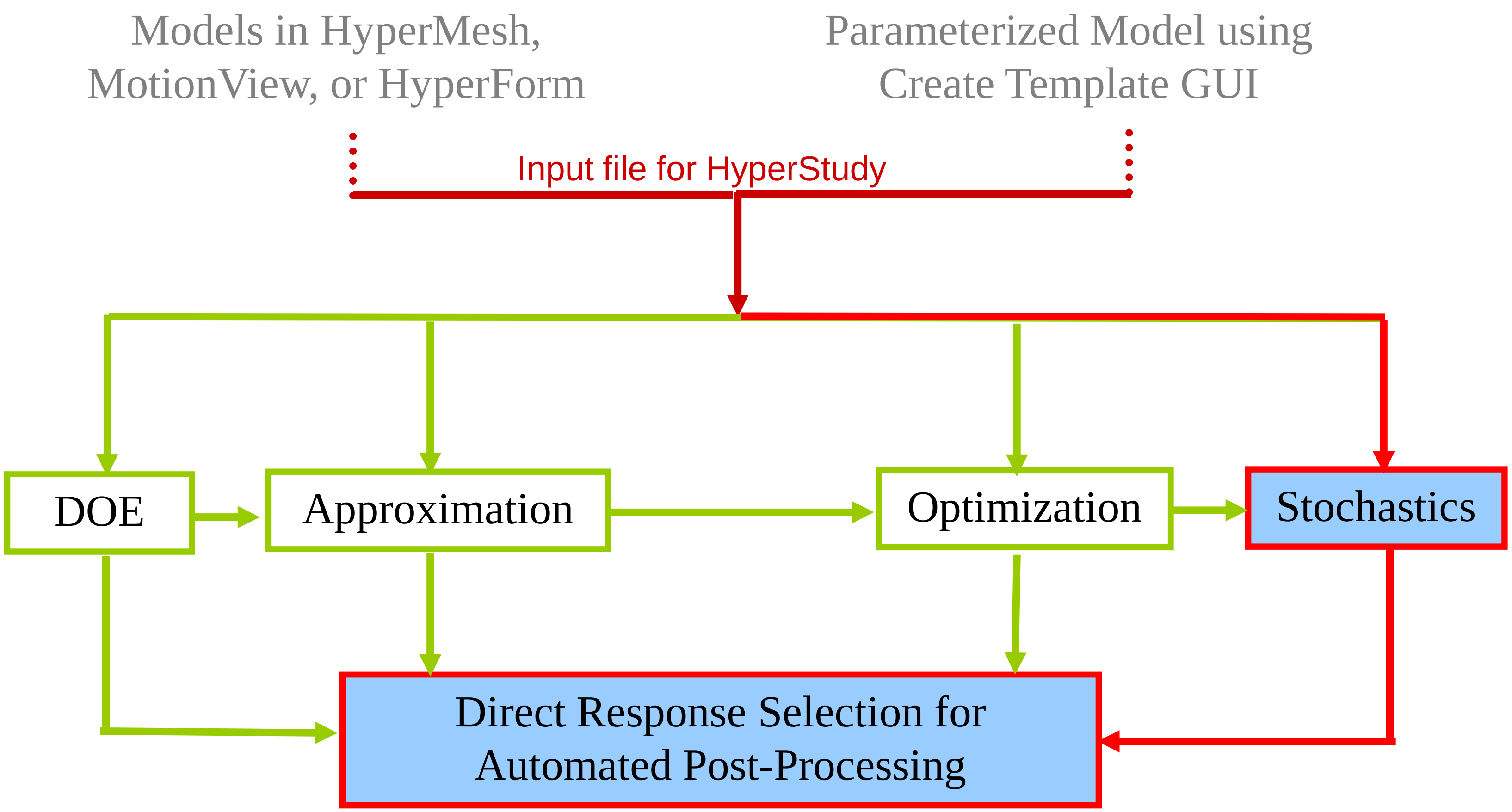
Exercise 4.3: Optimization Study Using an Excel Spreadsheet

Exercise 4.4: Shape Optimization Study using HyperForm

Exercise 4.5: Shape Optimization Study using ABAQUS

Chapter 5: Stochastic

Stochastic



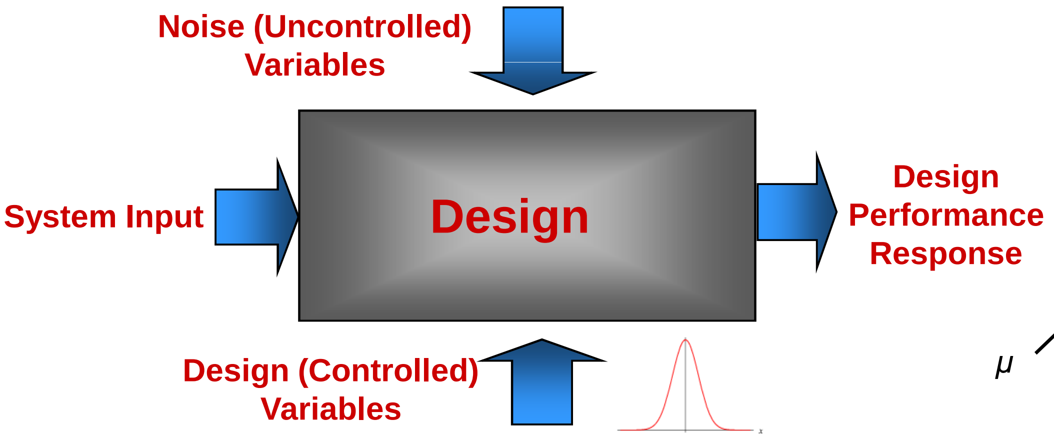
Models in HyperMesh, MotionView, or HyperForm
Parameterized Model using Create Template GUI
Input file for HyperStudy
DOE
Approximation
Optimization
Stochastics
Direct Response Selection for Automated Post-Processing

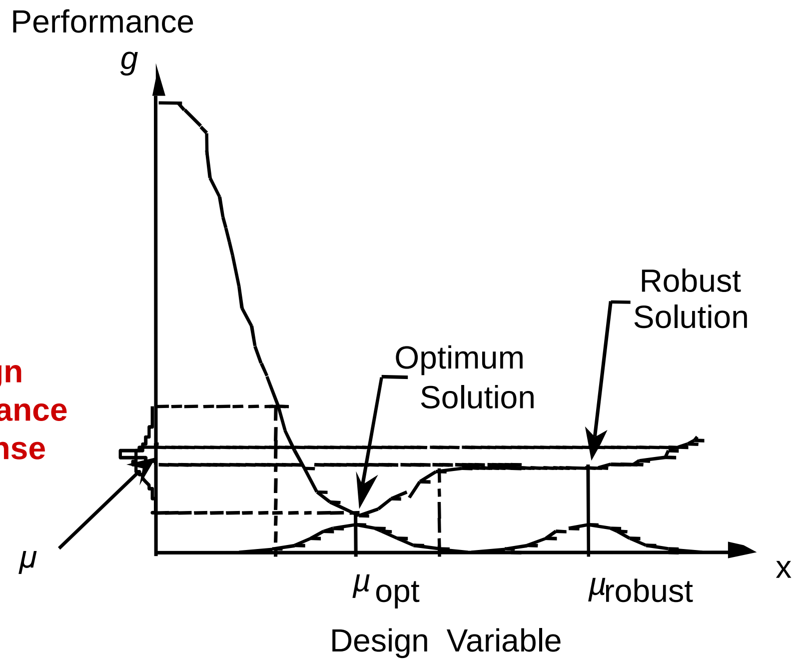
Uncertainty in Design

- **Physical uncertainty**
  - Loads
  - Boundary and Initial condition
  - Material properties
  - Geometry
- **Numerical simulation uncertainty**
  - Conceptual modeling
  - Mathematical modeling
- **Manufacturing**
  - Sheet metal thickness
  - Welds
  - Random design (controlled) variables
- **Loads**
  - Direction
  - Magnitude
  - Random noise (uncontrolled) variables
- **Material data**
  - Elastic properties
  - Failure
  - Random noise or design variables

Robust Design – Stochastic Study

Minimizing variations in performance caused by variations in design variables





Robustness Definition

- The design should be invariant against small changes of the parameters which are inevitable in the normal design process .
- Scattering of design parameters due to real world conditions in manufacturing and boundary conditions should not reduce the performance of the product.
- Robustness is the degree to which a system is insensitive to effects that are not considered in the design.
- Robust statistical procedures has to be designed to reduce the sensitivity of the parameter estimates to unreliability in the assumptions of the model.

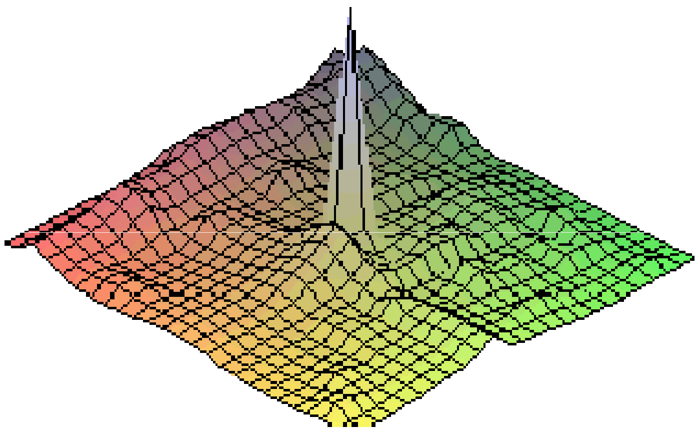
Robust Design

Type I Robust Design
(Flexible Specifications)

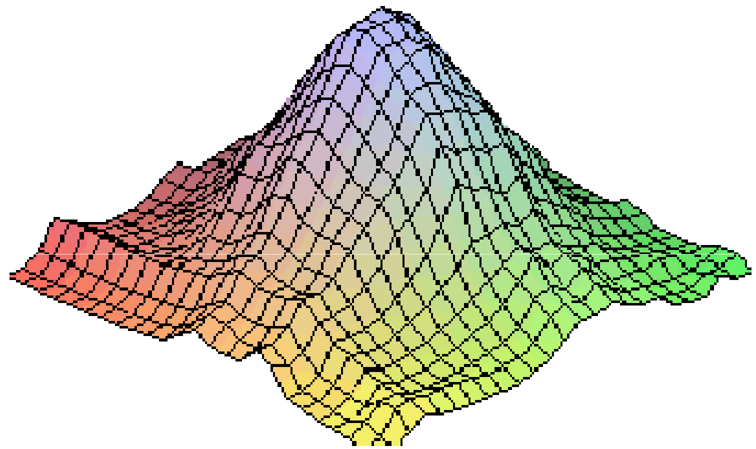
Given
Model f(x)
Find
A range of control factors x,
Δx (top-level specifications)
Satisfy
System constraints
Goals
Bring the mean μ_y on target
Minimize the variance σ_y
Minimize
Deviation Functions

Robustness Definition

Non-robust Design



Robust Design



Deterministic Optimization Approach

- Find acceptable solution to the design problem
- Design variables are continuous or discrete quantities
- Environmental conditions are given as deterministic load cases

$$f(\mathbf{x}) \Rightarrow \min$$

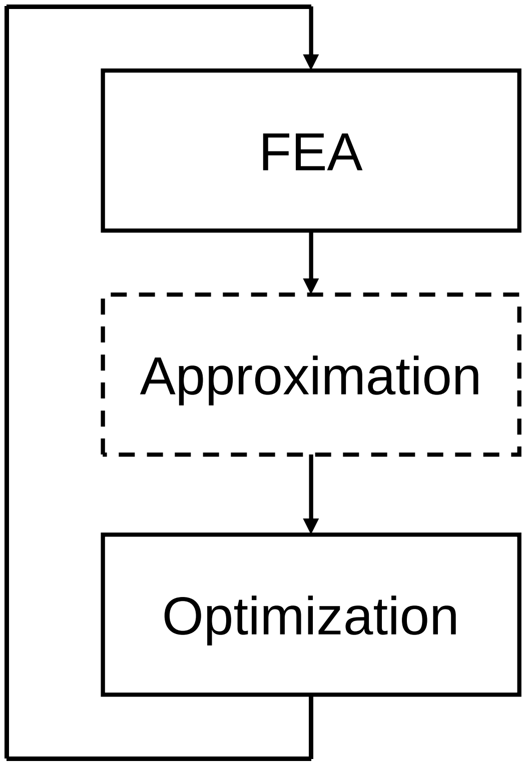
- Objective

$$g_j(\mathbf{x}) \leq 0$$

- Constraints

$$\mathbf{x}^L \leq \mathbf{x} \leq \mathbf{x}^U$$

- Design space



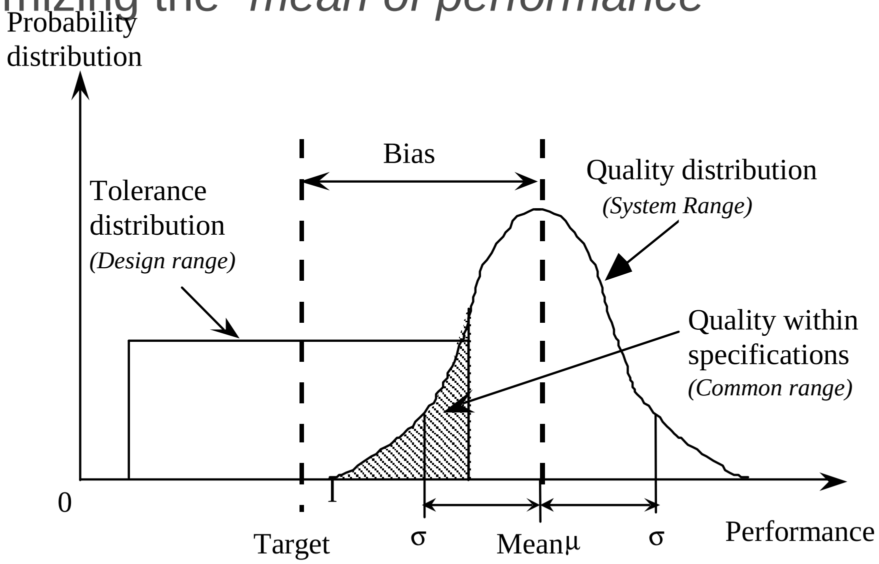
Deterministic Optimization Approach

- Deterministic optimization problem formulation does not incorporate uncertainty of design variables
- Optimal solutions for systems exhibiting highly non-linear responses can be misleading
- Surrogate models (response surfaces) built using certain approximation techniques tend to smooth the noisy behavior
- Most optimization algorithms push the constraints to the bounds in search of optimal solution

Robustness Optimization Approach

- Since the objective of any robust design problem is to bring the mean on target and minimize the variance, robust design problems should be modeled as bi-objective problem

- The two objectives are :
  - Minimize the variance of performance
  - Bring the mean on target or optimizing the "*mean of performance*"



Robust Design Optimization Formulation

- Objective $\quad \sigma(f(\mathbf{x},\mathbf{r})) \Rightarrow \min$

- Constraints

$$\mu(\mathbf{x},\mathbf{r}) - f_{\text{Target}} \le 0$$

$$\mu_j(\mathbf{x},\mathbf{r}) - g_{j\text{Target}} \le 0$$

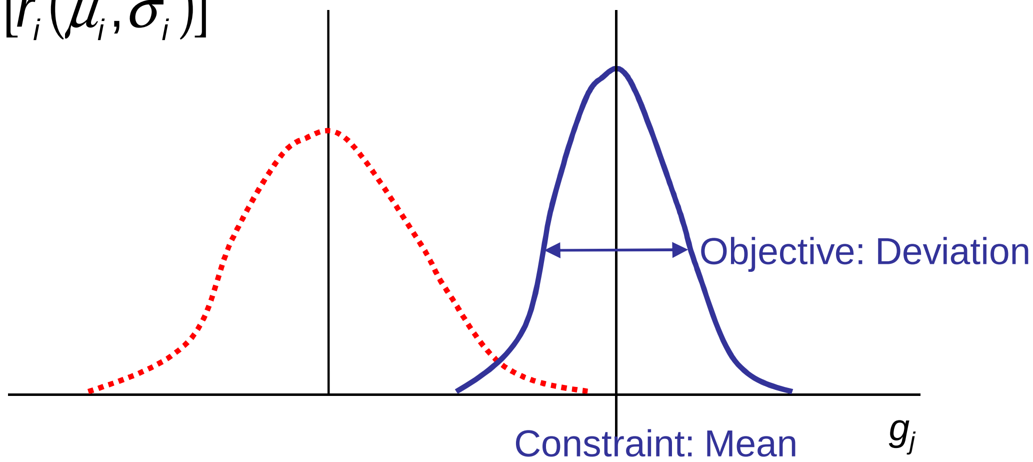
?

$$\mathbf{x}^L \le \mathbf{x} \le \mathbf{x}^U$$

- Design variables

$$\mathbf{r} = [r_i(\mu_i, \sigma_i)]$$

- Random variables



Exercise 5.1: Stochastic Study of a Rail Joint

Exercise 5.2: Stochastic Study of a Plate Model

REFERENCES

‘Structural Optimization in Occupant Safety and Crash Analysis’, U Schramm, D Schneider and H Thomas, OptiCON’98, California, Oct. 1998.

‘HyperOpt User’s Manual’, Version 6.0, Altair Engineering, Jan. 1999

‘AutoDv User’s Manual’, Version 3.0, Altair Engineering, June 1998

‘Validation of the Hybrid III Head and Neck Models for the Pole Test of FMVSS 201 Using Optimization Techniques’, L Fredriksson and A Hirth, CAD-FEM User’s Meeting, Oct. 1998

‘Crashworthiness optimization Study of a Bumper System’, K Simpson and G Harte, HyperWorks Conf. Proceedings, Jan 1999

‘Structural optimization - An Efficient Tool in Automotive Design’, U Schramm, ATZ WorldWide, pg 17-20, 100 (1998) 6.

‘Shape Optimization using Star-CD and HyperStudy’, M Ratzel and C E Baumann, STAR American Conference, Detroit, May 2006.